Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

A digital simulation platform with human-interactive immersive design for navigation, motion, and teleoperated manipulation of work-class remotely operated vehicle*#

Fanghao HUANG†1,2,3, Xiao YANG1,3, Xuanlin CHEN1,3, Deqing MEI1,2,3, Zheng CHEN†‡1,2,3

[1]*State Key Laboratory of Ocean Sensing, Zhejiang University, Hangzhou 310058, China*

[2]*Key Laboratory of Intelligent Robot for Operation and Maintenance of Zhejiang Province, Hangzhou 311121, China*

[3]*Ocean College, Zhejiang University, Zhoushan 316021, China*

†E-mail: huangfanghao@zju.edu.cn; zheng_chen@zju.edu.cn

Received June 6, 2024; Revision accepted Dec. 1, 2024; Crosschecked July 10, 2025

Abstract: Digital simulation of the full operation of a remotely operated vehicle (ROV) is an economically feasible way for algorithm pretesting and operator training prior to the actual underwater tasks, due to the huge difficulties encountered during the underwater test, high equipment cost, and the time-consuming nature of the process. In this paper, a human-interactive digital simulation platform is established for the navigation, motion, and teleoperated manipulation of work-class ROVs, and provides the human operator with the visualized full operation process. Specially, two mechanisms are presented in this platform: one provides the virtual simulation platform for operator training; the other provides real-time visual and force feedback when implementing the actual tasks. Moreover, an open data interface is designed for researchers for pretesting various algorithms before implementing the actual underwater tasks. Additionally, typical underwater scenarios of the ROV, including underwater sediment sampling and pipeline docking tasks, are selected as the case studies for hydrodynamics-based simulation. Human operator can operate the manipulator installed on the ROV via the master manipulator with the visual and force feedback after the ROV is navigated to the desired position. During the full operation, the dynamic windows approach (DWA)-based local navigation algorithm, sliding mode control (SMC) controller, and the teleoperation control framework are implemented to show the effectiveness of the designed platform. Finally, a user study on the ROV operation mode is carried out, and several metrics are designed to evaluate the superiority and accuracy of the digital simulation platform for immersive underwater teleoperation.

Key words: Underwater teleoperation; Telepresence; Navigation and motion control; Virtual reality; Visual and force assistance

https://doi.org/10.1631/FITEE.2400486 **CLC number:** TP242

1 Introduction

With the development of robotics (Hokayem and Spong, 2006; Liu et al., 2017; Chen et al., 2023), remotely operated vehicles (ROVs) have attracted increasing attention, whereby work-class ROVs can effectively replace human operators during the execution of remote and complex underwater tasks from a safe place onboard (Forbrigger and Pan, 2018;

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. 52075476 and 52301404) and the Zhejiang Provincial Natural Science Foundation of China (No. LR23E050001)

Electronic supplementary materials: The online version of this article (https://doi.org/10.1631/FITEE.2400486) contains supplementary materials, which are available to authorized users

ORCID: Fanghao HUANG, https://orcid.org/0000-0003-3710-114X; Zheng CHEN, https://orcid.org/0000-0003-0961-8758

Wang et al., 2020; Lu et al., 2022). Therefore, work-class ROVs are widely used in offshore and deep-sea operations, such as sampling and pipeline docking. The full operation process of a work-class ROV starts with navigation to the designated position (Long et al., 2022). Once the ROV reaches the designated operation area, it transitions from the navigation mode to the operation mode. At this stage, the human operator takes over, and teleoperates the manipulator installed on the ROV to perform precise tasks (Zhang JJ et al., 2018). To adapt to various operational environments and to improve the safety and precision of underwater tasks, tailored ROV navigation, motion control algorithms, and teleoperation control algorithms need to be developed for work-class ROVs. However, there still remain challenges in algorithm development for work-class ROVs, mainly due to the difficulties encountered during the underwater test, high equipment cost, and the time-consuming nature of the process. To address these issues, the establishment of a digital simulation platform becomes a feasible solution for algorithm pretesting and operator training for the full operation process before undertaking the actual tasks (Li et al., 2022).

Due to the complex terrain and the limited light in underwater environments, autonomous navigation and motion control of ROVs can effectively assist human operators in controlling the ROV to reach the desired positions (Kinsey et al., 2014; Zhao et al., 2014; Manzanilla et al., 2019; Tani et al., 2023). The typical method to test the navigation and motion control algorithm is to apply offline simulators, which have high calculation efficiency (Huang et al., 2023). However, it is difficult to support interaction with real data or sensors of the ROV in these offline simulators, which may result in inaccuracies during the algorithm testing process. As a result, algorithm testing for navigation with only the offline simulator is insufficient for direct application in a real ROV, which poses a challenge for the design of a digital simulation platform for the navigation and motion control of an ROV.

After the ROV reaches the desired position, the operation mode starts, and the human operator takes over to teleoperate the manipulator for the precise underwater tasks. Teleoperation technology plays a significant role for manipulator to perform underwater tasks, and thus, it is commonly used on ROV digital simulation platforms for operator training and algorithm pretesting (Cárdenas and Dutra, 2016; Zhang J et al., 2017a; Xia et al., 2023). During the underwater teleoperation process, telepresence is an important issue to evaluate whether and how well the human operator can receive the immersive feeling from a remote underwater environment, helping guide the slave underwater manipulator to accomplish the remote tasks immersively and effectively (Ferreira and Mavroidis, 2006; Yuan and Sun, 2023). The traditional way of implementing underwater tasks is with direct visual feedback on an ROV digital simulation platform, which offers the human operator the environment-related information on the slave side (Khatib et al., 2016; Reichherzer et al., 2018; Sivčev et al., 2018). For example, a virtual underwater manipulator and its carrier have been previously developed and presented in the form of a video animation (Zhang J et al., 2017b). However, the single vision-based environment perception and rebuilding process is insufficient to meet the demands for fully and immersively rebuilding the underwater environment. The use of force information as feedback is another effective solution to provide intuitive and immersive feeling from the underwater environment for the human operator (Chen et al., 2020, 2024; Sun WC and Yuan, 2023), such as providing the interaction force between the slave manipulator and the underwater environment (Zhang DW et al., 2020). Therefore, comprehensive perception of the visual and force information of the underwater environment becomes an effective way to improve the telepresence on ROV digital simulation platforms.

Additionally, existing research lacks comprehensive designs of a digital simulation platform that simultaneously considers both the teleoperation control of the manipulator and the navigation and motion control of an ROV, thus failing to provide the full operation process of a work-class ROV. For instance, Zhang J et al. (2017a) considered only the motion control of the manipulator during the underwater task, and Khadhraoui et al. (2016) focused solely on the navigation and motion control of the ROV.

Considering these challenges encountered in the development of a work-class ROV, a virtual reality (VR)-based digital simulation platform is designed in this paper for the full operation process of work-class ROVs. By the comprehensive perception

of visual and force information of the underwater environment, the designed platform provides the human operator with quality-improved underwater virtual training using the recorded in-situ data (described as Mechanism 1), which avoids expensive costs of real underwater tests, and helps improve operator skills before implementing the real tasks. In this mechanism, an open data interface is provided for the ROV navigation and motion control algorithm and the teleoperation control algorithm, offering the human operator a custom platform for algorithm testing, development, and verification. Moreover, it provides the visual and force feedback with immersive feeling when implementing the actual underwater tasks (described as Mechanism 2), which is designed as an additional function of this platform. In this way, a novel solution is proposed via this digital simulation platform, which can effectively reduce the training cost and enhance underwater operability with visual and force assistance.

The main contributions of this work are listed as follows:

1. Two mechanisms are proposed for work-class ROVs in the digital simulation platform, with one providing a virtual simulation with virtual scenarios to achieve human operator training, and the other providing real-time visual and force feedback when implementing the actual tasks. In the meantime, an open data interface is designed for researchers, which enables algorithm pretesting before undertaking the actual underwater tasks.

2. Based on the real video signal or the recorded in-situ data, a virtual scene with hydrodynamics effect is accurately created, and the motion of the ROV can be simulated in the virtual scene, which provides the human operator with intuitive and immersive visualization from the underwater environment.

3. A ROV navigation and motion control unit is presented to drive the ROV to the working position. This unit is designed with open data interfaces in both the navigation and the control modules, allowing researchers to test and evaluate their own navigation and control algorithms for the motion of ROVs. The dynamic windows approach (DWA)-based local navigation algorithm and sliding mode control (SMC) controller are proposed to implement navigation and motion control during the full process of ROV operation.

4. A teleoperation control framework is designed, wherein a modified wave-variable architecture is proposed to provide the human operator with accurate force feedback, and simultaneously ensure the stability of the teleoperation system. A radial basis function neural network (RBFNN)-based slave controller is proposed to improve the tracking performance of slave manipulator.

5. A user study on the ROV operation mode is carried out, which evaluates and verifies that the digital simulation platform can accurately simulate the underwater operation tasks with recorded in-situ data, and can effectively train the human operator with immersive telepresence via the visual and force feedback during the manipulation task.

2 Framework of the digital simulation platform

In this paper, typical underwater teleoperation scenarios of an ROV are selected, where the human operator can control the master manipulator (a PHANTOM Omni haptic device) from a safe place, and send the required commands to the slave manipulator (a manipulator installed on the ROV) to accomplish the remote tasks after navigating the ROV to the desired position. To obtain the telepresence during the teleoperation process, a human-interactive digital simulation platform is established in this section, as shown in Fig. 1. It can provide the human operator with an immersive feeling from the remote environment during the full operation process, and guide the slave manipulator to better accomplish the tasks with the visual and force feedback assistance. This digital simulation platform mainly consists of a virtual scene unit, an ROV navigation and motion-control unit, a teleoperated manipulation unit, an environment perception and rebuilding unit, and an in-situ data-monitoring unit.

However, considering the extreme difficulty, harsh conditions, and expensive equipment of underwater tests, it is usually challenging to frequently implement the teleoperation tasks of ROV in the real underwater environment. Thus, the designed digital simulation platform has two mechanisms:

Mechanism 1. This involves virtual training of the full operation process of work-class ROVs with the recorded in-situ data. This mechanism allows the human operator to test and improve his/her algorithms by simulating the full operation process,

and hone his/her skills before implementing the real underwater test. Moreover, this mechanism allows the digital simulation platform to restore the actual operation process of the ROV using the recorded in-situ data, and can be used to find any problems and gain experience before implementing the next actual underwater test.

Mechanism 2. This facilitates the real-time visualization and operation of a work-class ROV and its installed manipulator with an immersive feeling. This mechanism provides the human operator with the visualized ROV data during the navigation process along with the visual and force feedback with in-situ data during the real underwater teleoperation process, and guarantees human-interactive immersive guidance for the slave manipulator to accomplish the tasks after the ROV navigation process.

In detail, based on the target position obtained from the real video signal and the current ROV state data, the ROV navigation and motion control unit can simulate the motion of the ROV to the target position. After arriving at the target position, the ROV can adjust its posture and start the operation using its installed manipulators. In the operation mode, based on the proposed teleoperation control framework in the teleoperated manipulation unit, the human operator is able to execute the task through the manipulator installed on the ROV under time delay. The environmental force is transmitted in the form of a wave variable to guarantee the stability of the teleoperation system, which helps the human operator sense the real environmental force. Subsequently, the virtual scene unit can transmit the target position and joint angles of the manipulator to the slave side, and these are displayed on the screen. The virtual scene unit enables the human operator to observe and simulate the tasks of the ROV in various perspectives. Additionally, the in-situ data-monitoring unit can monitor and display the in-situ ocean data (e.g., temperature, salinity, and depth), the ROV state data, and the manipulator state data, which are obtained from the slave side or the virtual scene.

Thus, based on the units in Fig. 1, the virtual scene and the in-situ ocean data are displayed on the screen to visualize the full operation process of the ROV, while the force feedback is provided by the master manipulator, improving the human operator's telepresence with the visual and force feedback.

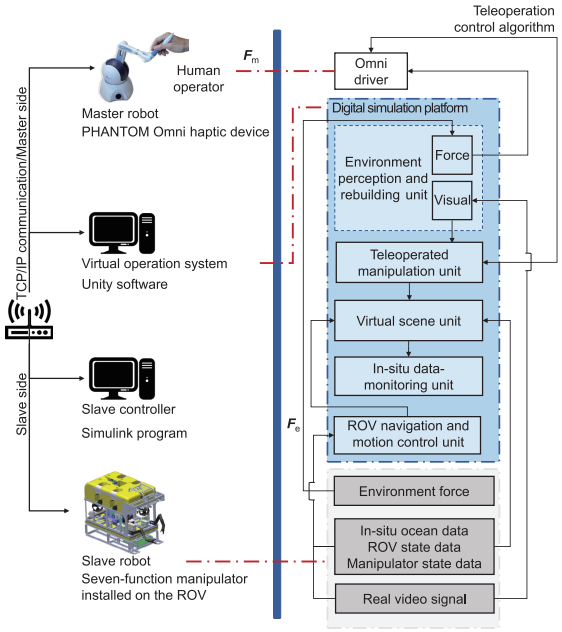


Fig. 1 Framework of the digital simulation platform, where red dashed lines indicate a correspondence relationship, and black arrow lines represent the signal flow between different modules. F_m: force feedback on the master side; F_e: real environmental force on the slave side. References to color refer to the online version of this figure

Remark 1 The traditional underwater operation (Lin and Kuo, 2001) usually uses a joy stick to control the slave robot only in the virtual environment without any remote information feedback to the human operator, which mainly results in poor telepresence. Therefore, accurately rebuilding the physical working environment on the master side becomes quite difficult because the human operator is usually far away from the working environment. Thus, the virtual environment in most of existing virtual training systems cannot be accurately modeled since the actual environment information is not accurately obtained, while the same joy stick used in the training cannot provide telepresence for the human operator. Different from previous works on virtual training, a novel visual-and-force-assisted virtual operation system is developed in this paper, where the force feedback and the VR-based visual feedback are added. Two mechanisms are designed, with one providing the simulation training with recorded in-situ data, and the other providing the real-time visual and force feedback when implementing the actual tasks. Based on the data interface designed

in this system, the in-situ data can be recorded in real time and used to accurately create the in-situ data-driven virtual environment as much as possible. Furthermore, the physical interaction, including the hydrodynamics and dynamics of the ROV, can be simulated based on the force feedback, and the force feedback provides a relatively immersive feeling for the human operator when implementing the virtual training. Moreover, this data interface allows the system to test and improve the teleoperation algorithms during training, while finally implementing them in the actual tasks. Based on the preceding analysis, the virtual operation system developed in this paper can accurately build the in-situ data-driven virtual environment with visual and force feedback, wherein more immersive training can be provided to help the human operator gain the human–robot collaboration experience.

3 Design of each unit of the digital simulation platform

3.1 Design of virtual scene unit

Based on the real video signal obtained by the ROV, the virtual scene is rebuilt in the digital simulation platform, and actually creates the real underwater environment of the ROV. The virtual scene unit mainly includes the ROV three-dimensional (3D) modeling and the underwater scenario modeling parts; the ROV 3D modeling part includes the proportional ROV 3D model established using the SolidWorks software, and the underwater scenario modeling part includes the underwater terrain, underwater environment, collision detection, and data mapping.

3.1.1 ROV 3D modeling

Due to the limitations of the modeling capabilities of the Unity software, a proportional ROV 3D model is built using the SolidWorks software based on the real ROV structure, as shown in Fig. 2. This model is equipped with two seven-function manipulators to simulate the ROV's real underwater operation tasks (Zhang J et al., 2017b). Supposing that the positive direction of the X axis for the ROV model is the positive direction of motion, the ROV model is dragged and dropped in the Unity software screen to guarantee that the heading direction of ROV is always the positive direction of the X axis.

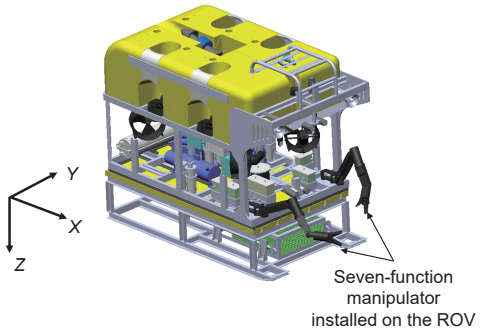


Fig. 2 The 3D model of the ROV

3.1.2 Underwater scenario modeling

Based on the real video signal, the real underwater scenario is selected and rebuilt in the digital simulation platform. As shown in Fig. 3, the underwater terrain rebuilt in the virtual scene is quite similar to the original one.

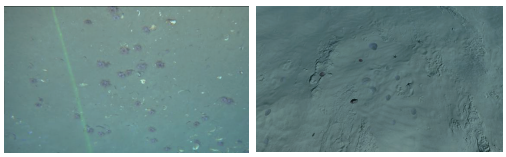


Fig. 3 Underwater terrain in the real video (a) and the virtual scene (b)

The procedure to rebuild the underwater scenario in the virtual scene is as follows:

Underwater terrain. The point cloud of the underwater terrain is first obtained by the RGB-D camera mounted on the ROV. Since the information obtained by the RGB-D camera requires larger storage space to restore the unprocessed point cloud data, and consumes more computation resources to complete the filtering operation during error correction, it may be not suitable for real-time modeling of the underwater terrain. Since the digital elevation model (DEM) file integrates all the underwater terrain data with only one filtering operation, the point cloud information combined with the DEM is much easier to be restored. Based on this idea, the DEM file is grayed, and the grayscale value is obtained as follows:

$$\boldsymbol{F} = r\boldsymbol{R}_{\mathrm{r}} + g\boldsymbol{G}_{\mathrm{r}} + b\boldsymbol{B}_{\mathrm{r}}, \tag{1}$$

where $\boldsymbol{F}$ is the grayscale value representing the average depth of the area in the underwater terrain, $\boldsymbol{R}_{\mathrm{r}}$,

$\boldsymbol{G}_{\mathrm{r}}$, and $\boldsymbol{B}_{\mathrm{r}}$ represent the matrices of the RGB components, and the grayscale coefficients are selected as $r = g = b = 1/3$.

To meet the requirement of the Unity software application, a 16-bit gray map file is generated by the grayscale value obtained earlier, wherein the horizontal axis corresponds to the longitude, the vertical axis corresponds to the latitude, and the gray value represents the terrain height. Subsequently, in the Unity software, a mesh plane with medium polygon density is created, while the grayscale map file is converted to the RAW file and imported, and the height of each vertex along the Z axis is changed with regard to the corresponding gray value in the mesh plane. Finally, based on the location of the ROV, the preset texture materials (e.g., rock, sealant, seashell, and other underwater creatures) are given to the mesh plane, thus completing the underwater terrain creation in the virtual scene, as shown in Fig. 3b.

Underwater environment. For the visual effect, the seawater color and atomization effect are considered. First, a material file with the gradient effect is added to the virtual scene, which can achieve a gradient effect of seawater color from shallow to deep; second, the atomization effect, which has the effect of being shrouded in a cloud of smoke, is added to the virtual scene to simulate the real underwater environment. The atomization effect includes the atomization color and atomization parameter f, the RGB values of the atomization color are selected as $R = 2$, $G = 49$, and $B = 58$, and the atomization parameter f is designed as follows:

$$f = \mathrm{e}^{-(d\rho)^2}, \tag{2}$$

where d is the distance between the target and the location of the ROV, ρ is the density parameter with the range of $[0, 1]$, and the value of ρ in this virtual scene is set as 0.005. Finally, using the atomization effect, the mixing of the virtual scene color and the atomization color is achieved, which means that the smaller the distance d, the larger the parameter f, and the clearer the virtual scene; while the larger the distance d, the smaller the parameter f, and the fuzzier the virtual scene. Detailed atomization effect is shown in Fig. 3b.

The hydrodynamics effect (Sun YX et al., 2022) is also considered while simulating the underwater environment in the physical aspect. The underwater flow is generated by the linear superimposition of each vortex in the X and Y directions. Assume that the underwater flow has the same velocity at different depths, since the 3D underwater flow can be simplified to two-dimensional (2D) underwater flow with multilayer superimposition, and the underwater flow models present similar flow fields at different depths. Subsequently, the flow speed can be described as follows:

$$\begin{cases} c_x(\boldsymbol{h}) = \sum\limits_{i=1}^{n} -\frac{\Gamma_i(h_y - y_i)}{2\pi r_i(\boldsymbol{h})}\left(1 - \exp\left(\frac{-r_i(\boldsymbol{h})}{\delta_i^2}\right)\right), \\ c_y(\boldsymbol{h}) = \sum\limits_{i=1}^{n} \frac{\Gamma_i(h_x - x_i)}{2\pi r_i(\boldsymbol{h})}\left(1 - \exp\left(\frac{-r_i(\boldsymbol{h})}{\delta_i^2}\right)\right), \end{cases} \tag{3}$$

where $c_x(\boldsymbol{h})$ and $c_y(\boldsymbol{h})$ represent the flow speeds at $\boldsymbol{h} = [h_x, h_y]$. Here, $[x_i, y_i]$, $\Gamma_i \in (6, 15)$, and $\delta_i \in (1, 3)$ represent the center position, intensity, and the radius of the i^{th} vortex, respectively. The term $r_i(\boldsymbol{h})$ denotes the distance between $\boldsymbol{h}$ and the i^{th} vortex's center position $[x_i, y_i]$; i.e., $r_i(\boldsymbol{h}) = \|\boldsymbol{h} - [x_i, y_i]\|$. $n = 50$ represents the number of vortices. To better simulate the underwater flow in this virtual scene, the speeds $c_x(\boldsymbol{h})$ and $c_y(\boldsymbol{h})$ are linearly superimposed with a limitation from 0 to 0.25 m/s, directly acting on the ROV dynamics modeled in Section 3.2.

Collision detection. To achieve the underwater operation tasks, the mesh-based collision body is attached to the ROV and the underwater terrain, which allows the virtual scene to have the ability to detect collision and rigid body contact.

Data mapping. The in-situ data (e.g., temperature, salinity, and depth) and the underwater terrain are matched, whereby the data relationship among the temperature, salinity, and the depth of the ROV is mapped.

3.2 Design of the ROV navigation and motion control unit

The ROV navigation and motion control process primarily consists of the mathematical modeling of the ROV, the navigation process, and the motion control process. This unit simulates the motion of the ROV to the target position, which is achieved through the following steps: First, based on the target position obtained from the real video signal, along with the current ROV state data obtained from the in-situ data-monitoring unit, the navigation module generates real-time desired velocity commands that can drive the ROV to the specified position. Then, the motion control module

generates control inputs based on the desired velocity commands. Finally, based on the control inputs of the ROV, the motion module simulates the real-time trajectory of movement of the ROV, which ultimately reaches the target position. After arriving at the target position, the ROV can adjust its posture and start the operation using its installed manipulators.

3.2.1 Modeling of the ROV

The motion module includes the mathematical model of the ROV in Fig. 2. The model uses five hydraulic motor-driven ducted propellers: one is arranged vertically above the ROV, and the others are arranged in the horizontal plane at certain angles, to achieve the motion of the ROV in the longitudinal, lateral, vertical, and heading directions.

Since in the teleoperation process, the main concern is the installed manipulators of the ROV, the ROV's six-degree-of-freedom mathematical model is simplified to a four-degree-of-freedom model (Zhang HP et al., 2022), which includes the translation on the X, Y, Z axes and rotation along the Z axis. The simplified four-degree-of-freedom mathematical model of ROV can be written as follows:

$$\dot{\boldsymbol{X}}_{\mathrm{R}} = R(\boldsymbol{X}_{\mathrm{R}})\boldsymbol{V}_{\mathrm{R}}, \tag{4}$$

$$\dot{\boldsymbol{V}}_{\mathrm{R}} = \boldsymbol{A}_{\mathrm{t}}\boldsymbol{V}_{\mathrm{R}} + \boldsymbol{C}_{\mathrm{t}}^{-1}\boldsymbol{T}_{\mathrm{t}} + \boldsymbol{D}_{\mathrm{t}}, \tag{5}$$

where $\boldsymbol{X}_{\mathrm{R}} = [x_{\mathrm{R}}, y_{\mathrm{R}}, z_{\mathrm{R}}, \phi_{\mathrm{R}}]^{\mathrm{T}}$, x_{R}, y_{R}, and z_{R} denote the positions of the ROV on the X, Y, and Z axes, respectively, and ϕ_{R} denotes the yaw angle. $\boldsymbol{V}_{\mathrm{R}} = [u, v, w, r]^{\mathrm{T}}$, where u, v, and w are the velocities of the ROV along the X, Y, and Z axes, respectively; r is the angular velocity of the ROV along the Z axis. $\boldsymbol{D}_{\mathrm{t}} = [d_u, d_v, d_w, d_r]^{\mathrm{T}}$ represents the external disturbances. $\boldsymbol{T}_{\mathrm{t}} = [X_{\mathrm{t}}, Y_{\mathrm{t}}, Z_{\mathrm{t}}, N_{\mathrm{t}}]^{\mathrm{T}}$ is the thrust force, which is regarded as the control input. $R(\boldsymbol{X}_{\mathrm{R}})$ is the rotation matrix and can be defined as follows:

$$R(\boldsymbol{X}_{\mathrm{R}}) = \begin{bmatrix} \cos(\phi_{\mathrm{R}}) & -\sin(\phi_{\mathrm{R}}) & 0 & 0 \\ \sin(\phi_{\mathrm{R}}) & \cos(\phi_{\mathrm{R}}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Furthermore, $\boldsymbol{A}_{\mathrm{t}} = \boldsymbol{C}_{\mathrm{t}}^{-1}\boldsymbol{P}_{\mathrm{t}}$, where $\boldsymbol{C}_{\mathrm{t}}$ and $\boldsymbol{P}_{\mathrm{t}}$ are defined as follows:

$$\boldsymbol{C}_{\mathrm{t}} = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 & 0 \\ 0 & m - Y_{\dot{v}} & 0 & 0 \\ 0 & 0 & m - Z_{\dot{w}} & 0 \\ 0 & 0 & 0 & I_z - N_{\dot{r}} \end{bmatrix}, \tag{6}$$

$$\boldsymbol{P}_{\mathrm{t}} = \begin{bmatrix} X_{u|u|}|u| & 0 & 0 & X_{vr}v \\ 0 & Y_1 & 0 & Y_r u - m \\ 0 & 0 & Z_1 & 0 \\ 0 & N_v u & 0 & N_1 \end{bmatrix}. \tag{7}$$

Herein, $Y_1 = Y_{v|v|}|v| + Y_v u$, $Z_1 = Z_{w|w|}w + Z_w u$, $N_1 = N_r u + N_{r|r|}|r|$, and m is the mass of the ROV; I_z is the moment of inertia of the ROV along the Z axis; $X_{\dot{u}}$, $X_{u|u|}$, X_{vr}, $Y_{\dot{v}}$, $Y_{v|v|}$, Y_r, $Z_{\dot{w}}$, $Z_{w|w|}$, Z_w, $N_{\dot{r}}$, $N_{r|r|}$, N_r, and N_v are the hydrodynamic coefficients.

3.2.2 Design of the navigation module

The navigation module is designed to generate the desired velocity command for the ROV to guide it toward the specified position. This module provides an open data interface for researchers since any navigation algorithm under development can be programmed for verification and improvement. Here, a DWA-based local navigation method is applied to guide the ROV to reach the target position.

Considering the four-degree-of-freedom model represented by Eq. (4) of the ROV, the navigation process is decomposed into a horizontal navigation process and a vertical navigation process. In the horizontal navigation process, the DWA-based navigation algorithm is used to guide the ROV's movement to the plane coordinates of the target position. Once the ROV reaches the desired horizontal position, it transitions to the vertical navigation process, whereby it ascends or descends vertically at a constant velocity to reach the target position.

The DWA-based navigation process is shown in Algorithm 1, where the input is the target position $\boldsymbol{P}_{\mathrm{goal}}$, and the output is the desired instructions $\boldsymbol{U}_{\mathrm{d}} = [u_{\mathrm{d}}, r_{\mathrm{d}}]^{\mathrm{T}}$; this is transmitted to the motion control module in the closed loop. $[x_{\mathrm{c}}, y_{\mathrm{c}}, \phi_{\mathrm{c}}, u_{\mathrm{c}}, r_{\mathrm{c}}]^{\mathrm{T}}$ are the current states of the ROV in the coordinate plane, u_{Max} and u_{Min} are the maximum and the minimum of the surge velocity u, while r_{Max} and r_{Min} are the maximum and the minimum of the yaw angular velocity r; $a_{u,\mathrm{Max}}$ and $a_{r,\mathrm{Max}}$ are the maxima of

the surge acceleration and yaw angular acceleration, respectively. T is the predictive time of the DWA algorithm.

As shown in Algorithm 1, the detailed navigation process can be described as follows: First, at the horizontal navigation stage, the plane coordinates of the target position are defined as the desired position $\boldsymbol{P}_{\text{goal}}$. Then, a dynamic window is constructed, which represents a set of feasible velocity commands (including linear velocity u_t and angular velocity r_t), bounded by the physical constraints of the ROV and discretized with resolution δ_u and δ_r. Next, each set of velocity commands within the dynamic window is evaluated, and the set with the highest evaluation metric is selected as the current generated velocity command. Note that "Traj_Gen" means trajectory generation, and $\boldsymbol{U}_{\text{Vel}}$ is the velocity at each resolution, an intermediate variable for recording velocities. The evaluation metrics in the aforementioned steps are defined as follows:

$$H_{\text{e}}(u,r) = aW_{\text{ori}} + bW_{\text{dist}} + cW_{\text{vel}}, \tag{8}$$

where a, b, and c are positive constants. W_{ori}, W_{dist}, and W_{vel} are the sub-objective functions that can be defined as follows:

$$\begin{cases} W_{\text{ori}} = 180 - (\phi_{\text{pred}} - \dfrac{180}{\pi}) \angle (\boldsymbol{P}_{\text{goal}} - \boldsymbol{P}_{\text{pred}}), \\ W_{\text{dist}} = \text{Min}(D_{\text{obs}}, D_{\text{safe}}), \\ W_{\text{vel}} = |u_t|, \end{cases} \tag{9}$$

where $\boldsymbol{P}_{\text{pred}} = [x_{\text{pred}}, y_{\text{pred}}]^{\text{T}}$ and ϕ_{pred} are the location and the yaw angle predicted by the velocity command (u_t, r_t) with respect to Eq. (4). D_{obs} is the distance between the ROV and the nearest obstacle obtained by other units, and D_{safe} is the safe distance; $\text{Min}(\cdot)$ denotes the minimal value function.

When the ROV reaches the plane coordinates of the target position, it initiates the vertical navigation, whereby it ascends or descends vertically at a constant velocity v until it reaches the target position.

Remark 2 In Algorithm 1, all the feasible velocity commands in the dynamic window are evaluated by the evaluation metric provided in Eq. (8) as follows: First, for each velocity command (u_t, r_t), we calculate the predicted position $\boldsymbol{P}_{\text{pred}}$ and the heading angle ϕ_{pred} in the predictive time T with respect to Eq. (4). Second, the evaluation metrics are calculated based on $\boldsymbol{P}_{\text{pred}}$ and ϕ_{pred} with respect to Eq. (9). That is, W_{ori} is used to evaluate the approach of the ROV's predicted yaw angle towards the target orientation; if the predicted heading angle ϕ_{pred} is closer to the target orientation, the greater is the weight of this item. W_{dist} is used to evaluate the risk of collision; if the ROV is farther away from the obstacle, the weight of this item is greater. Note that W_{dist} is constrained by a boundary to prevent it from occupying an excessively large proportion of the total weight when there is no collision. The final item W_{vel} is used to evaluate the efficiency of motion; that is, the whole motion is expected to be achieved in the shortest time. Therefore, by selecting the velocity command with the greatest weight as the output $\boldsymbol{U}_{\text{d}}$, the local trajectory guided by it can reach $\boldsymbol{P}_{\text{goal}}$ with obstacle avoidance and high efficiency.

Algorithm 1 DWA-based navigation

Input $\boldsymbol{P}_{\text{goal}}$ and $\boldsymbol{X}_{\text{R}}$; **Output** $\boldsymbol{U}_{\text{d}}$;
while $\boldsymbol{X}_{\text{R}} \neq \boldsymbol{P}_{\text{goal}}$ **do**
  $\underline{u} = \text{Min}\{u_{\text{Max}}, u_{\text{c}} + a_{u,\text{Max}}T\}$;
  $\bar{u} = \text{Max}\{0, u_{\text{c}} - a_{u,\text{Max}}T\}$;
  $\underline{r} = \text{Min}\{r_{\text{Max}}, r_{\text{c}} + a_{r,\text{Max}}T\}$;
  $\bar{r} = \text{Max}\{r_{\text{Min}}, r_{\text{c}} - a_{r,\text{Max}}T\}$;
  $i \leftarrow 0$; $j \leftarrow 0$;
  for $u_t = \underline{u} : \delta_u : \bar{u}$ **do**
    for $r_t = \underline{r} : \delta_r : \bar{r}$ **do**
      $[\boldsymbol{P}_{\text{pred}}, \phi_{\text{pred}}] \leftarrow \text{Traj\_Gen}(u_t, r_t)$;
      $\boldsymbol{H}_{\text{eva}}(i,j) \leftarrow H_{\text{e}}$;
      $\boldsymbol{U}_{\text{Vel}}(1:2,i,j) \leftarrow (u_t, r_t)$;
      $j \leftarrow j + 1$;
    end
    $i \leftarrow i + 1$;
  end
  $\boldsymbol{U}_{\text{d}} \leftarrow \boldsymbol{U}_{\text{Vel}}(1:2, \text{Find\_Max\_Metric}(\boldsymbol{H}_{\text{eva}}))$;
end

3.2.3 Design of the ROV motion control module

The ROV motion control module is designed to calculate the control input $\boldsymbol{T}_{\text{t}}$, which can drive the ROV to track the desired velocity commands u_{d}, w_{d}, and r_{d}. Note that this module also provides an open data interface for researchers to verify and improve the user-defined control algorithms. Here, an SMC algorithm is applied to illustrate how the motion control module works.

We define the desired velocity commands as $\boldsymbol{V}_{\text{D}} = [u_{\text{d}}, 0, w_{\text{d}}, r_{\text{d}}]^{\text{T}}$, where the desired sway

velocity v_{d} is set as zero. The control objective is to drive the motion of the ROV to achieve the desired velocity commands $\boldsymbol{V}_{\mathrm{D}}$ obtained in the navigation module. The SMC-based controller can be designed as follows:

First, we define the sliding surface as

$$\boldsymbol{S} = \boldsymbol{\Lambda}\tilde{\boldsymbol{V}}_{\mathrm{R}} + \int \tilde{\boldsymbol{V}}_{\mathrm{R}}\,\mathrm{d}t, \tag{10}$$

where $\tilde{\boldsymbol{V}}_{\mathrm{R}} = \boldsymbol{V}_{\mathrm{D}} - \boldsymbol{V}_{\mathrm{R}}$, and $\boldsymbol{\Lambda}$ is the positive constant matrix.

Then, taking the derivative of Eq. (10) and substituting it in Eq. (4), one can obtain

$$\dot{\boldsymbol{S}} = \boldsymbol{\Lambda}\dot{\tilde{\boldsymbol{V}}}_{\mathrm{R}} + \tilde{\boldsymbol{V}}_{\mathrm{R}} = \boldsymbol{\Lambda}(\boldsymbol{A}_{\mathrm{t}}\boldsymbol{V}_{\mathrm{R}} + \boldsymbol{C}_{\mathrm{t}}^{-1}\boldsymbol{T}_{\mathrm{t}} + \boldsymbol{D}_{\mathrm{t}} - \dot{\boldsymbol{V}}_{\mathrm{D}}) + \tilde{\boldsymbol{V}}_{\mathrm{R}}. \tag{11}$$

Thus, $\boldsymbol{T}_{\mathrm{t}}$ can be designed as

$$\begin{aligned}\boldsymbol{T}_{\mathrm{t}} = &-\boldsymbol{C}_{\mathrm{t}}\boldsymbol{K}\boldsymbol{S} - \boldsymbol{C}_{\mathrm{t}}\boldsymbol{\Gamma}\mathrm{sign}(\boldsymbol{S}) \\ &+ \boldsymbol{C}_{\mathrm{t}}(\dot{\boldsymbol{V}}_{\mathrm{D}} - \boldsymbol{A}_{\mathrm{t}}\boldsymbol{V}_{\mathrm{R}} - \boldsymbol{\Lambda}^{-1}\tilde{\boldsymbol{V}}_{\mathrm{R}}),\end{aligned} \tag{12}$$

where $\boldsymbol{K} = \mathrm{diag}(k_1, k_2, k_3, k_4)$, $\boldsymbol{\Gamma} = \mathrm{diag}(\eta_1, \eta_2, \eta_3, \eta_4)$, $k_m > 0$, and $\eta_m > d_{\max}$ for $m =$ 1, 2, 3, and 4, and $d_{\max} = \max\{|d_u|, |d_v|, |d_w|, |d_r|\}$.

Theorem 1 Considering the dynamics of the ROV in Eq. (4) with the bounded disturbance D_t and the SMC-based controller, shown in Eq. (12), with $k_m > 0$ and $\eta_m > d_{\max}$ for $m = 1$, 2, 3, and 4, all signals of Eq. (11) can be guaranteed to be bounded and the system in Eq. (11) is stable, which implies that the velocity error $\|\tilde{\boldsymbol{V}}_{\mathrm{R}}\| \to 0$ as time $t \to \infty$; the control objective can be achieved.

3.3 Design of the teleoperated manipulation unit

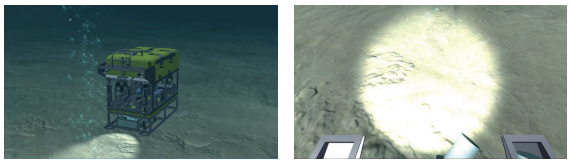
After navigation to the target position, manipulation of the ROV is usually achieved by human-interactive teleoperation of the slave manipulator installed on the ROV. To be specific, the operation mode starts, with the teleoperated manipulation unit receiving the commands given by the human operator and driving the manipulator to execute the tasks. The teleoperated manipulation unit mainly consists of the teleoperation system modeling and the teleoperation control framework (in Section 1 in the supplementary materials).

3.4 Design of the environment perception and rebuilding unit

The environment perception and rebuilding unit is mainly used to provide the human operator with the force and visual feedback of the underwater operation tasks on the slave side.

The vision-based perception and rebuilding unit can transmit the target position and joint angles of the slave manipulator to the teleoperated manipulation unit. Then, using the Omni driver, the asymmetric mapping algorithm, and the teleoperation control algorithm in the teleoperated manipulation unit, the ROV's real working environment can be accurately created in the virtual scene.

To better show the ROV's condition during the underwater operation tasks, various perspectives (e.g., the first- and the third-person perspectives) are designed in the visual perception and rebuilding unit, as shown in Fig. 4. In detail, the camera plug-in in the Unity software is used to select the specific view of the underwater operation tasks, where the view image is rendered and loaded by the RAW file; the procedure is the same as the underwater terrain discussed in Section 3.1.2. Thus, the viewed image can be displayed on the screen.



(a) (b)

Fig. 4 Multiple perspectives of a virtual scene: (a) third-person perspective; (b) first-person perspective

Remark 3 Based on the environment perception and rebuilding unit, the human operator can sense the interaction force between the slave manipulator and the working environment, which is achieved by the environmental force feedback on the master manipulator. Moreover, the human operator can immersively observe the underwater operation tasks on the screen, thereby improving the telepresence.

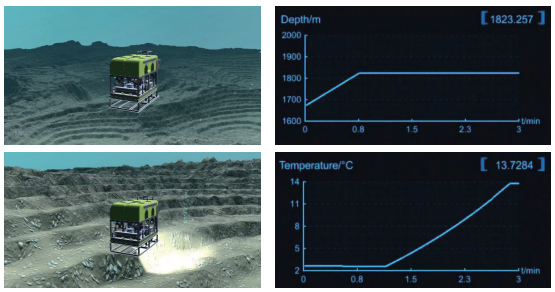
3.5 Design of the in-situ data-monitoring unit

The in-situ data-monitoring unit mainly includes the monitoring interface of in-situ data (e.g., temperature, salinity, and depth), the ROV state data, and the manipulator state data. By the

LineChart plug-in in the Unity software, the corresponding curves of in-situ data can be dynamically drawn in real time.

In detail, the monitoring interface of the in-situ data and the ROV state data displays the data from the relationship among the temperature, salinity, and depth of the ROV; that is, when the ROV moves in the virtual scene, the monitoring interface can display the in-situ data and the ROV state data at the current location of the ROV in real time.

For example, Fig. 5a depicts the motion of an ROV near a hydrothermal vent. When the ROV moves closer to the hydrothermal vent, the depth where it is located continues to increase, and the temperature continues to rise, which can be obtained from the in-situ data monitoring in Fig. 5b. Thus, this example verifies that the monitoring interface can accurately simulate the in-situ data of the working environment in which the ROV is located.



(a) (b)

Fig. 5 ROV motion near a hydrothermal vent: (a) ROV motion simulation; (b) in-situ data monitoring

Similarly, the manipulator state data-monitoring interface can display the joint angles of the slave manipulator in real time, which provides the human operator with real-time information about the states of the slave manipulator when executing the underwater operation tasks.

Remark 4 Along with the ROV navigation and motion control unit and the teleoperated manipulation unit, the proposed digital simulation platform is equipped with a DWA-based local navigation algorithm and an SMC controller for the navigation and motion control of the ROV. It is also equipped with the modified wave-variable architecture and an RBFNN-based slave manipulator controller for stability of the teleoperation system and precise position tracking during the ROV operation mode. Notably, these applied algorithms are served as examples in the platform. Moreover, researchers can design their own algorithms in these units with the designed data interface, which helps test and improve the algorithms in the virtual training mechanism, and finally implement these algorithms in the real underwater test.

4 Experiments on the full operation process of a work-class ROV: user study

In this section, two specific underwater operation tasks, including underwater sediment sampling and pipeline docking, are selected. Due to the huge difficulty, harsh and limited conditions, and expensive equipment costs of real underwater tests, the current conditions in our laboratories cannot frequently support the teleoperation tasks of ROVs in the real underwater environment. Instead, to obtain accurate virtual training as much as possible, we use recorded in-situ data and the real video signal obtained from the previous real underwater tests to accurately create the two specific underwater tasks, namely, sediment sampling and pipeline docking (described as Mechanism 1). Moreover, considering that the operation mode with teleoperated manipulation has a greater impact on the human operator, a user study with several metrics is designed to evaluate the accuracy of the proposed digital simulation platform and the effectiveness of training the human operator before the real underwater test.

4.1 Experimental setup

The experimental platform is shown in Fig. 6, where the human operator operates the master manipulator and commands the slave manipulator to accomplish the underwater operation tasks. In the meantime, the master manipulator provides the human operator with the force information via the Omni driver, and the screen shows the visual information of the underwater environment, which includes the virtual scene and the in-situ data. In particular, Fig. 7 shows an underwater scenario in the digital simulation platform, which is generated based on the underwater in-situ data of a certain area in Hainan Province, China. In Fig. 7, the orange point is the initial position of the ROV, and the red points are the sediment sampling point and the pipeline docking point.

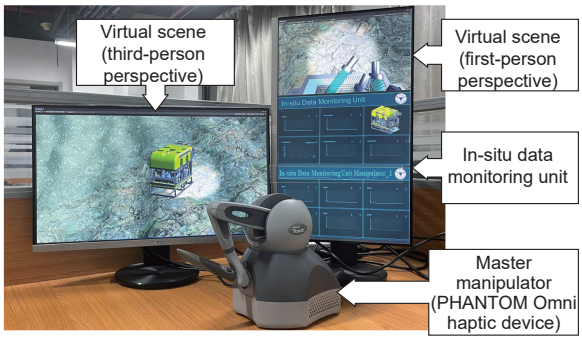


Fig. 6 The experimental platform

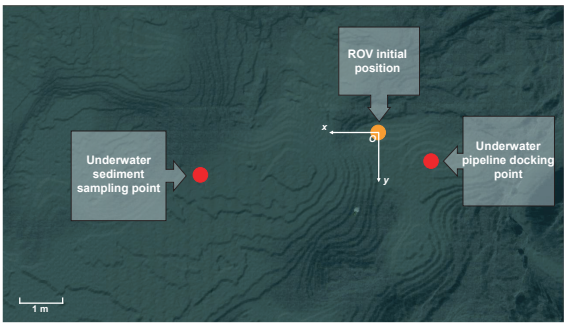


Fig. 7 Positions of the ROV in the virtual scene during the operation of the two tasks

In the ROV navigation and motion control unit, the parameters in the DWA-based local navigation algorithm are set as $a = 0.1$, $b = 5$, and $c = 1$, and the predictive time $T = 0.2$ s. The constant velocity of the ROV during the vertical navigation process is set as 1 m/s. The parameters of the SMC controller are set as $\boldsymbol{\Lambda} = \mathrm{diag}(0.1, 0.1, 0.1, 1)$, $\boldsymbol{K} = \mathrm{diag}(2.5, 2.5, 2.5, 2)$, and $\boldsymbol{\Gamma} = \mathrm{diag}(0.1, 0.1, 0.1, 1)$. In the teleoperated manipulation unit, the parameters of the wave-variable architecture are selected as $b = 10$, $\lambda = 0.5$, and $\tau = 0.05$. The desired matrices of the master trajectory generator are set as $\boldsymbol{M}_{\mathrm{md}} = \mathrm{diag}(4, 4, 2.5)$, $\boldsymbol{G}_{\mathrm{md}} = \mathrm{diag}(0, 0, 0)$, $\boldsymbol{G}_{\mathrm{md}} = \mathrm{diag}(0, 0, 20)$, and $\boldsymbol{v}_{\mathrm{m}} = \mathrm{diag}(1, 1, 1)$. As for selection of the RBFNN-based slave controller, $\boldsymbol{\sigma}_{\mathrm{s}} = \mathrm{diag}(5, 5, 5)$, $\boldsymbol{\kappa}_{\mathrm{s}} = \mathrm{diag}(1.5, 1.5, 1.5)$, $\zeta = 0.01$, and $\boldsymbol{\Gamma}_{\mathrm{s}} = \mathrm{diag}(2.5, 2.5, 2.5)$.

Moreover, to evaluate whether the human operator can efficiently complete the tasks in the digital simulation platform during the operation mode, six volunteers (aged 20–25 years, comprising three men and three women, with normal or corrected vision, no physical disability, and no virtual environment training experience earlier) are selected to carry out the user study. In the user study, five conditions have been designed to verify the effectiveness of each unit in the digital simulation platform to improve the telepresence, listed as follows:

C1: SP, without EFF, without AMA;

C2: VP, without EFF, with AMA;

C3: SP, with EFF, with AMA;

C4: VP, with EFF, without AMA;

C5: VP, with EFF, with AMA.

Here, SP represents single perspective, VP denotes various perspectives defined in Section 3.4, EFF denotes the environmental force feedback defined in Section 3.4, and AMA is the asymmetric mapping algorithm defined in Section 3.3.

Additionally, three metrics are designed to evaluate how good the virtual training of underwater operation tasks is, listed as follows:

1. The success ratio $p_{\mathrm{r}} = \frac{N}{M}$, where N is the number of successful times, and M is the total number of times, fixed as $M = 15$. In detail, the successful operation of the underwater sediment sampling task means that the mud picker is filled with the sediment and taken back to the collection box, while the successful operation of the underwater pipeline docking task means that the pipeline is docked via two manipulators.

2. The completion time p_{t}, which is the period to successfully complete a task once during the operation mode (e.g., the sediment sampling or pipeline docking), from the beginning to the successful operation of the task, excluding the rest time.

3. The work load $p_{\mathrm{o}} = \frac{1}{n}\sum_{i=1}^{n} I_i$, which is obtained by the NASA-task load index (NASA-TLX) (Hart and Staveland, 1988) with $n = 6$, where I_i contains the mental demand (I_1), physical demand (I_2), temporal demand (I_3), performance (I_4), effort (I_5), and frustration (I_6).

After providing the definitions of the conditions and the metrics, two setups of user studies are listed as follows:

Set 1. We select one volunteer randomly to implement the training process 0, 2, 4, 6, and 8 times under condition C5. After the training, this volunteer is required to implement the tasks one more time, with the completion time p_{t} measured.

Set 2. We select five other volunteers who have experienced training 20 times under conditions C1–C5. After training, each of them is required to implement the tasks another 15 times, with the average scores of the three metrics recorded.

4.2 Case 1: underwater sediment sampling

The underwater sediment sampling experiment is designed to verify the effectiveness of the digital simulation platform in the teleoperation of a single slave manipulator. For good presentation of the experiment, Set 2 with condition C5 is selected to show the experimental result.

The whole sampling process is shown in Fig. 8. First, assisted by the ROV navigation and motion control unit and the environment perception and rebuilding unit, the ROV moves from the initial position to the underwater sediment sampling point, as shown in Fig. 8a. Second, as shown in Fig. 8b, when the ROV arrives at the sampling point, its posture is adjusted, and the collection box (equipped with the mud pickers) is stuck out; then, the slave manipulator enters the operation mode, and is ready to start sampling. Subsequently, using the teleoperated manipulation unit, the master manipulator is directed to send the position signals of its end effector to the slave manipulator in the digital simulation platform via the transmission control protocol/Internet protocol (TCP/IP) communication mode. Finally, the slave manipulator is commanded to accomplish the tasks, which include the reaching of mud pickers' initial position to grasp the mud picker, the reaching of the sampling point to do the sediment sampling, the retraction of the mud picker, and the return to the initial position.

Remark 5 Fig. 8 includes the screenshots of the real video signal during the sediment sampling task. It can be concluded that the in-situ data-driven digital simulation platform can accurately simulate the actual tasks, which can be used for the virtual training of a human operator before the real underwater test. Specially, the real video signal only provides a small range and single perspective with dark view, and the sediment occasionally blocks the camera's view. Thus, the proposed platform provides a large range and various perspectives with bright view, in addition to removing the turbidity caused by the sediment, ensures the visibility of the operation, and further improves the telepresence for the human operator.

Figs. 8a and 8b show the navigation mode, wherein the ROV moves from its initial position to the underwater sediment sampling point using the DWA-based local navigation algorithm and the SMC

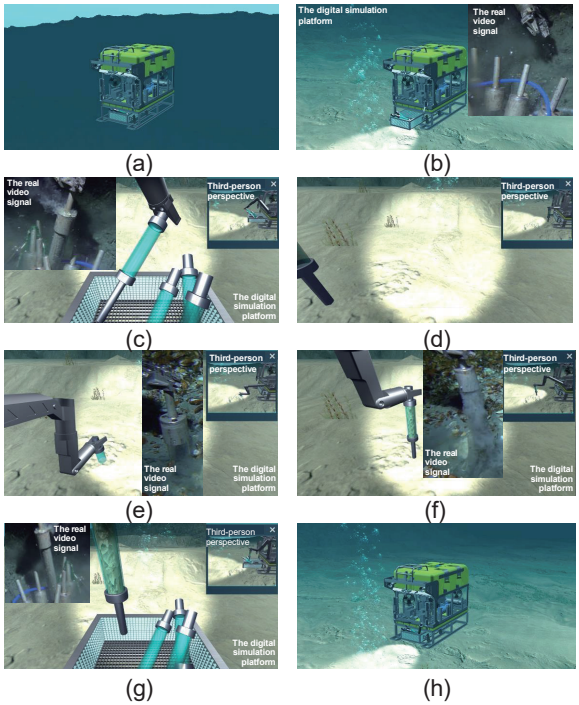


Fig. 8 The underwater sediment sampling process of ROV in the virtual scene: (a) ROV navigation and motion process; (b) ROV posture adjustment; (c) mud picker grasp; (d) ready to sample; (e) sediment sampling; (f) end of sampling; (g) retraction of the mud picker; (h) manipulator returns to the initial position

controller. Subsequently, during the operation mode, in the sampling preparation period (Figs. 8c–8e), the slave manipulator grasps the mud picker and inserts it into the sampling point under the teleoperation control framework; Figs. 8f–8h show the sampling operation period, during which the slave manipulator finishes the sampling, retracts the mud picker to the collection box, and returns to the initial position.

As shown in Fig. 9a, with the real-time planning of the DWA-based local navigation algorithm and the good control performance of the SMC controller, the ROV initially moves toward the target point (i.e., the underwater sediment sampling point) in the horizontal direction until it approaches the place above the target point. Subsequently, the ROV begins to descend vertically at a constant speed until it reaches the target point. To verify the effectiveness of the teleoperation control framework, the end effector's displacement of the master and slave manipulators is selected, shown in Fig. 9b. The two curves shown in Fig. 9b have the same shape, direction, and proportional size; this means that the

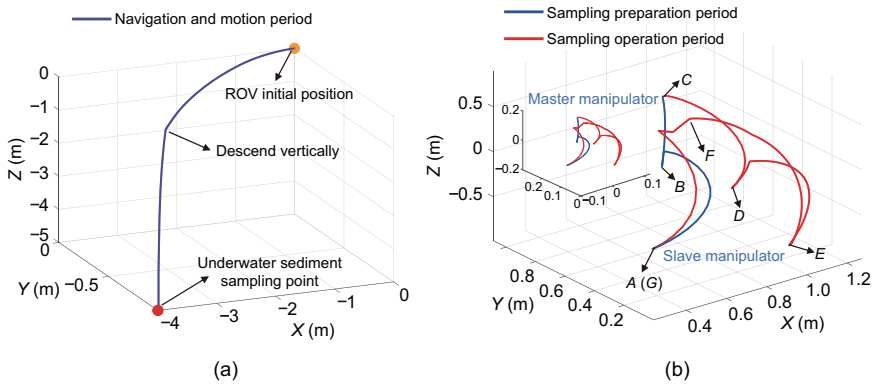


Fig. 9 The motion during the full operation process of the work-class ROV and its manipulator for the underwater sediment sampling task: (a) the planned practical trajectory of the work-class ROV; (b) the end effector's displacement of the teleoperated master and slave manipulators

RBFNN-based slave controller and workspace-based asymmetric mapping algorithm under the modified wave-variable architecture can achieve good accuracy during the teleoperation process. In Fig. 9b, the trajectory formed by A–C is the sampling preparation period, corresponding to Figs. 8c–8e. Subsequently, the trajectory formed by C–G represents the sampling operation period, corresponding to Figs. 8f–8h.

The results in Fig. 10 illustrate that the time delay of the underwater sediment sampling experiment is about 0.5 s, and the force feedback and the real environmental force are quite similar in the shape and value of the curves, which shows that the wave-variable architecture and the environmental force perception and feedback method can ensure great force feedback accuracy in the process of underwater sediment sampling. In detail, in the process of grasping the mud picker with the manipulator, there always exists an environmental force on the X and Y axes. Particularly, when the manipulator grasps the mud picker from the collection box from A to B in Fig. 9b, the environmental force on the Z axis gradually increases; when the manipulator grasps the mud picker and leaves the collection box, the environmental force on the Z axis gradually decreases; subsequently, when the manipulator descends vertically from D to E for sampling the sediment, the environmental force on the Z axis starts to increase again; lastly, when finishing sampling and pulling out the mud picker, the environmental force on the Z axis decreases. Based on the force perception and feedback method in Section 3.4, the human operator can obtain accurate force feedback via the Omni driver of the master manipulator.

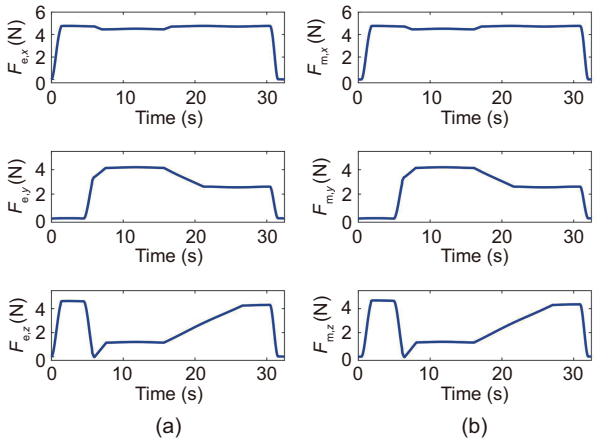


Fig. 10 Environmental force in the underwater sediment sampling process: (a) real environmental force F_e on the slave side; (b) force feedback F_m on the master side

4.3 Case 2: underwater pipeline docking

The experiment of underwater pipeline docking is designed to verify the effectiveness of the digital simulation platform in the teleoperation of multiple slave manipulators. For a good presentation of the experiment, Set 2 with condition C5 is selected to show the experimental result (in Section 2 in the supplementary materials).

4.4 User study analysis

Considering that the human operator's operational experience in the operation mode is important to the underwater tasks, a user study on the operation mode with teleoperated manipulation is

carried out. The results of the user study for these six volunteers are analyzed in this section.

The user study Set 1 is designed to verify the training effects of the proposed digital simulation platform for an inexperienced human operator, to increase the experience for implementing the tasks after training. The results of user study Set 1 are shown in Fig. 11, wherein the conclusions of the sediment sampling and pipeline docking tasks are basically the same. Specifically, the time required for a human operator without virtual training is 95.63 s and 121.58 s to complete the sampling and docking tasks, respectively. After eight rounds of training, the completion time decreases by 57.4% and 58.4%, respectively; that is, with the increase of training rounds, the human operator becomes more familiar with the process of sampling or docking, and the completion time can be significantly reduced. Thus, with regard to user study Set 1, it can be verified that the proposed digital simulation platform has good training effects for a human operator who is a green hand on the underwater operation tasks.

The user study Set 2 is designed to verify the telepresence improvement of the proposed platform; five volunteers are chosen, who are well trained with 20 rounds of training and who can thus be regarded as experts in later experiments. The results of user study Set 2 are shown in Fig. 12. C1 has only the SP without considering the EFF and AMA, so it has the lowest success ratio and the highest completion time and workload. Since C2, C3, and C5 add the proposed AMA, their completion time and the workloads are significantly decreased compared to those of C4. Particularly, the main difference between C4 and C5 is the presence of AMA. Thus, C5 shows a significant improvement in all three metrics when compared with C4; this is because the proposed AMA matches the workspace of the master and slave manipulators; as a result, the human operator can teleoperate to complete the sampling or docking tasks by commanding the master manipulator over a shorter distance, which reduces the workload and improves the operation efficiency. Additionally, since C5 adds the VP, C5 shows significant improvement in all three metrics when compared with C3. This is because the VP can assist the human operator to observe the operation process from different perspectives, which avoids the failure caused by the parallax effect of the camera. Since C5 adds the EFF, C5 also has a significant improvement in all three metrics when compared with C2; this is because the EFF can accurately provide the human operator with the force information from the interaction between the slave manipulator and the remote environment, which helps the sampling or docking tasks. Moreover, when comparing C2, C3, and C5 horizontally, the VP can greatly reduce the completion time, and the EFF can greatly reduce the workload. Thus, with regard to the user study Set 2, the proposed digital simulation platform (under condition C5) can provide a human operator with the optimal visual and force-assisted telepresence, and obtain the highest success ratio, the lowest completion time, and the lowest work load.

Particularly, Fig. 13 shows the results of the work load p_o in Set 2 obtained from the NASA-TLX. In detail, Fig. 13a indicates that the physical demand (I_2) in C4 during the sediment sampling task is 36.37% higher than that in C5, since C4 does not

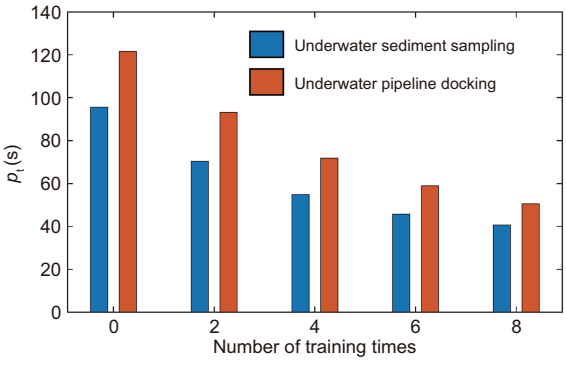


Fig. 11 Results of the user study Set 1

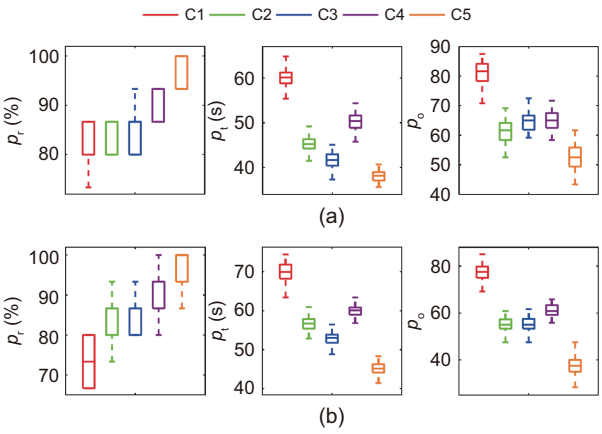


Fig. 12 Results of the user study Set 2: (a) underwater sediment sampling; (b) underwater pipeline docking

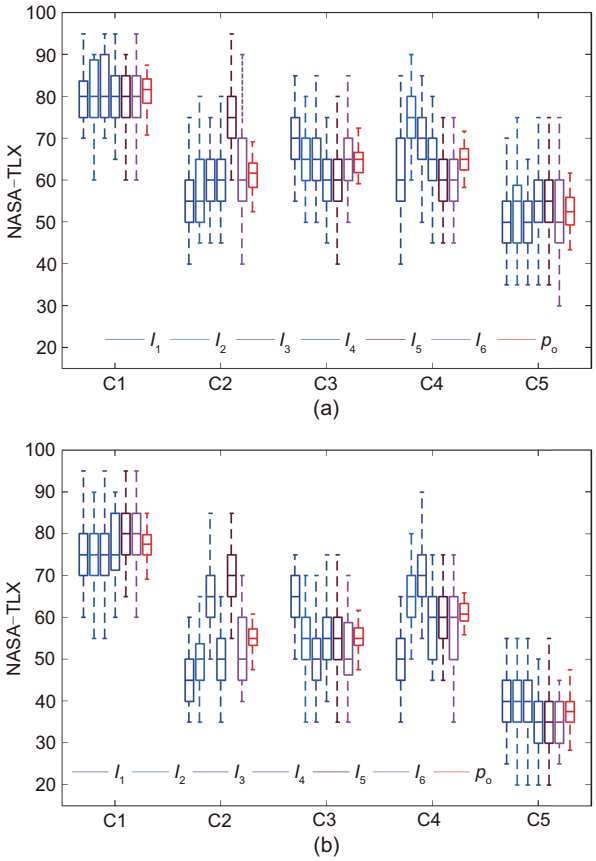


Fig. 13 Results of the NASA-TLX in Set 2: (a) underwater sediment sampling; (b) underwater pipeline docking

apply the proposed AMA, and the slave manipulator has a larger workspace compared with the master manipulator. If the workspace is not well matched, the human operator has to operate the master manipulator over a longer distance, which finally leads to higher I_2. When comparing C3 and C5, the mental demand (I_1) in C3 is 40% higher than that in C5; this is because the human operator can only observe the slave manipulator in a single perspective; this easily causes the parallax and renders it difficult to judge the position of the mud picker, which increases the I_1. Additionally, the effort (I_5) in C2 is 36.37% higher than that in C5; this is because C2 does not consider the EFF, so the operator needs to operate the slave manipulator more carefully to avoid excessive force or collision during the interaction between the manipulator and the working environment.

It can be seen from Fig. 13b that the results of the pipeline docking task are similar to that of the sediment sampling task. Specifically, when compared with C5, I_2 in C4 without the proposed AMA is 62.5% higher, I_1 in C3 without the SP is 62.5% higher, and I_5 in C2 without the EFF is 100% higher; that is, with the increase of the task complexity, the work load p_o of the human operator increases greatly. Thus, through the user study Set 2, it can be proved that the proposed digital simulation platform (C5) effectively reduces the work load of the human operator, and the more complicated the tasks, the more significant the reduction effect.

Remark 6 Based on the user study carried out to evaluate the virtual training process for the underwater sediment sampling and pipeline docking tasks, using the metrics listed in Section 4.1, the designed digital simulation platform is found to have good applicability. With regard to the comparison with the real video signal and the result of the user study, the visual information and the in-situ data are accurately created and displayed on the screen, while the force feedback is provided via the Omni driver. Thus, by using the proposed digital simulation platform, the human operator can effectively be trained to implement the underwater operation tasks with immersive telepresence.

5 Conclusions

This paper presents a digital simulation platform for the full operation process of work-class ROVs, which achieves low-cost and immersive underwater teleoperation training with quality-improved telepresence using the recorded in-situ data. To make this digital simulation platform more powerful, a data interface is designed for the recorded in-situ data virtual training and for helping test and improve the related algorithms. Several algorithms are implemented to ensure the integrity of the proposed platform, which shows the effectiveness of the simulation for navigation, motion, and teleoperated manipulation of work-class ROVs. To further evaluate the superiority and accuracy of the digital simulation platform for immersive underwater teleoperation training, a user study on the ROV operation mode is carried out with the specific underwater sediment sampling and pipeline docking tasks, which shows that the proposed digital simulation platform can provide the human operator with immersive visual and force-assisted telepresence, and thus helps him/her gain experience through training before the real underwater test.

In the future, we will investigate advanced navigation and control algorithms, in addition to topics related to more realistic ROV hydrodynamic modeling and underwater physical parameters, within the designed digital simulation platform.

Contributors

Fanghao HUANG designed the research. Fanghao HUANG, Xiao YANG, and Xuanlin CHEN processed the data and drafted the paper. Deqing MEI and Zheng CHEN helped organize the paper. All the authors revised and finalized the paper.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Cárdenas EF, Dutra MS, 2016. An augmented reality application to assist teleoperation of underwater manipulators. *IEEE Lat Am Trans*, 14(2):863-869. https://doi.org/10.1109/TLA.2016.7437233

Chen Z, Huang FH, Sun WC, et al., 2020. RBF-neural-network-based adaptive robust control for nonlinear bilateral teleoperation manipulators with uncertainty and time delay. *IEEE/ASME Trans Mech*, 25(2):906-918. https://doi.org/10.1109/TMECH.2019.2962081

Chen Z, Helian BB, Zhou Y, et al., 2023. An integrated trajectory planning and motion control strategy of a variable rotational speed pump-controlled electro-hydraulic actuator. *IEEE/ASME Trans Mech*, 28(1):588-597. https://doi.org/10.1109/TMECH.2022.3209873

Chen Z, Zhou SZ, Shen C, et al., 2024. Observer-based adaptive robust precision motion control of a multi-joint hydraulic manipulator. *IEEE/CAA J Autom Sin*, 11(5):1213-1226. https://doi.org/10.1109/JAS.2024.124209

Ferreira A, Mavroidis C, 2006. Virtual reality and haptics for nanorobotics. *IEEE Rob Autom Mag*, 13(3):78-92. https://doi.org/10.1109/MRA.2006.1678142

Forbrigger S, Pan YJ, 2018. Improving haptic transparency for uncertain virtual environments using adaptive control and gain-scheduled prediction. *IEEE Trans Haptics*, 11(4):543-554. https://doi.org/10.1109/TOH.2018.2839182

Hart SG, Staveland LE, 1988. Development of NASA-TLX (task load index): results of empirical and theoretical research. *Adv Psychol*, 52:139-183. https://doi.org/10.1016/S0166-4115(08)62386-9

Hokayem PF, Spong MW, 2006. Bilateral teleoperation: an historical survey. *Automatica*, 42(12):2035-2057. https://doi.org/10.1016/j.automatica.2006.06.027

Huang FH, Chen XL, Xu Y, et al., 2023. Immersive virtual simulation system design for the guidance, navigation and control of unmanned surface vehicles. *Ocean Eng*, 281:114884. https://doi.org/10.1016/j.oceaneng.2023.114884

Khadhraoui A, Beji L, Otmane S, et al., 2016. Stabilizing control and human scale simulation of a submarine ROV navigation. *Ocean Eng*, 114:66-78. https://doi.org/10.1016/j.oceaneng.2015.12.054

Khatib O, Yeh X, Brantner G, et al., 2016. Ocean one: a robotic avatar for oceanic discovery. *IEEE Rob Autom Mag*, 23(4):20-29. https://doi.org/10.1109/MRA.2016.2613281

Kinsey JC, Yang QJ, Howland JC, 2014. Nonlinear dynamic model-based state estimators for underwater navigation of remotely operated vehicles. *IEEE Trans Contr Syst Technol*, 22(5):1845-1854. https://doi.org/10.1109/TCST.2013.2293958

Li HJ, Huang FH, Chen Z, 2022. Virtual-reality-based online simulator design with a virtual simulation system for the docking of unmanned underwater vehicle. *Ocean Eng*, 266:112780. https://doi.org/10.1016/j.oceaneng.2022.112780

Lin Q, Kuo CG, 2001. On applying virtual reality to underwater robot tele-operation and pilot training. *Int J Virtual Reality*, 5(1):71-91. https://doi.org/10.20870/IJVR.2001.5.1.2670

Liu Y, Shen Q, Ma DL, et al., 2017. Steering control for underwater gliders. *Front Inform Technol Electron Eng*, 18(7):898-914. https://doi.org/10.1631/FITEE.1601735

Long CQ, Hu MJ, Qin XH, et al., 2022. Hierarchical trajectory tracking control for ROVs subject to disturbances and parametric uncertainties. *Ocean Eng*, 266:112733. https://doi.org/10.1016/j.oceaneng.2022.112733

Lu Y, Chen XY, Wu ZX, et al., 2022. A novel robotic visual perception framework for underwater operation. *Front Inform Technol Electron Eng*, 23(11):1602-1619. https://doi.org/10.1631/FITEE.2100366

Manzanilla A, Reyes S, Garcia M, et al., 2019. Autonomous navigation for unmanned underwater vehicles: real-time experiments using computer vision. *IEEE Rob Autom Lett*, 4(2):1351-1356. https://doi.org/10.1109/LRA.2019.2895272

Reichherzer C, Cunningham A, Walsh J, et al., 2018. Narrative and spatial memory for jury viewings in a reconstructed virtual environment. *IEEE Trans Vis Comput Graph*, 24(11):2917-2926. https://doi.org/10.1109/TVCG.2018.2868569

Sivčev S, Coleman J, Omerdić E, et al., 2018. Underwater manipulators: a review. *Ocean Eng*, 163:431-450. https://doi.org/10.1016/j.oceaneng.2018.06.018

Sun WC, Yuan YQ, 2023. Passivity based hierarchical multi-task tracking control for redundant manipulators with uncertainties. *Automatica*, 155:111159. https://doi.org/10.1016/j.automatica.2023.111159

Sun YX, Gu R, Chen XH, et al., 2022. Efficient time-optimal path planning of AUV under the ocean currents based on graph and clustering strategy. *Ocean Eng*, 259:111907. https://doi.org/10.1016/j.oceaneng.2022.111907

Tani S, Ruscio F, Bresciani M, et al., 2023. Development and testing of a navigation solution for autonomous underwater vehicles based on stereo vision. *Ocean Eng*, 280:114757. https://doi.org/10.1016/j.oceaneng.2023.114757

Wang J, Tang YG, Chen CX, et al., 2020. Terrain matching localization for hybrid underwater vehicle in the Challenger Deep of the Mariana Trench. *Front Inform Technol Electron Eng*, 21(5):749-759. https://doi.org/10.1631/FITEE.1900556

Xia PX, Xu F, Song ZY, et al., 2023. Sensory augmentation for subsea robot teleoperation. *Comput Ind*, 145:103836. https://doi.org/10.1016/j.compind.2022.103836

Yuan YQ, Sun WC, 2023. An integrated kinematic calibration and dynamic identification method with only static measurements for serial robot. *IEEE/ASME Trans Mechatron*, 28(5):2762-2773. https://doi.org/10.1109/TMECH.2023.3241302

Zhang DW, Yang G, Khurshid RP, 2020. Haptic teleoperation of UAVs through control barrier functions. *IEEE Trans Haptic*, 13(1):109-115. https://doi.org/10.1109/TOH.2020.2966485

Zhang HP, Zhu DQ, Liu CX, et al., 2022. Tracking fault-tolerant control based on model predictive control for human occupied vehicle in three-dimensional underwater workspace. *Ocean Eng*, 249:110845. https://doi.org/10.1016/j.oceaneng.2022.110845

Zhang J, Li W, Yu JC, et al., 2017a. Development of a virtual platform for telepresence control of an underwater manipulator mounted on a submersible vehicle. *IEEE Trans Ind Electron*, 64(2):1716-1727. https://doi.org/10.1109/TIE.2016.2557309

Zhang J, Li W, Yu JC, et al., 2017b. Study of manipulator operations maneuvered by a ROV in virtual environments. *Ocean Eng*, 142:292-302. https://doi.org/10.1016/j.oceaneng.2017.07.008

Zhang JJ, Liu WD, Gao LE, et al., 2018. The master adaptive impedance control and slave adaptive neural network control in underwater manipulator uncertainty teleoperation. *Ocean Eng*, 165:465-479. https://doi.org/10.1016/j.oceaneng.2018.07.055

Zhao B, Skjetne R, Blanke M, et al., 2014. Particle filter for fault diagnosis and robust navigation of underwater robot. *IEEE Trans Contr Syst Technol*, 22(6):2399-2407. https://doi.org/10.1109/TCST.2014.2300815

List of supplementary materials

1 Design of the teleoperated manipulation unit

2 Case 2: underwater pipeline docking

Video S1 The digital simulation platform

Table S1 The motion range of master and slave manipulators

Fig. S1 Teleoperation control framework

Fig. S2 The underwater pipeline docking process of an ROV in the virtual scene

Fig. S3 The motion during operation of the full process of the work-class ROV and its manipulator for the underwater pipeline docking task

Fig. S4 The environmental force in the underwater pipeline docking process