2016-08

Development of Kinect based teleoperation of Nao robot

Li, Chunxu

10.1109/icarm.2016.7606908
2016 International Conference on Advanced Robotics and Mechatronics (ICARM)
IEEE

2016 International Conference on Advanced Robotics and Mechatronics (ICARM)

Development of Kinect based Teleoperation of Nao Robot

Chunxu Li[1], Chenguang Yang*[2], Peidong Liang[3], Angelo Cangelosi[1], Jian Wan[4]

***Abstract*— In this paper, an online tracking system has been developed to control the arm and head of a Nao robot using Kinect sensor. The main goal of this work is to achieve that the robot is able to follow the motion of a human user in real time to track. This objective has been achieved using a RGB-D camera (Kinect v2) and a Nao robot, which is a humanoid robot with 5 degree of freedom (DOF) for each arm. The joint motions of the operator's head and arm in the real world captured by a Kinect camera can be transferred into the workspace mathematically via forward and inverse kinematics, realitically through data based UDP connection between the robot and Kinect sensor. The satisfactory performance of the proposed approaches have been achieved, which is shown in experimental results.**

I. INTRODUCTION

With the rapid development of robot technology, the application of teleoperation with robots for social activities has been widely used. The field of research with respect to motion capture of human based on tracking system attracted great attention during the past decades [1]. In [2], the author used a Microsoft Kinectint sensor intelligently to schedule a platform of multiple RGBD cameras especially in the Bayesian object tracking system. The kinematic validity of a Kinect based skeletal tracking system has been evaluated, which is designed to use with an upper limb virtual reality rehabilitation system [3]. In [4], a work on tracking the fingertips and palm centrer were developed based on Kinect sensor. However, researches focusing on head following is also necessary.

Head following technology has a wide range of application, especially in the video face recognition authentication, security, surveillance and human-robot interaction [5]. Due to various factors such as, the low resolution of huaman face in video, illumination change and motion blur factors, the head following involves a number of challanges [6]. According to [7], a multilayer neural-net (NN) controller for limb following is developed. In [8], a research team presents a model-based method for analyzing a human body motion in order to follow a robot arm model which represents a human body motion.

In recent years, the Kinect sensor with depth information has increasing number of applications in some special areas, such as, the game entertainment, health and fitness [9]. In early 2014, Microsoft company developed a new generation of sensor,which is Kinect v2, while providing Kinect development kit SDK, which makes Kinect v2 can collect more accurate color information with depth, become more suitable for the head following project [10].

NAO robot has been widely investigated owing to its opening programming framework [11]. In the field of scientific research, researchers use the Nao robot in computer science, mathematics, physics and artificial intelligence [12]. At the same time, people develop the programming of robots into the education with Nao robot because of its convenience, for example, one Swedish university encourages students to understand the knowledge of voice recognition and visual feedback through the establishment of laboratory with Nao robots [13]. In commercial field, Nao robots have been used in some large production launches to highlight the high technology of the products [14].

II. PRELIMINARIES

The control system consists of two main parts: Kinect v2 camera and the Nao robot. Fig.1 shows the control block of the system, which has been implemented to verify the performance of our approach.

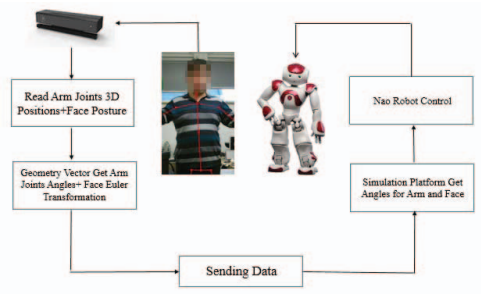


Fig. 1. Flowchart of the whole experiment process

A. Kinect v2

Kinect v2 produced by Microsoft, is an RGB-D device, which can capture depth, colour, and IR images (also sound) [15]. The depth, IR and colour image resolutions are significantly improved from that of the first generation Kinect v1. Raw colour, depth and IR images can be captured from the device using the Kinect 2.0 SDK (Software Development Kit) from Microsoft. Using the SDK, captured colour and depth information can be consolidated (transformed) into real-world co-ordinates, called Camera Space; these co-ordinates are referenced to the centre of the depth sensor [16]. Skeletal tracking can also be achieved with the use of the Kinect 2.0 SDK; although this feature is not used in this

*Corresponding author: C. Yang. Email: cyang@ieee.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]Center for Robotics and Neural Systems, Plymouth University, UK. [2]Zienkiewicz Centre for Computational Engineering, Swansea University, UK. [3]State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin 150001, China. [4]School of Marine Science and Engineering, Plymouth University, UK.

978-1-5090-3364-5/16/$31.00 ©2016 IEEE

project, it could be used by other researchers to track the position of an operator in front of device. Since the point specied by the operator on the Kinect colour image is 2 dimensional, the Kinect depth space image is required to determine the depth of this point. Both the colour image, and depth image transformed into a common frame, the origin of which is located at the centre of the depth camera, called Camera space; the co-ordinate system of this space follows a right-handed convention [17](as shown in Fig.2).

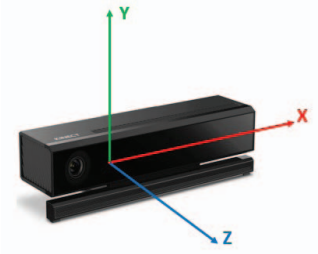


Fig. 2. Illustration of the origin of the Kinect v2s Camera space; the same as the depth sensor origin, modified from [18]

B. Nao Robot

In this project, we choose the Nao robot (shown in Fig.3), which is produced by the Aldebaran-Robotics in France and it is a humanoid robot with 5 DOF joints [19]. The Nao robot supports multiple sensors and controllers, such as, the head and jaw cameras, the chest sonar sensors, movement motors on the neck, hands and feet, three colour LEDs (in red, green and blue) on the eyes, and the head and feet tactile sensors [20]. NAO has a body mass index (BMI) of about 13.5 kg/m^2, hence, compared to other robots with the same heightit, Nao is quite light [20]. According to [21], Nao has a total of 25 degrees of freedom (DOF), wherein 11 joints are for the legs and pelvis and the rest are for the trunks, arms and head. In addition, each arm is supported by a 2 DOF shoulder, a 2 DOF elbow, a 1 DOF wrist and a 1 DOF hands gripper. The head is able to rotate on both yaw and pitch axes [21].

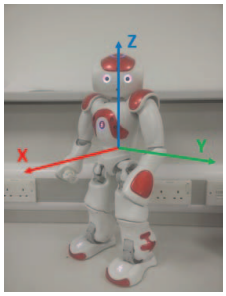


Fig. 3. Illustration of Nao Robot and coordinate system

III. CALIBRATION

The two angles of head for the Nao robot can not be used directly, which needs to be placed into the same coordinate system as the one read from Kinect, where we use the coordinate transformation. We use the following transformation as shown in (1) and then we can obtain Nao's coordinate transformation from the robot coordinates to the Kinect coordinates.

$$X_i = TX_i^{'} \tag{1}$$

where T represents the transformation matrix, we will give it concept as following $X_i = [x_i \quad y_i \quad z_i \quad 1]^T$, $X_i^{'} = [x_i^{'} \quad y_i^{'} \quad z_i^{'} \quad 1]^T$, which are the coordinates under both the robot and Kinect systems.

We can retain the transformation matrix T by calculating four points under both the robot and Kinect coordinates systems. Giving the assumption there are four points p_1, p_2, p_3 and p_4, whose coordinates are under the robot coordinate system and the Kinect coordinate system respectively. Fig.4 shows an example for one of the four points in relationship between the Nao robot and the Kinect coordinate system. (2) is the equation to calculate the transfer matrix T from the Kinect coordinates to the robot coordinates.

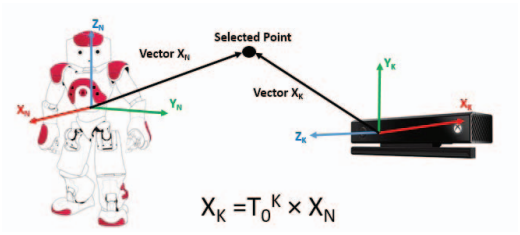
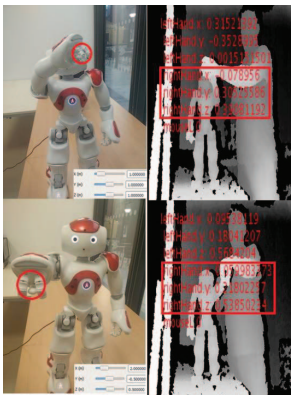


Fig. 4. Illustration of relationship between Nao Robot and Kinect coordinate system

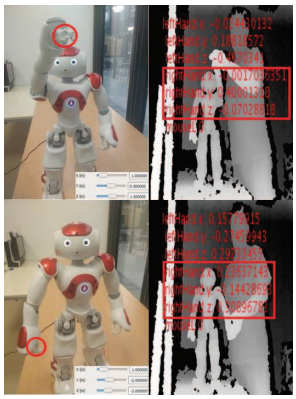
$$T = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1^{'} & x_2^{'} & x_3^{'} & x_4^{'} \\ y_1^{'} & y_2^{'} & y_3^{'} & y_4^{'} \\ z_1^{'} & z_2^{'} & z_3^{'} & z_4^{'} \\ 1 & 1 & 1 & 1 \end{bmatrix}^{-1} \tag{2}$$

In this project, we choose four positions under the same point on the central point of right hand for Nao robot which are placed on both Nao coordinate system and Kinect system as shown in Fig.5. To obtain the positions under Kinect coordinate system, we arrange a person stand in front of a Kinect sensor, which is used to obtain the coordinate data of human's right hand, and keep his right hand place on the same position as the right hand of Nao robot. The four positions under Nao robot coordinate system are (1, 1, 1), (2, -0.5, 0.5), (1, 0.5, 1.5), (1, -2, -3), under Kinect coordinate system are (-0.0790, 0.3053, 0.3908), (0.0600, 0.2180, 0.5385), (-0.0012, 0.4000, -0.0703), (0.2364, -0.1443, 0.3090).

Substituting all the values of positions both on Nao and Kinect system into (2) we can calculate the transformation matrix of Kinect under ground based coordinate system T_0^k, where we assumed the coordinate system of Nao robot is ground based because the Nao robot keeps stationary in this project.



(a) : The first and second positions in Nao and Kinect coordinate frame



(b) : The third and forth positions in Nao and Kinect coordinate frame

Fig. 5. Illustration of the four chosen positions under both Nao coordinate system and Kinect coordinate system

$$T_0^k = \begin{bmatrix} 1 & 2 & 1 & 1 \\ 1 & -0.5 & 0.5 & -2 \\ 1 & 0.5 & 1.5 & -3 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} -0.079 & 0.06 & -0.0012 & 0.2364 \\ 0.3053 & 0.218 & 0.4 & -0.1443 \\ 0.3908 & 0.5385 & -0.0703 & 0.309 \\ 1 & 1 & 1 & 1 \end{bmatrix}^{-1} \tag{3}$$

Then we would obtain:

$$T_0^k = \begin{bmatrix} 7.87 & 5.09 & 2.37 & 6.58 \\ -10.27 & -0.4 & -0.73 & -1.26 \\ 1.25 & 9.58 & 1.09 & -0.39 \\ -0 & 0 & 0.0001 & 4.72 \end{bmatrix} \tag{4}$$

After that the rotation matrix of Kinect under ground based coordinate system R_0^k can be obtained through its transformation matrix, which is given as following:

$$R_0^k = \begin{bmatrix} 7.87 & 5.09 & 2.37 \\ -10.27 & -0.4 & -0.73 \\ 1.25 & 9.58 & 1.09 \end{bmatrix} \tag{5}$$

IV. KINEMATICS METHODOLOGY

The image stream is captured from the Kinect v2 colour camera, such as the operators view of the Nao workspace. In order to select a point in the colour image which the operator desires the end effector of Nao to be transformed from the Kinect colour space to the Nao co-ordinate space. The pixels of the colour image could be transformed into the Camera space as follows:

$$[x_c \quad y_c \quad 1]^T = R \cdot [x_k \quad y_k \quad 1]^T + T_{Trans} \tag{6}$$

where x_c and y_c give the position of a transformed colour frame pixel (in Camera space), x_k and y_k give the position of the colour frame pixel, R is the rotational matrix, and T_{Trans} is the translation matrix, between the colour frame and Camera space. However, the Kinect v2 depth camera, suffers from inherent camera intrinsics, such as radial distortion (due to curvature of lens). Intrinsic parameters are given in one of the Kinect SDK examples, and were used to calculate the undistortion transformation matrix. It is now possible to transform any point in Camera space to Naos workspace, with Naos origin as its reference point. In order to match the orientations of the Camera space co-ordinate system and Nao co-ordinate system, the relevant axis of the Kinect co-ordinate system have been matched correctly with the axis of the Nao co-ordinate system.

A. Method to Get Orientation Angles of Head

The pose of a user's head, namely the orientation angles α_f^k, β_f^k, γ_f^k are able to be collected from the statistic of the experiment by using visual studio based on Kinect v2. In order to send the data of orientation angles for user's head, the angles need to be transformed into those under the basic ground coordinate, which is shown commonly in Fig.6. In this experiment, we keep the flat of Kinect camera be always parallel of the user's head, where the value of β_f^k equals zero. Then we have the matrix as follow, which represents rotation matrix:

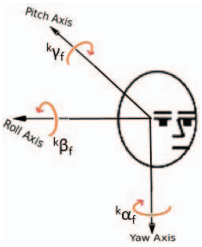


Fig. 6. Illustration of the conception for human head orientation angles

$$R_k^f(\alpha_f^k, \beta_f^k, \gamma_f^k) = \begin{bmatrix} c\alpha_f^k c\beta_f^k & c\alpha_f^k s\beta_f^k s\gamma_f^k - s\alpha_f^k c\gamma_f^k & c\alpha_f^k s\beta_f^k c\gamma_f^k + s\alpha_f^k s\gamma_f^k \\ s\alpha_f^k c\beta_f^k & s\alpha_f^k s\beta_f^k s\gamma_f^k + c\alpha_f^k c\gamma_f^k & s\alpha_f^k s\beta_f^k c\gamma_f^k - c\alpha_f^k s\gamma_f^k \\ -s\beta_f^k & c\beta_f^k s\gamma_f^k & c\beta_f^k c\gamma_f^k \end{bmatrix} \tag{8}$$

According to $R_0^f = R_0^K \times R_K^f$, wherein the R_0^K has already been obtained through calibration process, as shown in (5), the rotation matrix R_0^f for user's head under the ground base coordinate system could be found by (7), where we denote the abbreviations $s\alpha_f^k$ for s_a, $s\beta_f^k$ for s_b, $s\gamma_f^k$ for s_g, $c\alpha_f^k$ for c_a, $c\beta_f^k$ for c_b, $c\gamma_f^k$ for c_g.

Finally, the orientation angles of user's head under the basic ground coordinate can be calculated, which are shown in (9)-(11), where we give the abbreviation $R_0^f = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$.

$$\alpha_f^0 = arctan\frac{r_{21}}{r_{11}} \tag{9}$$

$$\beta_f^0 = arctan \pm (\frac{\sqrt{r_{11}^2 + r_{21}^2}}{r_{31}}) \tag{10}$$

$$\gamma_f^0 = arctan\frac{r_{32}}{r_{33}} \tag{11}$$

$$R_0^f = \begin{bmatrix} 7.87c_ac_b - 10.27(c_as_bs_g + s_ac_g) + 1.25(c_as_bc_g + s_as_g) & 5.09c_ac_b - 0.4(c_as_bs_g + s_ac_g) + 9.58(c_as_bc_g + s_as_g) \\ 7.87s_ac_b - 10.27(s_as_bs_g - c_ac_g) + 1.25(s_as_bc_g - c_as_g) & 5.09s_ac_b - 0.4(s_as_bs_g + c_ac_g) + 9.58(s_as_bc_g - c_as_g) \\ -7.87s_b - 10.27c_bs_g + 1.25c_bc_g & -5.09s_b - 0.4c_bs_g + 9.58c_bc_g \\ 2.37c_ac_b - 0.73(c_as_bs_g - s_ac_g) + 1.09(c_as_bc_g + s_as_g) & \\ 2.37s_ac_b - 0.73(s_as_bs_g + c_ac_g) + 1.09(s_as_bc_g - c_as_g) & \\ -2.37s_b - 0.73c_bs_g + 1.09c_bc_g & \end{bmatrix} \tag{7}$$

TABLE I
MODEL REPRESENTATION OF THE DH PARAMETER TABLE [22]

$LinkNumber$	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

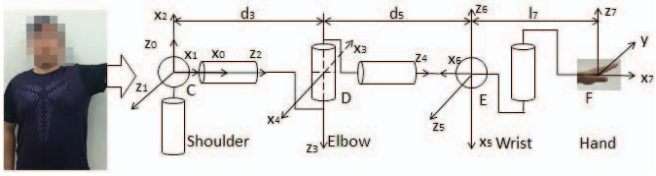


Fig. 7. Human arm model and its primary D-H coordinate system and the initial position of each joint angle

By using the Kinect sensor, we can receive the skeleton data of human directly, which include 25 joints 3D positions relative to the camera coordinate as shown in the left panel of Fig.8. Then, we create the geometry model of human arm as seen in the right panel of Fig.8. After that, we take the point Hip-Left as the origin of basic coordinate, while $x-axis$ is in the same direction of vector $\overrightarrow{AO}$. And $y-axis$ is along with vector $\overrightarrow{OC}$. It is easy to get the normal vector of each axis of the base coordinate,$\overrightarrow{X_0}$, $\overrightarrow{Y_0}$ and $\overrightarrow{Z_0}$:

$$\overrightarrow{X_0} = \frac{\overrightarrow{AO}}{|\overrightarrow{AO}|}; \overrightarrow{Y_0} = \frac{\overrightarrow{OC}}{|\overrightarrow{OC}|}; \overrightarrow{Z_0} = \overrightarrow{X_0} \times \overrightarrow{Y_0} \tag{12}$$

In order to calculate the angle of each joint, we consider two conditions: i) the left thumb must be in the same plane with the palm because we need the plane which consisted by points of thumb, wrist and hand to calculate some other joints angles. ii) The angles of two vectors can only change from 0 to 180°, so there must be some other solutions we ignored.

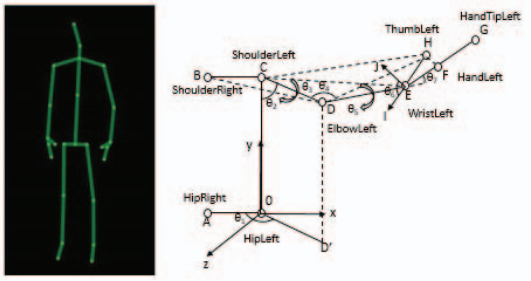


Fig. 8. Body skeleton map consisting of 25 joints captured by Kinect and Human arm motion model in joint space [22]

These seven angles are all in the joint space, and their initial positions are displayed in Fig.7. The geometry vector method we proposed is based on the principle of cosine value of two vectors shown in (17), in addition, the angle between two planes can be got by using their normal vector [22].

It is simple to find that angle θ_1 is the supplement of the angle between plane COD and xOy.

$$\theta_1 = \pi - < \overrightarrow{CO} \times \overrightarrow{CD}, \overrightarrow{CB} \times \overrightarrow{CO} > \tag{13}$$

Angle θ_2 is the angle between $y-axis$ and vector $\overrightarrow{CD}$, which is given below:

$$\theta_2 = < \overrightarrow{CO}, \overrightarrow{CD} > \tag{14}$$

In the same way, we can also get other angle: θ_3 is the angle of plane BCD and CDE.

$$\theta_3 = < \overrightarrow{CB} \times \overrightarrow{CD}, \overrightarrow{CD} \times \overrightarrow{CE} > \tag{15}$$

$$\theta_2 = < \overrightarrow{DC}, \overrightarrow{DE} > \tag{16}$$

Angle θ_5 is the angle of plane CDE and DEH.

$$\theta_3 = < \overrightarrow{EC} \times \overrightarrow{ED}, \overrightarrow{ED} \times \overrightarrow{EH} > \tag{17}$$

Angle θ_6 is the angle between vector $\overrightarrow{ED}$ and plane EFH.

$$\theta_3 = \Pi/2 + < \overrightarrow{EH} \times \overrightarrow{EG}, \overrightarrow{ED} > \tag{18}$$

However, θ_7 IS the yaw angle of wrist and it is difficult to calculate using the above method. θ_7 can be seen as the angle between $\overrightarrow{X_5}$ and $\overrightarrow{Y_7}$. While,

$$\overrightarrow{X_7} = \frac{\overrightarrow{EF}}{|\overrightarrow{EF}|}; \overrightarrow{Z_7} = \frac{\overrightarrow{EF} \times \overrightarrow{EH}}{|\overrightarrow{EF} \times \overrightarrow{EH}|}; \overrightarrow{Y_7} = \overrightarrow{Z_7} \times \overrightarrow{X_7} \tag{19}$$

So now the problem became to solve $\overrightarrow{X_5}$. We know that $\overrightarrow{X_5}$ must be in the plane of EFH, in addition, $\overrightarrow{X_5}$ is Perpendicular to $\overrightarrow{DE}$. Thus when suppose that:

$$\overrightarrow{X_5} = k_1\overrightarrow{EF} + k_2\overrightarrow{EH} \tag{20}$$

There are:

$$(k_1\overrightarrow{EF} + k_2\overrightarrow{EH}) \times \overrightarrow{DE} = 0; |k_1\overrightarrow{EF} + k_2\overrightarrow{EH}| = 1 \tag{21}$$

Therefore,

$$\theta_7 = < \overrightarrow{x_5}, \overrightarrow{x_7} > \tag{22}$$

Up to now, all the joint angles can be calculated.

V. EXPERIMENTAL STUDIES

In this section, some experiments have been designed to test the performance of following system. Experimental platform: computer system is windows 10. Kinect SDK for windows, Visual studio 2013 and OpenCV library. The experiment environment is an indoor and adequate illumination environment. Only one person stands forward to Kinect in the distance about 2 meters. Because the degree of freedom for Nao robot's arm is only 5, thus we choose the ShoulderPitch, ShoulderRoll, ElbowRoll, ElbowYaw and WristYaw joints from human arm to take the experiment.

A. Head Following Experiment

In this part, we run this experiment to test the stability of the head following system where we place a person in front of the Kinect camera. Then we run the program developed by the Kinect SDK 2.0 and send the command to an additional computer, which controls the Nao robot, to imitate the movement of human head. It has been shown in Fig.9, we run 4 times experiments with different postures in order to get the total performance, which are rotating 30° to the left direction, rotating 30° to the right direction, pitching up with 30° and pitching down with 30° respectively. The Fig.10 below shows the process of the first experiment. As a result, a graph can be drawn (seen in Fig.11), wherein the data has been collected from head angles for both operator and Nao robot. From the result of head following experiment, the movement of user's head can be followed accurately.

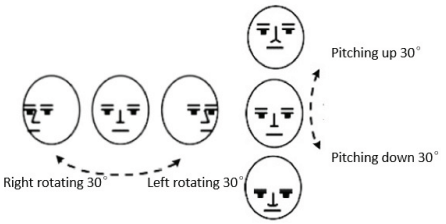


Fig. 9. Illustration of operator's head for different postures

From the first test above, we are able to conclude that the user's head with different postures can be captured quite smoothly. The Nao robot imitates the movement of human head well. The result illustrates that there is a fast and accurate response to follow the user's head and imitate the familiar movement with human in different postures.

Fig. 10. Process of imitation for Nao robot head following: Rotating 30° to left and right directions and pitching up and down with 30°

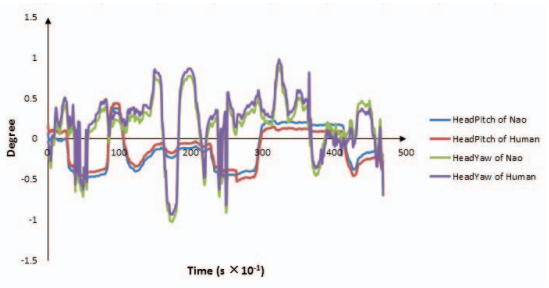


Fig. 11. Result of head following experiment for both human and Nao robot

B. Limb Following Experiment

In the second experiment, the operator keeps his body standstill and only puts his limb forwards at 90° and raises up at 180° and stretches his limb sidewards at 90°, with a low speed from the origin position to final position, for both of his right and left arms. Then we use the same method in the first experiment, after obtaining the data, we send it to the additional computer, which controls the Nao robot, to imitate the movement of human limb. The Fig.**??** below shows the process of the second experiment. As a result, a graph can be drawn (seen in Fig.13), wherein the data has been collected from arms for both operator and Nao robot. From the result above, we can conclude that all the movements of those 5 selected joints from user's arm can be followed completely.

VI. CONCLUSION

A system for tracking user's head and arm to interact with Nao robot via a Kinect v2 sensor has been developed in this paper. Kinect v2 is controlled to track the head's and arm's motion of the human operator. The operator can move their head and arm in a natural way to manipulate the position of the Kinect device. Additionally, the kinematics equations

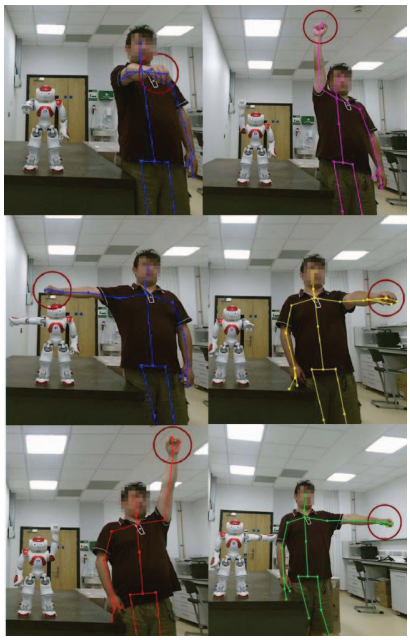


Fig. 12. Process of imitation for Nao robot limb following: Putting forwards at 90°, raising up at 180° and stretching sidewards at 90° for both arms

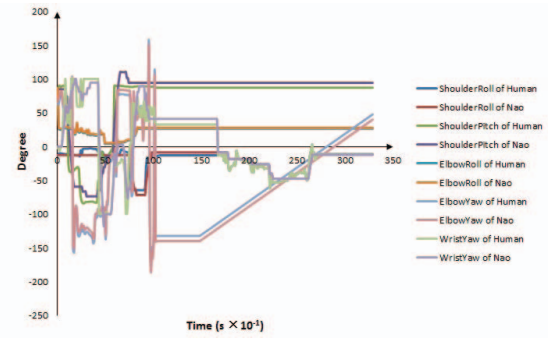


Fig. 13. Result of head following experiment for both human and Nao robot

carried with this research have been applied to transfer the Euler angles of user's head from the Kinect coordinate to the ground coordinate and calculate all the joints angles of human arm. We also run experiments and draw the curve according to the test data. The experimental result shows that the Nao robot imitates the human movement accurately. In the near future, we would focus on teleoperation of Nao robot with whole body and leg following would be added to the project.

VII. ACKNOWLEDGEMENT

We would like to thank Junshen Chen for his technical 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), 2012 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), 2013 International Conference on*, pp. 164–165, IEEE, 2013.

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

[5] N. Huang, G. Nanda, and M. R. Lehto, "Laser applications in safety and ergonomics," *Laser and Photonic Systems: Design and Integration*, p. 309, 2014.

[6] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM computing surveys (CSUR)*, vol. 35, no. 4, pp. 399–458, 2003.

[7] F. L. Lewis, A. Yesildirek, and K. Liu, "Multilayer neural-net robot controller with guaranteed tracking performance," *Neural Networks, IEEE Transactions on*, vol. 7, no. 2, pp. 388–399, 1996.

[8] M. Yamamoto and K. Koshikawa, "Human motion analysis based on a robot arm model," in *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pp. 664–665, IEEE, 1991.

[9] 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.

[10] 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.

[11] A. N. Panfir, R. G. Boboc, and G. L. Mogan, "Nao robots collaboration for object manipulation," in *Applied Mechanics and Materials*, vol. 332, pp. 218–223, Trans Tech Publ, 2013.

[12] J. Kulk and J. S. Welsh, "Evaluation of walk optimisation techniques for the nao robot," in *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pp. 306–311, IEEE, 2011.

[13] A. Louloudi, A. Mosallam, N. Marturi, P. Janse, and V. Hernandez, "Integration of the humanoid robot nao inside a smart home: A case study," in *Proceedings of the Swedish AI Society Workshop (SAIS). Linköping Electronic Conference Proceedings*, vol. 48, pp. 35–44, 2010.

[14] J. Han, N. Campbell, K. Jokinen, and G. Wilcock, "Investigating the use of non-verbal cues in human-robot interaction with a nao robot," in *Cognitive Infocommunications (CogInfoCom), 2012 IEEE 3rd International Conference on*, pp. 679–683, IEEE, 2012.

[15] 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.

[16] J. Webb and J. Ashley, *Beginning Kinect Programming with the Microsoft Kinect SDK*. Apress, 2012.

[17] 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.

[18] 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.

[19] D. Gouaillier, V. Hugel, P. Blazevic, C. Kilner, J. O. Monceaux, P. Lafourcade, B. Marnier, J. Serre, and B. Maisonnier, "Mechatronic design of nao humanoid," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pp. 769–774, IEEE, 2009.

[20] D. Gouaillier, V. Hugel, P. Blazevic, C. Kilner, J. Monceaux, P. Lafourcade, B. Marnier, J. Serre, and B. Maisonnier, "The nao humanoid: a combination of performance and affordability," *CoRR abs/0807.3223*, 2008.

[21] J. Kulk and J. Welsh, "A low power walk for the nao robot," *preprint, available at*, 2008.

[22] 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.