A Complementary Framework for Human–Robot Collaboration With a Mixed AR–Haptic Interface

Xiangjie Yan†, Yongpeng Jiang†, Chen Chen†, Leiliang Gong†, Ming Ge, Senior Member, IEEE, Tao Zhang, Senior Member, IEEE, and Xiang Li, Senior Member, IEEE

Abstract—There is invariably a tradeoff between safety and efficiency for collaborative robots (cobots) in human–robot collaborations (HRCs). Robots that interact minimally with humans can work with high speed and accuracy but cannot adapt to new tasks or respond to unforeseen changes, whereas robots that work closely with humans can but only by becoming passive to humans, meaning that their main tasks are suspended and efficiency compromised. Accordingly, this article proposes a new complementary framework for HRC that balances the safety of humans and the efficiency of robots. In this framework, the robot carries out given tasks using a vision-based adaptive controller, and the human expert collaborates with the robot in the null space. Such a decoupling drives the robot to deal with existing issues in task space [e.g., uncalibrated camera, limited field of view (FOV)] and null space (e.g., joint limits) by itself while allowing the expert to adjust the configuration of the robot body to respond to unforeseen changes (e.g., sudden invasion, change in environment) without affecting the robot’s main task. In addition, the robot can simultaneously learn the expert’s demonstration in task space and null space beforehand with dynamic movement primitives (DMPs). Therefore, an expert’s knowledge and a robot’s capability are explored and complement each other. Human demonstration and involvement are enabled via a mixed interaction interface, i.e., augmented reality (AR) and haptic devices. The stability of the closed-loop system is rigorously proved with Lyapunov methods. Experimental results in various scenarios are presented to illustrate the performance of the proposed method.

Index Terms—Collaborative robots, global adaptive control, human demonstration, null-space interaction.

Manuscript received 18 May 2023; accepted 20 July 2023. Date of publication 16 August 2023; date of current version 29 December 2023. This work was supported in part by the Science and Technology Innovation 2030 through the Key Project under Grant 2021ZD0201404; in part by the National Natural Science Foundation of China under Grant U21A20517 and Grant 52075290; in part by the Guangdong–Hong Kong–Macao Innovation Center, Guangzhou, through the Special Foundation for Applied Research, under Grant ITPRD-2021-279; in part by the Beijing National Research Center for Information Science and Technology under Grant 20201880382; and in part by the InnoHK funding launched by the Innovation and Technology Commission, Hong Kong, SAR. Recommended by Associate Editor Y. Yildiz.

(Received author version: Xiang Li.)

Xiangjie Yan, Yongpeng Jiang, Chen Chen, Tao Zhang, and Xiang Li are with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: yanxj20@mails.tsinghua.edu.cn; jyp19@mails.tsinghua.edu.cn; chenche20@mails.tsinghua.edu.cn; taozhang@tsinghua.edu.cn; xiangli@tsinghua.edu.cn).

Leiliang Gong is with Hong Kong Productivity Council (HKPC), Hong Kong, China (e-mail: leiliang@hkpc.org).

Ming Ge is with the Hong Kong Industrial Artificial Intelligence and Robotics Centre (FLAIR), and Hong Kong Productivity Council (HKPC), Hong Kong, China (e-mail: mingge@hkflair.org).

Digital Object Identifier 10.1109/TCST.2023.3301675

A COBOT usually shares its workspace with humans and thus directly (i.e., physically) or indirectly interacts with humans. It usually has two features: enhanced safety and ease of programming [1], and thus, it can operate near humans and be deployed flexibly on various tasks. The development of cobots in recent decades has increased production efficiency in manufacturing industries, which had previously reached a near-maximum level, to a new higher level.

The tradeoff between safety and efficiency is always an open issue for cobots. To guarantee the safety of humans, cobots must typically suspend the ongoing task of end-effector [2] and become passive to a human’s control efforts, regardless of whether the human intervenes intentionally (i.e., an expert who wishes to lead the task) or unintentionally. Not until the human ceases intervening can the robot continue its task. This operational process may affect task efficiency because the robot needs to transit between different working modes.

To address the open issue, this article proposes a new complementary framework for human–robot collaboration (HRC). Specifically, an illustrative scenario is considered in Fig. 1. First, a cobot grasps the target object in an interactive environment containing human workers. Then, it transfers the object to a desired position in an isolated setting, excluding workers for safety or cleanliness reasons. Such a scenario is commonly seen in factories, such as chemical factories [3], food-processing factories [4], and flat-panel display factories [5]. Cobots in these factories must often overcome one or several of the following challenges:

1) human workers invading their workspace;
2) inexacty known environment, such that the relationship between their workspace and the sensory space is uncalibrated;
3) joint angles that are subject to several limits (e.g., singularity or constrained environment) and features that leave the field of view (FOV) during large displacements.

This article considers the aforementioned illustrative scenario and proposes a new framework for HRC. The main novelty is its complementarity, which enables effective exploitation of the capabilities of a human expert (i.e., fast responses and smart decision-making) and those of a robot (i.e., high repetition and continuous working) and hence achieves a better balance between safety and efficiency. The contributions of this article can be summarized as follows.

1063-6536 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.
Task-space control has now become a standard method applied to robot manipulators. When a robot working in task space is subjected to a large displacement, its global stability is commonly limited by several open issues, i.e., joint limits, limited FOV, and uncalibrated camera. First, a robot’s joint angles may be subjected to several limitations due to singular configurations and constraints’ limits. Many studies have developed methods to keep the robot away from these limits, e.g., by replanning the trajectory beforehand [7], exploring the kinematic redundancy [8], or damping the robot’s motion when it is near the limit [9].

Second, the problem of limited FOV occurs when the visual feature leaves it during the task. In [10], a switching approach was proposed to switch the control input between a backward motion outside the FOV and a visual servoing method within it. In [11], a new weighted feature was proposed for vision-based control to allow some features to leave the FOV during manipulation. In [12], multiple visual features were kept within the FOV by regulating both the mean and also the variance of multiple features. The visual servoing scheme in [13] sets the FOV as visibility constraints in the predefined performance bound, and the robot was controlled to achieve the desired transient response and hence stay within the FOV.

Third, the parameters of a camera deployed in a task-space control system may be unknown, for lacking prior calibration or being subject to adjustment (e.g., changes in focal length) when undertaking different tasks. Li et al. [14] proposed a series of adaptive laws to estimate the parameters of uncalibrated cameras and robot dynamics concurrently. Without estimating the unknown camera parameters, Liang et al. [15] used three feature points distributed in a particular pattern, such that only pixel feedback from a fixed uncalibrated camera was needed to perform stabilization control for a nonholonomic mobile robot.

To address the aforementioned issues together, Li and Cheah [16] proposed an adaptive task-space controller with the feedback switching among joint space, Cartesian space, and vision space, to achieve global stability within the whole workspace. Nevertheless, this and other existing task-space control schemes are commonly applicable to isolated environments, or their global stability is affected by the issues of joint limits, limited FOV, or uncalibrated sensors (e.g., [9]).

**B. Human–Robot Collaboration**

The scenario of HRC can be found in some manufacturing applications, where humans and robots perform a task together [1], [17], [18], [19].

As the robot coexists with humans, it is very important to guarantee safety. The safety standard for HRC systems is defined in ISO 10218-1 and ISO/TS 15066, which are now used as guidelines for many real-world applications. In [20], an objective-switching method was adopted in an assembly task, which balanced the safety and time efficiency when the robot was approaching and avoiding the coworkers, respectively. In [21], an optimization-based trajectory planning framework with an iterative online safety module was proposed for HRC. A model recovering human-exerted forces was developed for dyadic cooperative object manipulation in [22].

**II. RELATED WORKS**

**A. Task-Space Control**

Task-space control directly specifies a feature or goal in task space, e.g., Cartesian space or vision space. This eliminates the need to solve an inverse kinematic problem, and thus task-space control has now become a standard method applied to robot manipulators. When a robot working in task space is subjected to a large displacement, its global stability is commonly limited by several open issues, i.e., joint limits, limited FOV, and uncalibrated camera. First, a robot’s joint angles may be subjected to several limitations due to singular configurations and constraints’ limits. Many studies have developed methods to keep the robot away from these limits, e.g., by replanning the trajectory beforehand [7], exploring the kinematic redundancy [8], or damping the robot’s motion when it is near the limit [9].

Second, the problem of limited FOV occurs when the visual feature leaves it during the task. In [10], a switching approach was proposed to switch the control input between a backward motion outside the FOV and a visual servoing method within it. In [11], a new weighted feature was proposed for vision-based control to allow some features to leave the FOV during manipulation. In [12], multiple visual features were kept within the FOV by regulating both the mean and also the variance of multiple features. The visual servoing scheme in [13] sets the FOV as visibility constraints in the predefined performance bound, and the robot was controlled to achieve the desired transient response and hence stay within the FOV.

Third, the parameters of a camera deployed in a task-space control system may be unknown, for lacking prior calibration or being subject to adjustment (e.g., changes in focal length) when undertaking different tasks. Li et al. [14] proposed a series of adaptive laws to estimate the parameters of uncalibrated cameras and robot dynamics concurrently. Without estimating the unknown camera parameters, Liang et al. [15] used three feature points distributed in a particular pattern, such that only pixel feedback from a fixed uncalibrated camera was needed to perform stabilization control for a nonholonomic mobile robot.

To address the aforementioned issues together, Li and Cheah [16] proposed an adaptive task-space controller with the feedback switching among joint space, Cartesian space, and vision space, to achieve global stability within the whole workspace. Nevertheless, this and other existing task-space control schemes are commonly applicable to isolated environments, or their global stability is affected by the issues of joint limits, limited FOV, or uncalibrated sensors (e.g., [9]).

**B. Human–Robot Collaboration**

The scenario of HRC can be found in some manufacturing applications, where humans and robots perform a task together [1], [17], [18], [19].

As the robot coexists with humans, it is very important to guarantee safety. The safety standard for HRC systems is defined in ISO 10218-1 and ISO/TS 15066, which are now used as guidelines for many real-world applications. In [20], an objective-switching method was adopted in an assembly task, which balanced the safety and time efficiency when the robot was approaching and avoiding the coworkers, respectively. In [21], an optimization-based trajectory planning framework with an iterative online safety module was proposed for HRC. A model recovering human-exerted forces was developed for dyadic cooperative object manipulation in [22].
so that only human-applied force was measured to control the robot while safety was guaranteed. However, most of the existing works have assumed that the perception of humans or obstacles is fully reliable, lacking the ability to deal with suddenly appearing or unforeseen changes.

Various HRC interfaces have also been developed for human involvement. In [23], EMG and IMU sensors were adopted to assess the human motion intention during physical interaction. Face and gesture recognition were integrated into a collaborative system for assembly tasks [24]. Moreover, some fluency evaluation methods were proposed in [25]. Nevertheless, the aforementioned works are commonly limited to specific and predefined tasks. A general HRC interface in an industrial setting is the teach pendant; However, it was found that this kind of interface has decreased the efficiency of and experience gained by humans [26]. Therefore, there is a demand for a human-oriented, intuitive, and general interface for HRC that facilitates convenient human involvement in robot-assisted tasks.

C. DMP for Robot Learning

Among various techniques of learning-from-demonstration (LfD), DMP has been proven to be an effective and efficient approach [27], [28], [29]. In DMP, movement is modeled using a spring–damper system, with the addition of a nonlinear forcing term to encode and modulate learning skills [27]. The DMP method has also been extended to the Cartesian orientation [30], force adaptation [31], and arbitrary via-point adaptation [32] variations.

A typical LfD setting is to construct a teleoperation system, such that skills from the human expert (the primary side) can be transferred to the robot (the secondary side) and then adjusted via DMP according to the given task. As the primary side is usually mechanically different from the secondary side, several issues on how to record and reproduce the demonstration for better execution performance were raised [27], [28]. To better teach the robot via DMP, Beik-Mohammadi et al. [33] developed a simulation environment where the human expert demonstrates a task using an AR device, and then transfers the demonstrated skills to the robot using highly transparent feedback. In another approach [34], a teleoperation control interface was developed for bilateral teleoperation, which consists of a 3 degrees-of-freedom (DOFs) HapticMaster robot and a stiffness control handle, which allows human-in-the-loop teaching, and hence results in a better trajectory encoding.

In summary, DMP methods that mainly or solely focus on the task of the robot end-effector have been developed in existing works. However, for a complex task such as that illustrated in Fig. 1, it is necessary to regulate both the robot end-effector and the robot’s body shape to suit the position of the collaborating human expert, avoid collisions, and so on.

III. PRELIMINARIES

For the convenience of expression, the robot in the subsequent development is always referred to as the collaborative robot (i.e., “cobot”). First, this article considers a robot with redundant joints, whose forward kinematic model can be described as

\[ r = h(q) \]  

where \( r \in \mathbb{R}^6 \) denotes the position and the orientation of the robot end-effector in the Cartesian space, \( q \in \mathbb{R}^n \) is the vector of joint angles, \( n > 6 \) is the number of DOFs, and \( h: \mathbb{R}^n \to \mathbb{R}^6 \) is a nonlinear function.

Then, the velocity of the robot end-effector in the Cartesian space is related to the joint-space velocity as follows [35]:

\[ \dot{r} = J(q)\dot{q} \]  

where \( J(q) \in \mathbb{R}^{6\times n} \) is the Jacobian matrix from the joint space to the Cartesian space.

The pseudoinverse matrix is defined as

\[ J^+(q) \triangleq J^T(q)J(q)J^T(q)^{-1} \in \mathbb{R}^{n\times 6} \]  

such that \( J(q)J^+(q) = I_n \), where \( I_n \in \mathbb{R}^{n\times n} \) is an identity matrix. Accordingly, the null-space matrix can be introduced as follows [36]:

\[ N(q) \triangleq I_n - J^+(q)J(q) \in \mathbb{R}^{n\times n} \]  

where \( I_n \in \mathbb{R}^{n\times n} \) represents an identity matrix. Equation (4) means that \( J(q)N(q) = 0 \), \( N(q)J^+(q) = 0 \), and \( N^2(q) = N(q) \), which implies that the null-space matrix \( N(q) \) is orthogonal to the Jacobian matrix \( J(q) \).

Because collaborative robots are typically lightweight and operate at relatively low speeds, the control input can be specified at the kinematic level, such that

\[ \dot{\theta} = u \]  

where \( u \in \mathbb{R}^n \) denotes the control input corresponding to the joint-space velocity.

When a camera is used to measure the robot end-effector in the vision space, the feature’s velocity in vision space is related to the end effector’s velocity in the Cartesian space [37], i.e.,

\[ \dot{x} = J_x(r)\dot{r} \]  

where \( x \) denotes the feature’s position (which is the position of the robot end-effector) in the vision space, and \( J_x(r) \in \mathbb{R}^{2\times 6} \) is the image Jacobian matrix. Due to the limited FOV, the visual feature is not available when it is initially outside the FOV or when it temporarily leaves the FOV during manipulation.

If the camera is not calibrated beforehand or if its parameters are adjusted to suit new tasks (e.g., camera autofocus, depth variation), the exact knowledge of the camera parameters may not be available, and hence the image Jacobian matrix is unknown and denoted as \( J_x(r) \).

\textbf{Problem Formulation:} The aim of this study is to design the control input (5) to guarantee the global stability of the robot and the convergence of task-space error to zero, in the presence of joint limits, uncalibrated camera, limited FOV, and human involvement.

\textbf{Remark 1:} The proposed interaction framework does not heavily rely on human expertise. Instead, it is designed such that the expert can safely intervene at any time to use his/her
expertise. In scenarios where such expertise can be simplified (e.g., using external sensors to detect the appearance of workers) or predefined (e.g., specifying the goal position), the expert is not necessarily present.

IV. MULTIPLE REGIONAL FEEDBACK

This article considers the scenario illustrated in Fig. 1. The overall structure of complementary collaboration under such a scenario is shown in Fig. 2. That is, the human expert interacts with the robot via the mixed interface. When the robot is controlled to grasp the target object in the coexisting environment, the human expert exerts control efforts in the null space to avoid potential collisions with workers. When the robot learns to place the object at the desired position, the human expert simultaneously demonstrates the reference trajectory in both the Cartesian space and null space. Note that the proposed structure can also be extended to many other scenarios involving human–robot interaction.

This section presents the regional feedback [16] for the grasping operation, which is used to solve problems (e.g., joint limits and limited FOV) that may arise during large-displacement transfers. Thus, a series of regional feedbacks are exploited to keep the robot in a displacement transfers. Thus, a series of regional feedbacks are used to drive the robot to move toward the feature, which can be treated as a repulsive force to keep the robot away from the limited configurations. The vector automatically if it is very close to the region boundary $f_i(q) = 0$. Hence, the second term in (8) is to decelerate the robot in advance and hence to alleviate the potential oscillation, where $k_{ri}$ is relatively small.

Now, a regional feedback vector can be specified in the joint space as

$$\xi_q = \frac{\partial P_i(q)}{\partial q}$$

which can be treated as a repulsive force to keep the robot away from the limited configurations. The vector automatically reduces to zero when the robot is outside the joint-space region.

B. Cartesian-Space Feedback

When the feature is not within the FOV, the Cartesian-space feedback is used to drive the robot to move toward the feature, such that the feature can be seen inside the FOV. Then, a region is formulated in the Cartesian space as

$$f_c(r) = \begin{bmatrix} f_{c1}(r_1) \\ f_{c2}(r_2) \\ f_{c3}(r_3) \end{bmatrix} = \begin{bmatrix} \left( \frac{r_1 - r_{c1}}{c_1} \right)^2 - 1 \\ \left( \frac{r_2 - r_{c2}}{c_2} \right)^2 - 1 \\ \left( \frac{r_3 - r_{c3}}{c_3} \right)^2 - 1 \end{bmatrix} \leq 0$$

where $k_{qi}$ and $k_{ri}$ are positive constants, and $f_{ri}(q) \leq 0$ is a reference region enclosing $f_i(q) \leq 0$.

The first term in (8) is to create a high potential energy barrier, such that the robot does not have enough kinetic energy to approach the limited configurations. Hence, $k_{qi}$ is set large to make the gradient of $P_i(q)$ steep. However, the steep gradient would cause oscillatory movement of the robot if it is very close to the region boundary $f_i(q) = 0$. Hence, the second term in (8) is to decelerate the robot in advance and hence to alleviate the potential oscillation, where $k_{ri}$ is relatively small.

Now, a regional feedback vector can be specified in the joint space as

$$\xi_q = \frac{\partial P_i(q)}{\partial q}$$

which can be treated as a repulsive force to keep the robot away from the limited configurations. The vector automatically reduces to zero when the robot is outside the joint-space region.
Next, the corresponding potential energy function for the above region is proposed as
\[
P_o(r) = \sum_{i=1}^{3} \frac{k_{ci}}{2} \left[ \max(0, f_{ci}(r)) \right]^2 \tag{11}
\]
where \(k_{ci}\) are positive constants. An illustration of the potential energy function is shown in Fig. 3(b). From (11) and Fig. 3(b), it can be seen that the potential energy drives the robot end-effector to enter the region where \(f_{ci}(r) \leq 0\) (which is also inside the FOV) and then reduces to zero.

However, the pose of the robot end-effector may not be suitable for grasping if only the position is regulated. To address the problem, another Cartesian-space region is introduced to control the orientation of the robot end-effector, i.e.,
\[
f_o(r) = \alpha_o \left[ \log \left( p \cdot p^{-1} \right) \right]_2 - 1 \leq 0 \tag{12}
\]
where \(\alpha_o\) is a positive constant which is related to the tolerance of orientation error, \(p_{gi}\) and \(p\) are the quaternions representing the goal and the robot end-effector, respectively, \((\cdot)\) denotes the Hamilton product, and \(\log()\) describes the quaternion logarithm. The use of quaternions avoids the representation singularity. A simple example for the orientation region (12) can be given as: \(\|\log(p \cdot p_{gi}^{-1})\|_2\), which describes the distance between \(p\) and \(p_{gi}\).

Similarly, the corresponding potential energy function is formulated as
\[
P_o(r) = \frac{1}{2} k_o \left[ \max(0, f_o(r)) \right]^2 \tag{13}
\]
where \(k_o\) is a positive scaling factor. The overall potential energy function in the Cartesian space is the sum of \(P_i(r)\) and \(P_o(r)\), i.e.,
\[
P_i(r) = P_i(r) + P_o(r) \tag{14}
\]

Next, the regional feedback vector is specified in the Cartesian space as
\[
\xi_r = \frac{\partial P_i(r)}{\partial r} = \frac{\partial P_i(r)}{\partial r} + \frac{\partial P_o(r)}{\partial r} \tag{15}
\]
which can be treated as an attractive force that drives the robot end-effector to enter the Cartesian-space regions, such that the end-effector becomes visible and its orientation can be adjusted to a configuration suitable for grasping.

C. Vision Feedback

The vision feedback is used by the robot end-effector to grasp the target object. First, a region function is specified in the vision space as
\[
f_v(x) = \begin{bmatrix} f_{v1}(x) \\ f_{v2}(x) \end{bmatrix} = \begin{bmatrix} \left( \frac{x_1 - x_d1}{b_1} \right)^2 - 1 \\ \left( \frac{x_2 - x_d2}{b_2} \right)^2 - 1 \end{bmatrix} \leq 0 \tag{16}
\]
where \(b_1, b_2 > 0\) are constants representing the half size of the FOV in the coordinates of \(x_1\) and \(x_2\), respectively, and \(x_d = [x_d1, x_d2]^T \in \mathbb{R}^2\) is the desired position, which is also the position of the target object in the vision space.

Accordingly, the potential energy function in the vision space is introduced as
\[
P_v(x) = \sum_{i=1}^{2} \left\{ \frac{k_v}{2} \left[ 1 - \min(0, f_{vi}(x))^2 \right] \right\} \tag{17}
\]
where \(k_v\) is a positive constant. The potential energy is shown in Fig. 4; its gradient is zero outside the region (i.e., \(f_v(x) > 0\)) and nonzero inside the region, which drives the robot end-effector to converge to the desired position for grasping.

The regional feedback vector in the vision space can now be specified in a similar way as
\[
\xi_x = \frac{\partial P_v(x)}{\partial x} \tag{18}
\]
which is activated inside the vision-space region where \(f_v(x) \leq 0\) (which is also inside the FOV). To ensure that the robot end-effector can move from the Cartesian-space region to the vision-space region, the Cartesian-space region can be set smaller than the corresponding FOV.

The connection between different types of feedback is illustrated in Fig. 5, where the use of multiple local regional feedback ensures the global movement within the whole workspace, e.g., from the initial position (blue triangle) to the goal position (red diamond).

Remark 2: The role of the Cartesian-space region is to drive the robot to move from outside to inside the FOV. In this sense, when the region is projected onto the image plane, it is not necessary to match the rectangular shape of the FOV. As long as the projected region is fully inside the FOV, the feedback can smoothly transit from the Cartesian-space information to the vision. In addition, specifying the Cartesian-space region...
is a one-time setup at the very beginning. Even if the worker may place the object at different positions inside the region every time, the robot can use the vision feedback to exactly approach the target object.

V. Vision-Based Global Adaptive Control

This section presents the global adaptive controller with multiple regional feedback, which drives the robot to interact with humans and grasp the target object in the presence of joint limits, limited FOV, and uncalibrated cameras. Specifically, the control input is proposed as

$$u = -J^+(q) \left( J_x^T(r) \hat{\xi}_x + \xi_r \right) + N(q) c_d^{-1} (d - \xi_q)$$

where $c_d$ is a positive scalar, and $d \in \mathbb{R}^n$ denotes the control efforts exerted by the human expert on robot joints via mixed interfaces (e.g., Microsoft HoloLens 2 in [38]). The first term on the right side of (19) is to drive the end-effector to carry out the main task in the vision space, and the second term is to regulate the redundant joints in the null space to collaborate with the expert and also avoid joint limits, without affecting the main task. The objective of the null-space control term can also be described as a damping model, i.e.,

$$N(q) (c_d \dot{q}) = N(q) (d - \xi_q)$$

where $c_d$ can be considered as the desired damping parameter.

Next, the entry of the unknown image Jacobian transpose is approximated with an adaptive neural network (NN) as

$$\hat{J}_i(r)_{i,j} = \hat{w}_{i,j}^T \theta(r)$$

where $\hat{J}_i(r)_{i,j}$ is the $(i,j)$th entry of the matrix $\hat{J}_i^T(r)$, $i = 1, 2, \ldots, m$, $j = 1, 2, \hat{w}_{i,j} \in \mathbb{R}^{n_k}$ is the corresponding weight, and $\theta : \mathbb{R}^m \rightarrow \mathbb{R}^{n_k}$ is the nonlinear function of neurons. Radial basis function (RBF) is used as the neuron, where the $i$th entry is

$$\theta_i(r) = \exp \left( - \frac{1}{2\sigma_i^2} \| r - c_i \|_2^2 \right)$$

where $c_i$ and $\sigma_i^2$, $i = 1, \ldots, n_k$ are the centers and the variances, respectively. These parameters are manually predefined.

For simplicity, we rewrite (21) in the following vectorized form:

$$\text{vec} \left( J^T_x(r) \right) = \hat{W} \theta(r)$$

where $\hat{W} = [\hat{w}_{1,1}^T; \ldots; \hat{w}_{m,1}^T; \hat{w}_{1,2}^T; \ldots; \hat{w}_{m,2}^T] \in \mathbb{R}^{2n \times n_k}$

Thus, the weight of NN is updated with the following online adaptation law:

$$\dot{\hat{W}} = \left[ L \theta(r) \left( J^T_x(r) \xi_x + \xi_r \right) \xi_x^T \right]^T$$

where $L \in \mathbb{R}^{n_k \times n_k}$ is a positive-definite matrix, and $\xi_x^T$ is a matrix that reformulates the entries of $\xi_x = [\xi_{x1}, \xi_{x2}]^T$ as

$$\xi_x^T = [\xi_{x1} I_m, \xi_{x2} I_m] \in \mathbb{R}^{m \times 2m}$$

which has the following property:

$$\xi_x^T \text{vec} \left( J^T_x(r) \right) = J^T_x(r) \xi_x.$$  \hspace{1cm} (26)

The advantages of the proposed control scheme (19) are summarized as follows.

1) When the robot nears the joint limits, the regional feedback vector $\xi_x$ is activated to drive the robot away.

2) The regional feedback vector $\xi_x$ is used to drive the robot end-effector to approach the desired position, such that both the feature and the desired position can be seen by the camera.

3) The regional feedback vector $\xi_x$ is activated only when both the feature and the target object are visible, such that the robot can grasp the target object in the presence of uncalibrated cameras.

4) The online adaptation is driven by the regional feedback to deal with the unknown parameters concurrently.

By substituting (19) into (5), the closed-loop equation is obtained as

$$\ddot{q} = -J^+(q) \left( J_x^T(r) \hat{\xi}_x + \xi_r \right) + N(q) c_d^{-1} (d - \xi_q).$$  \hspace{1cm} (27)

A. Null Space

Multiplying both sides of (27) by $N(q)$ and noting that $N(q) J^+(q) = 0$ and $N^2(q) = N(q)$, we have

$$N(q) \ddot{q} = N(q) c_d^{-1} (d - \xi_q)$$

that is,

$$N(q) (c_d \ddot{q} - d + \xi_q) = 0$$

which maps the desired damping model $c_d \ddot{q} = d - \xi_q$ into the null space of the Jacobian matrix, such that both the expert’s control efforts $d$ and the joint-space regional feedback $\xi_q$ work without affecting the robot end-effector.

We are now in a position to state the following statement.

B. Statement

If $d$ and $\xi_q$ do not work at the same time, $\xi_q$ does not lie in the null space of $N(q)$, then the human expert can intervene in the null space, i.e., $N(q) (c_d \ddot{q}) = N(q) d$, and the robot will eventually stay outside the joint region, i.e., $\xi_q \rightarrow 0$ as $t \rightarrow \infty$. 

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
This statement can be concluded by the following discussions.

1) When the robot leaves the joint-space region and hence stays away from joint limits or singularity, $\xi_q = 0$, (29) becomes $N(q)(c_q \dot{q}) = N(q) d$, such that the motion of redundant joints is solely determined by the expert.

2) When the robot is inside the joint-space region and the expert does not input control efforts, $d = 0$, (29) becomes $N(q)(c_q \dot{q}) = N(q) $ξq$. Although it is possible that $\xi_q \neq 0$ but $N(q)\xi_q = 0$, those cases are very rare. Because when the robot is not exactly located at the singular configuration, rank($N(q)$) = 1, $N(q) = a b^T$, where $a, b \in \mathbb{R}^7$ are vectors. That is, the aforementioned cases occur only when $\xi_q$ is exactly orthogonal to $b$, which is very rare in actual implementation. Hence, the robot will not stay inside the joint-space region (i.e., $\xi_q \neq 0$ and $\dot{q} = 0$) and will leave it by the end, i.e., $\xi_q \equiv 0$.

Hence, the control objective in null space is realized, that is, the robot reacts to the human’s interaction while avoiding the joint limits.

C. Task Space

Multiplying both sides of (27) by $J(q)$ and noting that $J(q) \dot{q} = \dot{r}$ and $J(q)N(q) = 0$, we have

$$\dot{r} = -\hat{J}_s r^T \xi_x - \xi_r.$$  (30)

We are now in a position to state the following theorem.

Theorem 1: The closed-loop system (27) gives rise to the global stability of the robot and also the convergence of $(\hat{J}_s r^T \xi_x + \xi_r) \rightarrow 0$ as $t \rightarrow \infty$.

Proof: See Appendix I.

From the above theorem, the convergence of $(\hat{J}_s r^T \xi_x + \xi_r) \rightarrow 0$ is obtained at steady state. When the regional feedback vector $\xi_q$ keeps the robot away from the joint limits, $\xi_q = 0$. Then, the regional feedback vector $\xi_q$ drives the robot to move from outside to inside the FOV. After it enters the FOV, the regional feedback vector $\xi_q$ is activated, and it reduces to zero only when the robot has reached the desired position (to grasp the target object). Hence, the convergence of $(\hat{J}_s r^T \xi_x + \xi_r) \rightarrow 0$ actually implies that the robot end-effector moves to and stays at the desired position, in the presence of limited FOV and uncalibrated camera.

VI. ROBOT LfD

After the robot grasps the object, it transfers the object to the desired position in an isolated environment, as illustrated in Fig. 2. The desired position and the trajectory to this position are learned from human demonstration via the DMP approach. Such a proposed formulation has the following advantages.

1) LfD can effectively exploit the expert’s knowledge to set the desired position (e.g., a specific grasping pose) and suit the constrained space (e.g., path planning in the environment in Fig. 1).

2) DMP allows the learned trajectory to be conveniently adjusted in response to the task generalization (e.g., modified goal positions on other shelves of the cabinet, or modified reaching speeds).

The proposed LfD method is illustrated in Fig. 6. Basically, the DMP model for learning the trajectory of a single joint can be described as

$$\tau \ddot{z} = \alpha z [\beta \theta (g - q) - \tau \dot{q}] + \xi(z)$$  (31)

$$\tau \dot{z} = -\alpha z \dot{z}$$  (32)

where (31) specifies a transformation system; (32) describes a canonical system; $\tau$ is a positive time constant, $q, \dot{q}, \ddot{q}$ represent the angle, angular velocity, and acceleration of the joint, respectively; and $\alpha, \beta, \alpha, \epsilon$ are positive gain constants.

In addition, $\xi(z)$ is the forcing term, which is formulated as a linear combination of nonlinear basis functions, i.e.,

$$\xi(z) = \sum_{i=1}^{N} \psi_i(z) \alpha_i$$  (33)

where $\omega_i$ is the weight of the $i$th basis function, $N$ is the total number of basis functions, $q_0$ is the initial position at $t = 0$, and $\psi_i(z)$ is the $i$th basis function. The latter is chosen as an RBF to allow discrete movement, i.e.,

$$\psi_i(z) = \exp \left[ -\frac{1}{\sigma_i^2} (z - c_i)^2 \right]$$  (34)

where $c_i$ and $\sigma_i^2$ denote the center and the variance, respectively. The generation of a learned trajectory consists of two steps: a learning phase and a reproducing phase.

A. Learning Phase

First, the human expert demonstrates a trajectory in terms of $[\dot{q}_{\text{demo}}(t), \ddot{q}_{\text{demo}}(t), \dot{q}_{\text{demo}}(t)]^T = 0$. By referring to this demonstration, a desired forcing term can be calculated by transposing (31) as

$$\xi_d = \tau^2 \dot{q}_{\text{demo}} - \alpha \beta [\beta \dot{q} (g - q_{\text{demo}}) - \tau \dot{q}_{\text{demo}}].$$  (35)

Next, the locally weighted quadratic error is defined as the optimization target, i.e.,

$$\text{Cost}_i = \sum_{t=1}^{T} \psi_i(z(t)) [\xi_d(t) - \omega_i z(t)(g - q_0)]^2$$  (36)

which forms a standard weighted linear regression problem, with the solution as

$$\omega_i = \frac{s_{\text{T}} \Gamma_i \xi_d}{s_{\text{T}} \Gamma_i s}$$  (37)

where $i = 1, \ldots, N$, and

$$s = [z(1)(g - q_0), z(2)(g - q_0), \ldots, z(T)(g - q_0)]^T$$  (38)

$$\Gamma_i = \text{diag} (\psi_i(1), \psi_i(2), \ldots, \psi_i(T))$$  (39)

$$\xi_d = [\xi_d(1), \xi_d(2), \ldots, \xi_d(T)]^T.$$  (40)
Fig. 6. Expert uses the proposed AR–haptic interface bimanually to carry out the demonstration in both the task space (for the end-effector) and null space (for redundant joints) in a simultaneous manner, and the demonstration is recorded and then parameterized with the DMP, such that it can be further modulated to suit other tasks.

B. Reproducing Phase

After the weights in the forcing term are learned, a new trajectory can now be generated by running (31) and (32). The learned trajectory can be modulated according to the given scenario. Specifically, $\tau$ can be adjusted to speed up or slow down the trajectory execution, and $g$ can be changed to set a new goal position while maintaining a similar transient movement to that position.

Remark 3: In this article, the human expert demonstrates the trajectory in both the task space of the robot end-effector and the null space of it in a bimanual and simultaneous way via the mixed interface, as illustrated in Figs. 1 and 2. The demonstration teaches the robot to not only find the correct desired pose in task space but also shape the overall body to avoid collisions in a constrained environment (which is also one of the improvements with respect to the conference version [6]).

Remark 4: The proposed framework can correct unintentional mistakes by the human expert during the demonstration. Specifically, if the learned Cartesian goal position is wrong, a new and correct Cartesian position can then be directly imposed in the reproduced trajectory by setting the parameter of $g$ in (35). If the learned trajectory to the Cartesian goal position is wrong, the robot can either refer to an external sensor (e.g., an RGBD camera) to detect the changes and then adjust its trajectory accordingly or add one or multiple waypoints [32] into the learned trajectory to suit the new environment. See Appendix II-A.

VII. EXPERIMENT

Experiments were conducted on a robotic manipulation system to validate the proposed method, as shown in Fig. 7(a). The overall system consisted of five modules: 1) a PC with Robot Operating System (ROS) [39] and Ubuntu 18.04 LTS, in which the algorithm was implemented; 2) a 7-DOF Franka robot with a two-fingered gripper, with ArUco markers attached to the gripper and the objects to aid perception [see Fig. 7(b)]; 3) a Basler ace acA1440-220uc camera with 1440 × 1080 resolution, which was fixed in the workspace of the robot but not calibrated; 4) an Omega 3, which is a haptic interface developed by Force Dimension; and 5) a HoloLens 2, which is a head-mounted AR device. Items 4) and 5) comprised a mixed interface, which enabled the expert to interact with the robot in both the task space and null space in a bimanual way.

The AR device allowed the human expert to exert control efforts on the robot manipulator in two ways: by pulling the virtual robot closer and directly manipulating a specific joint [see Fig. 7(d)] or by making the virtual robot overlap with the real robot and then using hand-ray and air-tap gestures to control the real robot remotely [see Fig. 7(e)]. These control efforts are represented as a vector [visualized in Fig. 7(d) and (e)] that is converted into a command velocity proportionally. Then, the control efforts $d$ in 19 are injected by projecting the velocity back to the joint space using the pseudoinverse of the Jacobian matrix of the selected joint. The method used to calculate $d$ is the same as that used in [6]. In addition, the AR device allowed the human expert to conveniently specify the Cartesian-space region [see Fig. 7(f)] by simply drawing a virtual region. The goal location in the placing task can also be set using hand-ray. In this article, the proposed control scheme is implemented at the kinematic level (i.e., the velocity input) as the robot moved at a relatively low velocity for HRC. The control algorithm is developed based on Franka interface and Frankapy control stack [40].

A user study with ten participants was conducted to evaluate the performance of the proposed mixed interface. The participants had no prior experience with the operation. During the experiment, they were required to use different interfaces to control the robot to place a grasped object onto the desired position inside the cabinet. Then, they were asked to rate the user experience of interfaces, including: 1) keyboard; 2) haptic-only device; and 3) the proposed mixed AR–haptic interface.

The experimental results are shown in Table I, where “success rate” is assessed in the task where an object is placed onto the desired position, “collision avoided” is used to describe the collision with the environment, “time” is the average
TABLE I

| index               | success rate | collision avoided | time | user experience |
|---------------------|--------------|-------------------|------|----------------|
| keyboard            | 20/20        | 17/20             | 38.8s| 6.2            |
| haptic-only device  | 20/20        | N.A.              | 25.5s| 8.0            |
| AR–haptic mixed interface | 20/20        | 20/20             | 25.7s| 8.1            |

Fig. 8. Statistical results of user study. (a) Completion time. (b) User experience (0–10).

completion time, and “user experience” is the average score from 0 to 10 (where higher score implies better experience). In addition, statistical results are given in Fig. 8.

While using the keyboard can successfully place the object onto the desired position, it is with the longest time and the lowest user-experience score. This is mainly because it provides a limited collection of specific movement commands only (e.g., move forward/downward), which is not flexible to realize the arbitrary motion of the robot. Such limitations also result in three failure cases during the obstacle avoidance. Next, the haptic-only device is time-efficient, but it lacks the function of redundant-joint adjustment and is hence not applicable to collision avoidance in a constrained environment. As a comparison, the AR–haptic mixed interface can achieve the 100% success rate for both object placement and collision avoidance, with a short operation time and high user experience score. The results prove that the proposed mixed interface is more effective and intuitive than others.

The following three experimental tasks were carried out to illustrate the performance of the proposed method. The purposes of the experiments are detailed as follows.

1) Placing Task: To demonstrate how the robot learned the desired trajectory in a complex environment.
2) Grasping Task: To validate the effectiveness of the global adaptive controller in the presence of a large-scale transition, an uncalibrated camera, and joint limits.
3) Collaboration Task: To illustrate the entire pipeline, in which the robot transferred an object from an interactive environment (i.e., humans coexist) to an isolated environment. Such a task is commonly performed in many factories, e.g., the transfer of hazardous chemicals.

A. Placing Task

In Experiment 1, the robot was already grasping an object and it was manipulated to place the object at a goal position on a shelf. The trajectory to the goal position was learned from human demonstration via the DMP method and the mixed interface (i.e., Omega 3 and HoloLens 2).

The goal position on the shelf is shown in Fig. 9. It was insufficient to define only the position and orientation of the robot end-effector, as an inappropriate body shape would result in the robot colliding with the cabinet (e.g., the collision of the fourth joint with the cabinet in Fig. 9). To solve this problem, the human expert used HoloLens 2 to define the motion of the redundant joint and thus “pulled” the joint away from the cabinet. During the demonstration, the goal position was on the second shelf of the cabinet. The reproduction results shown in Fig. 10 indicated that the robot reached the goal position by following the learned trajectory at a higher speed.

The goal position of the learned trajectory can be set to positions on other shelves such that the grasped object is placed there accordingly. Note that the robot’s movement was learned and reproduced in joint space, where the corresponding joint angles were computed based on the analytical inverse kinematics method [41].
From the joint motions illustrated in Fig. 11, the goal positions of the learned trajectories were successfully changed to other ones (i.e., the positions on the lower or higher shelves). These new trajectories were implemented in the subsequent task (see Section VII-C) to place multiple objects on different shelves without colliding with the cabinet.

B. Grasping Task

In Experiment 2, the robot started from a remote initial position to grasp a target object, as shown in Fig. 12(a). The proposed global adaptive controller defined by (19) and (24) was implemented to drive the robot to the object’s position for grasping, in the presence of joint limits, uncalibrated camera, and limited FOV.

Multiple regions were specified in the workspace of the robot: the joint-space region (7), the Cartesian-space region (10) and (12), and the vision region (16). The joint-space region was introduced to prevent the robot from approaching singular configurations or joint limits, which could have resulted in an emergency stop. The vision region was specified to cover the position of the target object, such that the robot was able to use the visual feedback for grasping. The Cartesian-space region was defined to dominate the remaining workspace of the robot, to ensure a smooth transition between different feedback. Specifically, the position region in the Cartesian space was defined by the human expert via the AR interface (based on the approximate location of the FOV), and the quaternion region in the Cartesian space was predefined. The combination of all the regional feedback guaranteed the movement of the robot within the whole workspace.

According to the singular configurations of the Franka robot [42], the joint-space regions are formulated as

\[ f_{1,\text{sing}}(q) = q_3^2 + (q_3 - \pi/2)^2 - R_1^2 \leq 0 \]  
\[ f_{2,\text{sing}}(q) = q_2^2 + (q_3 + \pi/2)^2 - R_2^2 \leq 0 \]  
\[ f_{3,\text{sing}}(q) = q_5^2 - R_3^2 \leq 0 \]  
\[ f_{4,\text{sing}}(q) = q_6^2 + (q_3 - \pi/2)^2 - R_4^2 \leq 0 \]  
\[ f_{5,\text{sing}}(q) = q_6^2 + (q_3 + \pi/2)^2 - R_5^2 \leq 0 \]  
\[ f_{6,\text{sing}}(q) = q_6^2 + (q_3 + \pi/2)^2 - R_6^2 \leq 0 \]

where \( R_1, R_2, \ldots, R_6 \) are positive constants, specifying the size of regions. The reference regions \( f_{i,\text{sing}}(q) \) for singularities are set similarly with constants \( R_1, R_2, \ldots, R_6 \). Besides, joint regions for avoiding joint limits are set as

\[ f_{i,\text{lim}} = (q_i - q_{\text{min}/\text{max}})^2 - R_i^2 \leq 0 \]  
\[ f_{i,\text{lim}} = (q_i - q_{\text{min}/\text{max}})^2 - R_i^7 \leq 0, \quad i = 1, \ldots, 7. \]  

Other parameters of region function and potential energy are listed in Table II. The initial image Jacobian \( J_a(r)_{l=0} \) was set randomly. In addition, the human expert did not exert additional control efforts in this experiment, and hence \( d = 0 \).

In the beginning, the human expert specified the size and the position of the Cartesian-space region \( (10) \) via the mixed interface to ensure that it was within the FOV [see Fig. 12(a)]; Then, the Cartesian-space feedback was used to transit from outside to inside the FOV. Subsequently, the vision feedback became available [see Fig. 12(b)] and was used to drive the end-effector to aim at the target object [see Fig. 12(c)]; Finally, the robot grasped the object and moved back to its home position to complete the task [see Fig. 12(d)].

In this experiment, the camera was not calibrated beforehand, and hence the exact information about the image Jacobian matrix was unknown. Thus, the adaptive NN was implemented to estimate the image Jacobian matrix via the online update law (24). The results with NN adaptation (where \( L \neq 0 \)) are shown in Fig. 13. The control input with NN adaption achieved a faster convergence than without it [see Fig. 13(b)]. The trajectory of the end-effector in 3-D space is shown in Fig. 14, proving a smooth transition of the robot among different regions.

Comparisons have also been performed in the trajectory-tracking task, and the results are shown in Fig. 15. The proposed adaptive control ensures both faster convergence and smaller tracking errors. The experiments for each trajectory were repeated ten times, and the statistical results are shown in Fig. 16, where the errors with adaptation are also smaller than those without it.

C. Collaboration Task

In Experiment 3, the robot conducted both grasping and placing tasks. The robot transferred an object from an interactive environment to an isolated environment while also collaborating with the human expert in the interactive environment. Such a mixed scenario is common in factories. For example, a worker hands over objects to a robot, which then transfers the objects to an unmanned laboratory where potentially hazardous experiments are conducted.

To fulfill the requirements, both the proposed learning scheme and the global adaptive controller were implemented and activated in different environments (i.e., the isolated and interactive environments in Fig. 1). The whole task of grasping–placing was performed three times in succession.
Fig. 12. Experiment 2—snapshots: the robot started from a remote initial position to grasp the target object in the presence of joint limits, uncalibrated camera, and limited FOV. The Cartesian-space region was defined by the expert via the AR interface and represented as a transparent cube with gray edges. (a) $t = 0.0$ s: the initial configuration. (b) $t = 5.8$ s: after the robot entered the Cartesian-space region, the marker appeared in the FOV. (c) $t = 9.2$ s: the robot used the visual feedback to adjust its pose to aim at the target object. (d) $t = 15.0$ s: the robot grasped the object and then returned to the home pose.

### Table II

**Table II**

| Control Parameters in Experiment 2 |
|------------------------------------|
| **Joint-Space Region** (8)         |
| $k_1, k_r$                         |
| $R_{ij}, R_{r,i}$ in (41)-(48)     |
| 10, 1                              |
| 0.1, 0.3                           |
| **Position Region in Cartesian Space** (10)(11) |
| $r_c$ (center)                     |
| $k_{c_1}, k_{c_2}, k_{c_3}$       |
| $c_1, c_2, c_3$ (size)            |
| by expert                         |
| $[4e - 4, 4e - 4, 4e - 5]$T        |
| by expert                         |
| **Orientation Region in Cartesian Space** (12)(13) |
| $\alpha_\theta$                   |
| $p_\theta$                        |
| $k_\theta$                        |
| 15                                |
| $[-0.28, 0.63, 0.66, 0.28]$T       |
| **Vision Region** (16)(17)         |
| $x_d$ (target object)              |
| $b_1, b_2$                        |
| $k_v$                              |
| by detection                      |
| $[1440, 1080]$                    |
| 0.3                               |
| **Human Control Input** (19)       |
| $c_d$                              |
| 3                                 |
| **Adaptive NN** (22),(23),(24)     |
| $c_i$                              |
| $\sigma$                          |
| $L$                               |
| $\tilde{W}_{t=0}$                 |
| $0.05, 0.35$T                     |
| $0.1$                             |
| $0.25I_9$                         |
| calculated according to $\tilde{J}, (r)_{t=0}$ |

Fig. 13. Experiment 2—results. (a) Path of the robot end-effector in the vision space. (b) Position errors in the vision space.

Fig. 14. Experiment 2—the path of the robot end-effector in 3-D space. i.e., three objects (with random initial positions within the FOV) were transferred to different shelves [see Fig. 7(b)].

For the grasping task, the positions of target objects were detected in the vision space and then set as the desired positions for the controller, as shown in Fig. 17(b). Every time a new object was to be grasped, the image Jacobian matrix was varied but well-estimated with the adaptive NN (24).

For the placing task, the robot followed the learned trajectories in Experiment 1. The trajectories had different goals
Fig. 16. Trajectory-tracking results: box plot of tracking errors.

Fig. 17. Experiment 3—results. (a) Trajectories demonstrated by human experts and reproduced with DMP. (b) Path of the ArUco marker (i.e., the feature of the target object) in the vision space. The initial position was at \( x_2 > 0 \) as the marker was occluded in the beginning.

Fig. 18. Experiment 3—snapshots. (a) \( t = 0.0 \) s: the initial configuration. (b) \( t = 3.4 \) s: the worker placed the object on the workbench and the robot moved to grasp the object. (c) \( t = 15.8 \) s: the worker had to collect markers from the table in an awkward position due to obstruction by the robot. (d) \( t = 26.3 \) s: the expert used the mixed interface to drag the fourth joint of the virtual robot forward and backward to adjust the body shape of the real robot, which guaranteed the worker’s safety and comfort. (e) \( t = 68.4 \) s: the robot transferred the object and placed it on a shelf. (f) \( t = 80.1 \) s: the worker placed another object at a different position on the workbench, and the robot moved to grasp and then placed the object again.

Snapsots of the experiment are shown in Fig. 18. From \( t = 0 \) to 10 s, the robot moved to grasp the first object. Meanwhile, the worker approached the robot and attempted to collect the scattered markers. The worker had to conduct the task in an awkward position. From \( t = 19 \) to 26 s and from \( t = 42 \) to 46 s, the human expert observed the situations and dragged the fourth joint of the virtual robot forward and backward to adjust the body shape of the real robot, which guaranteed the worker’s safety and comfort. The control efforts exerted by the expert are shown in Fig. 19, which were projected into the null space; thus, they did not affect the main task (translation and orientation) of the end-effector. Hence, the collaboration between the human expert and the robot was efficient, and the human expertise in the unstructured environment effectively complemented the robot’s large-scale transition ability.

All the experimental results can be found at https://youtu.be/zY3aPHQEx0E. Specifically, the video also shows that both the robot and the worker can perform tasks simultaneously in the shared space without affecting each other.

Fig. 19. Experiment 3—the translation (top) and orientation (middle) of the end-effector, and the human control efforts (bottom). The shaded area denotes the period when the human expert was exerting control efforts.

VIII. CONCLUSION

This article develops a new framework for HRC, where the main novelty is its complementary feature. This enables a human expert and a robot to collaborate in a more efficient way. Specifically, a new vision-based adaptive controller is proposed for the robot to ensure the global convergence of the end-effector, in the presence of joint limits, uncalibrated camera, and limited FOV. A mixed AR–haptic interface is developed to allow the expert to perform a demonstration in the task space and redundant joint space and perform collaboration to deal with unforeseen changes (e.g., suddenly appearing human walkers), without affecting the main task. Therefore, the proposed framework enables the robot to safely interact with other coexisting workers, in parallel to its ongoing work, and it also provides a natural and intuitive way for body and the shelves whenever a new object was being placed. The experimental results are shown in Fig. 17, which confirms that the consecutive grasping–placing was successfully realized.

Snapshots of the experiment are shown in Fig. 18. From \( t = 0 \) to 10 s, the robot moved to grasp the first object. Meanwhile, the worker approached the robot and attempted to collect the scattered markers. The worker had to conduct the task in an awkward position. From \( t = 19 \) to 26 s and from \( t = 42 \) to 46 s, the human expert observed the situations and dragged the fourth joint of the virtual robot forward and backward to adjust the body shape of the real robot, which guaranteed the worker’s safety and comfort. The control efforts exerted by the expert are shown in Fig. 19, which were projected into the null space; thus, they did not affect the main task (translation and orientation) of the end-effector. Hence, the collaboration between the human expert and the robot was efficient, and the human expertise in the unstructured environment effectively complemented the robot’s large-scale transition ability.

All the experimental results can be found at https://youtu.be/zY3aPHQEx0E. Specifically, the video also shows that both the robot and the worker can perform tasks simultaneously in the shared space without affecting each other.
the expert to deliver his/her knowledge and smart decision. The global stability of the closed-loop system is rigorously proved with Lyapunov methods, and the performance of the proposed scheme is validated in a series of transferring tasks in the hybrid environment (i.e., interactive and isolated). Future works will be devoted to marker-free perception and field application in factories.

APPENDIX I

First, a Lyapunov-like candidate is proposed as

\[ V = P_v(x) + P_r(r) + \text{tr}(\tilde{W}L^{-1}\tilde{W}^T) \]  

where \( \tilde{W} = W - \hat{W} \) is the approximation error. Differentiating (49) with respect to time yields

\[ \dot{V} = \dot{x}^T \frac{\partial P_v(x)}{\partial x} + \dot{r}^T \frac{\partial P_r(r)}{\partial r} - \text{tr}(\tilde{W}L^{-1}\dot{\tilde{W}}^T) \]

\[ = \dot{x}^T \xi_s + \dot{r}^T \xi_x - \text{tr}(\tilde{W}L^{-1}\dot{\tilde{W}}^T) \]

\[ = \dot{r}^T (\dot{J}_s(r)\xi_x + \xi_r) - \text{tr}(\tilde{W}L^{-1}\dot{\tilde{W}}^T). \]  

(50)

Substituting (30) into (50), it is obtained that

\[ \dot{V} = \left( \dot{J}_s(r)\xi_x + \xi_r \right)^T \times \left( \dot{J}_s(r)\xi_x + \xi_r \right) \]

\[ - \text{tr}(\tilde{W}L^{-1}\dot{\tilde{W}}^T) \]

\[ = \left( \dot{J}_s(r)\xi_x + \xi_r \right)^T \times \left( (\dot{J}_s(r) + \dot{J}_s(r))^T \xi_x + \xi_r \right) \]

\[ - \text{tr}(\tilde{W}L^{-1}\dot{\tilde{W}}^T) \]

\[ = \left( \dot{J}_s(r)\xi_x + \xi_r \right)^T \times \left( \dot{J}_s(r)\xi_x + \xi_r \right) \]

\[ - \left( \dot{J}_s(r)\xi_x + \xi_r \right)^T \dot{J}_s(r)\xi_x + \xi_r \]

\[ - \text{tr}(\tilde{W}L^{-1}\dot{\tilde{W}}^T). \]  

(51)

where \( \dot{J}_s(r) \triangleq J_s(r) - J_s^T(r) \). Making use of (23) and (26), it is clear that

\[ - \left( \dot{J}_s(r)\xi_x + \xi_r \right)^T \dot{J}_s(r)\xi_x \]

\[ = - \text{tr} \left[ \dot{J}_s(r)\xi_x \left( \dot{J}_s(r)\xi_x + \xi_r \right)^T \right] \]

\[ = - \text{tr} \left[ \xi_x^T \dot{\tilde{W}}(r) \left( \dot{J}_s(r)\xi_x + \xi_r \right)^T \right] \]

\[ = - \text{tr} \left[ \xi_x^T \dot{\tilde{W}}(r) \left( \dot{J}_s(r)\xi_x + \xi_r \right)^T \right]. \]  

(52)

Substituting the update law (24) into the last term of (51), it is obtained that

\[ - \text{tr}(\tilde{W}L^{-1}\dot{\tilde{W}}^T) = \text{tr} \left[ \dot{\tilde{W}}(r) \left( \dot{J}_s(r)\xi_x + \xi_r \right)^T \xi_x \right] \]

\[ = \text{tr} \left[ \xi_x^T \dot{\tilde{W}}(r) \left( \dot{J}_s(r)\xi_x + \xi_r \right)^T \right]. \]  

(53)

With (52) and (53), the last two terms in (51) can be canceled such that

\[ \dot{V} = - \left( \dot{J}_s(r)\xi_x + \xi_r \right)^T \times \left( \dot{J}_s(r)\xi_x + \xi_r \right) \leq 0. \]  

(54)

Since \( V > 0 \) and \( \dot{V} \leq 0 \), \( V \) is bounded, and the closed-loop system is stable. The boundedness of \( V \) ensures the boundedness of \( P_v(x) \), \( P_r(r) \), and \( \tilde{W} \). Hence, all the regional feedback vectors \( \xi_s, \xi_x, \xi_r \) are bounded. From (30), it can be seen that \( \dot{r} \) is bounded, which also ensures the boundedness of \( \dot{x} \) and \( \dot{q} \).

Hence, the term \( (\dot{J}_s(r)\xi_x + \xi_r) \) is uniformly continuous. From (54), it follows that \( (\dot{J}_s(r)\xi_x + \xi_r) \in L_2(0, +\infty) \).

Therefore, we have \( (\dot{J}_s(r)\xi_x + \xi_r) \to 0 \).

APPENDIX II

A. Corrections of Unintentional Wrong Demonstration

Since the human expert is responsible for the demonstration, some unintentional mistakes may arise, such that the robot learns a wrong or infeasible trajectory. It can be corrected as follows.

1) In the presence of a wrong Cartesian goal position, the robot cannot place the grasped object onto it [e.g., it has been occupied by another object as shown in Fig. 20(a)]. Since the learned trajectory is well-parameterized with the DMP approach, a new and correct Cartesian position can then be directly imposed in the reproduced trajectory by setting the parameter of \( g \) in (31), such that the robot will place the object there accordingly [see Fig. 20(b)].

2) In the presence of a wrong trajectory to a correct Cartesian goal position, the robot’s body may collide with the environment [which may be due to the change in the environment, as seen in Fig. 21(a)-(e)]. To address it, 1) the expert can choose another joint space goal with the same Cartesian goal position, where the joint space goal can be obtained by solving an optimization problem according to the environment change. Or, 2) one or multiple waypoints [32] can be added into the learned trajectory to suit the new environment. Both can shape the wrong trajectory and hence avoid the collision, as shown in Fig. 21(f)-(j).

B. Collision Avoidance Without Expert’s Involvement

Note that the collision avoidance with the worker can be automatically realized without the presence of the expert under the proposed framework. To achieve it, the control effort \( d \) in (19) is divided into two parts as

\[ d = d_h + d_a \]  

(55)

where \( d_h \in \mathbb{R}^7 \) is the control effort exerted by the human expert, and \( d_a \in \mathbb{R}^7 \) denotes another repulsive control term.
Fig. 21. Experiment 5—an example of the wrong trajectory to the goal position. (a)–(d) Tripod was placed in the middle of the trajectory to the goal position and treated as a new obstacle; when the robot followed the original trajectory, it collided with the obstacle. (e) Original joint-space trajectories. (f)–(i) Robot followed the corrected trajectory and successfully avoided the new obstacle. (j) Corrected joint-space trajectories.

Fig. 22. Experiment 6—collision avoidance with vision feedback. (a) \( t = 0.0 \) s: the robot was going to grasp the target object. (b) \( t = 9.5 \) s: the human’s right hand was approaching the robot; the robot used the 3-D vision sensor and detected the human, then it started to adjust the redundant joints without affecting the end-effector. The lower left figure is the point cloud image at the moment. (c) \( t = 16.0 \) s: the adjustment was finished. (d) \( t = 23.0 \) s: the robot grasped the object and returned to the original configuration.

An example of \( d_a \) can be given as [6], [32]

\[
d_a = \begin{cases} 
  k_a J_r^T \frac{p}{\|p\|}, & p \neq 0, \|p\| \leq \text{threshold} \\
  0, & \text{otherwise} 
\end{cases} 
\]  

(56)

where \( p \in \mathbb{R}^3 \) is a vector pointing from the worker’s hand to a redundant joint, which can be measured with a 3-D vision sensor, \( J_r \in \mathbb{R}^{3 \times n} \) is the translational part of the Jacobian matrix of that joint, and \( k_a \in \mathbb{R} \) is a scaling parameter.

When the expert is absent, \( d_a = 0 \). Then, \( d_a \) enables the robot to automatically repulse the human body (see Fig. 22) if the distance between the human body and the robot body (joint 4) is too close such that \( \|p\| \leq \text{threshold} \). The experimental results are illustrated in Fig. 23. It shows that the human was very close to the robot between \( t = 9.5 \) s and \( t = 16 \) s, where the distance \( p \) was below the threshold. The robot used the 3-D vision feedback to automatically avoid the collision with the human, without the involvement of a human expert. Note that such automatic adjustment does not affect the main task on the end-effector either. This helps guarantee the safety of the human worker.

C. Collaboration Without Human Experts

Collaboration without human experts was conducted to validate the claimed contribution and the generality of the proposed method. Specifically, a human worker entered the robot workspace and carried out tasks at the same time when the robot grasped and transferred the object, as shown in Fig. 24. The moving worker was treated as a dynamic obstacle and then avoided by the robot. During the grasping task of the robot, the worker intended to clean up scraps of paper on the table [see Fig. 24(a)]. As the robot was too close [see Fig. 24(b)], he moved his right hand to approach the robot’s fourth joint from the front [see Fig. 24(c)] and the back [see Fig. 24(f)] successively. The robot detected his movement and then adjusted the redundant joints to enlarge the operation space for the worker, without affecting the task on the end-effector [see Fig. 24(d)]. During the placing task of the robot,
the worker was measuring the workspace [see Fig. 24(g)]. As the worker was in the middle of the robot’s trajectory to the goal position, the previously learned trajectory became unavailable; Then, the robot shaped the learned trajectory by inserting one waypoint, hence avoiding the potential collision [see Fig. 24(h)]. Throughout the whole procedure, the expert was absent, and the robot used the external 3D vision sensor to achieve detection and collision avoidance automatically.

**Fig. 24.** Experiment 7—a human worker entered and coworked with the robot. (a) and (b) Worker intended to clean up two piles of paper scraps on the table. First, he cleaned the pile in front of the robot, but the robot was getting too close. (c) While the robot was placing the target object, the worker placed a new one. (f) Worker cleaned up the other pile of paper scraps behind the robot. During that, he adjusted the null space of the robot in the same way. (g) Worker was measuring the workspace when the robot was placing the object. (h) Robot followed the shaped trajectory to avoid the potential collision as well.

**REFERENCES**

[1] A. Weiss, A.-K. Wortmeier, and B. Kubicek, “Robots in industry 4.0: R roadmap for future practice studies on human–robot collaboration,” *IEEE Trans. Hum.-Mach. Syst.*, vol. 51, no. 4, pp. 335–345, Aug. 2021.

[2] X. Li, G. Chi, S. Vidas, and C. C. Cheah, “Human-guided robotic manipulation: Two illustrative scenarios,” *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 5, pp. 1751–1763, Sep. 2016.

[3] A. Hentout, M. Aouache, A. Maoudj, and I. Akli, “Human–robot interaction in industrial collaborative robotics: A literature review of the decade 2008–2017,” *Adv. Robot.*, vol. 33, nos. 15–16, pp. 764–799, Aug. 2019.

[4] J. Iqbal, Z. H. Khan, and A. Khalid, “Prospects of robotics in food industry,” *Food Sci. Technol.*, vol. 37, no. 2, pp. 159–165, May 2017.

[5] A. Sanderson, “Intelligent robotic recycling of flat panel displays,” M.S. thesis, Dept. Mech. Mechtron. Eng., Univ. Waterloo, Waterloo, ON, Canada, 2019.

[6] X. Yan, C. Chen, and X. Li, “Adaptive vision-based control of redundant robots with null-space interaction for human–robot collaboration,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 2803–2809.

[7] S. Chiaverini, “Kinematically redundant manipulators,” in *Springer Handbook of Robotics*. Berlin, Germany: Springer, 2008, pp. 245–268.

[8] C. Gosselin and L.-T. Schreiber, “Kinematically redundant spatial parallel mechanisms for singularity avoidance and large orientational workspace,” *IEEE Trans. Robot.*, vol. 32, no. 2, pp. 296–300, Apr. 2016.

[9] M. G. Carmichael, D. Liu, and K. J. Waldron, “A framework for singularity-robust manipulator control during physical human–robot interaction,” *Int. J. Robot. Res.*, vol. 36, nos. 5–7, pp. 861–876, Jun. 2017.

[10] G. Chesi, K. Hashimoto, D. Prattichizzo, and A. Vicino, “Keeping features in the field of view in eye-in-hand visual servoing: A switching approach,” *IEEE Trans. Robot.*, vol. 20, no. 5, pp. 908–914, Oct. 2004.

[11] N. García-Aracil, E. Malis, R. Aracil-Santonja, and C. Perez-Vidal, “Continuous visual servoing despite the changes of visibility in image features,” *IEEE Trans. Robot.*, vol. 21, no. 6, pp. 1214–1220, Dec. 2005.

[12] N. R. Gans, G. Hu, K. Nagarajan, and W. E. Dixon, “Keeping multiple moving targets in the field of view of a mobile camera,” *IEEE Trans. Robot.*, vol. 27, no. 4, pp. 822–828, Aug. 2011.

[13] C. P. Bechlioulis, S. Heshmati-Alamdari, G. C. Karras, and K. J. Kyriakopoulos, “Robust image-based visual servoing with prescribed performance under field of view constraints,” *IEEE Trans. Robot.*, vol. 35, no. 4, pp. 1063–1070, Aug. 2019.

[14] T. Li, J. Yu, Q. Qiu, and C. Zhao, “Hybrid uncalibrated visual servoing control of harvesting robots with RGB-D cameras,” *IEEE Trans. Ind. Electron.*, vol. 70, no. 3, pp. 2729–2738, Mar. 2023.

[15] X. Liang et al., “Purely image-based pose stabilization of nonholonomic mobile robots with a truly uncalibrated overhead camera,” *IEEE Trans. Robot.*, vol. 36, no. 3, pp. 724–742, Jun. 2020.

[16] X. Li and C. C. Cheah, “Global task-space adaptive control of robot,” *Automatica*, vol. 49, no. 1, pp. 58–69, Jan. 2013.

[17] S. Hjorth and D. Chrysostomou, “Human–robot collaboration in industrial environments: A literature review on non-destructive disassembly,” *Robot. Comput.-Int. Manuf.*, vol. 73, Feb. 2022, Art. no. 102208.

[18] A. K. Inkuulu, M. V. A. R. BabuBalendruni, A. Dara, and K. SankaranarayanaSamy, “Challenges and opportunities in human robot collaboration context of industry 4.0—A state of the art review,” *Ind. Robot. Int. J. Robot. Res. Appl.*, vol. 49, no. 2, pp. 226–239, Feb. 2022.

[19] A. C. Simões, A. Pinto, J. Santos, S. Pinheiro, and D. Romero, “Designing human–robot collaboration (HRC) workspaces in industrial settings: A systematic literature review,” *J. Manuf. Syst.*, vol. 62, pp. 28–43, Jan. 2022.

[20] A. Kanazawa, J. Kinugawa, and K. Kousage, “Motion planning for human–robot collaboration using an objective-switching strategy,” *IEEE Trans. Hum.-Mach. Syst.*, vol. 51, no. 6, pp. 590–600, Dec. 2021.

[21] A. Palleschi, M. Hamad, S. Abdolshah, M. Garabini, S. Haddadin, and L. Pallottino, “Fast and safe trajectory planning: Solving the cobot performance/safety trade-off in human–robot shared environments,” *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 5445–5452, Jul. 2021.

[22] E. Noohi, M. Žefran, and J. L. Patton, “A model for human–human collaborative object manipulation and its application to human–robot interaction,” *IEEE Trans. Robot.*, vol. 32, no. 4, pp. 880–896, Aug. 2016.

[23] W. Wang, R. Li, Z. M. Diekel, Y. Chen, Z. Zhang, and Y. Jia, “Controlling object hand-over in human–robot collaboration via natural wearable sensing,” *IEEE Trans. Hum.-Mach. Syst.*, vol. 49, no. 1, pp. 59–71, Feb. 2019.

[24] I. E. Makrini, K. Merckaert, D. Lefeber, and B. Vanderborght, “Design of a collaborative architecture for human–robot assembly tasks,” in *Proc. IROS*, 2017, pp. 1624–1629.

[25] G. Hoffman, “Evaluating fluency in human–robot collaboration,” *IEEE Trans. Hum.-Mach. Syst.*, vol. 49, no. 3, pp. 209–218, Jun. 2019.

[26] R. Bogue, “Europe continues to lead the way in the collaborative robot business,” *Ind. Robot. Int. J.*, vol. 43, no. 1, pp. 6–11, Jan. 2016.

[27] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, “A survey of robot learning from demonstration,” *Robot. Auto. Syst.*, vol. 57, no. 5, pp. 469–483, May 2009.

[28] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, “Recent advances in robot learning from demonstration,” *Annu. Rev. Control, Robot., Auto. Syst.*, vol. 3, no. 1, pp. 297–330, May 2020.
Leiliang Gong received the B.A. degree in mechanical engineering from Jilin University, Changchun, China, in 2015, and the Ph.D. degree from the National University of Singapore, Singapore, in 2020.

He currently serves as a Post-Doctoral Researcher with the Robotics and Artificial Intelligence Division, Hong Kong Productivity Council (HKPC), Hong Kong, China. His current research interests include intelligent robot systems, simultaneous localization and mapping (SLAM), and augmented reality.

Ming Ge (Senior Member, IEEE) received the Ph.D. degree from The Chinese University of Hong Kong (CUHK), Hong Kong, in 2003.

Before re-entering the campus, he worked as an Engineer with Jardine and Swire joint venture, Hong Kong, being responsible for the system design and control algorithm development of the automation system at Super Terminal 1 in Hong Kong Airport. Later, he worked as a Post-Doctoral Research Fellow with the NSF Industry/University Joint Centre, USA. In 2004, he joined Hong Kong Productivity Council, Hong Kong, China, serving as a Consultant, a Senior Consultant, the Principal Consultant, and currently the General Manager. In 2019, he engaged the establishment of the Hong Kong Industrial Artificial Intelligence and Robotics Centre, Hong Kong. He is currently a Visiting Professor with Guizhou University, Guiyang, China, and a Visiting Research Fellow with the Shenzhen Institute of Robotics and Intelligent Manufacturing, CUHK. He has a monograph and has published more than ten articles in reputable international journals. He owns eight invention patents and more than ten utility model patents.

Dr. Ge is a fellow of Hong Kong Institution of Engineers (HKIE) and the Chairperson of the CAI Division. He is also a member of the Editorial Board of the Journal of Applied Scientific Research.

Tao Zhang (Senior Member, IEEE) received the B.S., M.S., and first Ph.D. degrees from Tsinghua University, Beijing, China, in 1993, 1995, and 1999, respectively, and the second Ph.D. degree from Saga University, Saga, Japan, in 2002.

He is currently a Professor and the Head of the Department of Automation, School of Information Science and Technology, Tsinghua University. He is the author or a coauthor of more than 200 articles and 12 books. His current research interests include robotics, image processing, and control theory.

Xiang Li (Senior Member, IEEE) is currently an Associate Professor with the Department of Automation, Tsinghua University, Beijing, China. His current research interests include robotic manipulation, vision-based control, micro/nano robots, and human–robot interaction.

Prof. Li has received the Highly Commended Paper Award in 2013 International Federation for the Promotion of Mechanism and Machine Science (IFToMM), the Best Paper in Robotic Control in 2017 International Conference on Advanced Robotics (ICAR), the Best Application Paper Finalists in 2017 International Conference on Intelligent Robots and Systems (IROS), and the T. J. Tarn Best Paper in Robotics in 2018 IEEE International Conference on Robotics and Biomimetics (ROBIO). He is the Program Chair of IEEE 2023 International Conference on Real-Time Computing and Robotics. From 2019 to 2021, he was an Associate Editor of IEEE Robotics and Automation Magazine and International Conference on Robotics and Automation (ICRA). He has been an Associate Editor of IEEE ROBOTICS AND AUTOMATION LETTERS since 2022.