2017-09

Neural learning and Kalman filtering enhanced teaching by demonstration for a Baxter robot

Li, Chunxu

10.23919/iconac.2017.8081985
2017 23rd International Conference on Automation and Computing (ICAC)
IEEE

Neural Learning and Kalman Filtering Enhanced Teaching by Demonstration for a Baxter Robot

Chunxu Li[1], Chenguang Yang*[1,2], Jian Wan[2], Andy Annamalai[3] and Angelo Cangelosi[2]

Abstract— **In this paper, Kalman filter has been successfully carried out to fuse the data obtained from a Kinect sensor and a pair of MYO armbands. To do this, the Kinect sensor is used to capture movements of operators which is programmed by Microsoft Visual Studio. Operator wears two MYO armbands with the inertial measurement unit (IMU) embedded to measure the angular velocity of upper arm motion for the human operator. Additionally a neural networks (NN) control upgraded Teaching by Demonstration (TbD) technology has been designed and it also has been actualized on the Baxter robot. A series of experiments have been completed to test the performance of the proposed technique, which has been proved to be an executed approach for the Baxter robot's TbD has been designed.**

I. INTRODUCTION

With all the rapid growth and development of robot technology, the utilization of robotic areas may be traditionally enhanced. The field of research regarding TbD (Teaching by Demonstration) attracted great attention from the public during the past decade. TbD has been utilized in industrial fields to promote the working efficiency, where the motor skills will work to transfer all the motor behaviours into tutees automatically when it has been have captured by a computer. The need for TbD has attracted an increasing number of schools, companies and also researchers devoting themselves into those areas.

Nowadays, human operators are required to acchieve the new motor skills with a high frequence owing to the fast development of technologies and tools [1]. Hence, it is significant for tutees to decrease the training process [2]. In [3], an approach was developed for transferring skills between tutor and tutee. In order to do this, the strategies of tutor was captured by a sensory-based computational model, and then the online master commands were produced for the learning process of the less-skilled tutees. According to [4], a research group used the neural network to record the human skills in a computer, and exam the ability of human to simulate the transfer skills between human tutor and tutees with a non-consuming time process.

In recent years, motion capture technology has developed rapidly, and has been widely used in some areas, such as, HRI (human-robot interation), computer animation and 3D film production [5]. It can accurately record the 3D motion trajectory of each part of the moving object [6]. Based on these trajectory information, people can accurately model the moving objects and semantic analysis of the movement or use, such as animation [3]. However, motion capture is an extremely time-consuming process [7]. The captured action data requires a lot of post-processing, which requires manual segmentation of the action sequences into segments as needed and identification of semantics for each segment.

In [8], an approach to overcome the shortcomings of Wiener filtering is proposed, which is called as Kalman filtering (KF). KF is widely used as it may estimate the past, current state, along with the future state signal, set up exact nature with the unknown model. Essentially, filtering is a technique of signal processing and transformation. Moreover, there are uncertainties of the robot, especially the dynamic uncertainties, which will influence the consequence of teaching and learning experience. There are many methods that are of help for developing the control plan of robotic voice to bargain with the dynamic instabilities, for example, Fuzzy Logic and Neural Networks (NN). The NN control continues to be widely examined in the discrete-time system [9], [10] as well as the continuous time control system [10], [11].

This paper builds up a strategy to compute the five required joints and angles by vector approach, then applies Kalman filter based sensor combination to yield a progression of more exact information with less mistakes. In order to do this, a Kinect based body following framework has been used additionally, MYO armbands worn on the operator's lower arm and upper arm are used to recognize and measure the angular rate of human arm development. Furthermore, the control frameworks have been successfully connected to NN for the teaching and learning process with the Baxter robot.

II. PRELIMINARIES

A. Kinect sensor

Kinect uses structured light coding to obtain the depth of the captured image, including an RGB camera, an infrared camera and an infrared transmitter [12]. Compared to traditional cameras, Kinect has low price, pixel level resolution and donot need additional computing because of the use of the image processor, which let to a lower configuration to achieve real-time requirements [13]. The coordinate arrangement of this space takes after a privilege gave tradition [14](as shown in Fig.1).

B. MYO Armband

Thalmic Labs has introduced a body-wearable device named MYO armband (shown in Fig.2), which allows users to easily manipulate computers, cell phones when they do

*Corresponding author: C. Yang. Email: cyang@theiet.org

This work was supported in part by Engineering and Physical Sciences Research Council (EPSRC) under Grants EP/L026856/2 and EP/J004561/1.

[1]Zienkiewicz Centre for Computational Engineering, Swansea University, UK. [2]Center for Robotics and Neural Systems, Plymouth University, UK. [3]University of Highlands and Islands, UK.

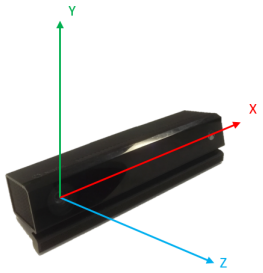

Fig. 1. Image of Kinect v2 sensor and its Camera space, modified from [15]

different gestures. MYO armband is designed with a muscle movement sensor inside, including a three-axis acceleration sensor, three-axis gyroscope. When the user wears on the arm, the sensor in the arm band can detect the electric pulse generated by the forearm muscles when the operator's hand or finger is being actuated, and the different gestures will be transformed into different digital commands to control the matching computer, cell phone or remote control device.

Fig. 2. Image of MYO armbands

C. Baxter Research Robot

In the experimental stage of the study, the Baxter robot (as shown in Fig.3) is used. The humanoid Baxter robot introduced by Rethink Robotics is a very loving cooperative robot, and its appearance reduces the threshold of factory automation. It runs under the ROS and Linux operating systems. Thanks to the advanced path planning and a new generation of force sensors, Baxter works with humans and avoids the risk of harming humans.

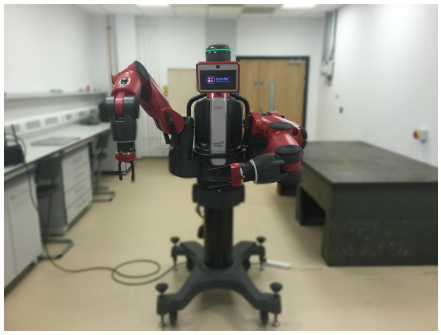

Fig. 3. Image of Baxter research robot

D. Kalman Filter

Kalman filter is the best linear filtering approach to solve the criterion of mean square error, which estimates the current value of the signal based on the previous estimate and the last observation data. It is estimated by the state equation and the recursive method, and its solution is given in the form of an estimate of its signal model, which is derived from the state equation and the measurement equation and can be shown by the equation below [5]

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + G\omega(t) \\ y(t) &= Hx(t) + v(t) \end{aligned} \tag{1}$$

where x and u are state variables; y is the measurement vector; A is the system matrix; G and B are voice matrixes; H is the measurement matrix; ω is the white noise vector; v is the continuous value of the white noise vector. The continuous time KF updating equation, according to [5] is demonstrated in (2),

$$\begin{aligned} &\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + K\left[y(t) - H\hat{x}\right] \\ &K(t) = P(t)H^T r^{-1}(t) \\ &\dot{P}(t) = P(t)H^T + AP(t) - P(t)H^t r^{-1} HP(t) \\ &+ Gs(t)G^T \end{aligned} \tag{2}$$

where K is the filter gain matrix, $\hat{x}$ is the estimated value of x, and P is the estimated covariance matrix.

E. RBF Neural Networks

A continuous function can be estimated by using Linear parameter RBF neural networks, such as $F(z) : R^m \rightarrow R$, over a minimized set $\Omega_z \subset \mathbf{R}^m$, can be imitated as, according to [16]

$$F(z) = W^{*\mathrm{T}} S(z) + \epsilon_z \ \forall_z \in \Omega_z \tag{3}$$

where $W^* = [w_1^*, w_2^*, \cdots, w_l^*]^{\mathrm{T}} \in R^l$ is the weight vector, $z \in \Omega_z$ is the information vector with $\Omega_z \subset \mathbf{R}^m$ being a minimal set, l is the NN hub number, and ϵ_z is the estimation mistake. $S(z) = [S_1(\|z - \mu_1\|), \cdots, S_l(\|z - \mu_l\|)]^T$, is the regressor vector, with a spiral premise work $S_i(\cdot)$, and μ_i $(i = 1, \cdots, l)$ a central inside $S_i(\cdot)$. The Gaussian capacities pick as

$$S_i(\|z - \mu_i\|) = \exp\left[\frac{-(z - \mu_i)^T (z - \mu_i)}{\varsigma^2}\right] \tag{4}$$

where ς is the variance and $\mu_i = [\mu_{i1}, \mu_{i2}, \cdots, \mu_{im}]^T \in R^m$ represents the center of each receptive area.

If here is a quite huge number of neurons, according to [16], then here is a weight matrix W^* shown as

$$Q(z) = W^{*\mathrm{T}} S(z) + \epsilon(z) \tag{5}$$

where $W^* = [W_1^*, W_2^*, \cdots, W_n^*] \in R^{N \times n}$ is the weight grid of RBFNN. The evaluated weight $\hat{W}$ will be utilized as a part of practice to swap W^* for the guess of a consistent capacity in this way $Q(z) = \hat{W} S(z)$, where $\hat{W}$ is the learning law which will be determined later.

TABLE I

MODEL REPRESENTATION OF THE DH PARAMETER TABLE [17]

$LinkNumber$	θ_i	$d_i(m)$	$a_i(m)$	$\alpha_i(rad)$
1	θ_1	0	0	$\pi/2$
2	θ_2	0	0	$\pi/2$
3	θ_3	d_3	0	$\pi/2$
4	θ_4	0	0	$\pi/2$
5	θ_5	d_5	0	$\pi/2$
6	θ_6	0	0	$\pi/2$
7	θ_7	0	a_7	0

III. METHODOLOGY

A. Calculation for Joints Angles of Arm

In order to estimate the 7-DOF model of human arm, the Denavit-Hartenberg (D-H) coordinate framework has been created in Fig.4 with a D-H parameter of the kinematic model (human arm) in I.

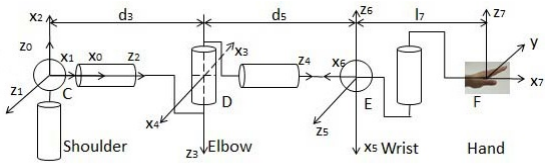

Fig. 4. The D-H coordinate framework and the initial position for each joint of human arm

Kinect emits infrared rays and detects infrared light reflections so that the depth values of each pixel in the field of view can be calculated, which is the depth data. Wherein, the object body and shape are first extracted. Then using the information above, the position of each joint can be obtained shown in Fig.5.

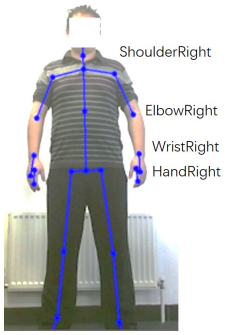

Fig. 5. Image of body skeleton captured by Kinect

To obtain the angle of rotation of the human shoulder, we can use the joints of ShoulderRight and ElbowRight in 3D coordinates (x, y, z) to calculate. Assuming three-dimensional coordinates of ElbowRight is (x_1, y_1, z_1), and the two skeletal nodes in the three-dimensional space to form a straight line (l_1). As the shoulder joint is unchanged during the rotation in the z-coordinate, there is just the xoy plane need to be considered, the linear equation is given below:

$$y = k_1 x + b_1 \tag{6}$$

where k_1=$tan\theta_1 = \frac{|y_2 - y_1|}{x_2 - x_1}(x_1 \neq x_2)$, $b1$ is not given in the calculation of the angle of rotation, so the formula is not given here. Assuming the angle between l_1 and y axis is θ_1, as shown in Fig.6, θ_1 is the human shoulder rotation angle, the formula is:

$$\theta_1 = arctank_1 = arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \tag{7}$$

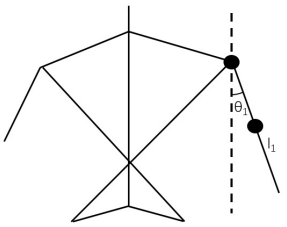

Fig. 6. Angle of rotation of shoulder joint

To obtain the angle of rotation of the human elbow, we can use joints of ElBowRight and WristRight in 3D coordinates to work out. Assuming the 3D coordinates of WristRight is (x_2, y_2, z_2), similarly, there is a straight line constituted by ElBowRight and WristRight (l_2), the linear equation is given below:

$$y = k_2 x + b_2 \tag{8}$$

where k_2=$tan\theta_2 = \frac{|y_3 - y_2|}{x_3 - x_2}(x_2 \neq x_3)$, $b2$ is not given in the calculation of the angle of rotation, so the formula is not given here. Assuming the angle between l_2 and l_1 is θ_2, as shown in Fig.7, θ_2 is the human elbow rotation angle, the formula is:

$$\theta_2 = arctank_2 = arctan\left(\frac{k_1 - k_2}{1 + k_1 k_2}\right) \tag{9}$$

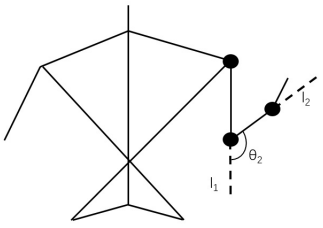

Fig. 7. Angle of rotation of elbow joint

Similarly, for the angle of rotation of human wrist, which is the angular between l_3 and l_2 defined as θ_3 shown in Fig.8, and the formula is:

$$\theta_3 = arctank_3 = arctan\left(\frac{k_2 - k_3}{1 + k_2 k_2}\right) \tag{10}$$

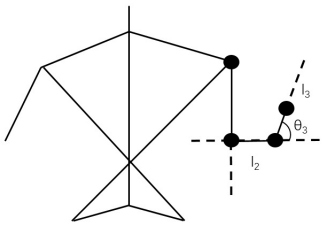

Fig. 8. Angle of rotation of wrist joint

B. Method to Obtain the Angular Velocity

The joint angles are gotten by figuring the basic for point speed. Any places of human operator's arms can be utilized as the underlying position, where the joint angles are thought to be zero, as indicated by [5]. At the point when the operator moves his arm to another posture P, the joint angles are the stance P with particular indicate the underlying stance in[5].

As shown in Fig.9, the angle (X_1, Y_1, Z_1) speaks to the introduction of MYO armband in the underlying position [5]. The frame (X_2, Y_2, Z_2) speaks to the present introduction of the MYO [5]. From the primary MYO armband worn on the upper arm, we can get three angles' angular speed $v1_x$, $v1_y$, $v1_z$, which speak to shoulder roll, pitch and yaw individually [5]. From the second MYO armband worn on the forearm, we can get the angles' velocity $v2_x$, $v2_y$, which represent elbow roll and pitch [5].

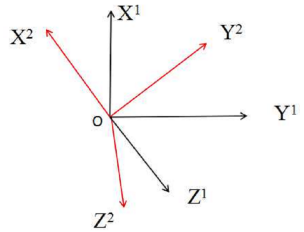

Fig. 9. The orientation of the MYO in the initial pose and the current pose

In the joint precise speed measured by the joint point, there will be errors, however here the mistakes will be superimposed [5]. The shoulder joint mistake will be superimposed on the elbow joint, bringing about a more prominent elbow error [5]. Here, the strategy for estimation of the precise speed was reached out from the technique for estimation for the points utilizing MYO armbands specified in past research [16]. In this paper, MYO armband is utilized to quantify angular speed for each joint, and Kinect is utilized to get the points of each joint.

C. Kalman Filtering based Sensor Fusion

The signal and noise in the Kalman filter are represented by the state equation and the measurement equation. The design of the Kalman filter therefore requires a known state equation and a measurement equation. It can be used for both smooth and unstable random process, but also can be applied to solve non-time-varying and time-varying systems. We assume that every single joint of human arms is taken into account separately to research, which gives that all the KF factors are the first order, hence, here A=0, B=1, G=1 and H=1. Then the KF equations are simplified as below,

$$\begin{aligned} \dot{x}_i &= u_i + \omega \\ y_i &= x_i + v \end{aligned} \tag{11}$$

where according to [5], in this special case, y_i is the angular position of the number of i joint collected from the Kinect sensor, we give it conception as following: y_i=$q_{di}{}'$. And u_i is the angular velocity of the number of i joint of operator's arm motion.

$$\begin{aligned} \dot{\hat{x}}_i &= u_i + k\left(y_i - \hat{x_i}\right) \\ k &= pr^{-1} \\ \dot{p} &= p - pr^{-1}p + s \end{aligned} \tag{12}$$

where k is the filter gain matrix, p is the estimated covariance matrix, $\hat{x}_i$ is the required (satisfied) data obtained from KF based sensor fusion, which is also the statistic that needs to be sent to development workstation via UDP [5].

IV. NEURAL NETWORKS BASED CONTROL SYSTEM

Here the concept of the data output from the result of KF based sensor fusion as q_{di}, in addition, the desired joint space trajectory q_d is defined as

$$\begin{aligned} q_{di} &= \hat{x}_i \\ q_d &= [q_{d1}, q_{d2}, q_{d3}, q_{d4}, q_{d5}]^T \in R^5 \end{aligned} \tag{13}$$

furthermore, here, we have to apply NN control procedure to achieve the copying control of the joint space direction. Hence, the controller should keep the framework to be quick, steady and exact. The precise framework $q_d \in R^5$ is conveyed by the MYO, which is dealt with as the reference flag. As the given criticism by the robot, the angular framework $q \in R^5$ is the genuine estimation of angles. At that point, as indicated by [16], we can acquire the dynamic conditions for the controller shown in (14).

$$\tau = I(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + U(q) \tag{14}$$

where $I(q) \in R^{5\times5}$ is the inertia matrix, $C(q,\dot{q}) \in R^{5\times5}$ is the Coriolis matrix, $G(q) \in R^{5\times1}$ is the gravity terms and $U(q) \in R^{5\times1}$ is unmodeled elements owing to interchangeable system vulnerabilities and robot gripper.

Characterize $z = \dot{e}_q + \Lambda e_q$, $q_r = \dot{q}_d - \Lambda e_q$, where $e_q = q - q_d$, $\Lambda = diag(\lambda_1, \lambda_2, \ldots, \lambda_n)$. Here the dynamic equation (14) can be changed as (15) from [16].

$$\tau - U(q) = I(q)\dot{z} + C(q,\dot{q})z + G(q) + I(q)\dot{q}_r + C(q,\dot{q})q_r \tag{15}$$

Here the conception of the adaptive controller as (16) [16].

$$\tau = \hat{H}(q) + \hat{I}(q)\dot{q}_r + \hat{C}(q,\dot{q})q_r - Kz \tag{16}$$

where $\hat{H}(q)$, $\hat{I}(q)$ and $\hat{C}(q,\dot{q})$ are the predicting values of $G(q) + U(q)$, $I(q)$ and $C(q,\dot{q})$ individually.

Then the closed circling framework dynamic equation is given below as (17) [16].

$$I\dot{z} + Cz + Kz = (\hat{I} - I)\dot{q}_r \tag{17}$$

The presidence function affinity approach is applied [16].

$$\begin{aligned} I(q) &= W_I^{*\mathrm{T}} S_I(q) + \epsilon_I \\ C(q,\dot{q}) &= W_C^{*\mathrm{T}} S_C(q,\dot{q}) + \epsilon_C \\ H(q) &= W_H^{*\mathrm{T}} S_H(q) + \epsilon_H(z) \end{aligned} \tag{18}$$

where W_I^*, W_C^* and W_H^* are the weight matrix; $S_I(z)$, $S_C(z)$ and $S_H(z)$ are the basis function matrix, and $\epsilon_I(z)$, $\epsilon_C(z)$ and $\epsilon_H(z)$ are the affinity errors.

Then we can write the equation for basis function matrix as follow [16],

$$\begin{aligned} S_I(q) &= \mathrm{diag}\left(S_q, \ldots, S_q\right) \\ S_C(q,\dot{q}) &= \mathrm{diag}\left(\begin{bmatrix} S_q \\ S_{\dot{q}} \end{bmatrix}, \ldots, \begin{bmatrix} S_q \\ S_{\dot{q}} \end{bmatrix}\right) \\ S_H(q) &= \begin{bmatrix} S_q^{\mathrm{T}} & \ldots & S_q^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \end{aligned} \tag{19}$$

where

$$\begin{aligned} S_q &= \begin{bmatrix} F(\|q-q_1\|) & F(\|q-q_2\|) & \ldots & F(\|q-q_n\|) \end{bmatrix}^{\mathrm{T}} \\ S_{\dot{q}} &= \begin{bmatrix} F(\|\dot{q}-\dot{q}_1\|) & F(\|\dot{q}-\dot{q}_2\|) & \ldots & F(\|\dot{q}-\dot{q}_n\|) \end{bmatrix}^{\mathrm{T}} \\ F(r) &= e^{-(\epsilon r)^2} \end{aligned} \tag{20}$$

The predicting values of $I(q)$, $C(q,\dot{q})$ and $H(q)$ can be obtained as (21).

$$\begin{aligned} \hat{I}(q) &= \hat{W}_I^{\mathrm{T}} S_I(q) \\ \hat{C}(q,\dot{q}) &= \hat{W}_C^{\mathrm{T}} S_C(q,\dot{q}) \\ \hat{H}(q) &= \hat{W}_H^{\mathrm{T}} S_H(q) \end{aligned} \tag{21}$$

Now, let (21) into (17), then the previous equations are simplified as

$$\begin{aligned} I\dot{z} + Cz + Kz = \tilde{W}_I^{\mathrm{T}} S_I(q)\dot{q}_r \\ + \tilde{W}_C^{\mathrm{T}} S_C(q,\dot{q}) q_r + \tilde{W}_H^{\mathrm{T}} S_H(q) \end{aligned} \tag{22}$$

where $\tilde{W}_I^{\mathrm{T}} = \hat{W}_I^{\mathrm{T}} - W_I^{*\mathrm{T}}$, $\tilde{W}_C^{\mathrm{T}} = \hat{W}_C^{\mathrm{T}} - W_C^{*\mathrm{T}}$ and $\tilde{W}_H^{\mathrm{T}} = \hat{W}_H^{\mathrm{T}} - W_H^{*\mathrm{T}}$.

Here, the Lyapunov function is used, which is shown as,

$$\begin{aligned} V =& \frac{1}{2} z^{\mathrm{T}} I z \\ &+ \frac{1}{2} tr\left(\tilde{W}_I^{\mathrm{T}} Q_I \tilde{W}_I + \tilde{W}_C^{\mathrm{T}} Q_C \tilde{W}_C + \tilde{W}_H^{\mathrm{T}} Q_H \tilde{W}_H\right), \end{aligned} \tag{23}$$

where Q_I, Q_C and Q_H are positive fixed weight matrix. Then V the derivative, which shows

$$\begin{aligned} \dot{V} =& - z^{\mathrm{T}} K z - \\ & tr\left[\tilde{W}_I^{\mathrm{T}}\left(S_I(q)\dot{q}_r z^{\mathrm{T}} + Q_I \dot{\hat{W}}_I\right)\right] + \\ & tr\left[\tilde{W}_C^{\mathrm{T}}\left(S_C(q,\dot{q}) q_r z^{\mathrm{T}} + Q_C \dot{\hat{W}}_C\right)\right] + \\ & tr\left[\tilde{W}_H^{\mathrm{T}}\left(S_H(q) z^{\mathrm{T}} + Q_H \dot{\hat{W}}_H\right)\right] \end{aligned} \tag{24}$$

According to [16], the upgraded principle is given as follow,

$$\begin{aligned} \dot{\hat{W}}_I &= -Q_I^{-1}\left(S_I(q)\dot{q}_r z^{\mathrm{T}} + \sigma_I \hat{W}_I\right) \\ \dot{\hat{W}}_C &= -Q_C^{-1}\left(S_C(q,\dot{q}) q_r z^{\mathrm{T}} + \sigma_C \hat{W}_C\right) \\ \dot{\hat{W}}_H &= -Q_H^{-1}\left(S_H(q) z^{\mathrm{T}} + \sigma_H \hat{W}_H\right) \end{aligned} \tag{25}$$

where $\sigma_I, \sigma_C, \sigma_H$ are predesigned positive constants.

Combining (25) and (24), then we obtain

$$\begin{aligned} \dot{V} =& - z^{\mathrm{T}} K z - \sigma_I tr\left(\tilde{W}_I^{\mathrm{T}} \hat{W}_I\right) - \sigma_C tr\left(\tilde{W}_C^{\mathrm{T}} \hat{W}_C\right) \\ & - \sigma_H tr\left(\tilde{W}_H^{\mathrm{T}} \hat{W}_H\right) \end{aligned} \tag{26}$$

Applying Young's inequality, (26) can be extended to

$$\begin{aligned} \dot{V} =& - z^{\mathrm{T}} K z + \frac{\sigma_I tr(W_I^{*\mathrm{T}} W_I^*)}{2} - \frac{\sigma_I tr(\tilde{W}_I^{\mathrm{T}} \tilde{W}_I)}{2} \\ & + \frac{\sigma_C tr(W_C^{*\mathrm{T}} W_C^*)}{2} - \frac{\sigma_C tr(\tilde{W}_C^{\mathrm{T}} \tilde{W}_C)}{2} \\ & + \frac{\sigma_H tr(W_H^{*\mathrm{T}} W_H^*)}{2} - \frac{\sigma_H tr(\tilde{W}_H^{\mathrm{T}} \tilde{W}_H)}{2} \end{aligned} \tag{27}$$

Finally we obtain

$$\dot{V} \leq -\eta V + \kappa \tag{28}$$

where $\eta = \min[2K, \sigma_I/\left(\lambda_{\max}(Q_I)\right), \sigma_C\left(\lambda_{\max}(Q_C)\right), \sigma_H/\left(\lambda_{\max}(Q_H)\right)]$, $\kappa = \frac{1}{2}\mathrm{tr}(\sigma_I W_I^{*\mathrm{T}} W_I^* + \sigma_C W_C^{*\mathrm{T}} W_C^* + \sigma_G W_G^{*\mathrm{T}} W_G^*)$. Since $V > 0$ and κ is the result of the predesigned constants and weight matrix that we given, the length of $\kappa \leq \eta$, we can have $\dot{V} \leq 0$. As indicated by Lyapunov method, the closed-loop stability is guaranteed.

V. EXPERIMENTAL STUDIES

A. Experimental Processes

Fig. 10. Image of designed TbD system

Here in this section, we set up two experimental processes to test the accuracy of the designed control system as shown in Fig.10, which are the demonstrated control without NN learning and the demonstrated control with NN learning.

And there was a heavy object attached to the end-effector of the Baxter and the task was limb following. Meanwhile, the operator standing in front of the Kinect sensor with a pair of MYO armbands moved his arm smoothly. Significantly, before everything gets started, it is crucial to calibrate the MYO armband.

B. Results of Experiment

The torque inputs of the designed control system for both with NN learning and without NN learning have been shown in Fig.11(a) and Fig.11(b), where the torque inputs are obtained from the designed Simulink program. The tracking performance of the designed TbD is shown in Fig.11(c), which demonstrated the variation for the desired joint space trajectory of 5 joints with NN learning and without NN learning respectively. And their NN learning weighs have been shown in Fig.11(d).

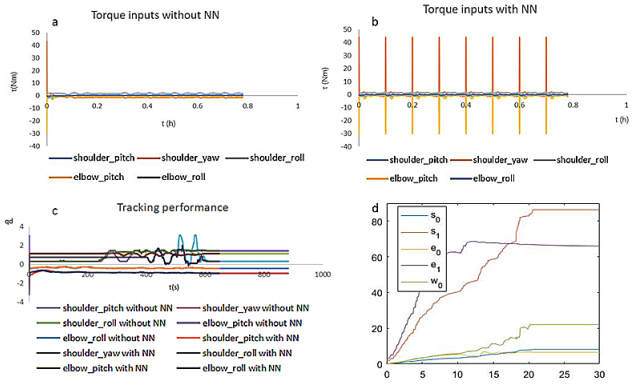

Fig. 11. Torque inputs of the control system with NN. (a) Torque inputs of the control system without NN. (b) Torque inputs of the control system with NN. (c) Tracking performance of the designed system for both without NN and with NN. (d) NN learning weighs of every single joints.

From the experimental statistics above, it is easy to find that the errors of the control inputs during the teaching and learning process are the smallest by applying the NN, which satisfies the proposed designing goal.

VI. CONCLUSION

In this paper, a NN learning enhanced TbD technology entitled to the KF based sensor fusion has been developed to acquire an enhanced performance. In order to do this, a Kinect sensor is used to catch the movement of operator arm with vector approach. The vector approach can exactly compute the precise information of human arm joints, by choosing 5 of the 7 joints each arm. At that point, the precise speed of human operator's arm can be measured by a couple of MYO armbands worn on the operator's arm. The continuous KF method produced the outlined information with less error, and then forward connected to the joints of Baxter robot for demonstration by a NN learning upgraded KF based technology. Several experiments have been applied to compare the effect of the unknown payload under both NN learning and without NN learning, where the results have proved the NN learning applied control system teaches Baxter with the smallest error.

VII. ACKNOWLEDGEMENT

We would like to thank Yong Li for his arithmetic assistance.

REFERENCES

[1] H. Yang, L. Shao, F. Zheng, L. Wang, and Z. Song, "Recent advances and trends in visual tracking: A review," *Neurocomputing*, vol. 74, no. 18, pp. 3823–3831, 2011.

[2] F. Faion, S. Friedberger, A. Zea, and U. D. Hanebeck, "Intelligent sensor-scheduling for multi-kinect-tracking," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, pp. 3993–3999, IEEE, 2012.

[3] G. Tao, P. S. Archambault, and M. Levin, "Evaluation of kinect skeletal tracking in a virtual reality rehabilitation system for upper limb hemiparesis," in *Virtual Rehabilitation (ICVR), International Conference on*, pp. 164–165, IEEE, 2013.

[4] M. Kranz, A. Möller, N. Hammerla, S. Diewald, T. Plötz, P. Olivier, and L. Roalter, "The mobile fitness coach: Towards individualized skill assessment using personalized mobile devices," *Pervasive and Mobile Computing*, vol. 9, no. 2, pp. 203–215, 2013.

[5] C. Li, C. Yang, J. Wan, A. S. Annamalai, and A. Cangelosi, "Teleoperation control of baxter robot using kalman filter-based sensor fusion," *Systems Science & Control Engineering*, vol. 5, no. 1, pp. 156–167, 2017.

[6] J. L. Raheja, A. Chaudhary, and K. Singal, "Tracking of fingertips and centers of palm using kinect," in *Computational intelligence, modelling and simulation (CIMSiM), the third international conference on*, pp. 248–252, IEEE, 2011.

[7] C. Zhang and Z. Zhang, "Calibration between depth and color sensors for commodity depth cameras," in *Computer Vision and Machine Learning with RGB-D Sensors*, pp. 47–64, Springer, 2014.

[8] K. Li, Y. L. Zhang, and Z. X. Li, "Application research of kalman filter and svm applied to condition monitoring and fault diagnosis," in *Applied Mechanics and Materials*, vol. 121, pp. 268–272, Trans Tech Publ, 2012.

[9] C. Yang, S. S. Ge, C. Xiang, T. Chai, and T. H. Lee, "Output feedback nn control for two classes of discrete-time systems with unknown control directions in a unified approach," *IEEE Transactions on Neural Networks*, vol. 19, no. 11, pp. 1873–1886, 2008.

[10] Y.-J. Liu, C. P. Chen, G.-X. Wen, and S. Tong, "Adaptive neural output feedback tracking control for a class of uncertain discrete-time nonlinear systems," *IEEE Transactions on Neural Networks*, vol. 22, no. 7, pp. 1162–1167, 2011.

[11] W. Chen and L. Jiao, "Adaptive tracking for periodically time-varying and nonlinearly parameterized systems using multilayer neural networks," *IEEE Transactions on Neural Networks*, vol. 21, no. 2, pp. 345–351, 2010.

[12] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of kinect depth data for indoor mapping applications," *Sensors*, vol. 12, no. 2, pp. 1437–1454, 2012.

[13] C. Li, C. Yang, P. Liang, A. Cangelosi, and J. Wan, "Development of kinect based teleoperation of Nao robot," in *Advanced Robotics and Mechatronics (ICARM), International Conference on*, pp. 133–138, IEEE, 2016.

[14] R. Butterfield, G. T. Houlsby, and G. Gottardi, "Standardized sign conventions and notation for generally loaded foundations," *Géotechnique*, vol. 47, no. 5, pp. 1051–4, 1997.

[15] C. Amon, F. Fuhrmann, and F. Graf, "Evaluation of the spatial resolution accuracy of the face tracking system for kinect for windows v1 and v2," in *Proceedings of the 6th Congress of the Alps Adria Acoustics Association*, 2014.

[16] C. Yang, J. Chen, and F. Chen, "Neural learning enhanced teleoperation control of baxter robot using imu based motion capture," in *Automation and Computing (ICAC), 22nd International Conference on*, pp. 389–394, IEEE, 2016.

[17] P. Liang, L. Ge, Y. Liu, L. Zhao, R. Li, and K. Wang, "An augmented discrete-time approach for human-robot collaboration," *Discrete Dynamics in Nature and Society*, vol. 2016, 2016.