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 unified shared control architecture for underwater vehicle–manipulator systems using task priority*#

Zhangpeng TU[1], Yuanchao ZHU[1,3], Xin WU[1], Canjun YANG‡[1,2]

[1]*State Key Laboratory of Fluid Power and Mechatronic Systems, Zhejiang University, Hangzhou 310027, China*

[2]*Ningbo Research Institute, Zhejiang University, Ningbo 315100, China*

[3]*Nanjing Research Institute of Electronics Technology, Nanjing 210007, China*

E-mail: zp.t@zju.edu.cn; ychzhu@zju.edu.cn; xinwu@zju.edu.cn; ycj@zju.edu.cn

Received June 2, 2024; Revision accepted Apr. 6, 2025; Crosschecked July 10, 2025

Abstract: It is challenging for underwater vehicle–manipulator systems (UVMSs) to operate autonomously in unstructured underwater environments. Relying solely on teleoperation for both underwater vehicle (UV) and underwater manipulator (UM) imposes a considerable cognitive and physical load on the operator. In this paper, we propose a unified shared control (USC) architecture for the UVMS, integrating divisible shared control (DSC) and interactive shared control (ISC) to alleviate the operator's workload. By applying task priority based on DSC, we divide the whole-body task into constraints, operation, and posture optimization subtasks. The robot autonomously avoids self-collisions and adjusts its posture according to the user's visual preferences. ISC incorporates haptic feedback to enhance human–robot collaboration, seamlessly integrating it into the operation task via a whole-body controller for the UVMS. Simulations and pool experiments are conducted to verify the feasibility of the method. Compared to manual control (MC), the proposed method reduces completion time by 17.50%, operator input length by 25.00%, and cognitive load by 35.53% in the simulations, with corresponding reductions of 22.73%, 40.00%, and 29.91% in the pool experiments. Subjective measurements demonstrate the reduction in operator workload with the proposed method.

Key words: Unified shared control; Underwater vehicle–manipulator system; Human–robot interaction; Task priority

https://doi.org/10.1631/FITEE.2400471 **CLC number:** TP242

1 Introduction

The proliferation of ocean exploration by humans has led to an increasing demand for underwater missions, including persistent underwater observation (Lin and Yang, 2020) and underwater manipulation (Petillot et al., 2019). Underwater tasks encompass a wide range of missions, such as scientific expeditions, marine rescue, and offshore industries (Ridao et al., 2015). Skilled divers are capable of undertaking some underwater interventions but face limitations in deep or prolonged underwater missions. Consequently, underwater vehicle–manipulator systems (UVMSs) have become essential for underwater operations (Sahoo et al., 2019). In the early stages of UVMS development, operators remotely control the vehicle and manipulator independently. However, full teleoperation requires operators to maintain sustained focus and patience for precise movements (Shim et al., 2010). This process

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (No. 52071292)

Electronic supplementary materials: The online version of this article (https://doi.org/10.1631/FITEE.2400471) contains supplementary materials, which are available to authorized authors

ORCID: Zhangpeng TU, https://orcid.org/0000-0001-6095-7363; Canjun YANG, https://orcid.org/0000-0002-3712-0538

places significant physical and cognitive load on operators, rendering it difficult to maintain during prolonged periods (Capocci et al., 2018).

In addressing this challenge, numerous UVMSs have been developed. Some are fully autonomous, such as GIRONA 500 (Ribas et al., 2015) and Aquanaut (Manley et al., 2018). Others, like Ocean One (Khatib et al., 2016) and DexROV (Di Lillo et al., 2021) operate in semi-autonomous mode. Notably, task priority control (Simetti et al., 2018) is widely used in UVMSs, and offers a hierarchical motion resolution framework for redundant robotic systems through the prioritization of control objectives. For instance, using task priority control, GIRONA 500 completed underwater tasks such as object recovery, valve turning (Maurelli et al., 2016), and pipeline inspection tasks (Cieślak and Ridao, 2018). This approach enables a UVMS to adapt to various operational scenarios featuring diverse control objectives and activation functions. Although underwater manipulation autonomy has been improved significantly (Zhang et al., 2017), fully automated control is still inadequate for complex and delicate operations in open and unstructured underwater environments. Underwater operations continue to require human intervention (Sivčev et al., 2018).

Human–machine hybrid augmented intelligence (HMAI) has the potential to significantly enhance the capabilities of complex systems (Wang et al., 2022). Human–robot shared control (SC) technology has garnered significant attention over the past decade. SC leverages human high-level cognition and decision-making while integrating robotic capabilities for low-level task execution (Yang et al., 2022b). SC methods are typically classified into divisible shared control (DSC) and interactive shared control (ISC), depending on the nature of human–machine collaboration (Samuel and Tee, 2019).

DSC divides the overall task into multiple subtasks, where humans are responsible for a specific subset of inputs (direction or several degrees of freedom (DoF) (Rakita et al., 2018)), while robots independently handle the remaining inputs (Nicolis et al., 2018). Task priority is a common variant of DSC methods in underwater robotics. Ocean One was developed with a force feedback teleoperation control method based on DSC (Brantner and Khatib, 2021). The robot controlled the constraint and optimization tasks, while the operator directly teleoperated the essential operational task. Another form of DSC is supervisory control, where the operator provides high-level task commands, and the robot performs the corresponding low-level actions, such as SAUVIM (Simpkins, 2014). Within the DSC framework, humans and robots collaborate as partners.

ISC considers the operator and the robot as distinct input sources, combining their control inputs via an arbitration function (linear, game theoretic, or data-driven strategies) (Yang et al., 2022b) to determine the relative contributions of each agent. In this case, humans and robots collaborate as partners. The DexROV Project (Birk et al., 2018) implemented an SC method where the robot was capable of learning actions from the teleoperation workspace. Its corrective actions increased as it approached the target. A human–machine target fusion method for the underwater humanoid robot has been proposed for controlling underwater manipulators (Yang et al., 2022a). The control weight between human and robot was dynamically adjusted based on the predicted accuracy of the target.

Our previous work (Zhu et al., 2023) proposed an SC method with haptic feedback, which determines control weights using fuzzy rules based on the operator's preferences and the confidence levels of both human and robotic agents.

In this paper, to further alleviate the operator's control workload, we introduce a unified shared control (USC) architecture that combines DSC and ISC for the UVMS based on task priority. This work contributes the following:

1. Proposal of a USC, in which humans and robots control specific subtasks independently by applying task priority derived from DSC. An ISC-based approach is integrated into the operational task to fully use the strengths of both human and robotic capabilities.

2. Development of methods to reduce the operator's burden, including an input fusion method that incorporates human intention and a motion distribution scheme based on distance, both of which use the same trajectory-related measurements during the operational task. Additionally, a haptic feedback guidance function is developed, and the posture of the UVMS is optimized based on operator inputs.

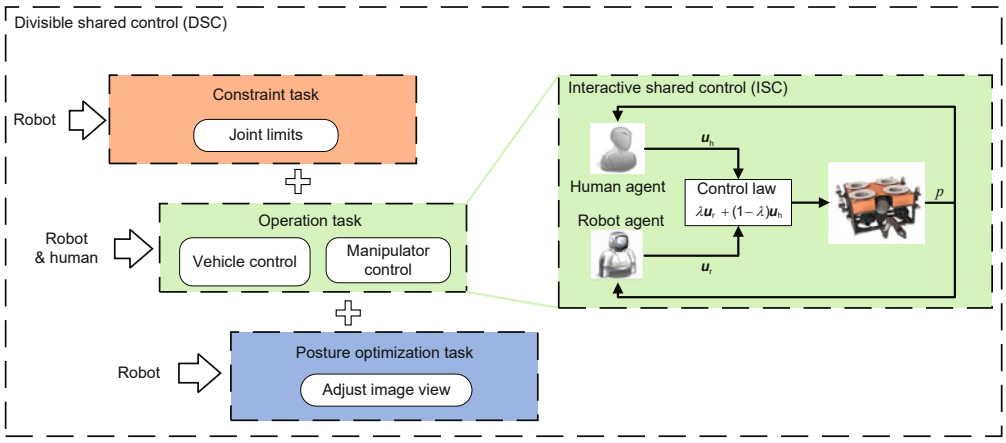

Fig. 1 Unified shared control (USC) architecture

2 Unified shared control architecture

In this section, we provide an overview of the USC architecture, as depicted in Fig. 1. The UVMS exhibits highly redundant DoF, enabling it to handle multiple tasks simultaneously. Following the DSC concept, the overall UVMS task is divided into several subtasks assigned to the human agent (HA) and the robot agent (RA). Considering UVMS safety and the dynamic response characteristics of the vehicle and manipulator, we decompose the whole-body task into a constraint task, an operation task, and a posture optimization task. The RA is responsible for both the constraint task and the posture optimization task. Through the implementation of an ISC method, the RA and HA combine their strengths to execute the operation task effectively.

Task priority is introduced as the DSC method to establish a hierarchical structure among these tasks. Task priority serves as a redundant control method, enabling secondary tasks to be executed in the null space of primary tasks. This ensures the completion of primary tasks while minimizing errors in lower-priority tasks.

To ensure robot safety and prevent collisions with surrounding objects, we prioritize the constraint task at the highest level. The hierarchical control focuses on the operation task. Given the UVMS's physical interaction with the external environment through its manipulator, we directly control the end-effector (EE) of the underwater manipulator (UM) using a whole-body controller. The vehicle autonomously follows the manipulator, which minimizes human input. The remaining controllable subspace is used to optimize the UVMS posture to track the target object. By adjusting the vehicle posture, we ensure that the target object remains within the desired perspective aligned with the HA's preferences in the camera's field of view. The task priority control law is as follows:

$$\boldsymbol{u} = \underbrace{(\boldsymbol{u}_{\mathrm{c}})}_{\text{constraint}} + \underbrace{(\boldsymbol{u}_{\mathrm{t|c}})}_{\text{operation}} + \underbrace{(\boldsymbol{u}_{\mathrm{p|t|c}})}_{\text{posture}}, \tag{1}$$

where $\boldsymbol{u}$ on the left-hand side of the equation represents the total input of the robot. The right-hand side of the equation contains the inputs for the three tasks.

In the operation task, we combine the HA input $\boldsymbol{u}_{\mathrm{h}}$ and the RA input $\boldsymbol{u}_{\mathrm{r}}$ using a control weight coefficient λ through a linear function:

$$\boldsymbol{u}_{\mathrm{s}} = \lambda \boldsymbol{u}_{\mathrm{r}} + (1-\lambda)\boldsymbol{u}_{\mathrm{h}}, \tag{2}$$

where $\boldsymbol{u}_{\mathrm{s}}$ denotes the SC input for the operation task. λ is a parameter that varies with the robot state, the historical trajectory, and the operator input.

3 Methodology

3.1 System modeling

The UVMS in this study consists of a 6-DoF underwater vehicle (UV), a 4-DoF UM, a 6-DoF master manipulator (MM), and a target object, as shown in Fig. 2. The reference frames $\boldsymbol{\Sigma}_{\mathrm{w}}$, $\boldsymbol{\Sigma}_{\mathrm{v}}$, $\boldsymbol{\Sigma}_{\mathrm{ub}}$, $\boldsymbol{\Sigma}_{\mathrm{ue}}$, $\boldsymbol{\Sigma}_{\mathrm{c}}$, and $\boldsymbol{\Sigma}_{\mathrm{g}}$ denote the frame of the world, UV, the base of the UM, the EE of the UM, camera, and the target object, respectively. $\boldsymbol{\Sigma}_{\mathrm{mb}}$ and $\boldsymbol{\Sigma}_{\mathrm{me}}$ are the base frame and EE frame of the MM, respectively.

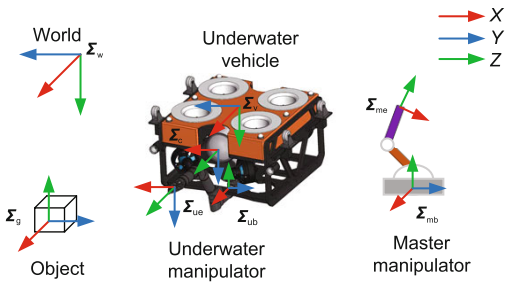

Fig. 2 System composition and reference frames

The configuration of the 10-DoF UVMS can be described by $\boldsymbol{q}_{\text{uvms}} = [\boldsymbol{\eta}_{\text{v}}^{\text{T}}, \boldsymbol{q}_{\text{m}}^{\text{T}}]^{\text{T}}$, where $\boldsymbol{\eta}_{\text{v}} = [x_{\text{v}}, y_{\text{v}}, z_{\text{v}}, \gamma_{\text{v}}, \beta_{\text{v}}, \alpha_{\text{v}}]^{\text{T}}$ denotes the position and roll–pitch–yaw angles of the UV with respect to (w.r.t.) $\boldsymbol{\Sigma}_{\text{w}}$, and $\boldsymbol{q}_{\text{m}} = [\theta_1, \theta_2, \theta_3, \theta_4]^{\text{T}}$ refers to the joint angles of the UM. We assume that $\boldsymbol{\mathcal{V}}_{\text{v}} = [\boldsymbol{v}_{\text{v}}^{\text{T}}, \boldsymbol{\omega}_{\text{v}}^{\text{T}}]^{\text{T}} \in \mathbb{R}^6$ represents the input linear and angular velocity of the vehicle w.r.t. its frame $\boldsymbol{\Sigma}_{\text{v}}$. The relationship between the UVMS velocity $\boldsymbol{\mathcal{V}}_{\text{uvms}} = [\boldsymbol{\mathcal{V}}_{\text{v}}^{\text{T}}, \dot{\boldsymbol{q}}_{\text{m}}^{\text{T}}]^{\text{T}} \in \mathbb{R}^{10}$ and EE velocity is

$$\begin{aligned}\boldsymbol{\mathcal{V}}_{\text{ue}} = [\boldsymbol{J}_{\text{v}}\boldsymbol{J}_{\text{w,v}}, \boldsymbol{J}_{\text{m}}]\dot{\boldsymbol{q}}_{\text{uvms}} &= [\boldsymbol{J}_{\text{v}}, \boldsymbol{J}_{\text{m}}]\boldsymbol{\mathcal{V}}_{\text{uvms}} \\ &= \boldsymbol{J}_{\text{uvms}}\boldsymbol{\mathcal{V}}_{\text{uvms}},\end{aligned} \tag{3}$$

where $\boldsymbol{\mathcal{V}}_{\text{ue}} = [\boldsymbol{v}_{\text{ue}}^{\text{T}}, \boldsymbol{\omega}_{\text{ue}}^{\text{T}}]^{\text{T}}$ denotes the EE velocity of the UM w.r.t. $\boldsymbol{\Sigma}_{\text{ue}}$, $\boldsymbol{J}_{\text{w,v}} \in \mathbb{R}^{6\times6}$ represents the velocity transformation matrix from $\boldsymbol{\Sigma}_{\text{w}}$ to $\boldsymbol{\Sigma}_{\text{v}}$, and $\boldsymbol{J}_{\text{m}} \in \mathbb{R}^{6\times4}$ is the Jacobian of UM w.r.t. $\boldsymbol{\Sigma}_{\text{ue}}$.

The Jacobian of vehicle $\boldsymbol{J}_{\text{v}} \in \mathbb{R}^{6\times6}$ w.r.t. $\boldsymbol{\Sigma}_{\text{ue}}$ can be expressed as

$$\boldsymbol{J}_{\text{v}} = [\text{Ad}_{\boldsymbol{T}_{\text{ue,v}}}] = \begin{bmatrix} \boldsymbol{R}_{\text{ue,v}} & \boldsymbol{0} \\ [\boldsymbol{p}_{\text{ue,v}}]\boldsymbol{R}_{\text{ue,v}} & \boldsymbol{R}_{\text{ue,v}} \end{bmatrix}, \tag{4}$$

where $[\text{Ad}_{\boldsymbol{T}_{\text{ue,v}}}]$ is the adjoint representation (Lynch and Park, 2017) of the transformation matrix $\boldsymbol{T}_{\text{ue,v}}$.

The inverse kinematics for the UVMS is as follows:

$$\boldsymbol{\mathcal{V}}_{\text{uvms}} = \begin{bmatrix} \boldsymbol{\mathcal{V}}_{\text{v}} \\ \dot{\boldsymbol{q}}_{\text{m}} \end{bmatrix} = \boldsymbol{J}_{\text{uvms}}^{\dagger}\boldsymbol{\mathcal{V}}_{\text{ue}}, \tag{5}$$

where $\boldsymbol{J}_{\text{uvms}}^{\dagger}$ refers to the pseudoinverse of the Jacobian $\boldsymbol{J}_{\text{uvms}}$, and $\boldsymbol{J}_{\text{uvms}}^{\dagger} = \boldsymbol{J}_{\text{uvms}}^{\text{T}}(\boldsymbol{J}_{\text{uvms}}\boldsymbol{J}_{\text{uvms}}^{\text{T}})^{-1}$. We select the velocity input as the robot command to maintain motion consistency, because velocity commands can be seamlessly integrated into position commands. $\boldsymbol{\mathcal{V}}_{\text{ue}}$ can be generated both by the HA and the RA.

3.2 Task priority-based motion planning

We employ the task-priority control method for hierarchical control of the multi-task redundant robot. The augmented null space projection (Siciliano and Slotine, 1991) of task priority enhances the orthogonality of all tasks, enabling more efficient handling of conflicting tasks.

For the two constrained tasks $\boldsymbol{x}_1 \in \mathbb{R}^{m_1}$ and $\boldsymbol{x}_2 \in \mathbb{R}^{m_2}$, the solution of the augmented null space projection gives

$$\boldsymbol{u} = \boldsymbol{J}_1^{\dagger}\dot{\boldsymbol{x}}_1 + (\boldsymbol{J}_2\boldsymbol{N}_1)^{\dagger}(\dot{\boldsymbol{x}}_2 - \boldsymbol{J}_2\boldsymbol{J}_1^{\dagger}\dot{\boldsymbol{x}}_1), \tag{6}$$

where $\boldsymbol{J}_i$ is the Jacobian of task $\boldsymbol{x}_i$, and $\boldsymbol{N}_1 = \boldsymbol{I} - \boldsymbol{J}_1^{\dagger}\boldsymbol{J}_1$ denotes the nullspace of task $\boldsymbol{x}_1$. This formula can also be recursively extended to n tasks $(i = 1, 2, ..., n)$:

$$\boldsymbol{u}_{i+1} = \boldsymbol{u}_i + (\boldsymbol{J}_i\boldsymbol{P}_{i-1})^{\dagger}(\dot{\boldsymbol{x}}_i - \boldsymbol{J}_i\boldsymbol{u}_i), \tag{7}$$

where $\boldsymbol{P}_i = \boldsymbol{I} - \boldsymbol{J}_i^{\dagger}\boldsymbol{J}_i$ represents the null space projection matrix of the augmented Jacobian $\boldsymbol{J}_i = [\boldsymbol{J}_1^{\text{T}}, \boldsymbol{J}_2^{\text{T}}, \cdots, \boldsymbol{J}_i^{\text{T}}]^{\text{T}}$.

Our study deploys three control objectives: constraint, operation, and posture optimization tasks.

3.3 Constraint task

Taking into account collision avoidance, we set joint limits to avoid undesired damage to the robotic system. The joint limit objective function (Scheurer et al., 2016) for the UM is defined as (the function diagram can be seen in Fig. S1 in the supplementary materials)

$$\boldsymbol{H}_{\text{cm}}(\boldsymbol{q}) = \begin{cases} \alpha\left(\text{e}^{\beta(\boldsymbol{q}-\boldsymbol{q}_{\min_{\text{set}}})^2} - 1\right), & \boldsymbol{q} < \boldsymbol{q}_{\min_{\text{set}}}, \\ \boldsymbol{0}, & \boldsymbol{q}_{\min_{\text{set}}} \le \boldsymbol{q} \le \boldsymbol{q}_{\max_{\text{set}}}, \\ \alpha\left(\text{e}^{\beta(\boldsymbol{q}-\boldsymbol{q}_{\max_{\text{set}}})^2} - 1\right), & \boldsymbol{q} > \boldsymbol{q}_{\max_{\text{set}}}, \end{cases} \tag{8}$$

where $\boldsymbol{q}_{\min_{\text{set}}} = \boldsymbol{q}_{\min} + \boldsymbol{q}_{\text{th}}$ and $\boldsymbol{q}_{\max_{\text{set}}} = \boldsymbol{q}_{\max} - \boldsymbol{q}_{\text{th}}$ represent the maximum and minimum values of the set joints, respectively. $\boldsymbol{q}_{\max}$ and $\boldsymbol{q}_{\min}$ represent the maximum and minimum physical limits of joints respectively, and $\boldsymbol{q}_{\text{th}}$ is the safety threshold for joints. α and β are constants, and we set $\alpha = \beta = 1$ in the actual use. Derive the joint limit objective function $\boldsymbol{H}_{\text{cm}}(\boldsymbol{q})$ to compute the joint limit Jacobian $\boldsymbol{J}_{\text{cm}}(\boldsymbol{q}) = \text{d}\boldsymbol{H}_{\text{cm}}(\boldsymbol{q})/\text{d}\boldsymbol{q}$.

To keep the joint of the manipulator within its limits, the velocity in the task space is

$$\dot{\boldsymbol{x}}_{\text{cm}} = \boldsymbol{0} - \boldsymbol{H}_{\text{cm}}(\boldsymbol{q}). \tag{9}$$

For the UV, we limit the position of the vehicle (x, y, depth) to avoid collisions:

$$\boldsymbol{H}_{\mathrm{cv}}(\boldsymbol{d})=\begin{cases}\frac{1}{2}(\boldsymbol{d}-\boldsymbol{d}_{\mathrm{th}})^2, & \boldsymbol{d}\le\boldsymbol{d}_{\mathrm{th}},\\ \boldsymbol{0}, & \boldsymbol{d}>\boldsymbol{d}_{\mathrm{th}},\end{cases} \tag{10}$$

where $\boldsymbol{d}$ indicates the distance from obstacles (walls, sea bottom, etc.) and $\boldsymbol{d}_{\mathrm{th}}$ represents the safe distance threshold.

The task velocity of the vehicle constraint is

$$\dot{\boldsymbol{x}}_{\mathrm{cv}}=\boldsymbol{0}-\boldsymbol{H}_{\mathrm{cv}}(\boldsymbol{q}). \tag{11}$$

The input for the joint limit task can be obtained as

$$\boldsymbol{u}_{\mathrm{c}}=[\boldsymbol{J}_{\mathrm{v,w}},\boldsymbol{J}_{\mathrm{cm}}^{-1}]\begin{bmatrix}\dot{\boldsymbol{x}}_{\mathrm{cv}}\\ \dot{\boldsymbol{x}}_{\mathrm{cm}}\end{bmatrix}. \tag{12}$$

3.4 Operation task

The operation task focuses on the motion control of the EE of the UM. The EE is controlled through the ISC method, and the ISC method incorporates inputs from both HA and RA. A whole-body controller is employed to coordinate the movements of the UV and the UM, allowing the operator to concentrate only on controlling the EE.

3.4.1 Human input

The HA input $\boldsymbol{\mathcal{V}}_{\mathrm{hd}}=[\boldsymbol{v}_{\mathrm{hd}}^{\mathrm{T}},\boldsymbol{\omega}_{\mathrm{hd}}^{\mathrm{T}}]^{\mathrm{T}}$ is

$$\boldsymbol{\mathcal{V}}_{\mathrm{hd}}=\boldsymbol{\mathcal{V}}_{\mathrm{m}}+\begin{bmatrix}\boldsymbol{p}_{\mathrm{u,d}}-\boldsymbol{p}_{\mathrm{u}}\\ [\log(\boldsymbol{R}_{\mathrm{u}}\boldsymbol{R}_{\mathrm{u,d}}^{-1})]^{\vee}\end{bmatrix}, \tag{13}$$

where $\boldsymbol{p}_{\mathrm{u,d}}$ and $\boldsymbol{R}_{\mathrm{u,d}}$ denote the desired EE position and orientation of the UM, respectively. $\boldsymbol{p}_{\mathrm{u}}$ and $\boldsymbol{R}_{\mathrm{u}}$ denote the current position and orientation of the EE, respectively. $\boldsymbol{\mathcal{V}}_{\mathrm{m}}$ denotes the EE velocity of the MM. The notation log stands for the matrix logarithm, and operator "$[\cdot]^{\vee}$" indicates the extraction of the orientation angle error vector from the antisymmetric matrix.

The desired EE position of the UM $\boldsymbol{p}_{\mathrm{u,d}}$ is determined by matching the displacement of the MM relative to its initial position:

$$\boldsymbol{p}_{\mathrm{u,d}}=\boldsymbol{p}_{\mathrm{m},0}+k_{\mathrm{s}}(\boldsymbol{p}_{\mathrm{m}}-\boldsymbol{p}_{\mathrm{m},0}), \tag{14}$$

where $\boldsymbol{p}_{\mathrm{m}}$ and $\boldsymbol{p}_{\mathrm{m},0}$ denote the current and the initial EE positions of MM respectively, and k_{s} denotes the scale factor.

The desired EE orientation of the UM $\boldsymbol{R}_{\mathrm{u,d}}$ gives

$$\boldsymbol{R}_{\mathrm{u,d}}=\boldsymbol{R}_{\mathrm{ub,mb}}\boldsymbol{R}_{\mathrm{mb,me}}\boldsymbol{R}_{\mathrm{me,ue}}, \tag{15}$$

where $\boldsymbol{R}_{\mathrm{ub,mb}}$ denotes rotation from $\boldsymbol{\Sigma}_{\mathrm{ub}}$ to $\boldsymbol{\Sigma}_{\mathrm{mb}}$, $\boldsymbol{R}_{\mathrm{mb,me}}$ represents rotation from $\boldsymbol{\Sigma}_{\mathrm{mb}}$ to $\boldsymbol{\Sigma}_{\mathrm{me}}$, and $\boldsymbol{R}_{\mathrm{me,ue}}$ indicates rotation from $\boldsymbol{\Sigma}_{\mathrm{me}}$ to $\boldsymbol{\Sigma}_{\mathrm{ue}}$.

3.4.2 Robot input

The desired RA input $\boldsymbol{\mathcal{V}}_{\mathrm{rd}}=[\boldsymbol{v}_{\mathrm{rd}}^{\mathrm{T}},\boldsymbol{\omega}_{\mathrm{rd}}^{\mathrm{T}}]^{\mathrm{T}}$ can be calculated by a proportional–integral–derivative (PID) controller as

$$\boldsymbol{\mathcal{V}}_{\mathrm{rd}}=K_{\mathrm{p}}\boldsymbol{e}+K_{\mathrm{i}}\int\boldsymbol{e}\mathrm{d}t+K_{\mathrm{d}}\frac{\mathrm{d}\boldsymbol{e}}{\mathrm{d}t}, \tag{16}$$

where $\boldsymbol{e}=[\boldsymbol{e}_{\mathrm{p}}^{\mathrm{T}},\boldsymbol{e}_{\mathrm{q}}^{\mathrm{T}}]^{\mathrm{T}}\in\mathbb{R}^6$ represents the error between the EE and the target object. $\boldsymbol{e}_{\mathrm{p}}$ and $\boldsymbol{e}_{\mathrm{q}}$ denote the position error and orientation error of the target object w.r.t. $\boldsymbol{\Sigma}_{\mathrm{ue}}$, respectively. K_{p}, K_{i}, and K_{d} are the proportional, integral, and differential coefficients, respectively.

3.4.3 Input fusion

We fuse the human input $\boldsymbol{\mathcal{V}}_{\mathrm{hd}}$ and the robot input $\boldsymbol{\mathcal{V}}_{\mathrm{rd}}$ through a control weighting coefficient λ. This coefficient λ is calculated by establishing the control confidence of the RA c_{r} considering the trajectory history of the robot and the human intention.

Given the trajectory $\boldsymbol{\xi}_{\boldsymbol{p}_0\to\boldsymbol{p}_{\mathrm{c}}}$ from the initial state $\boldsymbol{p}_0$ to the current state $\boldsymbol{p}_{\mathrm{c}}$ as shown in Fig. 3, the control confidence of the RA is defined as the probability of the target $P(g|\boldsymbol{\xi}_{\boldsymbol{p}_0\to\boldsymbol{p}_{\mathrm{c}}})$. $P(g|\boldsymbol{\xi}_{\boldsymbol{p}_0\to\boldsymbol{p}_{\mathrm{c}}})\propto P(\boldsymbol{\xi}_{\boldsymbol{p}_0\to\boldsymbol{p}_{\mathrm{c}}}|g)P(g)$ is derived from the maximum posterior probability estimate. Thus,

$$\lambda=c_{\mathrm{r}}=P(\boldsymbol{\xi}_{\boldsymbol{p}_0\to\boldsymbol{p}_{\mathrm{c}}}|g)P(g), \tag{17}$$

where $P(g)$ is the prior probability of the target, indicating the RA's certainty about the target. $P(g)$ is formulated based on the actions of the HA.

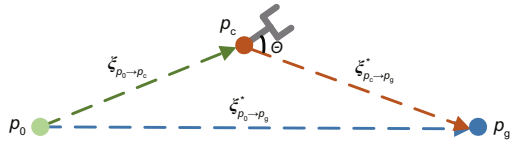

Fig. 3 Calculation of the confidence of the robot agent

The probability is calculated by approximating the integral along the trajectory following the

method (Dragan and Srinivasa, 2013):

$$P(\boldsymbol{\xi}_{\boldsymbol{p}_0\to\boldsymbol{p}_\mathrm{c}}|g)\approx\frac{\mathrm{e}^{-C_\mathrm{g}(\boldsymbol{\xi}_{\boldsymbol{p}_0\to\boldsymbol{p}_\mathrm{c}})-C_\mathrm{g}(\boldsymbol{\xi}^*_{\boldsymbol{p}_\mathrm{c}\to\boldsymbol{p}_\mathrm{g}})}}{\mathrm{e}^{-C_\mathrm{g}(\boldsymbol{\xi}^*_{\boldsymbol{p}_0\to\boldsymbol{p}_\mathrm{g}})}},\quad(18)$$

where $\boldsymbol{\xi}^*_{\boldsymbol{p}_i\to\boldsymbol{p}_j}=\arg\min C(\boldsymbol{\xi}_{\boldsymbol{p}_i\to\boldsymbol{p}_j})$ represents the optimal trajectory from $\boldsymbol{p}_i$ to $\boldsymbol{p}_j$. The cost function $C(\boldsymbol{\xi})=\sum_{i=0}^{N}(\boldsymbol{\xi}_{i+1}-\boldsymbol{\xi}_i)$ is the cumulative distance of the trajectory. $P(\boldsymbol{\xi}_{\boldsymbol{p}_0\to\boldsymbol{p}_\mathrm{c}}|g)$ represents the degree to which the robot deviates from the optimal trajectory when the target is determined.

The intention of the HA is inferred from the position and orientation movements of the EE, which can be formulated as

$$I_\mathrm{p}=\frac{C(\boldsymbol{\xi}^*_{\boldsymbol{p}_{\mathrm{h_c}}\to\boldsymbol{p}_{\mathrm{h_g}}})}{C(\boldsymbol{\xi}^*_{\boldsymbol{p}_{\mathrm{h_{c-1}}}\to\boldsymbol{p}_{\mathrm{h_g}}})},\quad(19)$$

where $\boldsymbol{p}_{\mathrm{h_c}}$ denotes the current position input from the HA, $\boldsymbol{p}_{\mathrm{h_{c-1}}}$ denotes the last position input, and $\boldsymbol{p}_{\mathrm{h_g}}$ denotes the target position, which is mapped back from the EE to the workspace of the MM according to Eq. (14). When $I_\mathrm{p}<1$, the operator is moving towards the target, and vice versa.

Because I_p typically falls within a narrow range around 1, we use the activation function $\tanh(x)$ to amplify the changes in I_p to the range of $[-1,1]$. In Fig. 3, θ represents the angle between the Z axis of the gripper and the line connecting the gripper and the target, which expresses the orientation intention of the HA:

$$P(g)=\alpha_\mathrm{w}\tanh(-2(I_\mathrm{p}-1))+(1-\alpha_\mathrm{w})\cos\theta,\quad(20)$$

where α_w is the constant weight coefficient between the position and orientation inputs, which is set as $\alpha_\mathrm{w}=0.75$ to place more emphasis on changes in the position intention. The value of $P(g)$ is limited to the range $[0,1]$. The greater the tendency of the HA to move toward the target and the smaller the angle θ, the greater the robot's certainty about the target.

Because it is difficult for the RA to autonomously determine the appropriate grasping posture, the orientation is controlled solely by the operator. Mixing inputs from the HA and the RA results in the following:

$$\begin{cases}\boldsymbol{v}_\mathrm{s}=\lambda\boldsymbol{v}_\mathrm{rd}+(1-\lambda)\boldsymbol{v}_\mathrm{hd},\\ \boldsymbol{\omega}_\mathrm{s}=\boldsymbol{\omega}_\mathrm{hd},\end{cases}\quad(21)$$

where $\boldsymbol{\mathcal{V}}_\mathrm{s}=[\boldsymbol{v}_\mathrm{s}^\mathrm{T},\boldsymbol{\omega}_\mathrm{s}^\mathrm{T}]^\mathrm{T}$ denotes the shared EE velocity of the UM.

3.4.4 Motion distribution

Considering the different dynamic characteristics of the UV and the UM, motion distribution is essential to improve the accuracy and efficiency of underwater operations. Weight coefficients are introduced into the Jacobian $\boldsymbol{J}_\mathrm{uvms}$ for motion distribution, which can be expressed as

$$\tilde{\boldsymbol{J}}_\mathrm{uvms}=\boldsymbol{U}_{\boldsymbol{x}}^\mathrm{T}\boldsymbol{J}_\mathrm{uvms}{\boldsymbol{U}_{\boldsymbol{q}}^\mathrm{T}}^{-1},\quad(22)$$

where $\boldsymbol{U}_{\boldsymbol{x}}$ and $\boldsymbol{U}_{\boldsymbol{q}}$ denote the decomposition matrices of the weight matrices $\boldsymbol{W}_{\boldsymbol{x}}$ and $\boldsymbol{W}_{\boldsymbol{q}}$. $\boldsymbol{W}_{\boldsymbol{x}}$ represents the weight matrix in the task space, and $\boldsymbol{W}_{\boldsymbol{q}}$ represents the weight matrix in the joint space.

The weight matrices are designed as

$$\boldsymbol{W}_{\boldsymbol{x}}=\boldsymbol{U}_{\boldsymbol{x}}\boldsymbol{U}_{\boldsymbol{x}}^\mathrm{T}=\begin{bmatrix}\beta_\mathrm{w}\boldsymbol{I}_{3\times3}&\boldsymbol{0}_{3\times3}\\ \boldsymbol{0}_{3\times3}&\boldsymbol{I}_{3\times3}\end{bmatrix},\quad(23)$$

$$\boldsymbol{W}_{\boldsymbol{q}}=\boldsymbol{U}_{\boldsymbol{q}}\boldsymbol{U}_{\boldsymbol{q}}^\mathrm{T}=\begin{bmatrix}\delta\boldsymbol{I}_{6\times6}&\boldsymbol{0}_{6\times4}\\ \boldsymbol{0}_{4\times6}&(1-\delta)\boldsymbol{I}_{4\times4}\end{bmatrix},\quad(24)$$

where the constant β_w adjusts the weight between the position and orientation of the EE Cartesian task. We set $\beta_\mathrm{w}=1$ to give equal importance to position and orientation. δ adjusts the motion distribution weight between the UV and the UM.

Given the fast response speed and high control precision of the UM, it is suitable for close-range, accurate operations. The UV, with its high inertia, is suitable for the approach phase of the operation process. The motion distribution coefficient δ is defined as a function related to the distance from the target object:

$$\delta=\frac{1-\max(0,\frac{d_\mathrm{s}-d_\mathrm{c}+L}{d_\mathrm{s}})}{P(\boldsymbol{\xi}_{\boldsymbol{p}_0\to\boldsymbol{p}_\mathrm{c}}|g)}\in[0,1],\quad(25)$$

where d_s denotes the initial distance between the EE and the target, d_c denotes the current distance between the EE and the target, and L denotes the maximum working distance of the UM. When the distance between the EE and the object is less than L, the manipulator is moved exclusively to ensure working accuracy. The smaller $P(\boldsymbol{\xi}_{\boldsymbol{p}_0\to\boldsymbol{p}_\mathrm{c}}|g)$ is, the more the UVMS deviates from the optimal trajectory. Consequently, more weight is assigned to the UV to quickly approach the target.

The input for the operation task can be obtained using Eq. (5):

$$\boldsymbol{u}_t=\tilde{\boldsymbol{J}}^\dagger_\mathrm{uvms}\boldsymbol{\mathcal{V}}_\mathrm{s}.\quad(26)$$

3.5 Posture optimization task

A better camera view assists the RA in measuring the distance to the target object, reducing both the operation time and cognitive load of the HA. This is facilitated by the image-based visual servoing (IBVS) method.

Assume that $\boldsymbol{s} = [s_{\mathrm{u}}, s_{\mathrm{v}}]^{\mathrm{T}}$ denotes the feature point $\boldsymbol{p}_{\mathrm{c,o}}$ projected on the image plane, and $\boldsymbol{\mathcal{V}}_{\mathrm{c}} = [v_{x,\mathrm{c}}, v_{y,\mathrm{c}}, v_{z,\mathrm{c}}, \omega_{x,\mathrm{c}}, \omega_{y,\mathrm{c}}, \omega_{z,\mathrm{c}}]^{\mathrm{T}}$ denotes the position and orientation velocity of the camera w.r.t. $\boldsymbol{\Sigma}_{\mathrm{c}}$. The relationship between the camera velocity and the image feature point velocity can be simplified as follows:

$$\dot{\boldsymbol{s}} = \boldsymbol{L}_p \boldsymbol{\mathcal{V}}_{\mathrm{c}}, \tag{27}$$

where $\boldsymbol{L}_p$ is the image Jacobian, also known as the interaction matrix, and p is the feature point on the image plane. The image feature point velocity is given as

$$\dot{\boldsymbol{s}} = \begin{bmatrix} s_{\mathrm{u,d}} - s_{\mathrm{u,c}} \\ s_{\mathrm{v,d}} - s_{\mathrm{v,c}} \end{bmatrix}, \tag{28}$$

where $\boldsymbol{s}_{\mathrm{d}} = [s_{\mathrm{u,d}}, s_{\mathrm{v,d}}]^{\mathrm{T}}$ and $\boldsymbol{s}_{\mathrm{c}} = [s_{\mathrm{u,c}}, s_{\mathrm{v,c}}]^{\mathrm{T}}$ denote the desired and current positions of the image feature point, respectively.

The process for selecting the optimal image feature point $\boldsymbol{s}_{\mathrm{d}}$ is illustrated in Fig. 4. We predefine n feature points on the image plane as $[\boldsymbol{s}_1, \boldsymbol{s}_2, \cdots, \boldsymbol{s}_n]$ (with $n = 2$ for demonstration purposes). Knowing the transformation between the camera and the EE, the origin and the Z-axis direction of the end-effector of the underwater manipulator (UE) frame are projected onto the image plane as $\boldsymbol{s}_{\mathrm{ue}}$ and ${}^*\boldsymbol{Z}_{\mathrm{ue}}$, respectively. By calculating the angle φ_i between the vectors $\boldsymbol{s}_{\mathrm{ue}}\boldsymbol{s}_i$ and ${}^*\boldsymbol{Z}_{\mathrm{ue}}$, the feature point $\boldsymbol{s}_i$ with the smallest angle φ_i is selected as the optimal feature point $\boldsymbol{s}_{\mathrm{d}}$:

$$\boldsymbol{s}_{\mathrm{d}} = \arg\min_{\boldsymbol{s}_i} \ \varphi_i(\boldsymbol{s}_{\mathrm{ue}}\boldsymbol{s}_i, {}^*\boldsymbol{Z}_{\mathrm{ue}}), \quad i = 1, 2, \cdots, n. \tag{29}$$

Because the EE direction of the UM is directly controlled by the HA, the UVMS automatically adjusts its posture to provide appropriate camera view according to the operator.

The camera velocity $\boldsymbol{\mathcal{V}}_{\mathrm{c}}$ is converted to the UV velocity $\boldsymbol{\mathcal{V}}_{\mathrm{v}}$:

$$\dot{\boldsymbol{s}} = \boldsymbol{L}_p[\mathrm{Ad}_{\boldsymbol{T}_{\mathrm{c,v}}}]\boldsymbol{\mathcal{V}}_{\mathrm{v}}. \tag{30}$$

The input for the posture optimization task can be obtained as

$$\boldsymbol{u}_p = (\boldsymbol{L}_p[\mathrm{Ad}_{\boldsymbol{T}_{\mathrm{c,v}}}])^{-1}\dot{\boldsymbol{s}}. \tag{31}$$

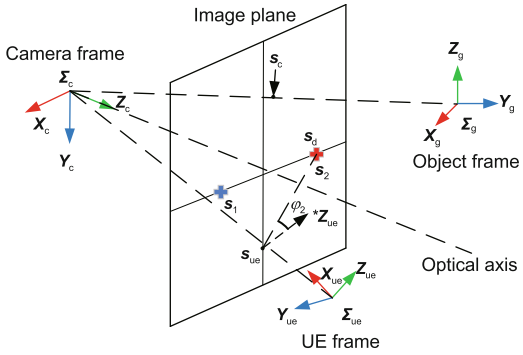

Fig. 4 Feature point $\boldsymbol{s}_{\mathrm{o}}$ selection based on the direction of the UE

3.6 Haptic feedback

When the HA is operating remotely, force feedback is provided to help the HA understand the robot's intentions. The MM is modeled as

$$\boldsymbol{M}(\boldsymbol{x})\ddot{\boldsymbol{x}} + \boldsymbol{C}(\boldsymbol{x}, \dot{\boldsymbol{x}})\boldsymbol{x} = \boldsymbol{F}_{\mathrm{m}} + \boldsymbol{F}_{\mathrm{h}}, \tag{32}$$

where $\boldsymbol{F}_{\mathrm{m}}$ is the force input by the HA to the force feedback device, and $\boldsymbol{F}_{\mathrm{h}}$ is the force feedback to the HA.

Force feedback is designed with two goals. First, it should guide the HA away from the constraints of the robotic system (such as joint limits). Second, the HA should easily perceive the desired position of the SC system. The force feedback function is designed as follows:

$$\boldsymbol{F}_{\mathrm{h}} = \lambda_{\mathrm{f}}\boldsymbol{K}_{\mathrm{m}}(\boldsymbol{p}_{\mathrm{m,d}} - \boldsymbol{p}_{\mathrm{m}}) + \lambda_{\mathrm{f}}\boldsymbol{B}_{\mathrm{m}}(\dot{\boldsymbol{p}}_{\mathrm{m,d}} - \dot{\boldsymbol{p}}_{\mathrm{m}}) + \boldsymbol{F}_{\mathrm{l}}, \tag{33}$$

where $\boldsymbol{p}_{\mathrm{m,d}}$ denotes the desired position in the MM workspace, mapped from the desired position in the UM workspace, and λ_{f} denotes the force feedback coefficient. The positive definite matrices $\boldsymbol{K}_{\mathrm{m}} \in \mathbb{R}^{3\times3}$ and $\boldsymbol{B}_{\mathrm{m}} \in \mathbb{R}^{3\times3}$ denote the stiffness and damping matrices, respectively. $\boldsymbol{F}_{\mathrm{l}}$ is the feedback force that guides the HA away from joint limits. Attractive force cues at the desired position indicate the SC system's intent. $\boldsymbol{p}_{\mathrm{m,d}}$ is calculated as

$$\boldsymbol{p}_{\mathrm{m,d}} = \boldsymbol{p}_{\mathrm{m},0} + \frac{1}{k_{\mathrm{s}}}(\boldsymbol{p}_{\mathrm{u}} - \boldsymbol{p}_{\mathrm{u},0}). \tag{34}$$

Considering that only three motors of the MM can provide force feedback, the joint limit feedback

force is projected onto the MM workspace using the linear part of the Jacobian matrix $\boldsymbol{J}_{\mathrm{m,v}}$:

$$\boldsymbol{F}_{\mathrm{l}} = -(\boldsymbol{J}_{\mathrm{m,v}}^{\mathrm{T}})^{\dagger}\frac{\partial \boldsymbol{H}_{\mathrm{cm}}(\boldsymbol{q})}{\partial \boldsymbol{q}}. \tag{35}$$

The control framework for the UVMS is shown in Fig. 5. $\boldsymbol{\eta}_{\mathrm{d}}$ denotes the desired position and orientation of the UVMS, $\boldsymbol{\eta}_{\mathrm{e}}$ denotes the end-effector position, and $\boldsymbol{\tau}_{\mathrm{d}}$ denotes the input torque calculated by the dynamic controller. The dynamic controllers of the UM and the UV are both PID controllers.

4 Simulations

We first verified the method's efficacy in aiding the operator's task completion through a grasping simulation. Subsequently, a user study was conducted to evaluate three control methods: manual control (MC), USC, and automatic control (AC). In MC, the operator used a whole-body controller via a joystick to control the position and orientation of the EE, viewing only the robot's camera feed displayed on a screen. In the AC method, the robot autonomously grasps the object using the approach described in Cieślak et al. (2015). The approach is also based on task priority and has been proven to successfully perform tasks such as underwater grasping and valve rotation (Simetti et al., 2014) without human intervention. This method serves as a benchmark for AC methods to illustrate the advantages of USC by incorporating human collaboration. USC integrates MC and AC to accomplish tasks through human–robot collaboration.

4.1 Simulation setup

The simulation setup is illustrated in Fig. 6. The UVMS used in the simulations consisted of a self-designed 6-DoF UV and a 4-DoF UM with a two-finger gripper. Geometry Touch served as the MM, capturing human input and outputting feedback forces. Object detection and IBVS algorithms were implemented using visual servoing platform (ViSP) (Marchand et al., 2005). The entire control system was developed on the robot operating system (ROS). A Tobii Eye Tracker 5 was attached to the monitor to record the operator's eye movements. The target object-grasping simulation using the UUV simulator (Manhães et al., 2016) was conducted to verify the proposed control method. The entire grasping process is illustrated in Fig. 7.

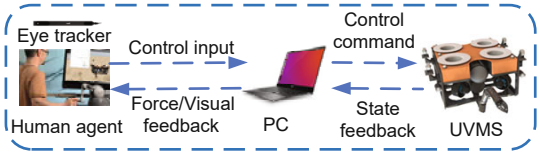

Fig. 6 Simulation setup

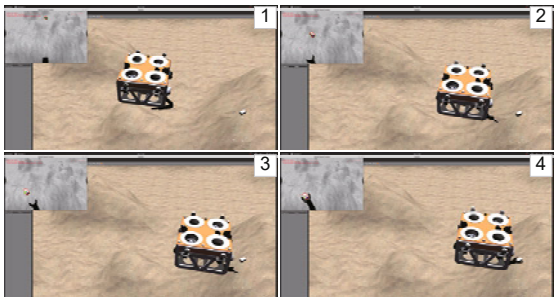

Fig. 7 Underwater grasping process

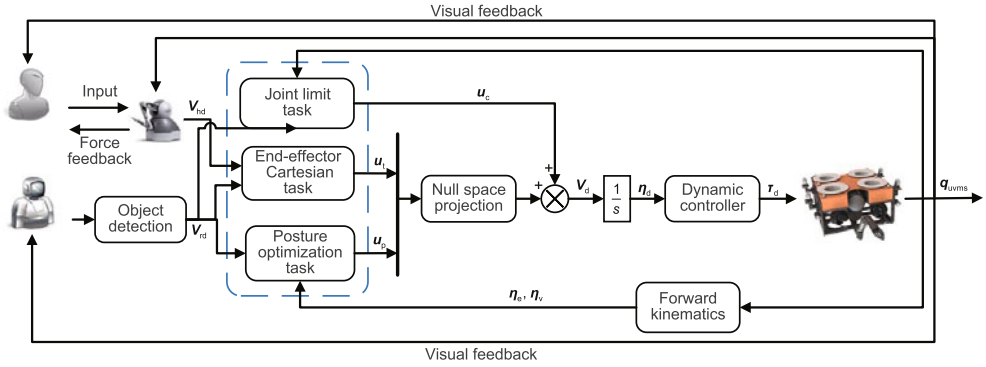

Fig. 5 Control framework. $1/s$ is the integral operation

4.2 Hypotheses

Prior literature indicates that more autonomy leads to better performance (Javdani et al., 2015). Based on this, we formulated the following hypotheses for different control methods, measured using objective metrics:

Hypothesis 1 Methods with more autonomy will lead to faster task completion.

Hypothesis 2 Methods with more autonomy will result in less human input.

Hypothesis 3 Methods with more autonomy will reduce the cognitive load of the operator.

4.3 Procedure

We recruited 13 participants (ten males, three females) for simulation from a university campus aged 23–32 years (mean = 25.7, standard deviation (StD) = 2.77). All participants are science, technology, engineering, mathematics (STEM) students pursuing a master's or doctoral degree. Participants had low familiarity with teleoperated robots (mean = 1.92, StD = 0.95, surveyed via a seven-point Likert scale). Three participants reported prior experience with robotics research.

After obtaining written consent, the participants were given 8 min to get familiar with both MC and USC. To simulate a real environment, we introduced a random disturbance at a specific stage near the object. Because most underwater disturbances were lateral, we added random disturbances lasting 10 s to the robot's X-axis, Y-axis, and yaw directions as $F_x, F_y, M_z \sim U(0, 10)$. The values of the disturbances were extracted in advance through the random function. The same disturbances were applied to each operator using the three methods. Table S1 in the supplementary materials shows the disturbances imposed on each group. Objective measurements (root mean square error (RMSE), success rate, execution time, operator input, and operator's eye movements) were adopted for simulation evaluation.

4.4 Simulation results

4.4.1 Method validation simulation

We conducted single-shot object-grasping simulations to verify the effectiveness of the USC control method. To demonstrate the effectiveness of the constraint task, we intentionally moved each joint close to its limit. A specific velocity was set for each joint to approach its limit. As shown in Fig. 8, the joints stopped moving forward once they reached the safe threshold. The joint limit constraint protected the joints from physical damage. The safe threshold was set at $q_{\text{th}} = 0.2$ rad.

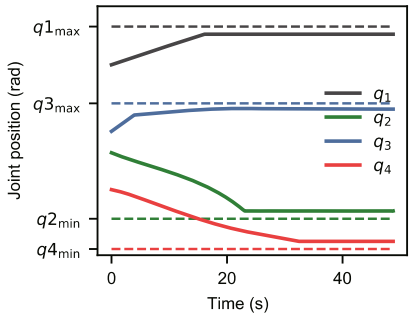

Fig. 8 Behavior of the joints under the constraint task

The EE trajectory of the UM under the USC method is depicted in Fig. 9. Fig. 9a shows that the position error decreased from the initial 0.9 m to about 0.01 m. The entire grasping process took approximately 72 s. The 3D trajectory of the vehicle and the EE is shown in Fig. 9b. The UV and the UM moved simultaneously and coordinated to complete the grasping task. There remained a certain distance between the final position of the EE and the target position because the target position recognized by the camera was the center of the cuboid. However, the EE had already made contact with the target object.

Fig. 10a shows the behavior of the motion distribution coefficient δ. As the distance to the target object decreased, δ gradually decreased from 0.7 to nearly 0. This was consistent with the trend of the distance d_{r} between the robot and the target in Fig. 10c. The control weight of the UV was large when the distance d_{r} was large. The motion weight of the UM gradually increased as the distance d_{r} became smaller.

The behaviors of the confidence and the weight coefficient of the SC are shown in Fig. 10b. The weight coefficient λ ranged from 0 at the beginning to about 0.9 at the end. In Fig. 10c, the decrease in d_{r} and d_{h} caused the weight coefficient to gradually increase. This indicates that control was eventually managed primarily by the RA, which is expected

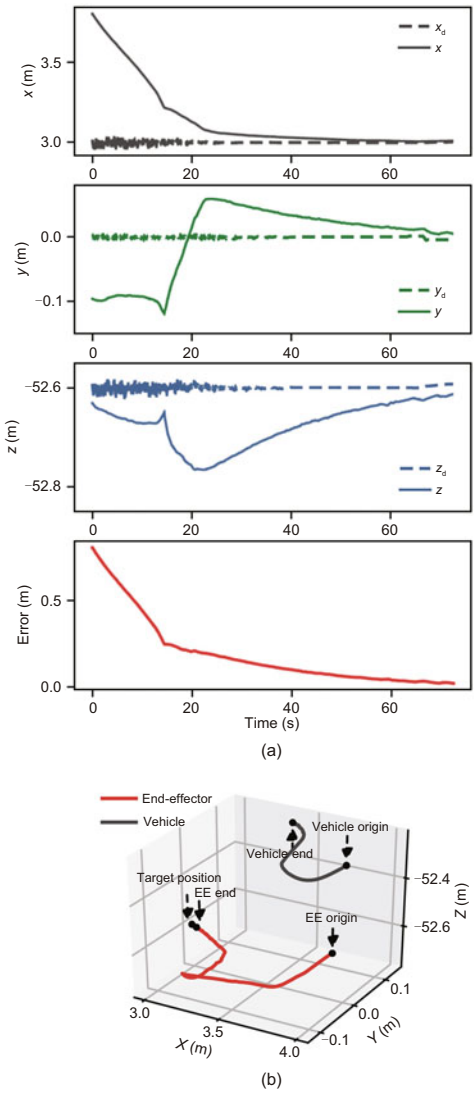

Fig. 9 EE trajectory during the target gripping process: (a) tracking error; (b) 3D trajectory of the UV and the UE. x_d, y_d, and z_d in (a) refer to the desired positions

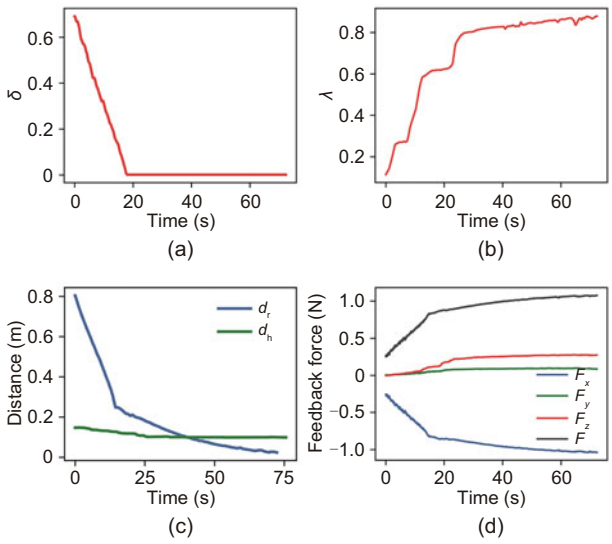

Fig. 10 Results of the SC in the operation task: (a) behavior of the motion distribution coefficient δ; (b) behavior of the weight coefficient λ; (c) distance from target in the robot workspace d_r and in the human operation workspace d_h; (d) behavior of the feedback force. SC: shared control

because the RA specializes in final-stage refinement. The behavior of the feedback force is depicted in Fig. 10d. The trend of the feedback force was consistent with the trend of the weight coefficient λ, proving that the force feedback design proposed in Eq. (33) is effective. The reason why the feedback force remained large at the end was that the HA stayed still in the final phase to let the RA adjust the final grasping position. There was a deviation between the current position of the MM and the desired position in the UM workspace mapped back to the MM workspace.

To verify the effect of the posture optimization method, we presented two feature points $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ on the image plane. At the beginning of the process, the EE was oriented towards the feature point $\boldsymbol{s}_2$ on the right plane of the image. Then the EE turned toward $\boldsymbol{s}_1$ on the left image plane.

Fig. 11a shows the angles φ_1, φ_2 between the feature point $\boldsymbol{s}_1, \boldsymbol{s}_2$ and the Z-axis direction of the EE. φ_2 was smaller than φ_1 before 43 s. The UVMS adjusted the posture to move the feature point of the target object to $\boldsymbol{s}_2$ in the camera view. When φ_1 became smaller, $\boldsymbol{s}_1$ became the new target. This is verified by the sudden change in feature error at 43 s in Fig. 11b. The process of UVMS posture adjustment and grasping is shown in Fig. 11c.

4.4.2 Comparative simulation

Objective measurements were analyzed using one-way analysis of variance (ANOVA) with a significance threshold $\alpha_c = 0.05$. All data passed the Shapiro-Wilk normality test. The results are shown in Fig. 12. The ANOVA results can be found in Table S2 in the supplementary materials.

Fig. 12a shows the EE motion trajectory under the three control methods. Disturbances were added near the end position, significantly affecting the robot's trajectory. Fig. 12b shows that the RMSE values in position for MC, USC, and AC methods were 0.022, 0.016, and 0.010 m, respectively.

This suggests that methods with more autonomous assistance lead to higher control precision, consistent with common intuition.

Despite increased error with operator involvement, it can help mitigate the target loss problem due to perturbations. Fig. 12c indicates that the success rates of MC, USC, and AC were 1.00, 1.00, and 0.77, respectively. Human participation can help the robot regain sight of the target when it is lost, and reduce instability caused by external disturbances.

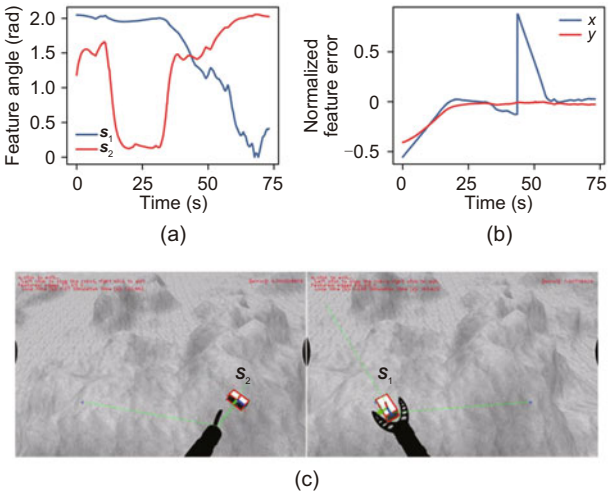

Fig. 11 Behavior of the feature point: (a) angle between the Z axis of the UE and the feature point; (b) IBVS feature error; (c) camera view

The average task execution time is 80 s for MC, 66 s for USC, and 37 s for AC. The execution time is reduced by 17.50% compared with MC. The task execution time differed significantly between control methods ($F(2,36) = 37.93$, $p < 0.001$). Increased autonomy resulted in faster task completion, and thus we accepted Hypothesis 1.

For the trajectory length of the operator input and eye movements, the data corresponding to AC are 0. The average length of the operator input was 0.3 for USC and 0.4 for MC, reduced by 25.00%. The operator input differed significantly between the two methods ($F(1,24) = 20.56$, $p < 0.01$). Force guidance played an important role in assisting the operator, and thus Hypothesis 2 could be accepted.

Fig. 12f shows that the trajectory lengths of the

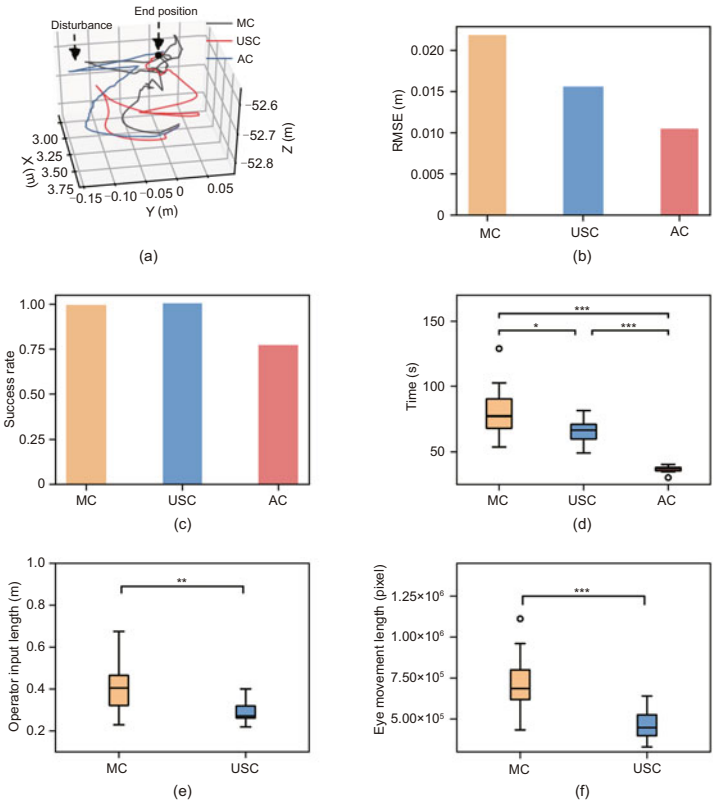

Fig. 12 Comparison results of different control methods: (a) motion trajectory; (b) RMSE of EE; (c) success rate; (d) task execution time; (e) operator input length; (f) eye movement length. *$p < 0.05$, **$p < 0.01$, and ***$p < 0.001$. Hollow circles refer to outliers in box plots

operator's eye movements for USC and MC were 470 577 and 729 865 pixel respectively, reduced by 35.53%. The operator's eye movements differed significantly between methods ($F(1,24) = 30.45$, $p < 0.001$). Increased autonomy can reduce the operator's cognitive load, and thus we found the support for Hypothesis 3.

Objective measurements substantiate our hypotheses that increased autonomy yields superior performance. The USC method outperforms MC, while AC offers advantages in execution efficiency. However, USC stands out as a more reliable option due to human intervention.

5 Experiments

To validate the effectiveness of the proposed USC method, we conducted grasping experiments in a pool using a UVMS. Similar to the simulations, we carried out a user study to evaluate three robot control methods: MC, USC, and SC based on a piecewise function (PSC). PSC was used as a comparative SC method for USC, with its piecewise function designed with reference to the baseline approach described in Dragan and Srinivasa (2013). The relationship between the control weight coefficient and control confidence is as follows:

$$\lambda = \begin{cases} 0, & c_{\mathrm{r}} < 0.2, \\ \frac{4}{3}\left(c_{\mathrm{r}} - 0.2\right), & 0.2 \le c_{\mathrm{r}} \le 0.8, \\ 0.8, & \text{otherwise.} \end{cases} \tag{36}$$

5.1 Experimental setup

The configuration of the UVMS employed in the experiments mirrors that of the simulations, as illustrated in Fig. 13a. VINS-Mono (Qin et al., 2018) served as the localization method in this experimental trial, and hand-eye calibration was conducted before the experiments. The console, illustrated in Fig. 13b, was configured to collect operational data during task execution. The drivers for the UV and the UM were implemented using ROS. Both the UV and the UM were equipped with PID controllers. The control algorithm can be easily adapted for deployment on an actual UVMS with minimal adjustments. The overall master–slave control framework of the robot is depicted in Fig. 14.

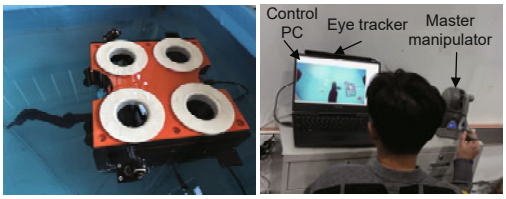

Fig. 13 Experimental setup: (a) underwater vehicle–manipulator system (UVMS); (b) control console

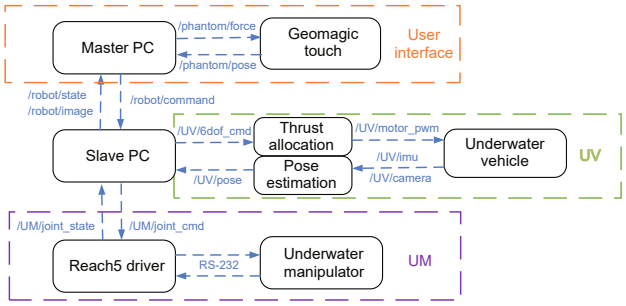

Fig. 14 Robot operating system (ROS) control framework of the underwater vehicle–manipulator system (UVMS)

5.2 Hypotheses

Following the assumption of the objective measurements above, we developed the following hypotheses regarding user-perceived workload, measured through subjective metrics:

Hypothesis 4 Participants will more strongly agree on reduced mental and physical demands with autonomous assistance.

Hypothesis 5 Participants will more strongly agree on reduced effort with autonomous assistance.

Hypothesis 6 Participants will more strongly agree on feeling satisfied with autonomous assistance.

5.3 Procedure

We recruited nine volunteers (seven males, two females) aged 23–29 years (mean = 25.67, StD = 2.0) to participate in the fake sea cucumber grasping experiments. All participants are STEM students pursuing a master's or doctoral degree. Participants had low familiarity with teleoperated robots (mean = 2.22, StD = 0.83). Three participants reported prior experience with robotics research.

Objective (completion time, success rate, operator input length, and trajectory length of operator eye movements) and subjective matrices (national aeronautics and space administration's task

load index, NASA-TLX) (Hart, 2006) were adopted for experimental evaluation. The NASA-TLX survey assesses the effectiveness and performance of the system by rating perceived workload. The workload is categorized into six subscales: (1) mental demand, (2) physical demand, (3) temporal demand, (4) performance, (5) effort, and (6) frustration. The meaning of each item in NASA-TLX is explained in the supplementary materials. Each subscale is rated within a 100-point range with 5-point steps. Lower scores were considered favorable for each subscale.

Each experimental session included the MC, PSC, and USC. Before the practical sessions, volunteers were allotted 15 min to acquaint themselves with both control methods. Volunteers were asked to fill out the NASA-TLX survey at the end of each task to collect subjective feelings.

5.4 Experimental results

A structured scene was arranged with objects placed in a specific order, using Aruco markers to aid in precise target positioning. A typical fake sea cucumber grasping process is presented in Fig. 15.

Fig. 16 represents the position trajectory of the EE throughout the grasping process. Specifically, Figs. 16a and 16b display the tracking error and 3D trajectory of the EE, respectively. As shown in Fig. 16a, minor manual adjustments were performed between approximately 10 and 15 s. During this time, the distance between the EE and the target increased. After fine-tuning, the target was successfully grasped within approximately 40 s.

Fig. 17 illustrates the behaviors of the motion coefficient δ, weight coefficient λ, and feedback force. The motion coefficient in Fig. 17a shifts towards the UM as the distance decreases. The HA intervention at approximately 15 s resulted in a change in the weight coefficient. Fig. 17b shows that the control weight of the RA starts at a low value of approximately 0.15 in the initial stage. Subsequently, the control weight of the RA increased rapidly. The HA intervened at about 15 s, causing a rapid drop in the RA control weight coefficient. After that, the control weight coefficient began to increase until the robot completely dominated the control at the end of the stage.

The change in feedback force is shown in Fig. 17c. The feedback force exhibits a gradually decreasing trend, and aligns with the changes in position error depicted in Fig. 16a. Additionally, the feedback force experiences a slight fluctuation of around 15 s due to the operator's control. Fig. 17d depicts the behaviors of the feature angles. After about 10 s of operator adjustment, the posture optimized toward the desired $\boldsymbol{s}_2$.

We compared the MC, PSC, and USC in underwater grasping tasks. The same objective measurements as in the simulations for control methods were recorded for ANOVA analysis. All data passed the Shapiro-Wilk normality test. The results of the analysis are presented in Fig. 18. The ANOVA results can be found in Table S3 in the supplementary materials.

As shown in Fig. 18a, the average task completion time was 44 s for MC, 39 s for PSC, and 34 s for USC. The execution time is reduced by 22.73% and 12.82% compared with that of MC and PSC, respectively. The completion time differed significantly between control methods ($F(2, 24) = 9.58$, $p < 0.01$). More autonomy resulted in faster task completion. Significant differences were observed between pairs of methods. Hence, we accepted Hypothesis 1.

Fig. 18b shows that volunteers can successfully complete the task, regardless of the control methods they used. This may be due to the sufficient time provided for the volunteers to become familiar with

Fig. 15 Sea cucumber grasping process

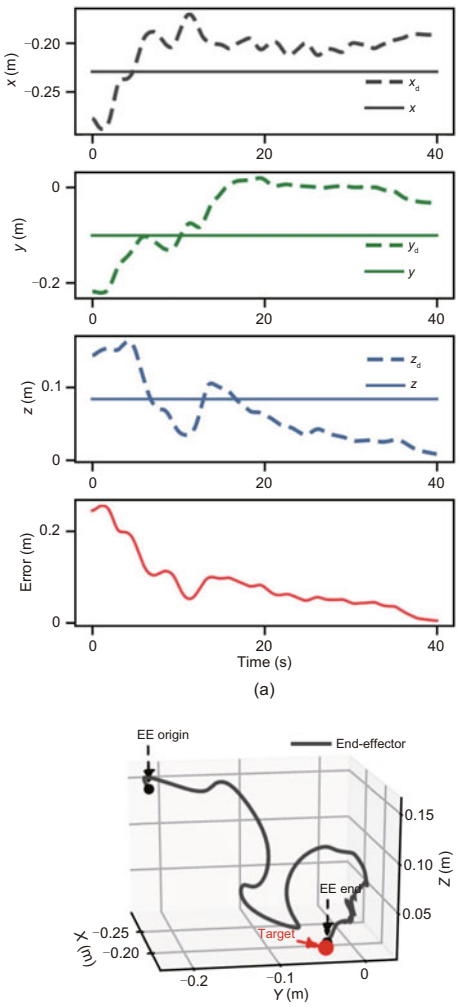

Fig. 16 EE trajectory: (a) tracking error; (b) 3D trajectory

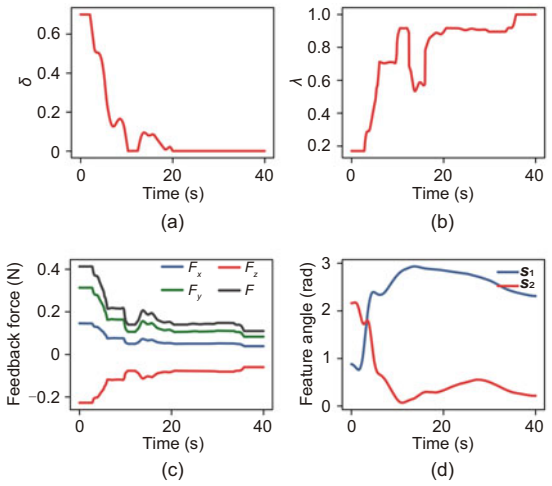

Fig. 17 Results of the USC in the experiments: (a) behavior of the motion coefficient δ; (b) behavior of the weight coefficient λ; (c) behavior of the feedback force; (d) behavior of the feature angle

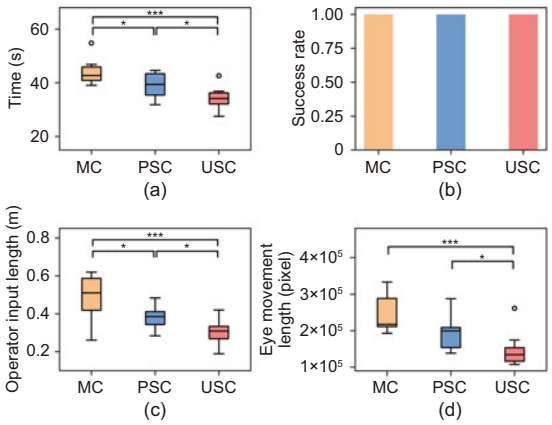

Fig. 18 Comparison results of different control methods in pool experiments: (a) task completion time; (b) success rate; (c) operator input length; (d) eye movement length. *$p < 0.05$ and ***$p < 0.001$. Hollow circles refer to outliers in box plots

the control system. USC does not demonstrate a distinct advantage in terms of success rate.

Fig. 18c presents the average trajectory lengths of the operator input, which were 0.5 m for MC, 0.4 m for PSC, and 0.3 m for USC. The input length is reduced by 40.00% and 25.00% compared with MC and PSC, respectively. The input trajectory length differed significantly between the control methods ($F(2, 24) = 8.44$, $p < 0.01$). There were significant differences between pairs of methods. Hence, we accepted Hypothesis 2.

Fig. 18d represents the average trajectory lengths of the operator eye movements, which were 225 124 for MC, 198 826 for PSC, and 157 780 for USC. This index decreased by 29.91% and 20.64% compared with MC and PSC, respectively. The average eye movement trajectory length differed significantly across the control methods ($F(2, 24) = 9.23$, $p < 0.01$). Significant differences were observed between all control methods except for MC versus PSC. Therefore, Hypothesis 3 was accepted.

Fig. 19 presents the subjective ratings of participants for each dimension of NASA-TLX. The SC method resulted in a slight increase (deterioration) in mental demand compared to MC. Among the SC approaches, PSC showed a higher mental demand than USC. However, the observed increase in mental demand was not as expected. Regarding physical demand, temporal demand, effort, and frustration, USC demonstrated notable improvements. In terms of performance, all three methods received comparable ratings.

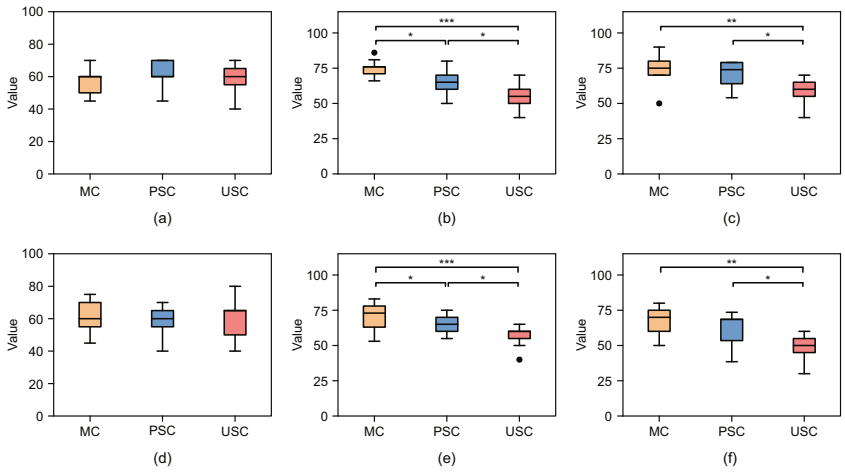

Fig. 19 Results of NASA-TLX: (a) mental demand; (b) physical demand; (c) temporal demand; (d) performance; (e) effort; (f) frustration. *$p < 0.05$, **$p < 0.01$, and ***$p < 0.001$. Solid circles refer to outliers in box plots

Physical demand differed significantly between control methods ($F(2, 24) = 10.63$, $p < 0.001$). USC was less physically demanding than MC ($p < 0.001$) and PSC ($p < 0.05$). However, due to the increase in mental demand, Hypothesis 4 is not supported.

Temporal demand also differed significantly between the control methods ($F(2, 24) = 5.93$, $p < 0.05$). USC resulted in a lower sense of time urgency when performing tasks compared to both MC ($p < 0.01$) and PSC ($p < 0.05$). In contrast, PSC did not lead to a significant reduction in time pressure.

Effort scores were 70 for MC, 65 for PSC, and 57 for USC. Effort differed significantly between the control methods ($F(2, 24) = 5.57$, $p < 0.01$). Significant differences were found between pairs of methods. Post-hoc analysis revealed that increased autonomy led to reduced user effort. Therefore, we accepted Hypothesis 5.

The average performance scores for the methods were relatively similar. There was no significant difference in performance between the control methods. Frustration differed significantly between methods ($F(2, 24) = 6.61$, $p < 0.01$). USC resulted in more positive emotions for operators compared to both MC ($p < 0.01$) and PSC ($p < 0.05$). Post-hoc analysis revealed that automated assistance can enhance participants' feelings and satisfaction while performing tasks. Thus, we found the support for Hypothesis 6.

Overall, subjective measurements indicate that increased automation reduces task load, with operator ratings following the order of USC > PSC > MC. This trend is consistent with the results from objective measurements.

6 Discussions

For objective measurements, the experimental results of grasping are consistent with the simulation results. As anticipated, USC with autonomous assistance outperformed the other methods. Significant differences were observed in most objective measures, particularly in execution time, operator input length, and length of operator eye movements, supporting Hypothesis 1–3. The objective results indicate that SC reduced control input length and cognitive demands on participants, thereby enhancing operational efficiency during underwater grasping tasks.

However, there remains some uncertainties in the subjective results. Significant differences were observed across most measurements, supporting Hypothesis 5–6. Unexpectedly, USC resulted in an increase in mental demand compared to MC and PSC. This increase is attributed to the need for collaboration between HA and RA. As one participant noted, "I need to be constantly aware of RA's intentions." In addition to haptic guidance, incorporating more intuitive interaction methods (such as VR or digital twins) could help users better understand the RA's intentions. Despite the slight increase in mental demand, there was a significant improvement in

physical demand and effort.

Users had varying opinions about USC, as reflected in the greater variance in performance scores. Some participants preferred direct MC, with one remarking, "While shared control is easier and more effective, it makes me feel inadequate." Others noted that the RA helped them focus on key parts, enhancing their performance. Regarding frustration, most users expressed more positive emotions towards USC than PSC, with comments such as, "It feels like having a partner to complete the task together." Assistance from others generally improved feelings of well-being. This suggests that exploring user preferences and adjusting the assistance method accordingly could be an interesting avenue for future research.

Although the underwater grasping task is relatively simple, both subjective and objective measurements demonstrate the effectiveness of the method. This study confirms that USC reduced operator workload and burden while improving the efficiency of underwater operations. In future research, we aim to develop an underwater robot digital twin system, offering an immersive sense of control and fostering mutual learning between humans and robots.

7 Conclusions

This study presents a USC architecture for UVMSs, which incorporates task priority. The approach combines DSC and ISC, establishing a hierarchical relationship among subtasks based on their priority. A haptic feedback system is designed to assist the operator in completing tasks. A simulated object-grasping experiment is conducted to verify the effectiveness of the proposed method in comparison with AC and MC methods. The simulation results demonstrate the superiority of the proposed method over MC, with reductions in completion time, operator input length, and cognitive load by 17.50%, 25.00%, and 35.53%, respectively. Although AC performs well in specific indicators, human intervention proves beneficial in managing external disturbances. Based on the objective and NASA-TLX subjective evaluation results from the pool experiments, the proposed method outperforms both MC and PSC. The method reduces completion time, operator input length, and cognitive load by 22.73%, 40.00%, and 29.91%, respectively, compared to MC. The proposed USC method effectively alleviates the operator load, and enhances efficiency in underwater operations.

Contributors

Zhangpeng TU designed the research. Zhangpeng TU and Xin WU processed the data. Zhangpeng TU drafted the paper. Yuanchao ZHU and Canjun Yang helped organize the paper. Zhangpeng TU and Yuanchao ZHU 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

Birk A, Doernbach T, Mueller C, et al., 2018. Dexterous underwater manipulation from onshore locations: streamlining efficiencies for remotely operated underwater vehicles. *IEEE Robot Autom Mag*, 25(4):24-33. https://doi.org/10.1109/MRA.2018.2869523

Brantner G, Khatib O, 2021. Controlling Ocean One: human–robot collaboration for deep-sea manipulation. *J Field Robot*, 38(1):28-51. https://doi.org/10.1002/rob.21960

Capocci R, Omerdic E, Dooly G, et al., 2018. Fault-tolerant control for ROVs using control reallocation and power isolation. *J Mar Sci Eng*, 6(2):40. https://doi.org/10.3390/jmse6020040

Cieślak P, Ridao P, 2018. Adaptive admittance control in task-priority framework for contact force control in autonomous underwater floating manipulation. Proc IEEE/RSJ Int Conf on Intelligent Robots and Systems, p.6646-6651. https://doi.org/10.1109/IROS.2018.8593542

Cieślak P, Ridao P, Giergiel M, 2015. Autonomous underwater panel operation by GIRONA500 UVMS: a practical approach to autonomous underwater manipulation. Proc IEEE Int Conf on Robotics and Automation, p.529-536. https://doi.org/10.1109/ICRA.2015.7139230

Di Lillo P, Simetti E, Wanderlingh F, et al., 2021. Underwater intervention with remote supervision via satellite communication: developed control architecture and experimental results within the DexROV Project. *IEEE Trans Contr Syst Technol*, 29(1):108-123. https://doi.org/10.1109/TCST.2020.2971440

Dragan AD, Srinivasa SS, 2013. A policy-blending formalism for shared control. *Int J Robot Res*, 32(7):790-805. https://doi.org/10.1177/0278364913490324

Hart SG, 2006. NASA-task load index (NASA-TLX); 20 years later. *Proc Hum Factors Ergon Soc Annu Meet*,

50(9):904-908.
https://doi.org/10.1177/154193120605000909

Javdani S, Srinivasa SS, Bagnell JA, 2015. Shared autonomy via hindsight optimization.
https://doi.org/10.48550/arXiv.1503.07619

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

Lin MW, Yang CJ, 2020. Ocean observation technologies: a review. *Chin J Mech Eng*, 33(1):32.
https://doi.org/10.1186/s10033-020-00449-z

Lynch KM, Park FC, 2017. Modern Robotics. Cambridge University Press, Cambridge, UK.
https://doi.org/10.1017/9781316661239

Manhães MMM, Scherer SA, Voss M, et al., 2016. UUV simulator: a Gazebo-based package for underwater intervention and multi-robot simulation. Proc OCEANS MTS/IEEE Monterey, p.1-8.
https://doi.org/10.1109/OCEANS.2016.7761080

Manley JE, Halpin S, Radford N, et al., 2018. Aquanaut: a new tool for subsea inspection and intervention. Proc OCEANS MTS/IEEE Charleston, p.1-4.
https://doi.org/10.1109/OCEANS.2018.8604508

Marchand E, Spindler F, Chaumette F, 2005. ViSP for visual servoing: a generic software platform with a wide class of robot control skills. *IEEE Robot Autom Mag*, 12(4):40-52. https://doi.org/10.1109/MRA.2005.1577023

Maurelli F, Carreras M, Salvi J, et al., 2016. The PANDORA project: a success story in AUV autonomy. Oceans, p.1-8. https://doi.org/10.1109/OCEANSAP.2016.7485618

Nicolis D, Palumbo M, Zanchettin AM, et al., 2018. Occlusion-free visual servoing for the shared autonomy teleoperation of dual-arm robots. *IEEE Robot Autom Lett*, 3(2):796-803.
https://doi.org/10.1109/LRA.2018.2792143

Petillot YR, Antonelli G, Casalino G, et al., 2019. Underwater robots: from remotely operated vehicles to intervention-autonomous underwater vehicles. *IEEE Robot Autom Mag*, 26(2):94-101.
https://doi.org/10.1109/MRA.2019.2908063

Qin T, Li PL, Shen SJ, 2018. VINS-Mono: a robust and versatile monocular visual-inertial state estimator. *IEEE Trans Robot*, 34(4):1004-1020.
https://doi.org/10.1109/TRO.2018.2853729

Rakita D, Mutlu B, Gleicher M, 2018. An autonomous dynamic camera method for effective remote teleoperation. Proc ACM/IEEE Int Conf on Human–Robot Interaction, p.325-333.

Ribas D, Ridao P, Turetta A, et al., 2015. I-AUV mechatronics integration for the TRIDENT FP7 project. *IEEE/ASME Trans Mechatron*, 20(5):2583-2592.
https://doi.org/10.1109/TMECH.2015.2395413

Ridao P, Carreras M, Ribas D, et al., 2015. Intervention AUVs: the next challenge. *Annu Rev Cont*, 40:227-241.
https://doi.org/10.1016/j.arcontrol.2015.09.015

Sahoo A, Dwivedy SK, Robi PS, 2019. Advancements in the field of autonomous underwater vehicle. *Ocean Eng*, 181:145-160.
https://doi.org/10.1016/j.oceaneng.2019.04.011

Samuel CMT, Tee KP, 2019. Unified human–robot shared control with application to haptic telemanipulation. Proc IEEE/RSJ Int Conf on Intelligent Robots and Systems, p.2221-2226.
https://doi.org/10.1109/IROS40897.2019.8967719

Scheurer C, Fiore MD, Sharma S, et al., 2016. Industrial implementation of a multi-task redundancy resolution at velocity level for highly redundant mobile manipulators. 47th Int Symp on Robotics, p.1-9.

Shim H, Jun BH, Lee PM, et al., 2010. Workspace control system of underwater tele-operated manipulators on an ROV. *Ocean Eng*, 37(11-12):1036-1047.
https://doi.org/10.1016/j.oceaneng.2010.03.017

Siciliano B, Slotine JJE, 1991. A general framework for managing multiple tasks in highly redundant robotic systems. Proc 5th Int Conf on Advanced Robotics, p.1211-1216. https://doi.org/10.1109/ICAR.1991.240390

Simetti E, Casalino G, Torelli S, et al., 2014. Floating underwater manipulation: developed control methodology and experimental validation within the TRIDENT project. *J Field Robot*, 31(3):364-385.
https://doi.org/10.1002/rob.21497

Simetti E, Casalino G, Wanderlingh F, et al., 2018. Task priority control of underwater intervention systems: theory and applications. *Ocean Eng*, 164:40-54.
https://doi.org/10.1016/j.oceaneng.2018.06.026

Simpkins CA, 2014. Introduction to autonomous manipulation: case study with an underwater robot, SAUVIM. *IEEE Robot Autom Mag*, 21(4):109-110.
https://doi.org/10.1109/MRA.2014.2379031

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

Wang FY, Guo JB, Bu GQ, et al., 2022. Mutually trustworthy human–machine knowledge automation and hybrid augmented intelligence: mechanisms and applications of cognition, management, and control for complex systems. *Front Inform Technol Electron Eng*, 23(8):1142-1157. https://doi.org/10.1631/FITEE.2100418

Yang CJ, Wu X, Zhu YC, et al., 2022a. Recent progress of an underwater robotic avatar. Proc 15th Int Conf on Intelligent Robotics and Applications, p.615-626.
https://doi.org/10.1007/978-3-031-13844-7_57

Yang CJ, Zhu YC, Chen YH, 2022b. A review of human–machine cooperation in the robotics domain. *IEEE Trans Hum-Mach Syst*, 52(1):12-25.
https://doi.org/10.1109/THMS.2021.3131684

Zhang T, Li Q, Zhang CS, et al., 2017. Current trends in the development of intelligent unmanned autonomous systems. *Front Inform Technol Electron Eng*, 18(1):68-85. https://doi.org/10.1631/FITEE.1601650

Zhu YC, Yang CJ, Tu ZP, et al., 2023. A haptic shared control architecture for tracking of a moving object. *IEEE Trans Ind Electron*, 70(5):5034-5043.
https://doi.org/10.1017/9781316661239

List of supplementary materials

Fig. S1 Joint limit function

Table S1 Disturbances for each group in the simulations

Table S2 Comparative measurement analysis in the simulations

Table S3 Comparative measurement analysis in the pool experiments