



An Incremental Learning Framework to Enhance Teaching by Demonstration Based on Multimodal Sensor Fusion

Jie Li¹, Junpei Zhong², Jingfeng Yang³ and Chenguang Yang^{4*}

¹ Key Laboratory of Autonomous Systems and Networked Control, School of Automation Science and Engineering, South China University of Technology, Guangzhou, China, ² School of Science and Technology, Nottingham Trent University, Nottingham, United Kingdom, ³ Shenyang Institute of Automation Guangzhou Chinese Academy of Sciences, Guangzhou, China, ⁴ Bristol Robotics Laboratory, University of the West of England, Bristol, United Kingdom

OPEN ACCESS

Edited by:

Bin Fang,
Tsinghua University, China

Reviewed by:

Eiji Uchibe,
Advanced Telecommunications
Research Institute International
(ATR), Japan

Jian Huang,

Huazhong University of Science and
Technology, China

Mingjie Dong,

Beijing University of Technology, China

*Correspondence:

Chenguang Yang
cyang@ieee.org

Received: 14 January 2020

Accepted: 09 July 2020

Published: 27 August 2020

Citation:

Li J, Zhong J, Yang J and Yang C
(2020) An Incremental Learning
Framework to Enhance Teaching by
Demonstration Based on Multimodal
Sensor Fusion.
Front. Neurobot. 14:55.
doi: 10.3389/fnbot.2020.00055

Though a robot can reproduce the demonstration trajectory from a human demonstrator by teleoperation, there is a certain error between the reproduced trajectory and the desired trajectory. To minimize this error, we propose a multimodal incremental learning framework based on a teleoperation strategy that can enable the robot to reproduce the demonstration task accurately. The multimodal demonstration data are collected from two different kinds of sensors in the demonstration phase. Then, the Kalman filter (KF) and dynamic time warping (DTW) algorithms are used to preprocessing the data for the multiple sensor signals. The KF algorithm is mainly used to fuse sensor data of different modalities, and the DTW algorithm is used to align the data in the same timeline. The preprocessed demonstration data are further trained and learned by the incremental learning network and sent to a Baxter robot for reproducing the task demonstrated by the human. Comparative experiments have been performed to verify the effectiveness of the proposed framework.

Keywords: incremental learning network, teaching by demonstration, teleoperation, data fusion, robot learning

INTRODUCTION

With the development of control theory and sensor technology, robots have been widely applied in various fields, especially in industry and social service. It plays an increasingly vital role in human daily life, such as entertainment, education, and home service, etc. In most cases (Billard et al., 2008; Yang et al., 2018; Fang et al., 2019), robots need to learn and execute many complex and repetitive tasks, which include learning the motion skills from observing humans performing these tasks, also known as teaching by demonstration (TbD). TbD is an efficient approach to reduce the complexity of teaching a robot to perform new tasks (Billard et al., 2008; Yang et al., 2018). With this approach, a human tutor demonstrates how to implement a task to a robot easily (Ewerthon et al., 2019). Then, the robot learns the key features from human demonstration and repeats it by itself. Obviously, the main issue of robot learning is how to learn more critical features from the demonstration to fulfill a certain task well. Therefore, it is essential to take account of some learning methods to learn much more useful features effectively. In this sense, robot learning contains two tasks: motion perception based on multiple sensors and features learning with efficient methods. Different modalities of sensors can enable obtaining an accurate description of the target motions

and enrich the information (Chavez-Garcia and Aycard, 2016), and the learning methods promote to learn the desirable features. In this paper, we developed a novel robot learning framework to enhance the performance of TbD. This framework combines the superiority of the incremental learning network and the multiple sensors fusion.

Multimodal sensor fusion is a promising technique to create more accurate, more complete, or more dependable data with less uncertainty (Elmenreich, 2002; Haghighat et al., 2011) to enrich the features of demonstration data. Compared with individual sensor data, the multi-sensor fusion data have a distinctive preponderance in four general aspects (Mitchell, 2007). First, a higher resolution and richer semantic become possible for usage in the representation of the data. Second, the fused sensory data or data from disparate sources can reduce the uncertainty of information than these sources are used individually. Besides, a more all-sided view regarding the object is allowed for a coherent space to enhance the completeness of the information. Last, if the data are noisy or have errors, the fusion process will reduce or eliminate noise and errors. Hence, the data through the fusion process are possible to achieve the desired result with enhanced reliability, extended parameter coverage, and improved resolution (Fung et al., 2017). The system with multiple sensors provides immense opportunities for applications in a wide variety of areas. Applications that benefit from the sensor fusion technology cover many engineering fields which include internet of things (Din et al., 2015; Bijarbooneh et al., 2016), automation systems (Iyengar et al., 2003; Caterina et al., 2015), computer vision (Eitel et al., 2015), target tracking (Smith and Singh, 2006), health care (Medjahed et al., 2011; Koshmak et al., 2016), mechatronics (Luo and Chang, 2012), and robotics (Chung et al., 2011).

Recently, the multimodal sensor fusion is widely engaged in human-robot interaction (HRI) to enhance the performance of interaction (Gui et al., 2017; Argyrou et al., 2018; Deng et al., 2018; Fang et al., 2019; Li C. et al., 2019). Gui et al. (2017) designed a multimodal rehabilitation HRI system, which combines the electroencephalogram (EEG)-based HRI and electromyography (EMG)-based HRI to assistant gait pattern, to enhance active participation of users for gait rehabilitation and to accomplish abundant locomotion modes for the exoskeleton. Argyrou et al. (2018) proposed a human-robot collaborative monitoring system that can fuse data from multiple sources to estimate the execution status of the tasks more accurately. Deng et al. (2018) proposed an improved HRI by fusing the operator's gesture and speech to control the movements of a robot. The fusion of gesture and speech improved the accuracy, efficiency, and naturalness of the proposed system. Li C. et al. (2019) developed an augmented reality interface based on HRI that the Kalman filter (KF) algorithm was used to fuse the position and velocity signals from the Leap Motion sensor and the Kinect sensor to improve the tracking performance, aiming to provide an easier and accurate interaction. Wan et al. (2017) developed an intelligent system to teach robots to do object assembly through multimodal vision for next-generation industrial assembly. Zeng et al. (2019) proposed a TbD system to teach the robots to learn specific tasks based on multiple sensor

fusion. Compared with single modal data, the multimodal data provide a more rich and complementary information source to facilitate the diversity of robot TbD. These applications benefit from sensor fusion technology because of multi-sensor-based data fusion algorithms. Due to the varieties of the nature of the fusion process, different algorithms are used to enable the different levels of sensor fusion, such as KF (Kalman, 1960), support vector machine (SVM) (Cortes and Vapnik, 1995; Waske and Benediktsson, 2007), particle filter (Crisan and Doucet, 2002), Bayesian inference method (Khaleghi et al., 2013), fuzzy sensor fusion approach (Gibson et al., 1994), and artificial neural network (Hu, 2010), etc. Studies showed that the KF is ideally suited to coping with multi-sensor estimation and data fusion problems. This is mainly because the algorithm runs best with well-defined state descriptions (such as positions, velocities) and for states where observation and time-propagation models are also well-understood. In this paper, the KF is used to fuse the positions and velocities of a humanoid robot to achieve an overall complete description of the joint positions with high accuracy and fewer uncertainties.

Sensor fusion can enable to obtain more accurate demonstration data, while effective learning methods can learn more desired features of data. A deep learning neural network, as a kind of popular feature learning algorithm, has been successfully applied in various fields because of its powerful approximation capability (Ciresan et al., 2012; Marblestone et al., 2016; Sze et al., 2017). Although this advantage makes it apply in amounts of areas, it often needs a large number of datasets to train the network. Due to this, a complicated network structure is needed to deal with them, and then the network will suffer from a time-consuming process. Apart from that, the network is also faced with the issue that entire retraining when new samples are inputted. Considering these problems of deep structure learning methods, Chen and Liu (2017) proposed an incremental learning method, which provides an alternative way for deep structure neural network (Liu and Chen, 2017). The incremental learning network can rapidly learn and model the target system without a retraining process if new samples are fed into it. Also, the structure of this network can be expanded flexibly in a wide sense. Like a deep structure neural network, the approximation capability of an incremental learning network is universal (Chen et al., 2018). Hence, it has been successfully engaged in different fields employing efficient modeling and fast learning ability. These applications mainly involved two aspects: classification and regression. Most researchers employ this algorithm in various kinds of classification (Zhang et al., 2018; Li J. et al., 2019). For example, Zhang et al. (2018) applied it to recognize facial expression to improve the accuracy of recognition. Based on this method, Wang et al. (2018) integrated it with the convolution neural network to classify EEG emotion which achieves the highest average recognition accuracy. The applications, which are involved in different curves fitting, were seldom. Luo et al. (2019) used it to estimate human intention by predicting the force of human hand. Chen et al. (2018) have proved that compared with function approximation and time series prediction, the incremental learning algorithm is superior to other learning algorithms, such as SVM, least squared

SVM, and extreme learning machine (ELM), on regression performance. It is noted that the incremental learning network and ELM algorithm (Huang et al., 2004) are similar among these methods. Both networks have the structure of a single layer. Also, both networks are thought to have potential advantages in learning rate and generalization ability (Huang et al., 2006; Chen et al., 2018). Apart from that, the incremental learning network can be employed in other scenarios, such as fault diagnosis (Zhao et al., 2019) and monkey oculomotor decision decoding (Shi et al., 2020). However, this method is seldom used in HRI to improve the performance of robot learning.

For the TbD system, we can teach a robot to move as the desired trajectory. However, human movement is not always necessarily optimal for the robot when it tries to repeat and accomplish a task. Therefore, teaching a robot remotely, there will be some deviations between the robot's trajectory and the target trajectory. Through learning based on a neural network, the robot's trajectory can approach the target trajectory. To achieve that, the incremental learning algorithm is used to learn the fused features of a certain task from different sensors to enhance the learning performance. Then, experiments are performed to verify the effectiveness of the proposed multimodal framework.

The main contribution of this paper is to develop a framework that integrates the advantages of the multiple modal information fusion with the approximation capability of the incremental learning algorithm to enhance the performance of the TbD system. The remainder of the paper is organized as follows. The System Outline section presents the whole architecture of the proposed framework. The details of the data collection, preprocessing, and learning methods are introduced in the Methodology section. The Experiments and Results section describes the experimental settings and explains the results of the experiments. The experimental results are discussed in the Discussion section. The Conclusions and Future Work section concludes this work.

SYSTEM OUTLINE

System Description

The proposed framework of the TbD is shown in **Figure 1**, which consists of three modules: the human demonstration module, the learning module, and the robot execution module.

Human demonstration module: This module, which is a virtual demonstration system, allows the human demonstrator to control the Baxter robot in Virtual Robot Experimentation Platform (V-REP) *via* human joint information. The human joint information including joint angles and joint angular velocities is recorded by the Kinect sensor and Myo armbands separately.

The learning module: This module includes two steps: data preprocessing and incremental learning. The target of data preprocessing is to align the time series information of the demonstrated tasks in the same timeline. After that, an incremental learning method is used to learn the preprocessed data.

The robot execution module: The main function of this module enables the robot to complete the task with the learned

data from the training module. To this end, a specific task will be performed by a robot to verify the effectiveness of the proposed framework.

System Principle

The principle of the overall system based on the proposed method with multimodal sensor data fusion is presented in **Figure 2**. As shown in **Figure 2**, it consists of a Kinect sensor, two Myo armbands, and a Baxter robot. Kinect sensor is a motion capture device which is used to capture the motion of the human body. Myo armband, as a wearable device, is used to capture the human joint angular velocities. Baxter is a versatile semi-humanoid robot which is equipped with several advanced sensing technologies (including force, position, and torque sensing) which allow it to be applied in scientific research. V-REP is a powerful open-source robot simulator with an integrated development environment, distributed control architecture, and rich user interface to make it be an ideal platform for robot simulations. The remote application programming interface (API) in V-REP can control the robot simulation from an external application or remote hardware. This work will simulate the Baxter robot and control it by two developed API clients in V-REP.

Figure 3 shows the communication links of the virtual TbD system. It is noted that the data collected from the Kinect sensor and Myo armbands are separately recorded by two computers. Two sensors both can recognize human hand gestures. To capture the joint angles and angular velocities simultaneously, the hand state is used to control the start or end of the data collecting. When the human demonstrator's hand state is open, the data of joint angles and angular velocities will be recorded and saved in different files. Instead, the data collecting work will stop.

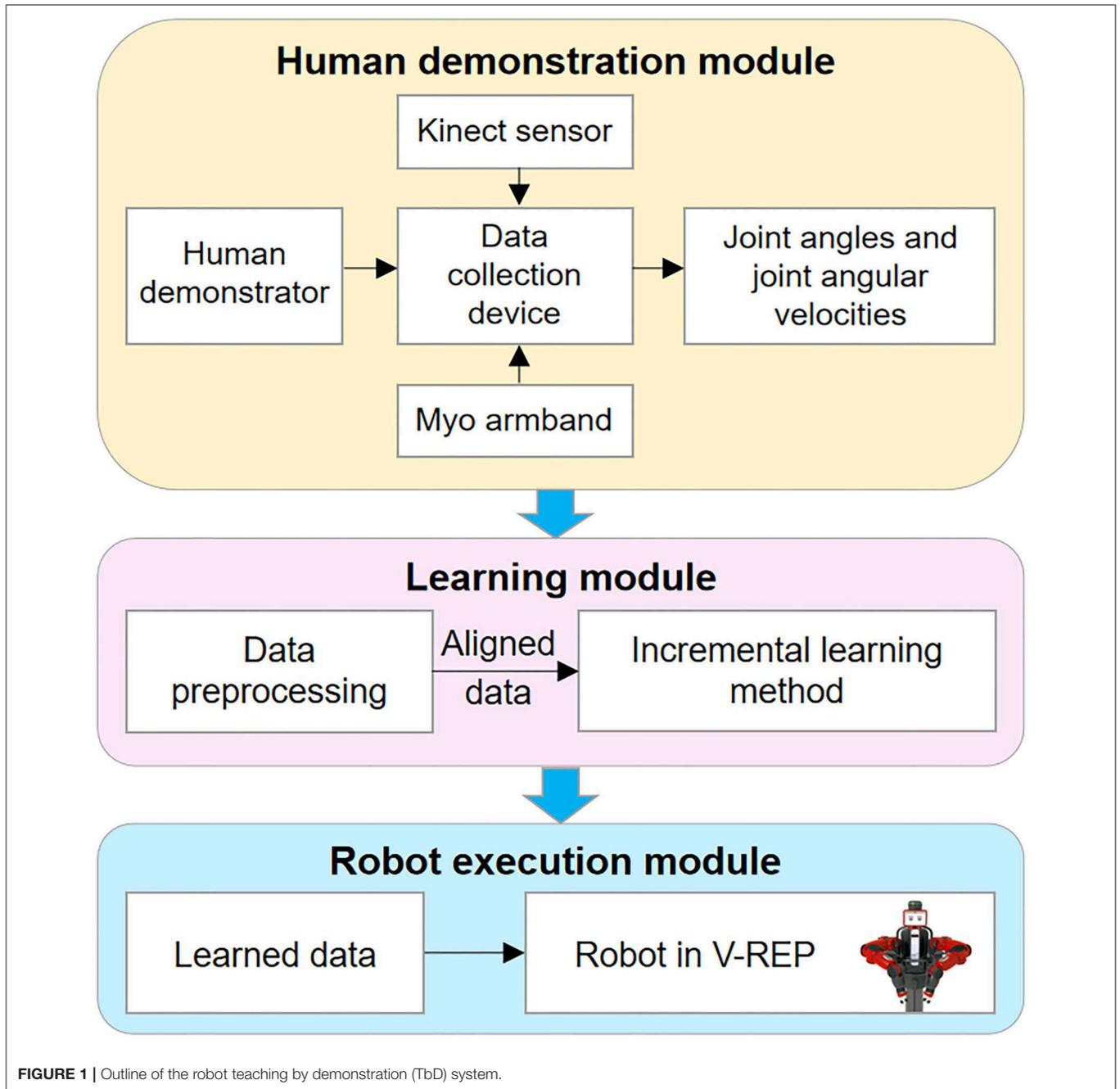
According to the designed human demonstration model, joint angles and angular velocities are recorded from the multiple demonstrations based on a specific task. Then, the raw demonstration data will be preprocessed and learned by the robot learning module. After that, the learned data are transferred to the Baxter robot in V-REP by MATLAB for execution. We can verify the effectiveness of the proposed method by the execution result of the Baxter robot.

METHODOLOGY

The proposed incremental learning framework includes three processes: data collection, data preprocessing, and data learning method, which correspond to the three modules mentioned above. In this section, the data collection and preprocessing processes will be introduced, and the details of the incremental learning network and the multi-sensor fusion algorithm KF also will be given.

Data Collection

In this section, we will introduce how to capture the human joint angles and angular velocities using a Kinect sensor and two Myo armbands in detail, respectively.



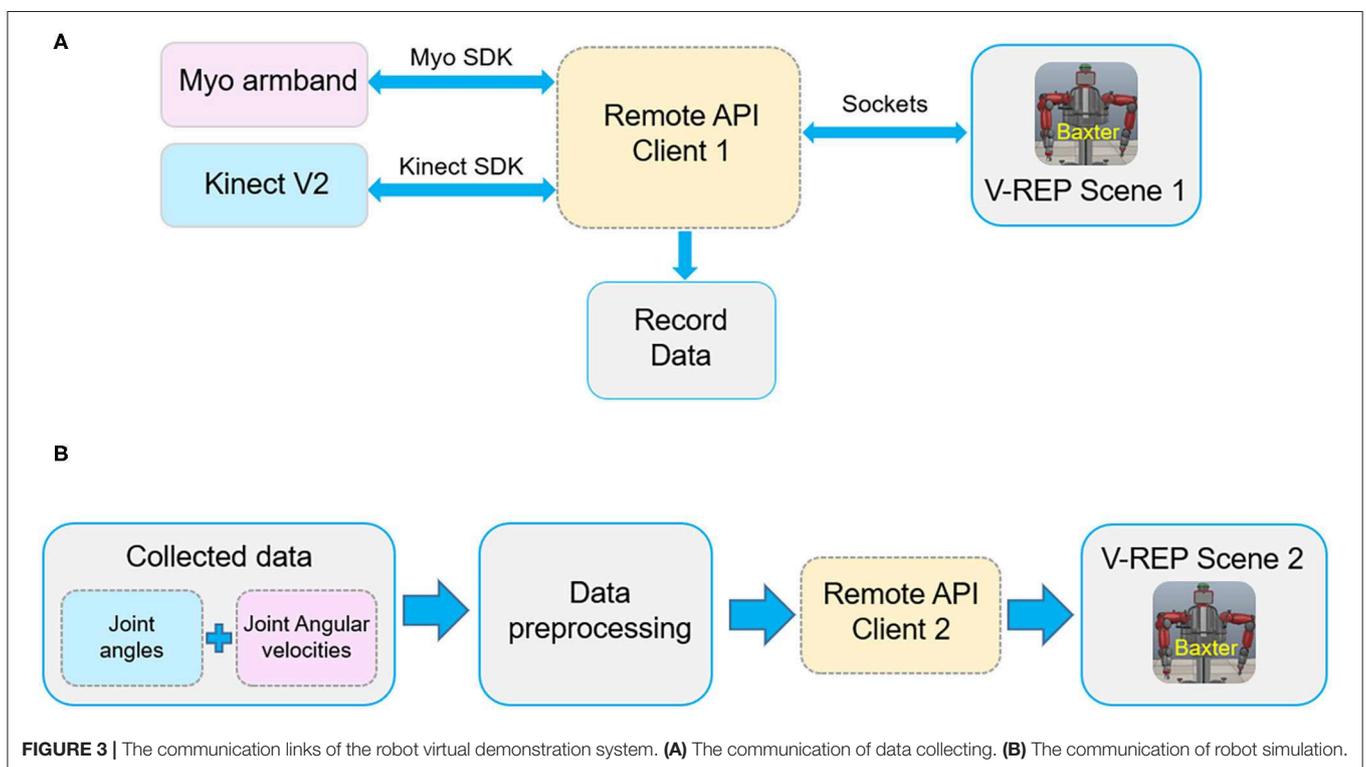
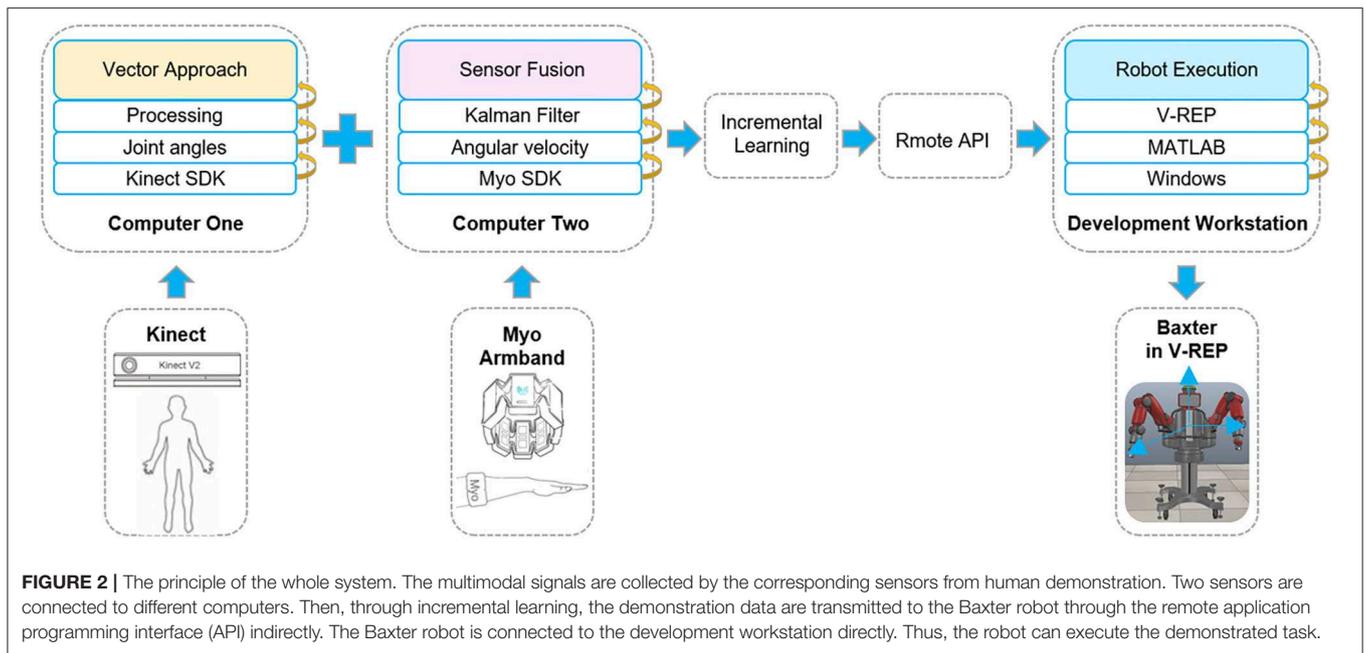
Calculation of Joint Angles Using the Space Vector Approach

Since we can get the three-dimensional (3D) joint coordinates of a human body using the Kinect sensor, the key to obtain the joint angles is how to convert these coordinates into corresponding angles. This problem can be addressed by the space vector approach. As we know, the distance between two specified 3D points $A(x_a, y_a, z_a)$ and $B(x_b, y_b, z_b)$ can be calculated by the following equation:

$$d_{AB} = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2 + (z_b - z_a)^2} \quad (1)$$

Essentially, the distance d_{AB} is equal to the norm of the vector $\vec{AB} = (x_b - x_a, y_b - y_a, z_b - z_a)$. In a 3D space, the law of cosines can be used to calculate the angles between two known vectors. In the Kinect coordinate, a joint can be expressed as a vector. So, the angle between joint 1 (\vec{PO}) and joint 2 (\vec{OQ}) can be computed as:

$$\cos(\vec{PO}, \vec{OQ}) = \frac{\vec{PO} \cdot \vec{OQ}}{|\vec{PO}| \cdot |\vec{OQ}|} \quad (2)$$



According to Equation (1), we can transform the coordinates returned by Kinect into corresponding vectors. Then, the angles of these vectors can be calculated by Equation (2).

The models of the full human body and the left arm are shown in **Figure 4**. The coordinate system of Kinect in Cartesian space is constituted by three directed straight lines AX, AY, and AZ, where point A is the origin of the coordinate. According to Equation (2),

the shoulder pitch angle $\angle AOC$ can be calculated by the vectors \vec{OA} and \vec{OC} from the position coordinates of points A, O, and C. The elbow pitch angle $\angle OCD$ is calculated using the same method. We can get the shoulder yaw angle $\angle JAK$ in a similar way. The difference is that the vectors \vec{AJ} and \vec{AK} are obtained by projecting vectors \vec{OB} and \vec{OC} to the XZ plane.

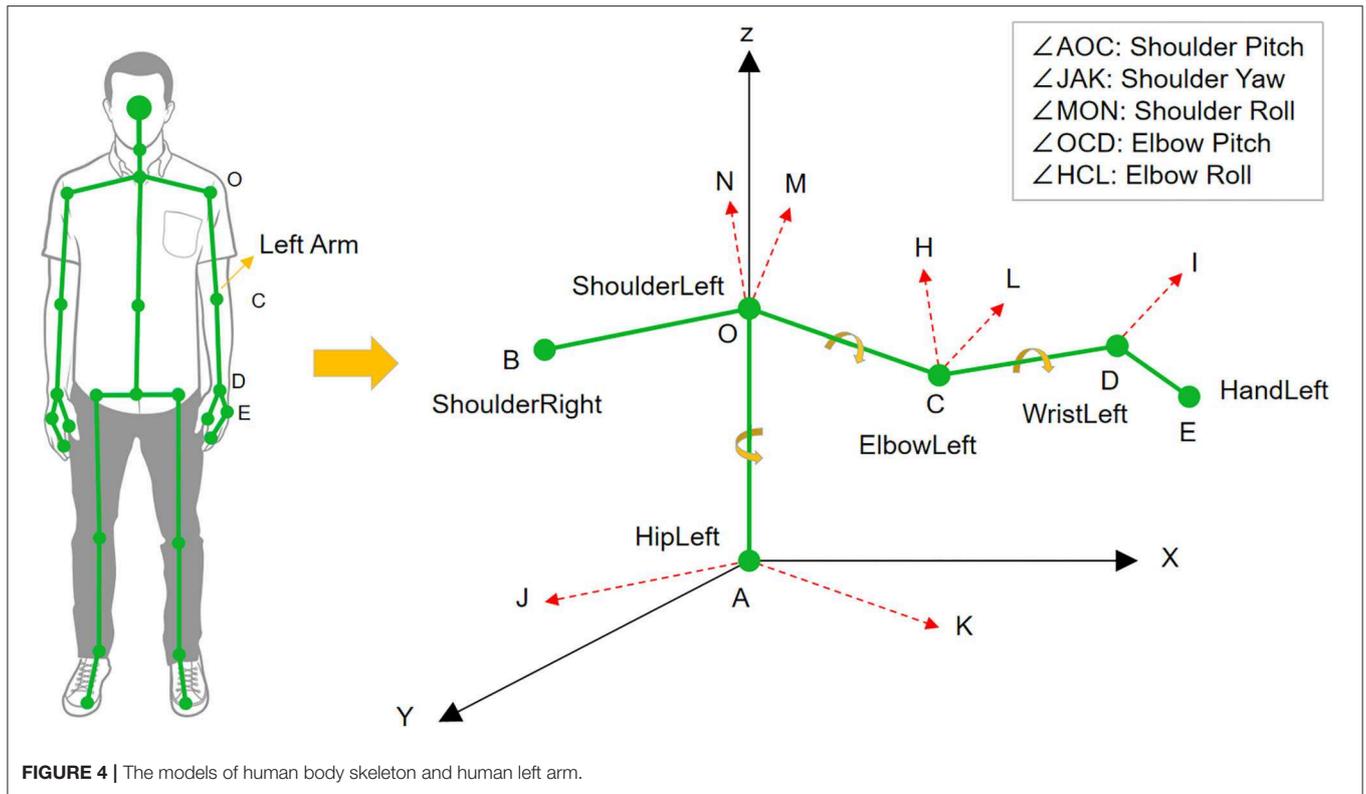


FIGURE 4 | The models of human body skeleton and human left arm.

To calculate the shoulder roll angle, the cross product is applied to get the normal vector of different planes. The normal vectors of the BOC and OCD planes can be calculated by:

$$\begin{cases} \vec{OM} = \vec{OB} \times \vec{OC} \\ \vec{CH} = \vec{CO} \times \vec{CD} \end{cases} \quad (3)$$

Then, translating the vector \vec{CH} along the vector \vec{CO} to point O can get the vector \vec{ON} . So, the calculation of the shoulder roll angle $\angle MON$ is addressed. Using the same method, we can get elbow roll angle $\angle HCL$, which is the angle between the planes of OCD and CDE.

Here, only three joint angles of the human arm involving the human shoulder and elbow are collected.

Calculation of Joint Angular Velocity From Myo Armband

To obtain the joint angular velocity, two Myo armbands are needed to wear on the user's upper arm and forearm. The quaternion method is used to obtain the joint angles. Then, the joint angular velocities can be computed based the difference of the joint angles. According to Yang et al. (2018), we can assume that the joint angle of the initial position is zero. When the user's arm is moved from a pose T to a new pose P , the angle from pose T to P is the rotation angle. For the pose P , pose T can be regarded as the initial pose, and the rotation angle is the joint angle.

Assume that the Myo armband's orientation is expressed by frame (x_0, y_0, z_0) in the initial position, the current orientation is expressed by frame (x_1, y_1, z_1) . Then, the angular velocities of the shoulder roll, shoulder yaw, and shoulder pitch can be obtained by the forearm armbands. The velocities of the elbow roll and pitch angles are acquired by the armbands worn on the upper arm. Thus, five joint angular velocities are obtained for each arm from a pair of Myo armbands.

Thus, we can obtain two different modalities information of human arm. After that, these joint angles and the joint angular velocities will be fused by the KF algorithm.

Data Preprocessing

The demonstration data from the Kinect sensor and Myo armband will be preprocessed before they are fed into the incremental learning method. Firstly, the data fusion method based on the KF is used to fuse the joint angles and joint angular velocities to obtain a more accurate and smooth dataset. Since the demonstration data are not matched in the timeline, then the dynamic time warping (DTW) algorithm is applied to align them. Here, the two preprocessing methods will be introduced briefly.

Data Fusion by Kalman Filter

KF, as one of the most powerful sensor fusion algorithms, can smooth noisy input data and optimize the estimation of the current state based on current measurements and the previously estimated state. These current measurements are often multiple sequential measurements from several sensors with noise. The

existing works have proved that the estimate of the system's state from multiple sensors is better than the estimate obtained from only one sensor (Gui et al., 2017). Therefore, the sensor fusion based on the KF is used to improve the accuracy of data.

This algorithm uses a series of state prediction and measurement update steps to update the state of the target object. The prediction and update steps are presented below. For a continuous simplified linear system, the dynamic model is described as follows (Davari et al., 2016):

$$\begin{aligned} \dot{x}(t) &= Fx(t) + Gu(t) + Mn(t) \\ z(t) &= Hx(t) + v(t) \end{aligned} \quad (4)$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ is the deterministic input vector, $z(t) \in \mathbb{R}^p$ is the measurement vector, $n(t) \in \mathbb{R}^q$ is the white noise term for the state vector with zero-mean and covariance S , and $v(t) \in \mathbb{R}^p$ is the noise term for measurement vector with zero-mean and covariance R . $F \in \mathbb{R}^{n \times n}$ and $G \in \mathbb{R}^{n \times m}$ are both system matrices. M and H are parameter matrices related to the noise and measurement, respectively. The KF model of the linear system can be expressed by the following equations (Simon, 2006):

$$\begin{aligned} \hat{x}(t) &= F\hat{x}(t) + Gu(t) + K(t)[z(t) - H\hat{x}(t)] \\ K(t) &= \Sigma(t)H^TR^{-1} \\ \dot{\Sigma}(t) &= F\Sigma(t) + \Sigma(t)F^T + MSM^T - \Sigma(t)H^TR^{-1}H\Sigma(t) \end{aligned} \quad (5)$$

where $K(t)$ is the filter gain, $\hat{x}(t)$ is the state estimation of x , and $\Sigma(t)$ is the estimation of covariance.

For the above equations, we assume that $x(0)$, n , and v are uncorrelated to each other, and all the KF parameters are first order. If each joint of human arm is considered separately, we have $F = 0$, $G = 1$, $M = 1$, and $H = 1$. Thus, Equations (4, 5)

can be simplified as:

$$\begin{aligned} \dot{x}_i &= u_i + n_i \\ z_i &= x_i + v_i \end{aligned} \quad (6)$$

$$\begin{aligned} \dot{\hat{x}}_i &= u_i + K(t)[z_i - \hat{x}_i] \\ K &= \Sigma R^{-1} \\ \dot{\Sigma} &= S - \Sigma R^{-1} \Sigma \end{aligned} \quad (7)$$

where u_i is the i th joint angular velocity of the human arm, z_i is the i th joint position (or the joint angles), and \hat{x}_i is the fused data of the i th joint. Note that the parameters K , Σ , R , and S are scalar values.

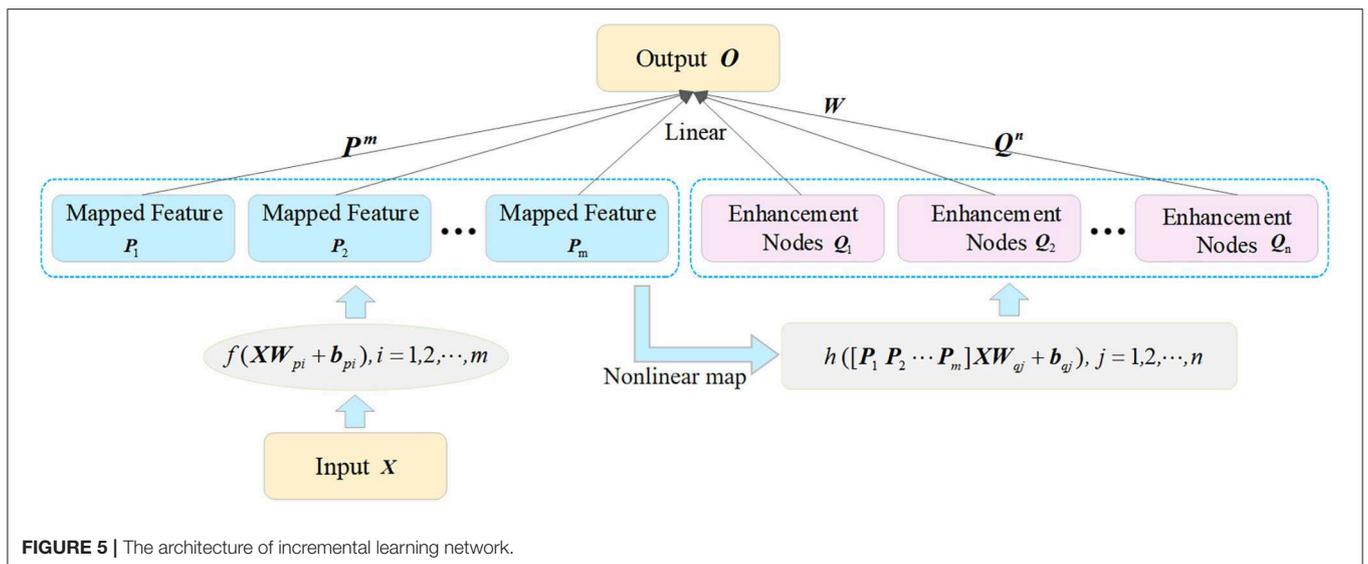
Data Preprocessing With Dynamic Time Warping

Through the human demonstration module, the angles and angular velocities of the human joints are collected from multiple demonstrations. As aforementioned, the time for every demonstration is not the same. We employ the DTW algorithm to align them in the same timeline.

DTW is a method to measure the similarity of two time series with different lengths. It has been widely used in processing the temporal sequences of video, audio, and graphics data. If two given temporal sequences g and k satisfy the boundary, monotonicity, and step size conditions, the objective of DTW can be transformed into the optimal match path problem between the two sequences. We expressed this optimal match path as:

$$DTW(y_1, y_2) = \min(d(y_1, y_2)) \quad (8)$$

where $d(y_1, y_2)$ represent the distance between sequences y_1 and y_2 . Then, the Dynamic programming is used to solve Equation (8). At the same time, an accumulated cost matrix E with the dimension of $m \times n$ is generated. The expression of matrix E is



written as follows:

$$E(l_1, l_2) = (y_1, y_2) + \begin{cases} 0 & \text{if } l_1 = 1 \text{ and } l_2 = 1 \\ E(l_1, l_2 - 1) & \text{else if } l_1 = 1 \text{ and } l_2 > 1 \\ E(l_1 - 1, l_2) & \text{else if } l_1 > 1 \text{ and } l_2 = 1 \\ \min(E(l_1, l_2 - 1), \\ E(l_1 - 1, l_2), \\ E(l_1 - 1, l_2 - 1)) & \text{otherwise} \end{cases} \quad (9)$$

where l_1 and l_2 are the length of the sequences y_1 and y_2 , respectively.

Incremental Learning Method

The incremental learning algorithm is essentially a single-layer neural network that the structure can be dynamically expanded in a wide sense. It is constructed based on the random vector functional link neural network (FLNN). The architecture of the incremental learning network is shown in **Figure 5**.

The input of this network is composed of two parts: the mapped features and the enhancement nodes. As shown in **Figure 5**, the original inputs are first transformed into a group of mapped features to extract the random features by some linear feature mappings. Then, the mapped features are extended to enhancement nodes by non-linear mappings. Further, the mapping features and the enhancement nodes in the input layer are both connected with the output linearly. Thus, the weights between the input layer and the output layer can be calculated by the ridge regression of the pseudo-inverse method.

The detailed process of the incremental network is presented as follows. For a given input dataset $\{X\}$ and m feature mapping function f_i , $i = 1, 2, \dots, m$, the i th mapped features can be calculated as:

$$P_i = f_i(XW_{p_i} + b_{p_i}), \quad i = 1, 2, \dots, m \quad (10)$$

where $X \in \mathbb{R}^{m \times n}$, m is the number of training samples; n is the size of each training sample; both the bias unit b_{p_i} and the weights W_{p_i} , which connect the original input and the mapped features, are randomly generated. It is noted that the functions f_i and f_l are equal for $i \neq l$. We denote the first i th groups of mapped features as $P^i \equiv [P_1 P_2 \dots P_i]$ and express the non-linear mappings connected the mapped features with enhancement nodes as h_j , $j = 1, 2, \dots, n$. Then, using the non-linear function h_j , the relationship between the mapped features P_i and Q_j , the enhancement nodes can be built. The j th group of enhancement nodes is expressed as:

$$Q_j = h_j(P^m W_{p_j} + b_{p_j}), \quad j = 1, 2, \dots, n \quad (11)$$

where W_{p_j} and b_{p_j} are randomly generated, and W_{p_j} are the weights connecting the mapped features and the enhancement nodes. Likewise, the first j th group of enhancement nodes is denoted as $Q^j \equiv [Q_1 Q_2 \dots Q_j]$. The enhancement nodes P_i together with the mapped features Q_j form the actual input of the incremental learning network $A \equiv [P_1, \dots, P_m, Q_1, \dots, Q_n] = [P^m Q^n]$. Hence, the output O of this network is computed as:

$$O = AW_m^n, \quad (12)$$

where the weights W_m^n connect the output layer and the input layer. Since the target output O is given, we can calculate the weights W_m^n as follows:

$$W_m^n = A^+ O, \quad (13)$$

Here, the rigid regression learning algorithm is used to solve the pseudo-inverse A in Equation (13). According to this algorithm, the pseudo-inverse A is obtained by the following equation:

$$A = \lim_{\lambda \rightarrow 0} (\lambda I + AA^T)^{-1} A^T, \quad (14)$$

Algorithm 1 presents the whole training process of the incremental learning network.

Algorithm 1 The procedure of the incremental learning network.

Input: Demonstration dataset X , mapped feature group m , and enhancement nodes group n .

Output: The parameter matrix W .

for $i = 1$ to m **do**

Randomly initialize the weights W_{p_i} and bias unit b_{p_i} ;

Calculate P_i according to Equation (10).

end

Set mapped features group $P^m \equiv [P_1 P_2 \dots P_m]$;

for $j = 1$ to n **do**

Randomly initialize the weights W_{p_j} and bias unit b_{p_j} ;

Calculate Q_j according to Equation (11).

end

Set the enhancement nodes group $Q^n \equiv [Q_1 Q_2 \dots Q_n]$;

Calculate weights W_m^n according to Equation (13).

As aforementioned, the ELM method and the incremental learning method both are single-layer neural networks, and the learning speed of two methods is also fast. For the incremental learning network, if the learning cannot reach the desired result, it can be addressed by inserting additional enhancement nodes in a wide sense not deep way to achieve a better performance. The increase of the enhancement nodes will result in the recalculation of weights. It is worth noting that only a part of the weights needs to be recalculated, not all weights. The new weights are calculated by the following equations:

$$W_m^{n+1} = \begin{bmatrix} W_m^n - DB^T O \\ O \end{bmatrix}, \quad (15)$$

where $C = h_{n+1}(P^m W_{p_{n+1}} + b_{p_{n+1}}) - A^n D$, $D = (A^n)^+ h_{n+1}(P^m W_{p_{n+1}} + b_{p_{n+1}})$, and

$$B^T = \begin{cases} (C)^+ & \text{if } C \neq 0 \\ (I + D^T D) B^T (A^n)^+ & \text{if } C = 0 \end{cases}, \quad (16)$$

Note that 0 is zero matrix, and O is the output of the network.

For the ELM network, the solution to improve performance is to increase the number of hidden layer neurons, which

results in more connecting parameters. Thus, a great number of parameters including all weights need to be updated. It means that the ELM network suffers from a complete relearning process. In this respect, the incremental learning network is different. Besides, the incremental learning network is allowed to increase the number of input samples without relearning all samples. Likewise, only the newly added samples need to be learned by the incremental learning network. It also implies that the incremental learning network can adapt to new data without forgetting its existing knowledge, instead of relearning all samples. This is the difference in the structural expansion between the two networks.

Furthermore, the mapped features of the incremental learning network are randomly generated from the original input dataset $\{X\}$. In other words, the mapped features are the results of feature representation for the original input data. Feature representation can capture the efficient characteristics of the data to achieve outstanding performance in supervised learning tasks (Chen et al., 2018). It explains why the incremental learning network can learn the desired features. Also, it shows that the actual input data of the two networks are different. This implies the difference between the two networks from another aspect.

As stated above, the motivation to use the incremental learning algorithm is its convenience in a specific scene and feature learning ability.

EXPERIMENTS AND RESULTS

Experimental Setup

We test our method on the Baxter robot. The experimental system is shown in **Figure 6**. The hardware devices consist of a Baxter robot, a Kinect sensor, and two Myo armbands. Based on

the platform, two tasks (wiping and pushing) are performed to verify the effectiveness of the proposed TbD system.

In the wiping task, the robot in V-REP follows human motion to raise his left arm, move toward the left, and then put it down along the path it passed. The difficulty of this task is that the trajectories of up and down motions should be consistent. The repetitive processes with the same task are performed more than 16 times.

The wiping task is performed under the following three conditions:

- **Condition 1: with Kinect sensor data and incremental learning method.** The demonstration data are only collected from the Kinect sensor but without Myo armbands. Through processing of DTW, the incremental learning network is used to learn it. There is no data fusion in this condition.
- **Condition 2: with two sensors data (Kinect and Myo armband) and incremental learning method.** The demonstration data are collected from both Kinect and Myo armbands. In this case, the sensor fusion process is added before data preprocessing with DTW algorithm. Later, the preprocessed data are learned by the incremental learning network.
- **Condition 3: with two sensors data (Kinect and Myo) and ELM algorithm.** The demonstration data collection and processing processes are the same as the second condition. The difference is that the learning method of these data is ELM algorithm (Huang et al., 2004) instead of the incremental learning network.

In summary, the first condition is to show the performance of incremental learning network with only joint angle information but without joint angular velocities. The second condition is set

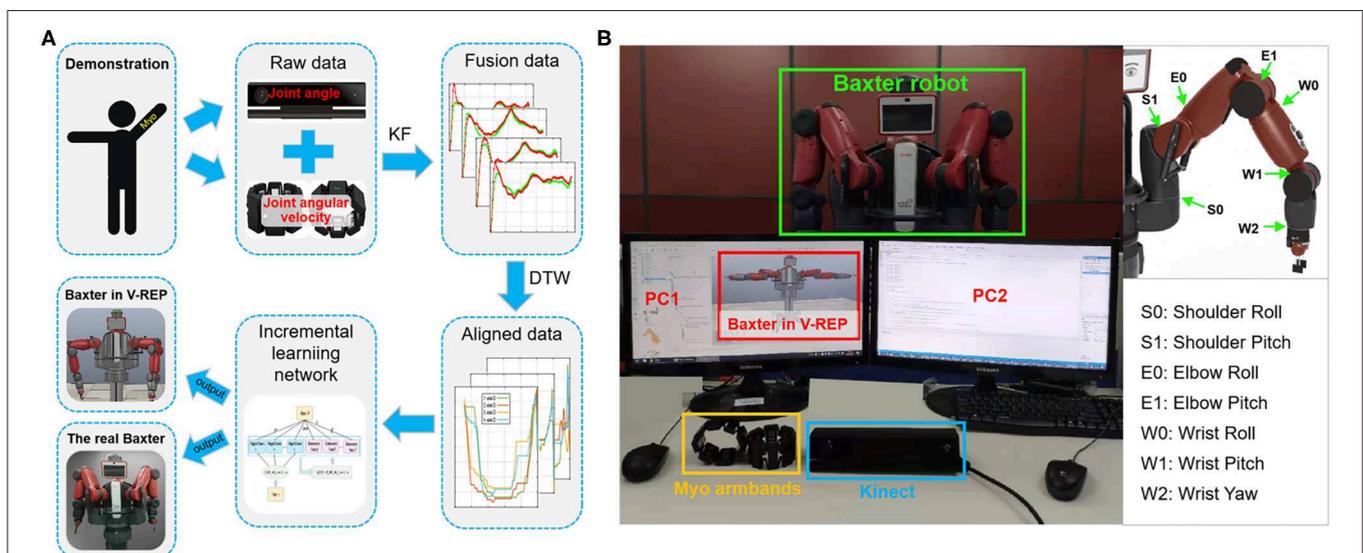
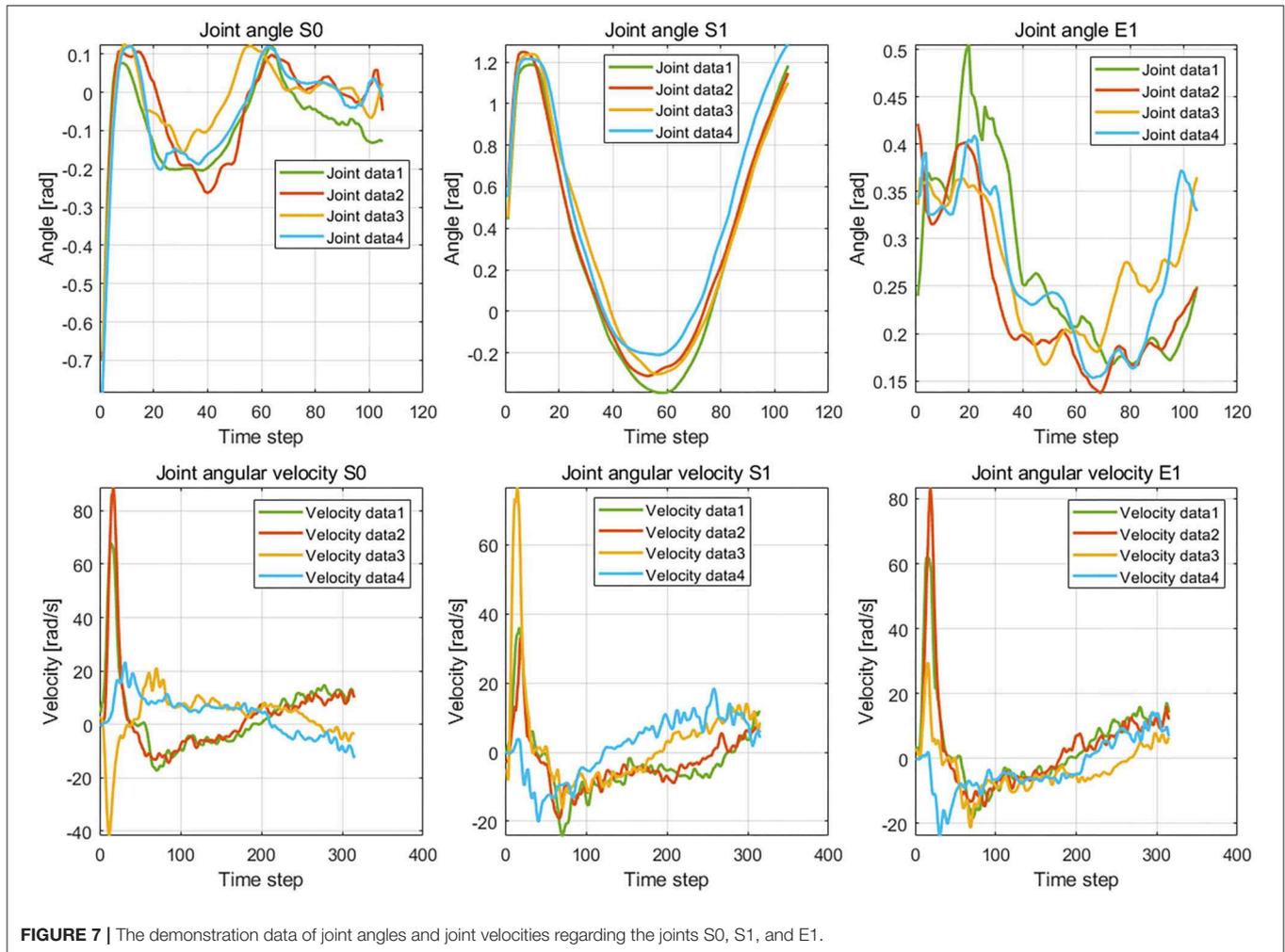


FIGURE 6 | The experimental system. **(A)** The diagram of the experimental system. **(B)** The experimental platform of the demonstration phase. During demonstration, the joint angles and joint angular velocities of the human arm are collected simultaneously by Kinect and Myo armband. Then, the raw demonstration data will be fused and aligned in the same timeline by the Kalman filter (KF) and dynamic time warping (DTW) algorithms in turn. After that, the incremental learning network is applied to learn the processed data. During the robot learning phase, the learned data are directly sent to the robot model in Virtual Robot Experimentation Platform (V-REP) and the real Baxter robot.



to validate the proposed incremental learning framework with sensor fusion, while the third condition is to test the performance of the ELM network with sensor fusion.

To find the optimal number of feature mapping group m and enhancement nodes group n , we change m and n from 1 to 50 for the incremental learning network. The result shows that the highest accuracy appears when m and n are 6 and 8, respectively. A similar test is conducted for the ELM algorithm. We can get that when the number of hidden layer neurons is 11, the ELM network has the best accuracy.

Experimental Results

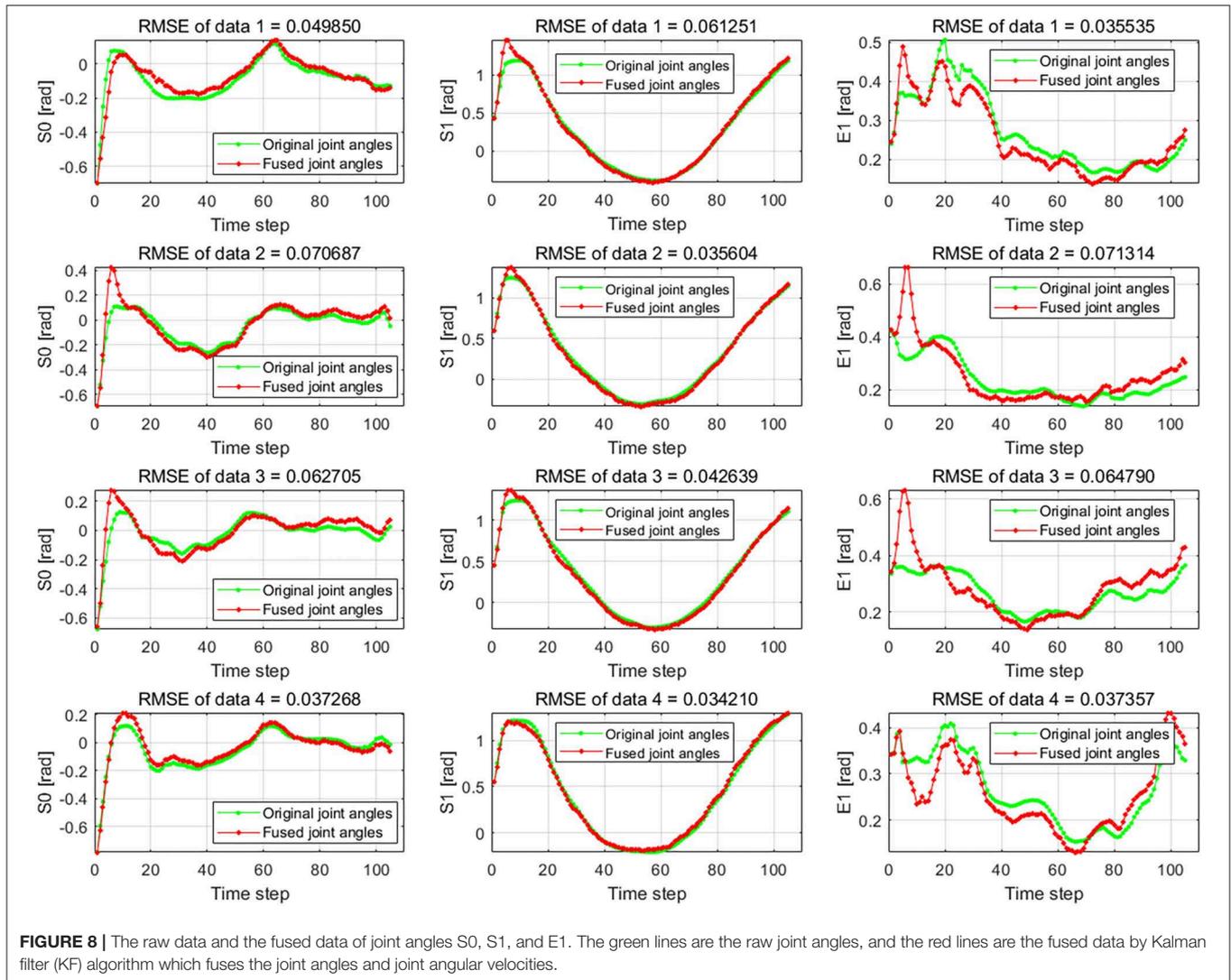
Results of the Wiping Task

The results of the wiping tasks are shown in **Figures 7–10**. In the demonstration phase, the raw multimodal data are recorded by different sensors. **Figure 7** presents four randomly selected samples of human demonstrations. The results of preprocessing are shown in **Figures 8, 9**. **Figure 8** shows the curves of the fusion datasets. Note that there are deviations between the raw joint angles and the fused data. **Figure 9** displays the aligned results of the fused datasets. Compared with the demonstration data without alignment in the timescale, the aligned data also retain

the primary characteristics through the aligning process by DTW algorithm. The aligned results prepare for the next training and learning of the neural network.

The difference between the first and second conditions is that the demonstration samples are different for the incremental learning network. Since deviations between the raw original joint angles and the fusion joint angles exist, the results of DTW aligning will be different. We can observe it from the images of the second and third rows in **Figure 9**. The dimension of the original raw demonstration data is 105. After processing by DTW, the dimensions of these datasets are 367 and 355 for the raw data and fusion data, respectively.

The trajectories learned by the incremental learning network and ELM network are shown in **Figures 10A–C** with red dotted lines. For the Baxter robot, all changes of seven joint angles are aimed to obtain a desired trajectory of the end effector because the robot execution eventually depends on the end effector. We recorded the trajectory of the end effector in Cartesian space during robot execution, which is shown in **Figure 10D**. Seven joint angles of the real Baxter's left arm are also recorded and shown in **Figure 10E**.



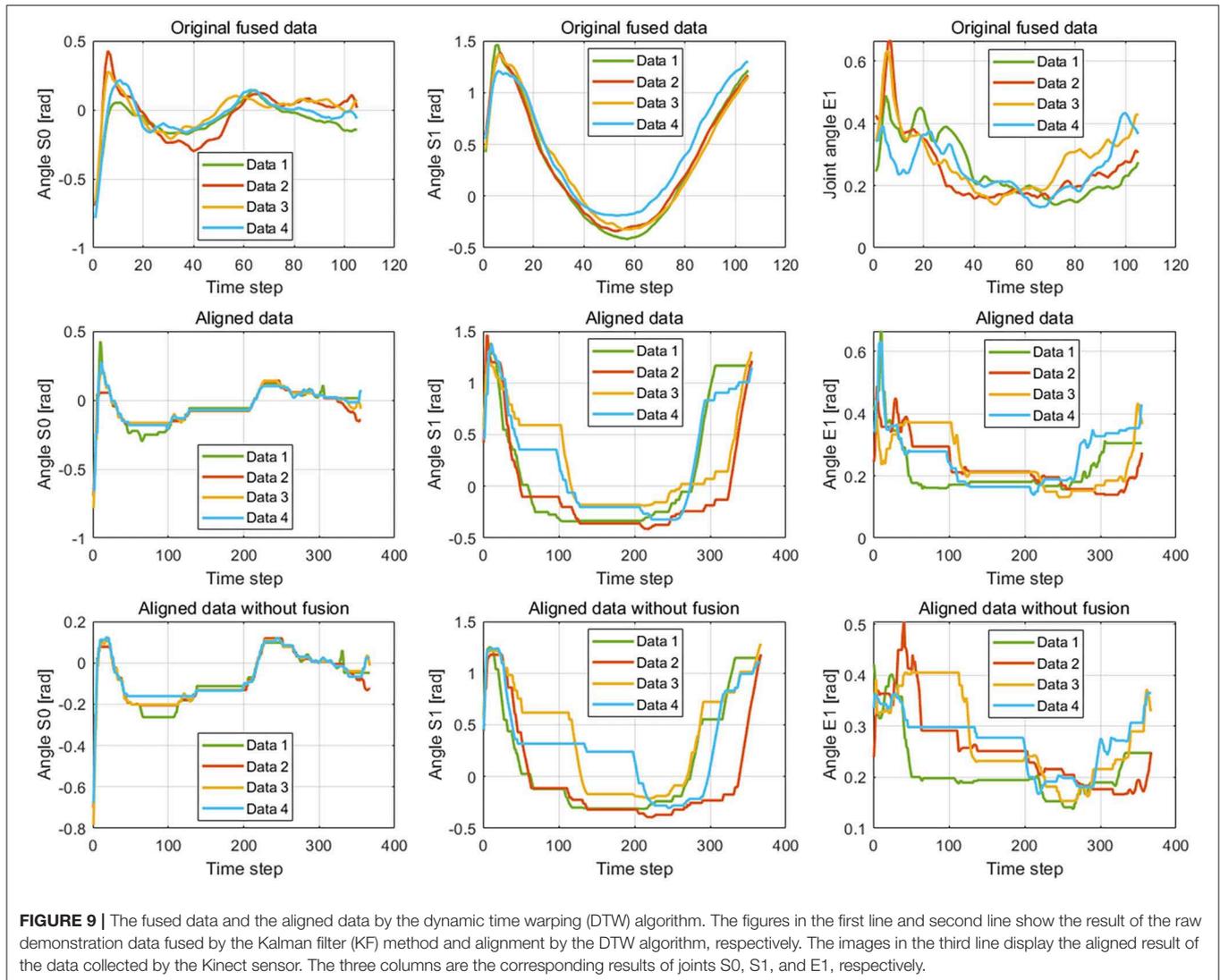
Based on the learned results by the incremental learning network, the Baxter robot can implement the wiping task. The robot implementation includes a simulation experiment of the Baxter robot in V-REP and an experiment for real Baxter robot. And the wiping task covers four directions of continuous and smooth movement: up, down, right, and left.

Results of the Pushing Task

To test the generalization ability of the proposed method, a pushing task is performed. The pushing task requires the robot to push two square workpieces over on the desk in sequence. In other words, the robot should firstly push the workpiece on the right to the desk. During pushing, the robot cannot touch the workpiece on the left. Then, the robot pushes the right one to the desk. The short distance between the two workpieces makes it more difficult for the robot to complete this task. Because the aims of the pushing task and the wiping task are different, the pushing task is only conducted under the above conditions

2 and 3. The experimental steps are the same as the wiping task. The experimental results are ultimately reflected in the trajectory of the robot end effector, which determines whether the robot can complete the demonstrated task. Hence, **Figure 11** only presents the trajectories of the real Baxter robot in the simulation scene and real environment, but not the results of data preprocessing.

As shown in **Figure 11**, the distance between the two tasks is very close. Any deviation in the trajectory of the robot end effector could result in the robot failing to complete the task. Nevertheless, we can find that the Baxter robot can complete the pushing task well from the experimental results. For this task, the results can directly reflect the performance of the two learning methods. Since the result of the robot end effector's trajectory under condition 3 is a failure to complete the pushing task, the corresponding result is not displayed in **Figure 11**. Also, the results for the wiping tasks are undiscussed in the next section. These results illustrate that the proposed method can not only improve the performance of TbD but also be applied in learning



different tasks for the robot. It implies that the proposed method has good generalization ability.

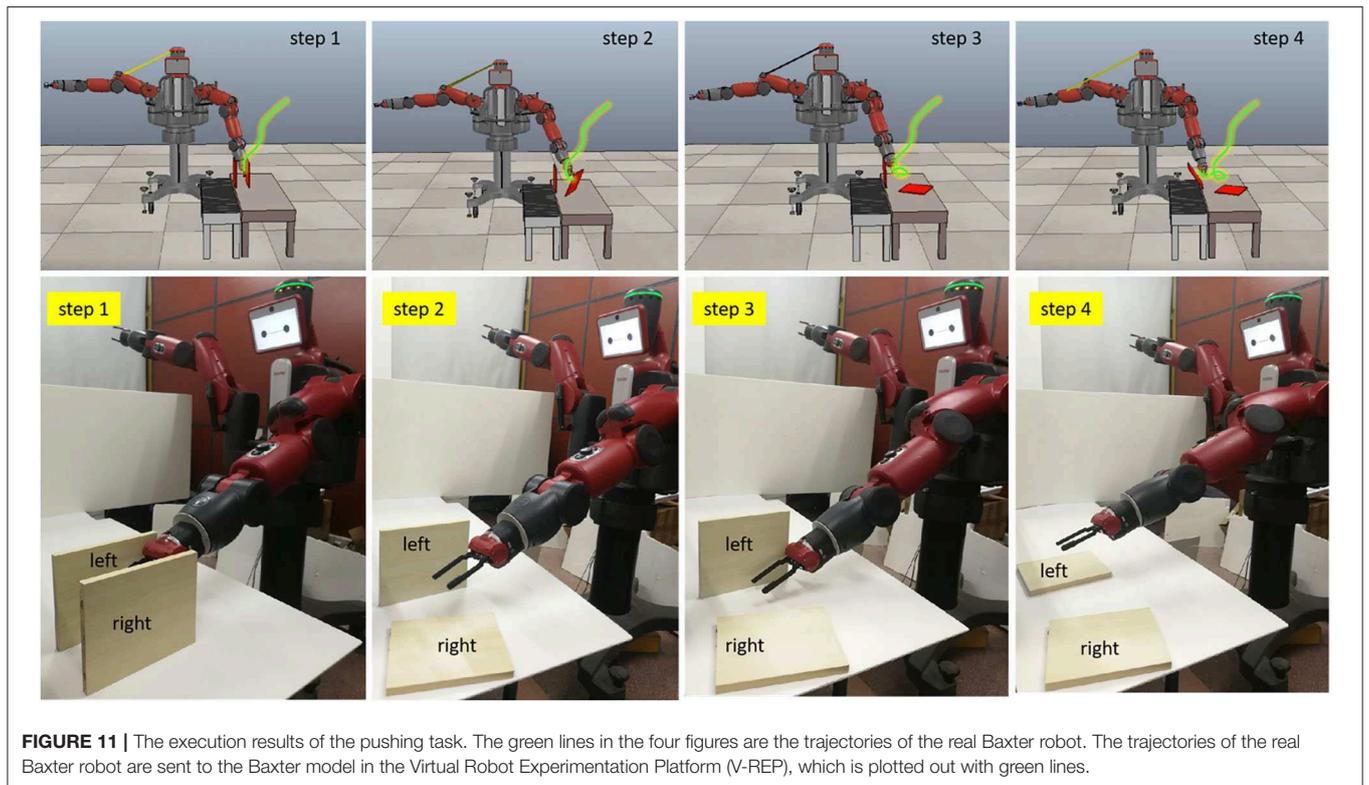
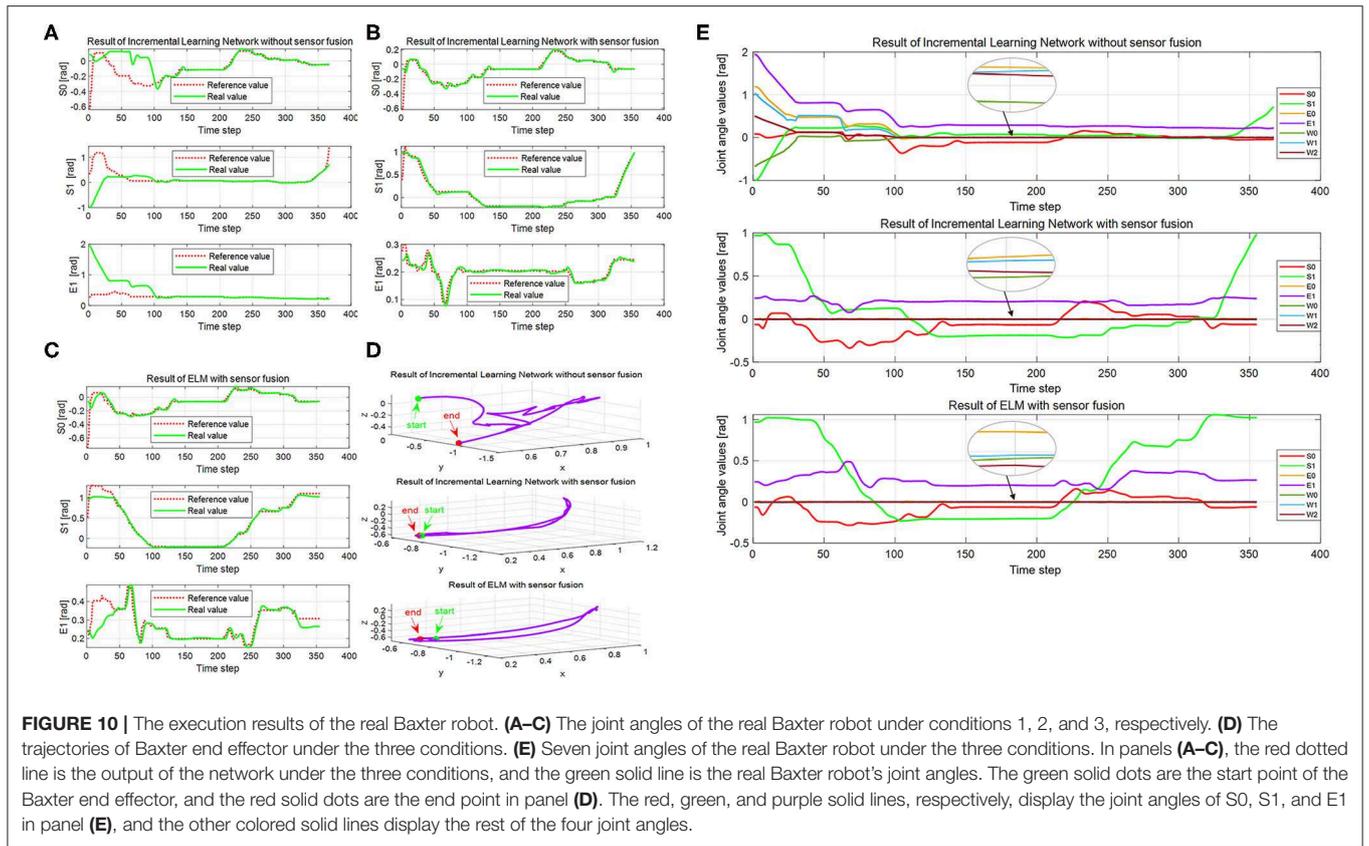
DISCUSSION

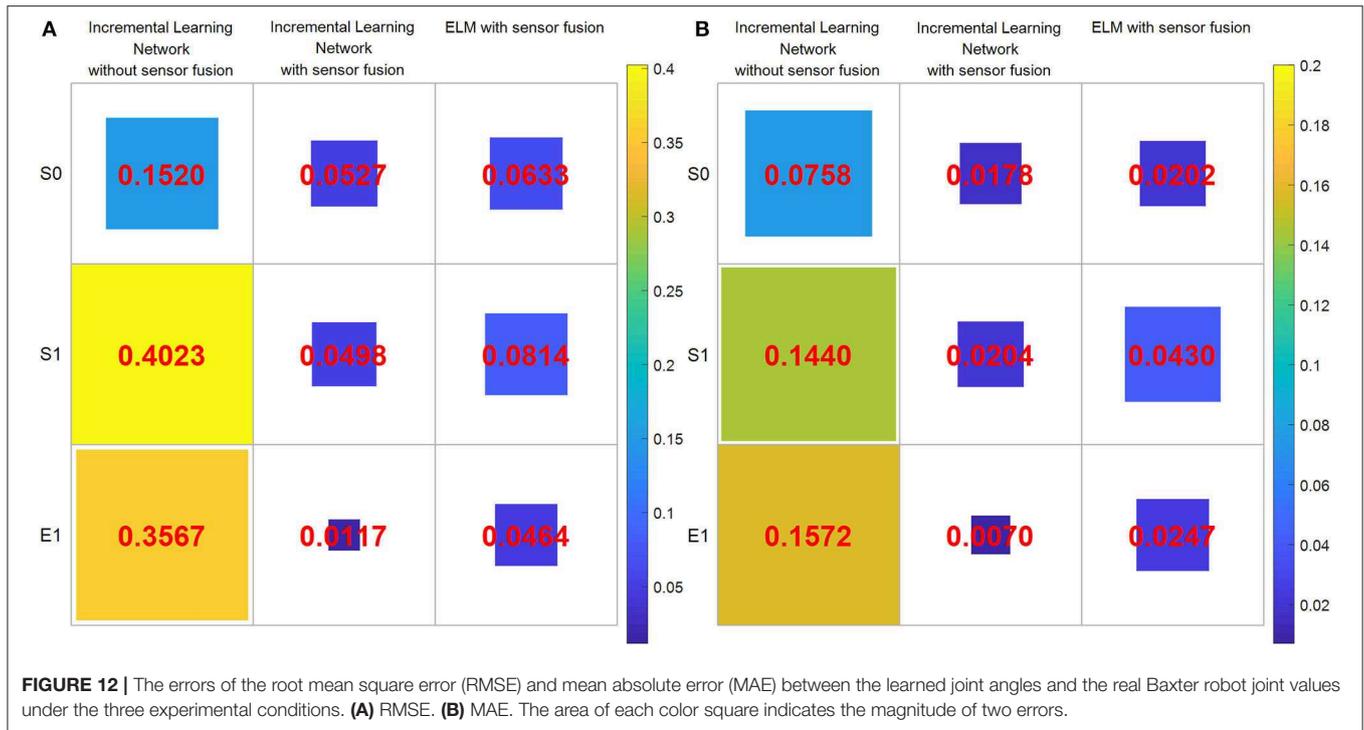
The purpose of this work is to investigate the practical effect of the proposed method on robot TbD, as well as to explore the impact on the result considering the fusion of multiple modality information. It is noted that only the results of the wiping task are discussed in this section. Because the pushing task requires more accurate execution for the Baxter robot, the performance of the learning method can be directly judged by the execution results of the Baxter robot. The experimental results of the pushing task clearly illustrate that using the incremental learning method can enable the robot to complete the pushing task well, while the ELM algorithm cannot.

Firstly, we examine the effectiveness of multimodal data fusion by comparing the results in **Figures 10A,B** under the first and

second conditions. It is clear that the bias between the reference trajectories and the real trajectories of the first condition is much larger than the second one, especially in the start phase of the interval (0, 100). And the curves of the real trajectories are inconsistent for the first and second conditions in **Figures 10A,B**. For joints S0 and S1, the trend of reference trajectories is almost the same under conditions 2 and 3. Concerning the joint E1, the differences between the curves are especially evident under the same conditions. In **Figure 10A**, the maximum difference value between the reference and the real value is already close to 2. But this value is not more than 0.6 under the second condition, which can be observed from **Figure 10B**. The trajectories of the real Baxter robot regarding the joints S0, S1, and E1 illustrate that the multimodal data fusion can promote a result that is much closer to the reference values.

Next, we discuss the superiority of the proposed method in comparison with another robot learning method. **Figures 10B,C** show the corresponding results by using the proposed method and the ELM method. The red dotted lines are the learning





result of the two networks. Obviously, the trends of red reference trajectories in **Figures 10B,C** are consistent. It indicates that both of the methods can learn the features of the joint angles to complete the wiping task. However, the biases between the reference trajectories and the real trajectories of the two methods are different. To analyze the result quantitatively, we calculate the mean absolute error (MAE) and root mean square error (RMSE), which are shown in the second and third columns of **Figures 12A,B**. The RMSE is calculated as follows:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{t=1}^N (y_t - \hat{y}_t)^2}, \quad (17)$$

where N is the size of the demonstration sequences, \hat{y}_t is the value fused by the KF algorithm, and y_t is the value measured by the Kinect sensor.

The calculation of MAE is as follows:

$$\text{MAE} = \frac{1}{N} \sum_{t=1}^N |y_t - \hat{y}_t|, \quad (18)$$

Noteworthy, RMSE and MAE are the errors between the reference data (namely, the output of the incremental learning network or the learned trajectories) and that of the real Baxter robot data. The areas of the squares using the proposed method are less than that of squares using the ELM method. The RMSE and MAE results imply that the errors of the incremental learning method are smaller. This shows that the experiment performance of the proposed method is better than the ELM method. We can

also find that the maximum error under the three conditions is from the result without data fusion. It also implies that through data fusion, both errors are diminished notably.

As aforementioned, the difficulty of the wiping task is how to ensure that the trajectories of upward and down motion are consistent. We find that the result under condition 1 is worst, and the trajectory is disordered in **Figure 10D**. On the contrary, the trajectories under the second and third conditions are smooth and orderly. Furthermore, the result of the second condition is better than that of the third one. It also proves that data fusion can improve the experiment performance in another way. Besides, we compute the distance between the start point and the end point for three conditions, which are 0.8825, 0.0135, and 0.0778, respectively. For the results, the shorter the distance, the better the performance. It is obvious that the distance for the second condition is the shortest. That is to say, both the trajectories of the end effector and the distance illustrate that the proposed method is better. These results suggest that the proposed method is superior to the ELM method, not only the joint angles but also the trajectories of the end effector.

Lastly, we compare all recorded joint angles of the robot's left arm under the three conditions. The desired result is that the other four joint angles are approximate to zero except for S0, S1, and E1, which is shown in **Figure 10E**. The four joint angles in the interval (0, 100) under the first condition are much bigger than zero, and then they gradually trend to zero. However, the four joint angles are much closer to zero under the second and third conditions from beginning to end. It also shows that sensor fusion can decrease demonstration errors.

To sum up, the demonstration data with multimodal information can significantly improve the experiment performance, and the proposed method can achieve a better execution result with smaller errors. This is probably because data fusion is beneficial to obtain a demonstration dataset close to the real value. At the same time, the KF algorithm smooths the raw data to some extent. All of these help the real robot move smoothly and efficiently. On the other hand, the incremental learning network can learn more effective features to enhance TbD performance.

CONCLUSIONS AND FUTURE WORK

In this paper, we propose an incremental learning framework to learn demonstration features by integrating different modality data. Using the proposed method and the KF algorithm, the TbD performance is remarkably improved. To verify the proposed method, comparative experiments involving the incremental learning network and ELM algorithm were conducted based on a Baxter robot in a real physical environment. Through the experiments, the robot achieved a better result with smaller errors using the proposed network on the basis of two modality information fusions. The effectiveness of the proposed method was verified by analyzing the learned data and the real robot data in comparison with ELM methods. As a result, the proposed method can learn more critical features to get the desired result. Since the TbD system is based on two modality information fusions, we also verify the effect of multimodal integration on the real robot. Compared with the results of single-modality data, the multimodal data with sensor fusion can achieve a better performance. It implies that the fusion

of modality information is beneficial to improve the accuracy of data. To test the generalization of the proposed method, a pushing task is performed. The successful experiment results show that the proposed method has the generalization ability in TbD. In the future, integrating modality information from different types of sensors, e.g., force, will be addressed to perform complex tasks online. We will further explore the complete time of a specific task for the real robot by employing other methods. Also, how to reduce the effect of demonstrations from different people on the experimental results is taken into account.

DATA AVAILABILITY STATEMENT

The datasets generated for this study are available on request to the corresponding author upon reasonable request.

AUTHOR CONTRIBUTIONS

JL, JZ, and CY conceived of the presented idea. JL implemented the framework and conducted the experiment. JL and CY contributed to results analysis. JL contributed to manuscript writing, original draft. JZ, JY, and CY contributed to review and provide critical feedback. All authors had read the manuscript and agreed with its content.

ACKNOWLEDGMENTS

The authors thank the participants of Mr. Jiahao Zhang, Mr. Cao Zeng and Mr. Xianfa Xue for their valuable time in data collecting.

REFERENCES

- Argyrou, A., Giannoulis, C., Sardelis, A., Karagiannis, P., Michalos, G., and Makris, S. (2018). A data fusion system for controlling the execution status in human-robot collaborative cells. *Procedia CIRP* 76, 193–198. doi: 10.1016/j.procir.2018.01.012
- Bijarbooneh, F. H., Du, W., Ngai, E. C.-H., Fu, X., and Liu, J. (2016). Cloud-assisted data fusion and sensor selection for internet of things. *IEEE Internet Things J.* 3, 257–268. doi: 10.1109/JIOT.2015.2502182
- Billard, A., Calinon, S., Dillmann, R., and Schaal, S. (2008). “Robot programming by demonstration,” in *Springer Handbook of Robotics*, eds B. Siciliano and O. Khatib (Berlin: Springer Press), 1371–1394. doi: 10.1007/978-3-540-30301-5_60
- Caterina, M., Bühlhoff, H. H., and Stegagno, P. (2015). “Autonomous vegetation identification for outdoor aerial navigation,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems* (Hamburg), 3105–3110.
- Chavez-Garcia, R. O., and Aycard, O. (2016). Multiple sensor fusion and classification for movingobject detection and tracking. *IEEE Trans. Intell. Transp. Syst.* 17, 525–534. doi: 10.1109/TITS.2015.2479925
- Chen, C. P., and Liu, Z. (2017). Broad learning system: an effective and efficient incremental learning system without the need for deep architecture. *IEEE Trans. Neural Networks Learn. Syst.* 29, 10–24. doi: 10.1109/TNNLS.2017.2716952
- Chen, C. P., Liu, Z., and Feng, S. (2018). Universal approximation capability of broad learning system and its structural variations. *IEEE Trans. Neural Networks Learn. Syst.* 30, 1191–1204. doi: 10.1109/TNNLS.2018.2866622
- Chung, H., Oh, S., Shim, D. H., and Sastry, S. S. (2011). Toward robotic sensor webs: algorithms, systems, and experiments. *Proc. IEEE* 99, 1562–1586. doi: 10.1109/JPROC.2011.2158598
- Ciresan, D., Meier, U., and Schmidhuber, J. (2012). “Multi-column deep neural networks for image classification,” in *IEEE Conference on Computer Vision and Pattern Recognition* (Providence, RI), 3642–3649. doi: 10.1109/CVPR.2012.6248110
- Cortes, C., and Vapnik, V. (1995). Support-vector networks. *Mach. Learn.* 20, 273–297. doi: 10.1007/BF00994018
- Crisan, D., and Doucet, A. (2002). A survey of convergence results on particle filtering methods for practitioners. *IEEE Trans. Signal Process.* 50, 736–746. doi: 10.1109/78.984773
- Davari, N., Gholami, A., and Shabani, M. (2016). Multirate adaptive kalman filter for marine integrated navigation system. *J. Navig.* 70, 1–20. doi: 10.1017/S0373463316000801
- Deng, Y., Li, F., and Xin, H. (2018). Research on multimodal human-robot interaction based on speech and gesture. *Comput. Electr. Eng.* 72, 443–454. doi: 10.1016/j.compeleceng.2018.09.014
- Din, S., Ghayvat, H., Paul, A., Ahmad, A., Rathore, M. M., and Shafi, I. (2015). “An architecture to analyze big data in the internet of things,” in *9th International Conference on Sensing Technology* (Auckland), 677–682. doi: 10.1109/ICSensT.2015.7438483
- Eitel, A., Springenberg, J. T., Spinello, L., Riedmiller, M., and Burgard, W. (2015). “Multimodal deep learning for robust RGB-d object recognition,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems* (Hamburg), 681–687. doi: 10.1109/IROS.2015.7353446

- Elmenreich, W. (2002). *Sensor fusion in time-triggered systems* (Ph.D. Thesis). Vienna University of Technology, Vienna, Austria.
- Ewerton, M., Arenz, O., Maeda, G., Koert, D., Kolev, Z., Takahashi, M., et al. (2019). Learning trajectory distributions for assisted teleoperation and path planning. *Front. Robot. AI* 6:89. doi: 10.3389/frobt.2019.00089
- Fang, B., Wei, X., Sun, F., Huang, H., Yu, Y., and Liu, H. (2019). Skill learning for human-robot interaction using wearable device. *Tsinghua Sci. Technol.* 24, 654–662. doi: 10.26599/TST.2018.9010096
- Fung, M. L., Chen, M. Z. Q., and Chen, Y. H. (2017). “Sensor fusion: a review of methods and applications,” in *The 29th Chinese Control and Decision Conference* (Chongqing), 3853–3860. doi: 10.1109/CCDC.2017.7979175
- Gibson, R. E., Hall, D. L., and Stover, J. A. (1994). “An autonomous fuzzy logic architecture for multisensor data fusion,” in *International Conference on Multisensor Fusion and Integration for Intelligent Systems* (Las Vegas, NV), 143–150. doi: 10.1109/MFI.1994.398450
- Gui, K., Liu, H., and Zhang, D. (2017). Toward multimodal human-robot interaction to enhance active participation of users in gait rehabilitation. *IEEE Trans. Neural Syst. Rehabil. Eng.* 25, 2054–2066. doi: 10.1109/TNSRE.2017.2703586
- Haghighat, M. B. A., Aghagolzadeh, A., and Seyedarabi, H. (2011). Multi-focus image fusion for visual sensor networks in DCT domain. *Comput. Electr. Eng.* 37, 789–797. doi: 10.1016/j.compeleceng.2011.04.016
- Hu, G. (2010). “Neural network applications in sensor fusion for a mobile robot motion,” in *WASE International Conference on Information Engineering* (Beidaihe), 46–49.
- Huang, G. B., Zhu, Q. Y., and Siew, C. K. (2004). “Extreme learning machine: a new learning scheme of feedforward neural networks,” in *IEEE International Joint Conference on Neural Networks* (Budapest), 985–990.
- Huang, G. B., Zhu, Q. Y., and Siew, C. K. (2006). Extreme learning machine: theory and applications. *Neurocomputing* 70, 489–501. doi: 10.1016/j.neucom.2005.12.126
- Iyengar, S. S., Sastry, S., and Balakrishnan, N. (2003). Foundations of data fusion for automation. *IEEE Instrum. Meas. Mag.* 6, 35–41. doi: 10.1109/MIM.2003.1251481
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Trans. ASME J. Basic Eng.* 82, 35–45. doi: 10.1115/1.3662552
- Khaleghi, B., Khamis, A., Karray, F. O., and Razavi, S. N. (2013). Multisensor data fusion: a review of the state-of-the-art. *Inf. Fusion* 14, 28–44. doi: 10.1016/j.inffus.2011.08.001
- Koshmak, G., Loutfi, A., and Linden, M. (2016). Challenges and issues in multisensor fusion approach for fall detection: review paper. *J. Sens.* 2016, 1–12. doi: 10.1155/2016/6931789
- Li, C., Fahmy, A., and Sienn, J. (2019). An augmented reality based human-robot interaction interface using kalman filter sensor fusion. *Sens.* 19:4586. doi: 10.3390/s19204586
- Li, J., Zhong, J., Chen, F., and Yang, C. (2019). “An incremental learning framework for skeletal-based hand gesture recognition with leap motion,” in *9th IEEE International Conference on CYBER Technology in Automation, Control and Intelligent System* (Suzhou), 13–18. doi: 10.1109/CYBER46603.2019.9066761
- Liu, Z., and Chen, C. P. (2017). “Broad learning system: structural extensions on single-layer and multi-layer neural networks,” in *International Conference on Security, Pattern Analysis, and Cybernetics* (Shenzhen), 136–141. doi: 10.1109/SPAC.2017.8304264
- Luo, J., Liu, C., Li, Y., and Yang, C. (2019). “A framework of human impedance recognition,” in *25th International Conference on Automation and Computing* (Lancaster), 1–6. doi: 10.23919/ICoAC.2019.8895250
- Luo, R. C., and Chang, C. C. (2012). Multisensor fusion and integration: a review on approaches and its applications in mechatronics. *IEEE Trans. Ind. Inf.* 8, 49–60. doi: 10.1109/TII.2011.2173942
- Marblestone, A. H., Wayne, G., and Kording, K. P. (2016). Toward an integration of deep learning and neuroscience. *Front. Comput. Neurosci.* 10:94. doi: 10.3389/fncom.2016.00094
- Medjahed, H., Istrate, D., Boudy, J., Baldinger, J. L., and Dorizzi, B. (2011). “A pervasive multi-sensor data fusion for smart home healthcare monitoring,” in *IEEE International Conference on Fuzzy Systems* (Taipei), 1466–1473. doi: 10.1109/FUZZY.2011.6007636
- Mitchell, H. B. (2007). *Multi-Sensor Data Fusion: An Introduction*. Heidelberg: Springer Science and Business Media Press.
- Shi, Z., Chen, X., Zhao, C., He, H., Stuphorn, V., and Wu, D. (2020). Multi-view broad learning system for primate oculomotor decision decoding. *IEEE Trans. Neural Sys. Rehabil. Eng.* 1. doi: 10.1109/tnsre.2020.3003342
- Simon, D. (2006). “The continuous-time kalman filter,” in *Optimal State Estimation*, ed D. Simon (Hoboken, NJ: John Wiley & Sons, Inc.), 229–262. doi: 10.1002/0470045345.ch8
- Smith, D., and Singh, S. (2006). Approaches to multisensor data fusion in target tracking: a survey. *IEEE Trans. Knowl. Data Eng.* 18, 1696–1710. doi: 10.1109/TKDE.2006.183
- Sze, V., Chen, Y.-H., Yang, T.-J., and Emer, J. S. (2017). Efficient processing of deep neural networks: a tutorial and survey. *Proc. IEEE* 105, 2295–2329. doi: 10.1109/JPROC.2017.2761740
- Wan, W., Lu, F., Wu, Z., and Harada, K. (2017). Teaching robots to do object assembly using multi-modal 3D vision. *Neurocomputing* 259, 85–93. doi: 10.1016/j.neucom.2017.01.077
- Wang, X., Zhang, T., Xu, X., Chen, L., Xing, X., and Chen, C. L. P. (2018). “EEG emotion recognition using dynamical graph convolutional neural networks and broad learning system,” in *IEEE International Conference on Bioinformatics and Biomedicine* (Madrid), 1240–1244. doi: 10.1109/BIBM.2018.8621147
- Waske, B., and Benediktsson, J. A. (2007). Fusion of support vector machines for classification of multisensor data. *IEEE Trans. Geosci. Remote Sens.* 45, 3858–3866. doi: 10.1109/TGRS.2007.898446
- Yang, C., Chen, C., He, W., Cui, R., and Li, Z. (2018). Robot learning system based on adaptive neural control and dynamic movement primitives. *IEEE Trans. Neural Networks Learn. Syst.* 30, 777–787. doi: 10.1109/TNNLS.2018.2852711
- Zeng, C., Yang, C., Zhong, J., and Zhang, J. (2019). Encoding multiple sensor data for robotic learning skills from multimodal demonstration. *IEEE Access* 7, 145604–145613. doi: 10.1109/ACCESS.2019.2945484
- Zhang, T., Liu, Z., Wang, X., Xing, X., Chen, C. L. P., and Chen, E. (2018). “Facial expression recognition via broad learning system,” in *IEEE International Conference on Systems, Man, and Cybernetics* (Miyazaki), 1898–1902. doi: 10.1109/SMC.2018.00328
- Zhao, H., Zheng, J., Xu, J., and Deng, W. (2019). Fault diagnosis method based on principal component analysis and broad learning system. *IEEE Access* 7, 99263–99272. doi: 10.1109/ACCESS.2019.2929094

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2020 Li, Zhong, Yang and Yang. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.