S.I. : HUMAN-IN-THE-LOOP MACHINE LEARNING AND ITS APPLICATIONS

Human–robot skill transmission for mobile robot via learning by demonstration

Jiehao Li[1] · Junzheng Wang[1] · Shoukun Wang[1] · Chenguang Yang[2]

Abstract

This paper proposed a skill transmission technique for the mobile robot via learning by demonstration. When the material is transported to the designated location, the robot can show the human-like capabilities: autonomous tracking target. In this case, a skill transmission framework is designed, which the Kinect sensor is utilized to distinguish human activity recognition to create a planned path. Moreover, the dynamic movement primitive method is implemented to represent the teaching data, and the Gaussian mixture regression is utilized to encode the learning trajectory. Furthermore, in order to realize the accurate position control of trajectory tracking, a model predictive tracking control is investigated, where the recurrent neural network is used to eliminate the uncertain interaction. Finally, some experimental tasks using the mobile robot (BIT-6NAZA) are carried out to demonstrate the effectiveness of the developed techniques in real-world scenarios.

Keywords Mobile robot · Human–robot skill transfer · Imitation learning · Learning by demonstration

1 Introduction

Most recently, with the development of manufactured technologies, human–cyber-robot systems (HCRS) have received extensive attention, especially in the field of intelligent manufacturing [6, 23, 25, 39]. Traditional man–machine methods cannot adapt to complex and changeable tasks, so that they cannot meet the needs of diversification and individualization. However, in the advanced HCRS, the advantages of humans (such as intelligence and dexterity) and robots (such as fast response and high efficiency) are efficiently when combined to achieve a deep integration of human, information, and robot systems. Therefore, human–robot skill transmission is to transfer human skills to the robot after a general description, and then realize the motion operation, which is currently a hot issue. Human–robot skill transmission via multi-sensor fusion is a new significant topic in human-assisted systems [19, 26, 32, 36]. More specifically, it is promising to control the mobile rescue robot for delivering the wounded or materials safely in real-world environment with the uncertain scenarios [12]. For example, to transport the patient with a virus, epidemic is the safest method. As with the new viruses COVID-19 that are currently suffering in the world, if mobile robots can autonomously carry patients to designated locations, it can reduce infections among medical staff. However, few studies have applied human demonstration teaching to mobile robots to receive learning skills. In particular, it is interesting to combine the Kinect sensor for the application of skills transfer. Thus, unlike most applications in robotic arms, this paper innovatively presents the human skill transmission for the mobile robot via imitation learning.

Human–robot skill transmission technology is the primary section to realize the skills from humans to robots and determine how people teach robots [10, 18, 30]. According

✉ Junzheng Wang
  wangjz@bit.edu.cn

  Jiehao Li
  jiehao.li@bit.edu.cn

  Shoukun Wang
  bitwsk@bit.edu.cn

  Chenguang Yang
  cyang@ieee.org

1 State Key Laboratory of Intelligent Control and Decision of Complex Systems, School of Automation, Beijing Institute of Technology, Beijing 100081, China

2 Bristol Robotics Laboratory, University of the West of England, Bristol BS16 1QY, United Kingdom

to different interactive interfaces, common transfer methods can be summarized in the following three forms: teleoperation-based [31], physical human–robot interaction [37], and vision-based [35]. For the teleoperation-based human–machine skill transfer, the instructor can operate the slave robot through the master device. For example, a robot learning framework via teleoperation in the visual interaction was considered in [33], and an extreme learning machine method was used to improve the teaching trajectory for the robot motion. Furthermore, in [38], in order to obtain the situation of virtual reality headsets, a deep imitation learning scheme was discussed to the teleoperate robots under complex manipulation tasks. The advantage of this teaching method based on remote operation is that it can be used in remote control scenes and working scenes that are not suitable for direct contact between the instructor and the robot. However, the teleoperation system often has delay problems and tremor problems. On the other hand, the physical human–robot interaction-based skill transmission, which the instructor directly contact with the robot, is another commonly used technique [17]. For instance, an optimal physical human–robot interaction approach using machine learning was considered in [1] for the humanoid robots. However, it cannot be applied to surgical scenes with infectious diseases through physical interaction. The visual-based skill transmission is a simple and effective method. The visual sensor (such as Kinect camera) captures and tracks human motion information and then uses the learning algorithms to model the motion state to obtain a generalized description. For example, a discrete-time method was presented in [22], which utilizes the Kinect camera to capture the joint angle of the teacher's arm during movement and then map the human joint angle to the joint-space of the robot. The vision-based skill transfer method is convenient for people to teach. Since the human body does not directly contact the robot, the movement of the teacher body is not restricted, especially for the mobile robot. Based on this, we use the visual-based method for the imitation learning.

On the other hand, there are some advanced technologies for human skill transmission via teaching by human demonstration [5, 34], such as dynamic motion primitives (DMP), hidden Markov model (HMM), Gaussian mixture model (GMM) [3], and Gaussian mixture regression (GMR) [4]. For example, the GMM model is introduced into the imitation learning framework of humanoid robots, and the movement information learned by GMM can be used to dynamically adjust the corresponding actions according to the movement information of the target object [11]. Calinon et al. [2] proposed an architecture based on the HMM–GMR model so that the robot can learn human motion skills. It is the HMM method to model human motion information, and the GMR method to regress the

motion control commands. This architecture is similar to GMM–GMR, but it can characterize richer motion information with effective robustness.

In general, DMP has the advantages of a simple model, high computational efficiency, and strong generalization, but DMP independently characterizes the information of each movement dimension and loses the associated information between each dimension. Furthermore, GMM and HMM cannot distinguish the residence time of each state, so it takes a long time to learn model parameters. However, the architecture combined with DMP and GMR can represent richer motion information so that the algorithm has stronger robustness and generalization capabilities, and this paper mainly centers on this method applying for the motion representation of mobile robots.

How to combine the Kinect sensor technology to achieve human-like control is the main challenge because robots and humans need effective collaboration. Meanwhile, it is interesting to deliver the patients and materials without colliding with people and equipment for the mobile rescue application. Therefore, this paper aims to bridge the human–robot cooperative control for the human activity recognition and robot-assisted systems via teaching by human demonstration. The main framework of this article is arranged as below:

1. The Kinect camera is adopted to recognize human movement points, and then, the imitation learning technique is implemented to generate the teaching trajectory for the mobile robot.
2. Aiming to high accuracy of tracking control, a model predictive tracking controller is developed to track the learning path, and a recurrent neural network is utilized to effectively evaluate the uncertain interaction simultaneously.
3. Different from the conventional application of imitation learning in manipulators, this article intends to establish a human-like control of mobile robot application, which enhances human skills transferred to the mobile robot.

The overview of this article can be arranged as follows: Sect. 2 investigates the methodology of human skill transfer framework based on the Kinect sensor, the trajectory regression, and the neural network-based predictive tracking controller. The experiment validation is presented in Sect. 3, and the summary is discussed in Sect. 4.

2 Methodology

2.1 Human movement recognition

The position information of the joint points of the human body in the three-dimensional space can be obtained by the

Kinect sensor [8]. The Kinect sensor can accurately obtain the three-dimensional coordinates of the 25 joint points of the human body in real time. According to the three-dimensional information of the joint position, the angle feature and the distance feature of each joint point of the human body can be expressed. The recognition process of human motion is presented in Fig. 1. The image of the human body captured by the Kinect camera can record the human movement and then scale the joints after being processed, subsequently normalize concerning scale and position, and then consider the posture analysis.

First of all, normalizing the standard action features and real-time action features can improve the calculation speed and accuracy. Considering $\mathcal{B}(k) = (x(k), y(k), z(k))$, and the depth image points $D(i, j)$ and color image $\vartheta(i, j)$, the probability $\mathcal{B}(t|\vartheta)$ can be calculated as below:

$$\mathcal{B}(t \mid \vartheta) = \frac{\mathcal{B}(\vartheta \mid t)\mathcal{B}(t)}{\mathcal{B}(\vartheta)} \tag{1}$$

where $\mathcal{B}(\vartheta)$ represents the movement point, and $\mathcal{B}(\vartheta|t)$ is the previous color probabilities.

At certain points, it is because of consciousness or lack of joint details that we need to add other roles that can provide human shape details to boost the classifier's accuracy. To obtain the positive $2D$ image and the profile obtained, we followed orthogonal Cartesian planes on the depth map. So the silhouette is converted by mapping Cartesian coordinates to polar coordinates [14].

$$\mathcal{R}_i = \sqrt{\left(x_i - x_j\right)^2 - \left(y_i - y_j\right)^2} \tag{2}$$

$$\theta_i = \tan^{-1}\frac{y_i - y_j}{x_i - x_j} \tag{3}$$

where (x_i, y_i) denotes the teacher position. $(\mathcal{R}_i, \theta_i)$ represents the corresponding angle and polar position. (x_j, y_j) denotes central reference point of human exoskeleton.

Finally, the average $\mathcal{E}_{mean}$ can be addressed as below:

$$\mathcal{E}_{mean} = \frac{1}{T}\sum_{t=1}^{T} I(x, y, z, t) \tag{4}$$

2.2 Dynamic movement primitive

The Kinect sensor is used to extract human exoskeleton data and then to obtain the distance characteristics and the angular characteristics. It is useful to dynamically match with the template animation in real time with a good recognition performance. At the same time, this method has simple algorithm and fast calculation speed, which can meet the real-time requirements for pose matching. Thus, the next process is to generate the human movement trajectory. Hence, we apply the imitation learning technique to encode the teaching position data. The DMP algorithm can be expressed as follows:
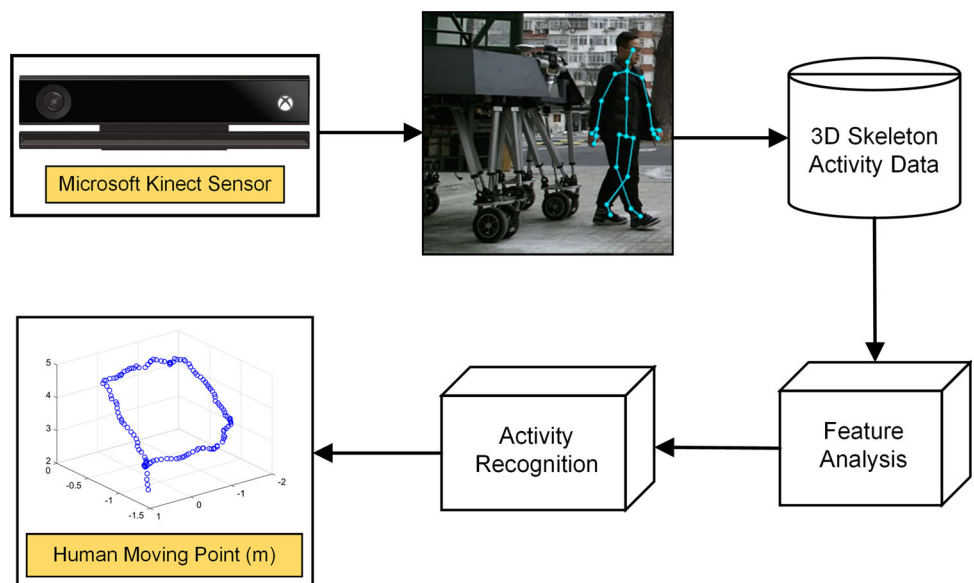
$$\begin{aligned} \ddot{\mathcal{C}}_t &= G^p(g - \mathcal{C}_t) - G^v\dot{\mathcal{C}}_t + F(v_t) \\ \dot{v}_t &= \alpha_v v_t \\ F(v_t) &= h_t^{\mathrm{T}}(v_t)\omega(g - \mathcal{C}_0) \end{aligned} \tag{5}$$

subjected to

$$h_t(v_t) = \frac{\sum_{i=1}^{N} \psi_i(v_t)v_t}{\sum_{i=1}^{N} \psi_i(v_t)_t} \tag{6}$$

$$\psi_i(v_t) = \exp\left(-\frac{1}{2\Xi_i}(v_t - c_i)^2\right) \tag{7}$$

Fig. 1 The recognition process of human motion

where $[\mathcal{C}_t, \dot{\mathcal{C}}_t, \ddot{\mathcal{C}}_t]$ is the position in the state of Cartesian, and G^v represents the item in this state. G^p denotes the structural stiffness. $\mathcal{C}_0$ presents the original state, and g denotes designed point of the mobile robot. Besides, α_v represents the order factor in established learning system. ω denotes the corresponding dynamic movement primitive factor. It should be noted that v_t gradually tend to zero. Finally, Ξ_i is the length of Gaussian function and the c_i is the home position, where i represents the order of the function.

In particular, there are two parameters in dynamic movement primitive procedure, including the item of a linear spring-damper section and the nonlinear section, that is, $G^p(g - \mathcal{C}_t) - G^v\dot{\mathcal{C}}_t$ and $F(v_t)$. In order to efficiently obtain the motion representation of the mobile robot, These items can be used to evaluate the motion process of the teacher for the method of imitation learning. Therefore, the intention of DMP method is to imitate the teacher motions combined with the point of g.

2.3 Gaussian mixture model

The method of dynamic movement primitive is a movement representation of human activities. Consequently, we apply the technology of Gaussian mixture model to encode the human movement position via the Kinect sensors. This model is based on the knowledge of statistical probability, which can calculate the probability density distribution of each Gaussian model data set. Thus, according to the dynamic movement primitive function presented in (5), the corresponding Gaussian mixture model can be addressed as follows:

$$\ddot{\mathcal{C}} = \sum_{k=1}^{G} h_k \left(G_k^p \left(\mu_k^{\mathcal{C}} - \mathcal{C} \right) - G_k^v \dot{X} + F \right) \tag{8}$$

where G_k^p represents the structural stiffness, and G_k^v denotes the corresponding item in Cartesian state.

At the same time, define the human demonstration in the Cartesian state as below: $v_j = (v_{t,j}, v_{\mathcal{C},j})(j = 1, \ldots, N)$, including the time set $v_{t,j}$ and the position set $v_{\mathcal{C},j}$.

Subsequently, according to the related motion point, we apply the following Gaussian mixture model.

$$p(v_j) = \sum_{k=1}^{G} p(k) p(v_j|k) \tag{9}$$

Among them, $p(k)$ and $p(v_j|k)$ represent the corresponding prior probability and conditional probability density, and k presents the related orders.

Hence, the items of Gaussian mixture model can be addressed as below:

$$p(k) = \lambda_k \tag{10}$$

$$p(v_j|k) = \frac{e\left(-\frac{1}{2} (v_j - \mu_k)^T \Xi_k^{-1} (v_j - \mu_k) \right)}{\sqrt{8\pi^3 |\Xi_k|}} \tag{11}$$

where λ_k is a priori probability, μ_k is the average value, Ξ_k is the variance value, and D_k is a posterior probability.

Then, according to statistical theory, the related D_k can be calculated as below:

$$D_k = \sum_{j=1}^{N} p(k|v_j) \tag{12}$$

subjected to

$$p(k|v_j) = \frac{p(k)p(v_j|k)}{\sum_{m=1}^{G} p(m)p(v_j|m)} \tag{13}$$

2.4 Gaussian mixture regression

The coefficients related to its probability have been obtained through the Gaussian mixture model, and then, the problem of the nonlinear part will be considered. How to effectively achieve the regression factor F is the next problem. After the Gaussian mixture model solves the probability, we utilize the technique of Gaussian mixture regression to reflect motion points and need to evaluate $v_{\mathcal{C}}$. Therefore, we have

$$\mu_k = \{\mu_{t,k}, \mu_{\mathcal{C},k}\} \tag{14}$$

$$\Xi_k = \begin{pmatrix} \Xi_{tt,k} & \Xi_{t\mathcal{C},k} \\ \Xi_{ct,k} & \Xi_{cc,k} \end{pmatrix} \tag{15}$$

where μ_k is the average value and Ξ_k is the variance value.

The next step is to calculate the desired distribution $v_{\mathcal{C},k}$, and then, we have

$$p(v_{\mathcal{C},k}|v_t, k) = \mathcal{N}(v_{\mathcal{C},k}; \hat{v}_{\mathcal{C},k}, \hat{\Xi}_{\mathcal{C}\mathcal{C},k}) \tag{16}$$

$$\hat{v}_{\mathcal{C},k} = \mu_{\mathcal{C},k} + \sum_{\mathcal{C}t,k} (\Xi_{tt,k})^{-1} (v_t - \mu_{t,k}) \tag{17}$$

$$\hat{\Xi}_{\mathcal{C}\mathcal{C},k} = \Xi_{\mathcal{C}\mathcal{C},k} - \Xi_{\mathcal{C}t,k} (\Xi_{tt,k})^{-1} \Xi_{t\mathcal{C},k} \tag{18}$$

where $\hat{v}_{\mathcal{C},k}$ and $\hat{\Xi}_{\mathcal{C}\mathcal{C},k}$ are determined by probability distribution.

Finally, the condition probability density can be defined as follows:

$$p(v_{\mathcal{C}}|v_t) = \sum_{k=1}^{G} h_k \mathcal{N}(v_{\mathcal{C}}; \hat{v}_{\mathcal{C},k}, \hat{\Xi}_{\mathcal{C}\mathcal{C},k}) \tag{19}$$

$$h_k = \frac{p(k)p(v_t|k)}{\sum_{i=1}^{G} p(i)p(v_t|i)} = \frac{\lambda_k \mathcal{N}(v_t; \mu_{t,k}, \Xi_{tt,k})}{\sum_{i=1}^{G} \lambda_i \mathcal{N}(v_t; \mu_{t,i}, \Xi_{tt,i})} \quad (20)$$

Therefore, the movement $\{\hat{v}_t, \hat{v}_\mathcal{C}\}$ can be generated by estimating $\{\hat{v}_\mathcal{C}, \hat{\Xi}_{\mathcal{CC}}\}$ at the time step v_t, that is,

$$\hat{v}_\mathcal{C} = \sum_{k=1}^{G} h_k \hat{v}_{\mathcal{C},k} \quad (21)$$

$$\hat{\Xi}_{\mathcal{CC}} = \sum_{k=1}^{G} h_k^2 \hat{\Xi}_{\mathcal{CC},k} \quad (22)$$

2.5 Tracking controller development

The Kinect camera collects human motion points, and then, the generated trajectory can be obtained with the imitation learning method. Finally, the next task is to control the mobile robot to track the human movement point using predictive tracking controller. At the same time, the neural networks are used to evaluate the uncertain interaction in the tracking process [7, 20, 24, 27, 28].

Figure 2 displays the kinematic model and turning model of the robot. The turning model can be defined as follows:

$$R = \frac{W}{2} + \frac{L}{2 \tan \theta_{in}} \quad (23)$$

$$R_{out} = \sqrt{\left(\frac{L}{2}\right)^2 + \left(W + \frac{L}{2 \tan \theta_{in}}\right)^2} \quad (24)$$

$$R_{in} = \sqrt{\left(\frac{L}{2}\right)^2 + \left(\frac{L}{2 \tan \theta_{in}}\right)^2} \quad (25)$$

where the inner steering angle is denoted by $\theta_{in} = \theta_1 = \theta_5$. The wheelbase and width are represented by L and W,

respectively. The wheel width is denoted by the letter W. The radius of centroid, inner, and outer is denoted by R, R_{in} and R_{out}, respectively. Thus, the robot speed and yaw rate are described as follows:

$$v_x = v \cos \varphi \quad (26)$$

$$v_y = v \sin \varphi \quad (27)$$

$$\omega_c = \frac{v}{R} = \frac{2v \tan \theta_{in}}{W \tan \theta_{in} + L} \quad (28)$$

where φ and v denote the course angle robot speed, respectively. Thus, the steering relationship can be represented as below:

$$\tan \theta_{out} = \frac{\frac{L}{2}}{\frac{W}{2} + R} \quad (29)$$

$$\theta_{out} = \arctan \frac{L}{W + 2R} \quad (30)$$

$$R_6 = R - W \quad (31)$$

$$R_{15} = \sqrt{\left(\frac{L}{2}\right)^2 + \left(R - \frac{W}{2}\right)^2} \quad (32)$$

$$R_{24} = \sqrt{\left(\frac{L}{2}\right)^2 + \left(W + \frac{L}{2 \tan \theta_{in}}\right)^2} \quad (33)$$

$$R_3 = R + W \quad (34)$$

where the R_6, R_{15}, R_{24}, R_3 represent the wheel number of 6, 1 and 5, 2 and 4 and 3, respectively.

The kinematic model of BIT-6NAZA mobile robot [21] can be addressed as follows:

$$\dot{Z} = \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\varphi}_c \end{bmatrix} = \begin{bmatrix} \cos \varphi_c & 0 \\ \sin \varphi_c & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_c \\ \omega_c \end{bmatrix} \quad (35)$$
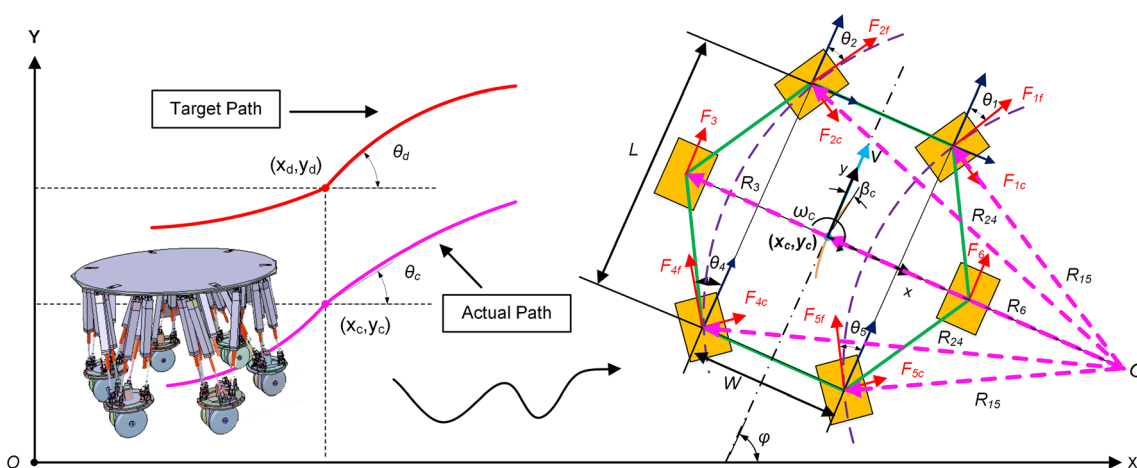

Fig. 2 Six-wheeled independent steering model

where (x_c, y_c, φ_c) denote the position in X–Y-axis, and the course angle, respectively. ω_c denotes yaw rate, and v_c represents the line velocity.

Then, the tracking error function exhibited can be indicated as follows:

$$
\dot{\mathcal{Z}}_e = \begin{bmatrix} \dot{x}_c - \dot{x}_d \\ \dot{y}_c - \dot{y}_d \\ \dot{\varphi}_c - \dot{\varphi}_d \end{bmatrix} = \begin{bmatrix} 0 & 0 & -v_c \sin \varphi_d \\ 0 & 0 & v_c \cos \varphi_d \\ 0 & 0 & 0 \end{bmatrix} \mathcal{Z}_e
$$
$$
+ \begin{bmatrix} \cos \varphi_d & 0 \\ \sin \varphi_d & 0 \\ \dfrac{2 \tan \delta}{L(1 + \tan \delta)} & \dfrac{2v_c}{L(1 + \tan \varphi)^2 \cos^2 \delta_d} \end{bmatrix} u_e \qquad (36)
$$

where (x_d, y_d, φ_d) and (v_d, δ_d) denote the desired state variables and control variables, respectively. L represents the wheel track of the robot, and δ denotes the turning angle.

Subsequently, the error function can be described as follows:

$$
\tilde{\mathcal{Z}}(k+1) = \mathcal{U}_{k,t} \tilde{\mathcal{Z}}(k) + \mathcal{K}_{k,t} \tilde{u}(k) \qquad (37)
$$

subjected to

$$
\mathcal{U}_{k,t} = \begin{bmatrix} 1 & 0 & -v_d T \sin \varphi_d \\ 0 & 1 & v_d T \cos \varphi_d \\ 0 & 0 & 1 \end{bmatrix} \qquad (38)
$$

$$
\mathcal{K}_{k,t} = \begin{bmatrix} T \cos \varphi_d & 0 \\ T \sin \varphi_d & 0 \\ \dfrac{2T \tan \delta}{L(1 + \tan \delta)} & \dfrac{2v_d T}{L(1 + \tan \varphi)^2 \cos^2 \delta_d} \end{bmatrix} \qquad (39)
$$

where T denotes the testing period.

Considering the objective function in [16], the optimization function can be selected as below:

$$
\mathcal{A}_d(k) = \sum_{i=1}^{P_p} \Psi(k+i \mid t) - \Psi_{ref}(k+i \mid t)_{\mathcal{L}}^2
$$
$$
+ \sum_{i=1}^{P_c - 1} \| \Delta U(k+i \mid t) \|_R^2 + \sigma \psi^2 \qquad (40)
$$

where P_p and P_e represent the constraint of prediction and control, respectively. σ and ψ denote the corresponding weight variables.

Finally, the specific constraint of the tracking controller can be defined as below:

$$
\begin{bmatrix} -0.4 \\ -25 \end{bmatrix} \le u \le \begin{bmatrix} 0.6 \\ 32 \end{bmatrix}
$$
$$
\begin{bmatrix} -0.03 \\ -0.07 \end{bmatrix} \le \Delta U \le \begin{bmatrix} 0.03 \\ 0.07 \end{bmatrix} \qquad (41)
$$

The unknown disruption in the trajectory control process

for the mobile robot needs to be managed in order to successfully pass the trajectory teaching by human demonstration [9, 13, 29]. We consider in this paper that two components are included in the uncertainty of the drum system: internal connection and outside complexity. Considering the following function $\mathcal{G}(\mathcal{K}) : R^q \rightarrow R$, it can determine the constraint of unknown dynamics.

$$
\mathcal{G}_{nn}(\mathcal{K}_{in}) = \mathcal{Q}^T \Theta(\mathcal{K}_{in}) \qquad (42)
$$

where $\Theta(\mathcal{K}_{in}) = [\Theta_1(\mathcal{K}_{in}), \Theta_2(\mathcal{K}_{in}), \ldots, \Theta_i(\mathcal{K}_{in})]^T$ and $\Theta_i(\mathcal{K}_{in})$ represent the corresponding Gaussian function. $\mathcal{Q} = [\xi_1, \xi_2, \ldots, \xi_B] \in R^B$ and $\mathcal{K}_{in} \in \Omega \subset R^q$ denote the hidden layer and the input of neural networks.

$$
\Theta_i(\mathcal{K}_{in}) = \exp \left[\frac{-(\mathcal{K}_{in} - u_i^T)(\mathcal{K}_{in} - u_i)}{\eta_i^2} \right] \qquad (43)
$$

where $i = 1, 2, \ldots, m$, $u_i = [u_{i1}, u_{i2}, \ldots, u_{iq}]^T \in R^q$, and η_i is the variance.

Therefore, we have

$$
\| \Theta(\mathcal{K}_{in}) \| \le \tau \qquad (44)
$$

where τ denotes a positive variable.

$$
\mathcal{G}_{nn}(\mathcal{K}_{in}) = \mathcal{Q}^{*T} \Theta(\mathcal{K}_{in}) + \varepsilon \qquad (45)
$$
$$
\mathcal{Q}^* = \arg \min_{\mathcal{K}_{in} \in R^q} \left\{ \sup | \mathcal{G}_{nn}(K_{in}) - \mathcal{Q}^T \Theta(\mathcal{K}_{in}) | \right\} \qquad (46)
$$

where $\mathcal{Q}^*$ subjects to $\Phi_{\mathcal{K}_{in}} \subset R^q$, and $\| \varepsilon \| \le \tau_c$.

3 Experiment validation

In this part, the experimental demonstration is performed to discuss the developed imitation learning in real-world application [15], and the experimental environment is presented in Fig. 3. There is one Kinect sensor (Microsoft Xbox ONE) used in the demonstration. Among them, the human is the leader, and the mobile robot is the follower. The primary objective of this investigation is that the mobile robot can operate the learning trajectory by human demonstration effectively.

At the same time, the main experiment parameter settings are as follows: mobile robot velocity is 400 r/min; prediction horizon and control horizon set as 25 and 10, respectively; controller sample time is $T = 0.01$ s; the weight variables of neural networks are defined as $\mathcal{Q}_1(0) = \mathbf{0} \in R^{3l_1 \times 3}$ and $\mathcal{Q}_2(0) = \mathbf{0} \in R^{3l_2 \times 3}$; the corresponding learning rate of neural networks are defined as 0.000004 and 0.000006, respectively. $\eta_i = 1.008$ and $u_i = [-2.5 \quad -2.0 \quad 0 \quad 2.0 \quad 3.0 \quad 4.0]$.

On the other hand, the procession is set as follows: The human follows the same trajectory with five times, and the Kinect sensor records the moving trajectory points. Then,

Fig. 3 The experimental environment in real-world scenario. The Kinect camera is used to detect the human movement, and then, the DMP combined with GMR is applied to encode the learning trajectory. The mobile robot finally follow the teaching trajectory
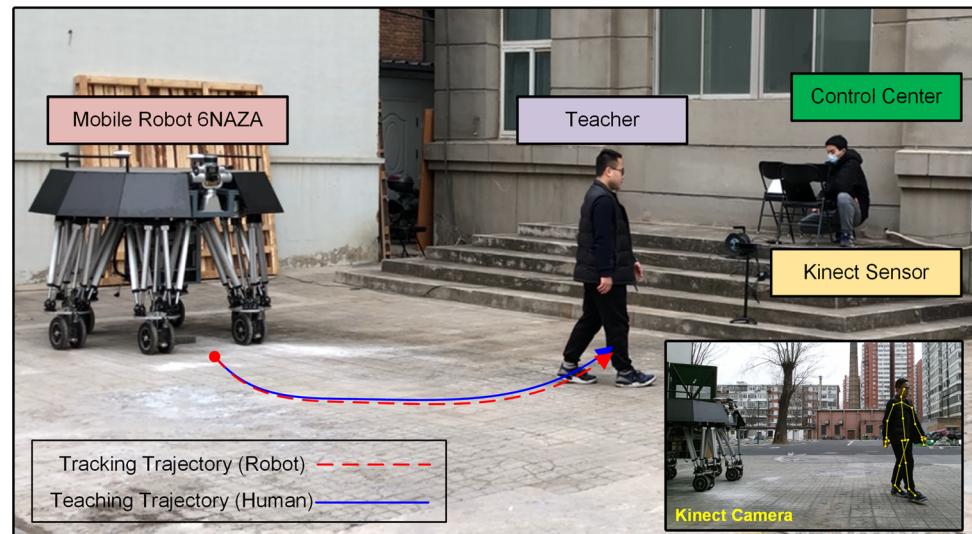
the method of DMP with GMR is used to generate a desired trajectory. Finally, the robot can be controlled to follow the teaching trajectory through the tracking controller. In this case, we set up two teaching tracks, including straight trajectory and C-shape trajectory.

The results are shown in Fig. 4. The Kinect sensor extracts the point information of the human exoskeleton and then selects the gravity center of the human as the reference point to record the set of track points during the teaching process. Figure 4a displays the regression process of teaching by human demonstration. In order to improve the regression accuracy of the Gaussian model, the trajectory dataset can be obtained by teaching five times. Then, the regression method of DMP with GMR is implemented to generate the teaching trajectory. Finally, the mobile robot can be control to follow the teaching trajectory using the model predictive tracking controller. As shown in Fig. 4b, there is the tracking results of x-error x_e, y-error y_e, course angle error θ_e, robot velocity v_{robot}, pitch angle, roll angle, and tracking trajectory in real world. It can be concluded that the horizontal and vertical position errors of the robot are basically constrained within the range of ± 0.05 meters, and the control accuracy is high. At the same time, the heading angle error is also constrained within $1°$, which shows that the proposed neural network can effectively eliminate external interference. Moreover, under the constraints of predictive controller, the speed of the robot is kept within the range of 400 r/min, and the attitude angles such as pitch angle and roll angle are also kept within a reasonable range, realizing the stability control of the robot.

Furthermore, the teaching and tracking experiment of C-shape trajectory is carried out, and the experimental performance is shown in Fig. 5. Similarly, five teaching movements of human are collected through the Kinect sensor, and then the gravity center is selected as the reference point to form a teaching set, as shown in Fig. 5a. The Gaussian function generates the desired trajectory of the robot through a reasonable number of iterations. The mobile robot achieves a human-like trajectory tracking performance by tracking the teaching trajectory. Figure 5b exhibits the tracking results of x-error x_e, y-error y_e, course angle error θ_e, robot velocity v_{robot}, pitch angle, roll angle, and tracking trajectory in real world. Benefited to the model predictive controller and the approximation of neural networks, the mobile robot can effective follow the desired trajectory. The position error of x-axis and y-axis is basically constrained within a reasonable range of ± 0.06 meters, and the course angle error can be controlled within $2°$. It is in line with the expected effect. In addition, the speed control and attitude control of the mobile robot also meet the requirement of the steady-state error, and there is no oscillation situation.

4 Conclusion

This paper studies a human–robot capability transfer method for controlling a mobile robot using learning by demonstration in real-world situations, with an emphasis on material transportation and wounded rescue. Learning by presentation, which is an ability learning focused on various teachings, is used to understand human–robot conversion technologies. A skill transmission framework is investigated in this situation, with the Kinect camera being used to discern human activity identification and establish an expected route. Furthermore, the dynamic movement primitive approach is used to represent the teaching results, and the learning curve is encoded using Gaussian mixture regression. On the other hand, a model predictive tracking
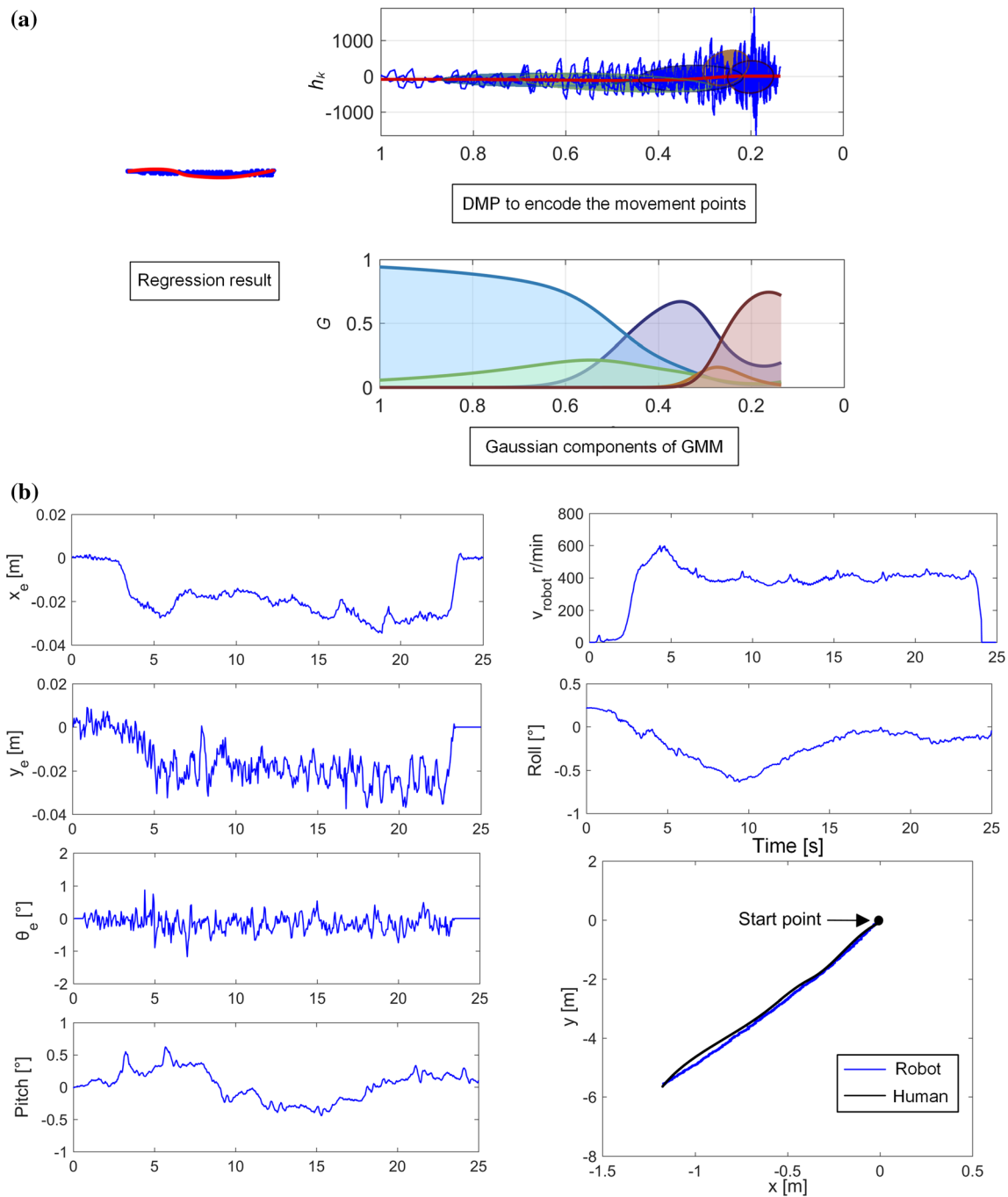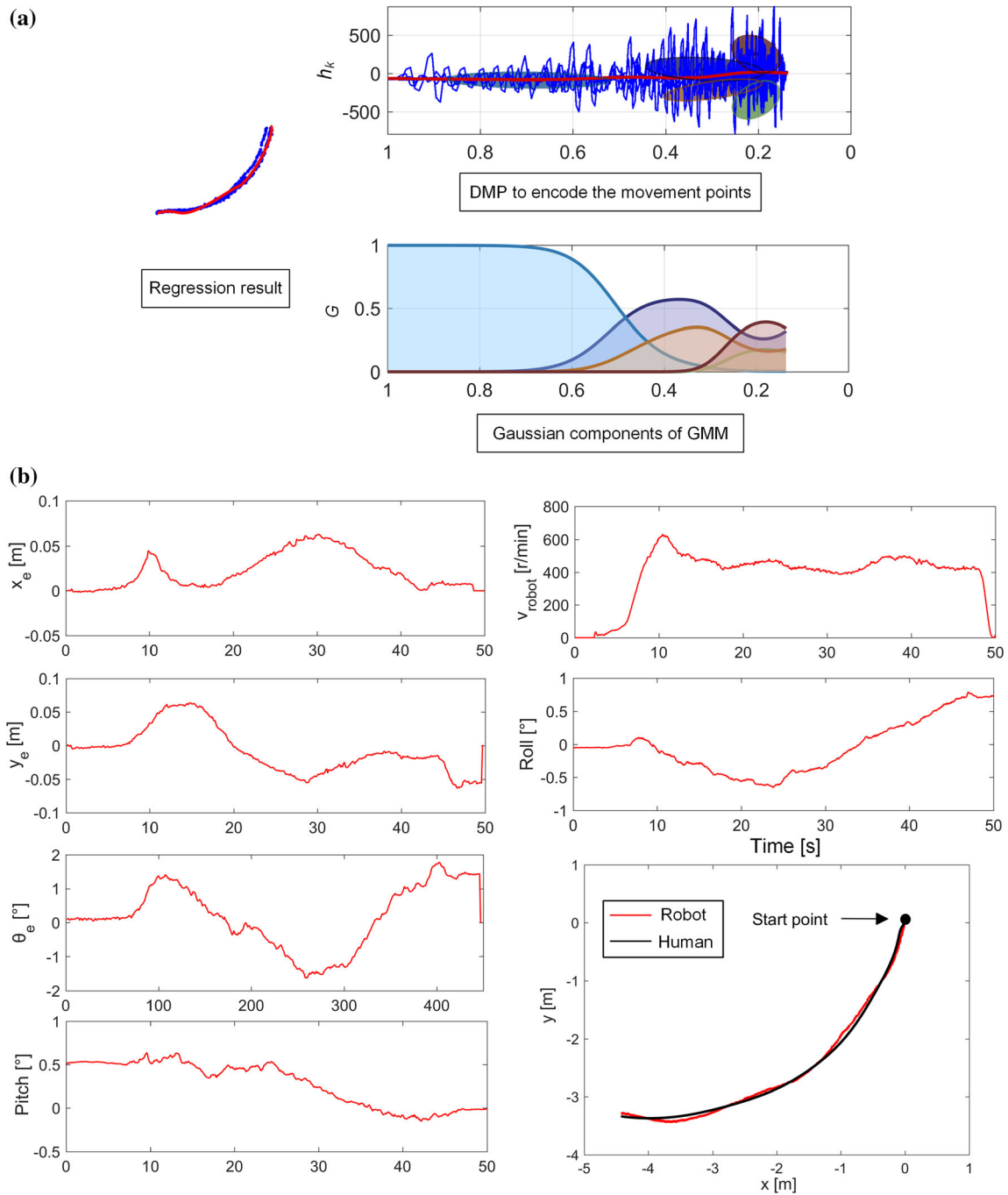
Fig. 4 a The regression process of straight line and **b** The tracking results of x-error x_e, y-error y_e, course angle error θ_e, robot velocity v_{robot}, pitch angle, roll angle and tracking trajectory in real-world

control is studied in order to achieve precise path tracking position control, where the recurrent neural network is used to eradicate the unknown interaction. Extensive demonstrations highlight the reasonable results in a real-world environment, and it offers a possible alternative for a mobile robot with human-like skill capacity. In future works, how to combine the Internet of thing and multisensors to achieve high performance of human activity recognition will be considered. At the same time, advanced tracking algorithms such as reinforcement learning control

Fig. 5 The tracking performance of teaching by demonstration under C-shape. **a** The regression process of C-shape; **b** The tracking results of x-error x_e, y-error y_e, course angle error θ_e, robot velocity v_{robot}, pitch angle, roll angle, and tracking trajectory in real-world

will be investigated to improve control accuracy and real-time performance.

Declaration

Conflict of interest All authors have participated in conception and design, or analysis and interpretation of the data. No conflict of interest exists in the submission of this manuscript.

References

1. Berger E, Müller D, Vogt D, Jung B, Amor HB (2014). Transfer entropy for feature extraction in physical human–robot interaction: detecting perturbations from low-cost sensors. In: IEEE/RAS international conference on humanoid robots. IEEE, pp 829–834
2. Calinon S, Billard A (2008) A probabilistic programming by demonstration framework handling constraints in joint space and task space. In: IEEE/RSJ international conference on intelligent robots and systems. IEEE, pp 367–372
3. Calinon S, Guenter F, Billard A (2007) On learning, representing, and generalizing a task in a humanoid robot. IEEE Trans Syst Man Cybern Part B 37(2):286–298
4. Calinon S, D'halluin F, Sauser EL, Caldwell DG, Billard AG, (2010) Learning and reproduction of gestures by imitation. IEEE Robotics Autom Mag 17(2):44–54
5. Chen J, Du C, Zhang Y, Han P, Wei W (2021) A clustering-based coverage path planning method for autonomous heterogeneous UAVs. IEEE Trans Intell Transp Syst 1–11. https://doi.org/10.1109/TITS.2021.3066240
6. Chen Z, Li J, Wang J, Wang S, Zhao J, Li J (2021) Towards hybrid gait obstacle avoidance for a six wheel-legged robot with payload transportation. J Intell Robotic Syst 1–21. https://doi.org/10.1007/s10846-021-01417-y
7. Chen Z, Wang S, Wang J, Xu K, Lei T, Zhang H, Wang X, Liu D, Si J (2021) Control strategy of stable walking for a hexapod wheel-legged robot. ISA Trans 108:367–380
8. Fankhauser P, Bloesch M, Rodriguez D, Kaestner R, Hutter M, Siegwart R (2015) Kinect v2 for mobile robot navigation: evaluation and modeling. In: International conference on advanced robotics (ICAR). IEEE, pp 388–394
9. Huang D, Yang C, Pan Y, Cheng L (2019) Composite learning enhanced neural control for robot manipulator with output error constraints. IEEE Trans Ind Inf 17(1):209–218
10. Huang H, Zhang T, Yang C, Chen CLP (2020) Motor learning and generalization using broad learning adaptive neural control. IEEE Trans Ind Electron 67(10):8608–8617
11. Khansari-Zadeh SM, Billard A (2011) Learning stable nonlinear dynamical systems with gaussian mixture models. IEEE Trans Robotics 27(5):943–957
12. Klamt T, Schwarz M, Lenz C, Baccelliere L, Buongiorno D, Cichon T, DiGuardo A, Droeschel D, Gabardi M, Kamedula M et al (2020) Remote mobile manipulation with the centauro robot: full-body telepresence and autonomous operator assistance. J Field Robotics 37(5):889–919
13. Li Z, Zhao T, Chen F, Hu Y, Su CY, Fukuda T (2017) Reinforcement learning of manipulation and grasping using dynamical movement primitives for a humanoidlike mobile manipulator. IEEE/ASME Trans Mechatron 23(1):121–131
14. Li Z, Huang B, Ye Z, Deng M, Yang C (2018) Physical human-robot interaction of a robotic exoskeleton by admittance control. IEEE Trans Ind Electron 65(12):9614–9624
15. Li J, Wang J, Peng H, Zhang L, Hu Y, Su H (2020) Neural fuzzy approximation enhanced autonomous tracking control of the wheel-legged robot under uncertain physical interaction. Neurocomputing 410:342–353
16. Li J, Wang J, Wang S, Peng H, Wang B, Qi W, Zhang L, Su H (2020) Parallel structure of six wheel-legged robot trajectory tracking control with heavy payload under uncertain physical interaction. Assem Autom 40(5):675–687
17. Li Y, Eden J, Carboni G, Burdet E (2020) Improving tracking through human-robot sensory augmentation. IEEE Robotics Autom Lett 5(3):4399–4406
18. Li Z, Xu C, Wei Q, Shi C, Su CY (2020) Human-inspired control of dual-arm exoskeleton robots with force and impedance adaptation. IEEE Trans Syst Man Cybern Syst 50(12):5296–5305
19. Li J, Qin H, Wang J, Li J (2021) Openstreetmap-based autonomous navigation for the four wheel-legged robot via 3d-lidar and CCD camera. IEEE Trans Ind Electron. https://doi.org/10.1109/TIE.2021.3070508
20. Li J, Wang J, Peng H, Hu Y, Su H (2021) Fuzzy-torque approximation enhanced sliding mode control for lateral stability of mobile robot. IEEE Trans Syst Man Cybern Syst. https://doi.org/10.1109/TSMC.2021.3050616
21. Li J, Wang S, Wang J, Li J, Zhao J, Ma L (2021) Iterative learning control for a distributed cloud robot with payload delivery. Assem Autom. https://doi.org/10.1108/AA-11-2020-0179
22. Liang P, Ge L, Liu Y, Zhao L, Li R, Wang K (2016) An augmented discrete-time approach for human–robot collaboration. Discret Dyn Nat Soc 2016:1–13
23. Peng G, Yang C, He W, Chen CP (2019) Force sensorless admittance control with neural learning for robots with actuator saturation. IEEE Trans Ind Electron 67(4):3138–3148
24. Peng H, Wang J, Wang S, Shen W, Shi D, Liu D (2020) Coordinated motion control for a wheel-leg robot with speed consensus strategy. IEEE/ASME Trans Mechatron 25(3):1366–1376
25. Qiao H, Li Y, Tang T, Wang P (2013) Introducing memory and association mechanism into a biologically inspired visual model. IEEE Trans Cybern 44(9):1485–1496
26. Qiao H, Wang M, Su J, Jia S, Li R (2014) The concept of "attractive region in environment'' and its application in high-precision tasks with low-precision systems. IEEE/ASME Trans Mechatron 20(5):2311–2327
27. Shi D, Xue J, Zhao L, Wang J, Huang Y (2017) Event-triggered active disturbance rejection control of DC torque motors. IEEE/ASME Trans Mechatron 22(5):2277–2287
28. Shi D, Xue J, Wang J, Huang Y (2019) A high-gain approach to event-triggered control with applications to motor systems. IEEE Trans Ind Electron 66(8):6281–6291
29. Su H, Hu Y, Karimi HR, Knoll A, Ferrigno G, De Momi E (2020) Improved recurrent neural network-based manipulator control

with remote center of motion constraints: experimental results. Neural Netw 131:291–299

30. Su H, Mariani A, Ovur Salih E, Menciassi A, Ferrigno G, De Momi E (2021) Towards teaching by demonstration for robot-assisted minimally invasive surgery. IEEE Trans Autom Sci Eng. https://doi.org/10.1109/TASE.2020.3045655

31. Su H, Qi W, Hu Y, Karimi HR, Ferrigno G, De Momi E (2020) An incremental learning framework for human-like redundancy optimization of anthropomorphic manipulators. IEEE Trans Ind Inf. https://doi.org/10.1109/TII.2020.3036693

32. Wang W, Huang H, Zhang L, Su C (2020) Secure and efficient mutual authentication protocol for smart grid under blockchain. Peer-to-Peer Netw Appl 1–13

33. Xu Y, Yang C, Zhong J, Wang N, Zhao L (2018) Robot teaching by teleoperation based on visual interaction and extreme learning machine. Neurocomputing 275:2093–2103

34. Yang C, Chen C, He W, Cui R, Li Z (2019) Robot learning system based on adaptive neural control and dynamic movement primitives. IEEE Trans Neural Netw Learn Syst 30(3):777–787

35. Yang C, Zeng C, Cong Y, Wang N, Wang M (2019) A learning framework of adaptive manipulative skills from human to robot. IEEE Trans Ind Inf 15(2):1153–1161

36. Yang C, Huang D, He W, Cheng L (2020) Neural control of robot manipulators with trajectory tracking constraints and input saturation. IEEE Trans Neural Netw Learn Syst 1–12 (2020). https://doi.org/10.1109/TNNLS.2020.3017202

37. Zeng C, Yang C, Cheng H, Li Y, Dai S (2020) Simultaneously encoding movement and sEMG-based stiffness for robotic skill learning. IEEE Trans Ind Inf 17(2):1244–1252

38. Zhang T, McCarthy Z, Jow O, Lee D, Chen X, Goldberg K, Abbeel P (2018) Deep imitation learning for complex manipulation tasks from virtual reality teleoperation. In: IEEE international conference on robotics and automation (ICRA). IEEE, pp 5628–5635

39. Zhong J, Peniak M, Tani J, Ogata T, Cangelosi A (2019) Sensorimotor input as a language generalisation tool: a neurorobotics model for generation and generalisation of noun-verb combinations with sensorimotor inputs. Auton Robot 43(5):1271–1290