Research Journal of Applied Sciences, Engineering and Technology 8(11): 1384-1388, 2014
DOI:10.19026/rjaset.8.1111
ISSN: 2040-7459; e-ISSN: 2040-7467
© 2014 Maxwell Scientific Publication Corp.
Submitted: July 24, 2014 Accepted: 11 October 2014 Published: September 20, 2014

Research Article

Motion Control of Robot by using Kinect Sensor

[1]Mohammed A. Hussein, [2]Ahmed S. Ali, [1]F.A. Elmisery and [3]R. Mostafa
[1]Department of Electronics Technology, Faculty of Industrial Education, Beni-Suef University,
[2]Department of Mech. Engineering, Faculty of Engineering, Assuit University,
[3]Department of Head of Automatic Control, Beni-Suef University, Egypt

Abstract: In the presented work, a remote robot control system is implemented utilizes Kinect based gesture recognition as human-robot interface. The movement of the human arm in 3 d space is captured, processed and replicated by the robotic arm. The joint angles are transmitted to the Arduino microcontroller. Arduino receives the joint angles and controls the robot arm. In investigation the accuracy of control by human's hand motion was tested.

Keywords: Arduino, gesture, human-robot interaction, Kinect, robotic arm

INTRODUCTION

In recent years, the development of human robot interaction in service robots has attracted the attention of many researchers. The most important applications of industrial robots are material handling, welding, assembling, dispensing and processing where the robotic arm manipulator needs to perform pick and place operations incessantly, one of such industrial standard robots is a generic serial arm which consists of a base, a link or series of links connected at joints and an end effectors. Generally, end effectors are a gripper which is at the end of the last link and the base is the first link of a serial arm.

In this study, Microsoft Kinect sensor is applied in a remote robot control system to recognize different body gestures and generate visual Human-Robot interaction interface without calculating complex inverse kinematics to make the robot arm follow the posture of human arm. This kind of system aims to enrich the interactive way between human and robots which help non-expert users to control the robot freely, making human-robot interaction much easier (Goodrich and Schultz, 2007; Cheng *et al*., 2012).

LITERATURE REVIEW

The concept of Gesture control to manipulate robots has been used in many earlier researches. Thanh *et al*. (2011) construct the system through which human user can interact with robot by body language. The Kinect camera is used as the visual device. Luo *et al*. (2013) use the Kinect sensor developed by the Microsoft as our motion capture device. Waldherr *et al*. (2000) describes a gesture interface for the control of a mobile robot equipped with a manipulator. The interface uses a camera to track a person and recognize gestures involving arm motion. The basic technique of the depth sensor is to project a structured infrared light continuously and calculate depth from the reflection of the light at different positions. By processing the depth image, user's skeleton joints can be captured and provided in real time. Human arm movement will be directly reflected in the action of robot arms. A lot of researches have been proposed about human motion capture through different sensor devices, e.g., cameras, depth sensors, inertial sensors or marker based vision system (Ott *et al*., 2008; Cole *et al*., 2007; Pons-Moll *et al*., 2010).

OBJECTIVES AND PROBLEM STATEMENT

The aim of study is to development a human-machine interface used for control robot arm, as shown in Fig. 1. In the presented work, a remote robot control system is described that utilizes Kinect based gesture recognition as human-robot interface. The presented work provided the evaluation of control robot arm using based gesture recognition as human-robot interface. The movement of the human arm in 3 d space is captured, processed and replicated by the robotic arm. The joint angles are transmitted to the Arduino microcontroller. Arduino receives the joint angles and controls the robot arm.

Kinect sensor: In the presented work, we use The Kinect Sensor was developed by Microsoft and Prime Sense, shown in Fig. 2, It is a hardware device used to control the Microsoft XBOX-360 game console without any kind of controller that the user has to hold or wear.

Corresponding Author: Mohammed A. Hussein, Department of Electronics Technology, Faculty of Industrial Education, Beni-Suef University, Egypt
This work is licensed under a Creative Commons Attribution 4.0 International License (URL: http://creativecommons.org/licenses/by/4.0/).

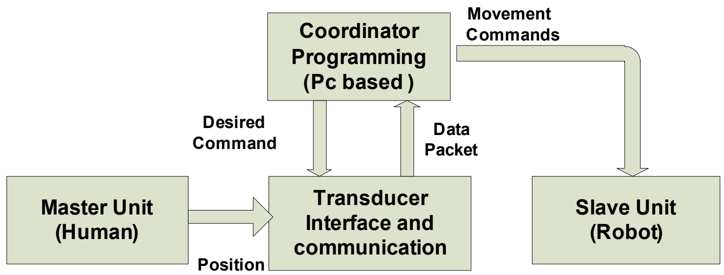

Fig. 1: General architecture of the system

Fig. 2: Kinect-sensor

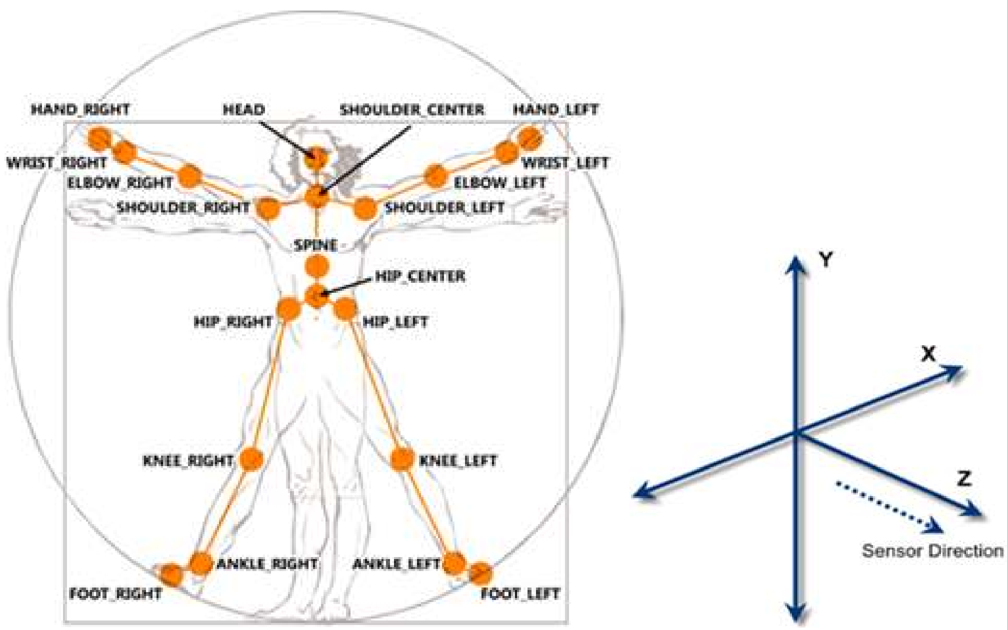

Fig. 3: Skeleton tracking

The official release of Kinect in North America was on November 4th, 2010, in Europe on November 10th, 2010.

The skeleton data consists of a set of joints. These joints are shown in Fig. 3. The coordinate system for the skeleton data is a full 3D system with values in meters. It is shown in the following diagram Fig. 1. There are operations for converting between Skeleton Space and Depth Image space. The Kinect for Windows SDK supports up to two players (skeletons) being tracked at the same time. A player index is inserted into the lower 3 bits of the depth data so that you can tell which depth pixels belong to which player. You must take these 3 bits into account when you want to examine the depth values (http://msdn.microsoft.com/en-us/library/hh438998.aspx).

STRUCTURE OF THE CONTROL SYSTEM

The structure of system is shown in Fig. 4 consist of user, Kinect sensor, computer, Arduino microcontroller and robot arm. This system consists of two parts: angle determination and transfer data to robot arm.

User: Is anyone standing in front of the Kinect at a certain distance committed to the rules dealing with Kinect.

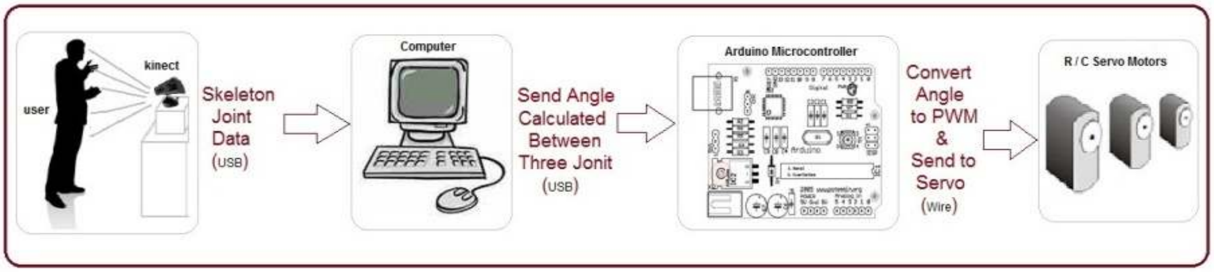

Fig. 4: The structure of the control system

Kinect sensor: Is used as input device. It captures the movement of the human arm in real time.

And send the skeleton joint data to computer via USB for processing.

Computer: It processes the information from the Kinect sensor and converts it into a skeletal image then calculates angle between joints and sending it Arduino microcontroller via USB.

Arduino microcontroller: Is logic board, which allows interfacing electronic devices to the computer quickly and easily. The version of the board we employ is named Arduino Uno, Depending on the receiving data (angles) the Arduino generates PWM signals designed to move a Servo-Motor to a specific angle.

Robot arm (R/C servo): Edubot100 Robotic Arm is a five-axis articulated robotic arm designed to teach the industrial robot technology in the simplest way. Received data from Arduino (PWM) the servo motor turn their axles to the angles which are received.

SOFTWARE

In the presented work we used c# to write the programs for our system, we employed the Microsoft Kinect as depth sensor, using the OpenNI APIs to interface it and the NITE framework for depth image analysis and control skeleton extraction.

The OpenNI framework is an open source SDK used for the development of 3D sensing middleware libraries and applications (http://www.openni.org).

To interface with the Kinect sensors the OpenNI-Framework was used. This is an open source package by Prime Sense. It is intended to make available the new opportunities offered by sensors like Kinect to a larger community, to accelerate new developments in natural interaction.

OpenNI provides a driver for Kinect and an Application Programming Interface (API). Also, it offers a lot of basic functionality for analysis of the scene watched by Kinect. The functionality that was used in this thesis consists of the following:

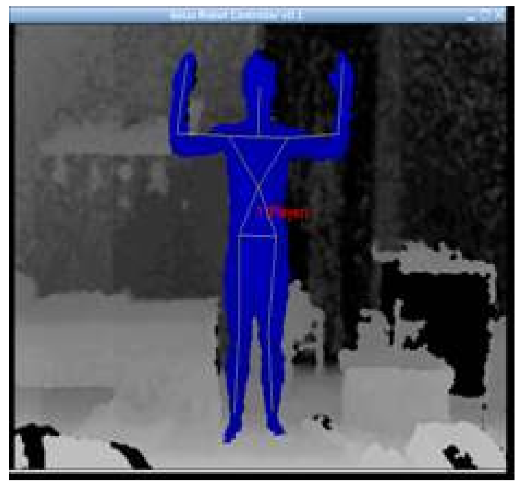

Fig. 5: Calibration position

Depth generator: The depth generator provides a depth map of the scene as an array of floats, even though the actual depth values are always natural numbers of the unit mm. OpenNI will give us this position in “real-world” coordinates http://kinectcar. ronsper.com/docs/openni/groupdepthgen.html). These are coordinates that are designed to correspond to the physical position of your body in the room.

User detection (calibration): To start the user work must stand in front of the sensor assuming the calibration position, that is, with arms parallel to the ground and forearms perpendicular (Fig. 5). This process might takes from 10 sec or a little bit more depending on the positions of the Kinect sensor. Once the calibration is done, Kinect tracks the joints and limbs position.

However, if the person stays out of the frame for too long, Kinect recognizes that person as a new user once she/he comes back and the calibration needs to be done again, in Fig. 6 shown steps of calibration using (OpenNI+NITE PrimeSense).

Arduino programming: The microcontroller is programmed using the Arduino programming language and the Arduino development environment (http://www.arduino.cc). The brain of robotic arm will

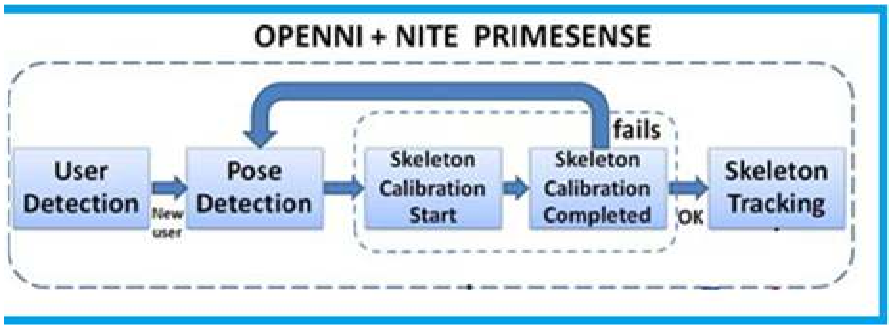

Fig. 6: Process of calibration

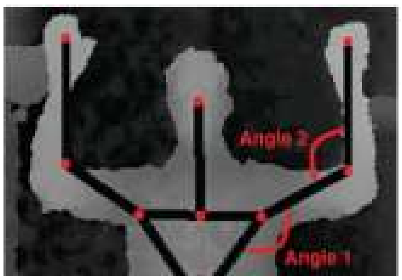

Fig. 7: Calculating angels

Fig. 8: The robot is controlled by human demonstration in real time using Kinect sensor

be an Arduino microcontroller. The Arduino is a small computer that can control simple electronics and sensors. Then communicate with the Arduino from Processing so we can position robotic arm based on the data that capture from the Kinect.

Calculating angels: Robot arm servos are going to reproduce the angles of the user's shoulder and elbow. When we refer to the angle of the shoulder (angle1) creates between the joints (torso, shoulder and elbow). Also, the angle of elbow (angle2) creates between the joints (shoulder, elbow and hand), shown in Fig. 7.

Kinect sensor provide the coordinates x, y, z of each joint on human body. Calculating the angle between three joints (two vectors) can define by:

$$\theta = \cos^{-1} \frac{\vec{A}\,.\,\vec{B}}{\|A\|.\,\|B\|}$$

where,

θ = The angle between three joints (two vectors)
A = Vector from joint hand to joint elbow
B = Vector from joint shoulder to joint elbow

Testing the algorithm: The test of the human motion imitation system with conductor motion and a sequence of photos are shown in Fig. 8. It demonstrates that the arm robot arm can be controlled by human demonstration in real time by using Kinect sensor.

CONCLUSION

This study presented rich the interactive way between human and robots and help non-expert users to control the robot freely, making human-robot interaction much easier.

The presented provides the evaluation of control robot arm using Kinect sensor where the joint angles are carried out. The joint angles are transmitted to the Arduino controller. Arduino controller receives the joint angles and controls the robot arm. The performance of the system is characterized using human input for different situations and the results show that system has the ability to control the robot by using Kinect sensor.

REFERENCES

Cheng, L., Q. Sun, H. Su, Y. Cong and S. Zhao, 2012. Design and implementation of human-robot interactive demonstration system based on Kinect. Proceeding of 24th Chinese Control and Decision Conference (CCDC, 2012), May 23-25, pp: 971-975.

Cole, J.B., D.B. Grimes and R.P.N. Rao, 2007. Learning full-body motions from monocular vision: Dynamic imitation in a humanoid robot. Proceeding of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS, 2007), pp: 240-246.

Goodrich, M.A. and A.C. Schultz, 2007. Human-robot interaction: A survey. Foundations Trends Hum. Comput. Interaction, 1(3): 203-275.

Luo, R.C., B.H. Shih and T.W. Lin, 2013. Real time human motion imitation of anthropomorphic dual arm robot based on Cartesian impedance control. Proceeding of IEEE International Symposium on Robotic and Sensors Environments (ROSE, 2013), pp: 25-30.

Ott, C., D. Lee and Y. Nakamura, 2008. Motion capture based human motion recognition and imitation by direct marker control. Proceeding of 8th IEEE-RAS International Conference on Humanoid Robots (Humanoids, 2008), pp: 399-405.

Pons-Moll, G., A. Baak, T. Helten, M. Müller, H.P. Seidel and B. Rosenhahn, 2010. Multi sensor-fusion for 3D full-body human motion capture. Proceeding of IEEE Conference on Computer Vision and Pattern Recognition (CVPR, 2010), pp: 663-670.

Thanh, N.N.D., D. Stonier, S.Y. Lee and D.H. Kim, 2011. A new approach for human-robot interaction using human body language. Proceeding of the 5th International Conference on Convergence and Hybrid Information Technology (ICHIT, 2011), September 22-24, pp: 762-769.

Waldherr, S., R. Romero and S. Thrun, 2000. A gesture based interface for human-robot interaction. Auton. Robot., 9(2): 151-173.