Imitation of Human Arm Movements by a Robotic Arm Through Vision

Sagar P Sujir[1], Rohith Prabhu, Shashank N, Rahul U Hebbar, Sandesh Kumar
Department. of ECE, SMVITM, Udupi
[1] sujir13sagar@gmail.com

Abstract— The use of manipulator robot controlled by a natural human computer interaction offers new possibilities in the execution of complex tasks in dynamic workspaces. Learning by demonstration is an intuitive and efficient way to let a humanoid robot learn a variety of human-like motions. So, for this purpose we build a robotic arm which mimics the motion of the arm of the user by hand tracking method. Usually most of the robotic arms are controlled by sensors and accelerometers which require direct connections to the human arm. This process is time inducing and cumbersome. This project focuses on a Kinect based real-time interactive control system implementation. Based on LabVIEW integrated development environment (IDE), which allows user to control the robotic arm in real time. The Kinect software development kit (SDK) provides a tool to keep track of human body skeleton and abstract it into 3-dimensional coordinates. Therefore, the Kinect sensor is used to detect the different user joints co-ordinate. Hence it does not require any device to be attached to the human arm which makes the task simple and efficient.

Index Terms—Human motion Imitation, Kinect, Robotic Arm, degrees of freedom.

1 Introduction

Imitation is an important way of skill transfer in biological agents. Many animals imitate their parents in order to learn how to survive. It is also a way of social interaction. Imitative capabilities of the biological agents increase with the complexity of the agent. It starts from simple mimicry to intention and goal-based imitation. The most complicated form of imitation is observed in humans. This shows that imitation requires higher mental capabilities. A sociable robot must have the capability to imitate the agents around it. In a human society, people generally teach new skills to other people by demonstration. We do not learn to dance by programming; instead we see other dancers and try to imitate them. Hence, Learning by demonstration is an intuitive and efficient way to let a humanoid robot learn a variety of human-like motions [1].

In order to let humanoid robots have similar behavior with human, motion planning is crucial. However, planning humanlike motions for robots is very complicated because it needs to handle multiple degrees of freedom (DOFs) simultaneously. The idea is to generate human-like motions by extracting information directly from human motion via a motion capture system, and it simplifies the process of programming and learning complex motions. A lot of researches have been proposed about human motion capture through different sensor devices [2], e.g. cameras, depth sensors [4], sensors attached to body [5], inertial sensors [3] or marker based vision system. In our work, we use the Kinect sensor developed by the Microsoft as our motion capture device. Kinect is a motion detection and recognition smart sensor which allows human/computer interaction without the need of any physical controllers. Indeed, Kinect sensor is a motion sensor that provides a natural user interface available for several applications in different fields including game based learning systems, stroke rehabilitation, helping visually impaired people, navigation systems, and other fields. Kinect sensor composed of a RGB camera and a depth sensor is widely used in motion capture system recently because of its relative low price. 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 provides a 3-dimensional coordinate in real time, these coordinate values can be provided to a computer environment such as LabVIEW.

2 Recogntion Method

The Kinect is shown in Fig.1, has two cameras and one IR Emitter. One camera gives RGB image and other camera calculates depth of image. IR projector projects IR laser pattern towards objects, which are bounced back towards IR camera. Kinect takes 30 frames per second for 640x480 resolutions. It can provide both RGB and depth images at the same time. Human hand tracking is carried out by continuously processing RGB images and depth images of an operator standing in front of a fixed Kinect. In fact, as one of 20 human skeleton points defined by Microsoft Kinect SDK, the hand center point can be easily obtained using the skeleton tracking method. However, the position precision of the hand center point can't be guaranteed since too many skeleton points are processed at the same time.

To generate the corresponding joint coordinates we are using computer environment LabVIEW. Based on Microsoft Kinect SDK, the Kinesthesia Toolkit for Microsoft Kinect is developed at University of Leeds initially for the medical rehabilitation and surgical tools. In addition, the toolkit helps the NI LabVIEW programmers to access and use the popular functions of the Microsoft Kinect camera such as RGB video, depth camera and skeletal tracking. The toolkit comes fully packaged using JKI's VI Package Manager.

Fig. 1. Microsoft Kinect for Windows

In addition, the toolkit allows the user to initialize and close the Kinect's different components through polymorphic VIs (RGB camera, depth camera, and skeletal tracking) dependent on the functionalities they require. The Operational Layout for the Kinect LabVIEW toolkit is described in Fig. 2.

Fig. 2: Operation overview of LabVIEW

The first step is to initialize the Kinect, this act opens the device and returns a reference.

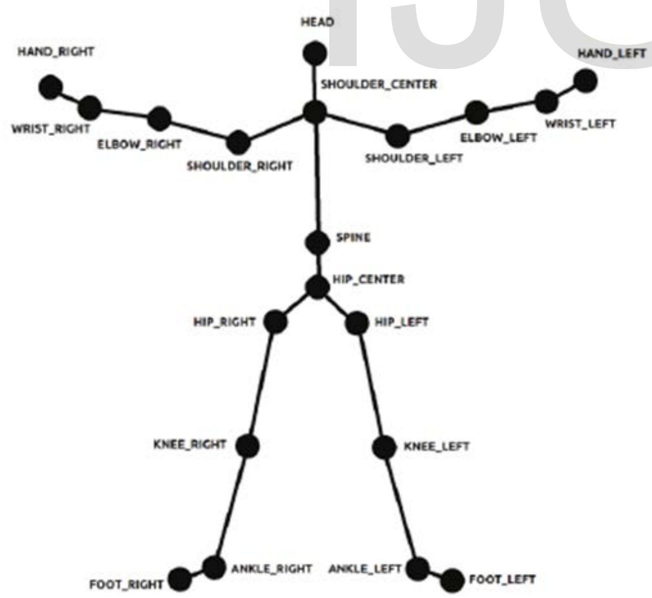

Fig. 3: Skeleton Joints Detected by Kinect

The second step is to configure the Kinect. In this step, we can choose the RGB stream format, enable the skeleton smoothing, and select the appropriate event (video feedback from the RGB camera, 3D skeleton information, or depth information). In the next step, we can get skeleton depth, and video information from the Kinect in the main loop such as while loop. The additional implementations can be put in the same loop. Finally, we stop the Kinect and close the communication.

The User Interface containing a skeleton display has been developed using Kinesthesia Toolkit using LabVIEW. Along with the screen for skeleton display, there is a screen of video recording from kinect for comparison. This is shown in Fig.4.

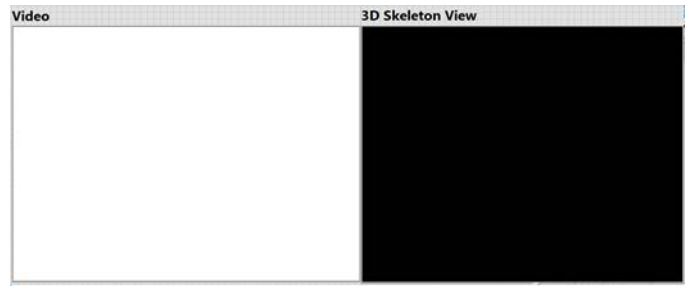

Fig. 4. Skeleton and Video Screen

3 Robotic Arm

This robotic Arm is a 4 Axis robotic arm with servo gripper. Robot arm has 4 degrees of freedom which includes Shoulder XZ axis rotation, Shoulder YZ axis rotation, Elbow XZ axis rotation, Elbow YZ axis rotation. It uses 5 metal gear Servo motors. Each Servo motor has a torque of 11 kg cm (6 V), The range of motion per axis is 180 degrees. So, we consider Shoulder XZ axis rotation as θ_1, Shoulder YZ axis rotation as θ_2, Elbow XZ axis rotation as θ_3, Elbow YZ axis rotation as θ_4 shown in Figure 4. These Servo motor are connected to the Arduino. Arduino is also programmed with the help of LabVIEW. There is also an toolkit present for interfacing the Arduino with LabVIEW. There is block for servo motor in Arduino toolkit which can be used.

Fig. 5: Arm Setup

4 Control Design

Robot manipulator has highly non linear dynamic movements. For this reason, the design of a robust controller is required. To we need 2 angles in each joint as shown in Fig. 6 for the angle between the shoulder and the elbow.

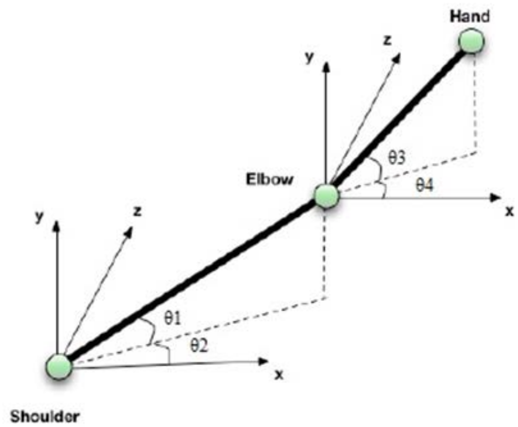

Fig. 6: Arm Movements Stick Figure

As there are 2 servos at each joint we have to find angle for 2 angles i.e. is θ1 and θ2. Same goes for the Elbow joint we are going to find the angle θ3 and θ4. Angle can be calculated by converting the 3D Cartesian coordinates to Spherical Coordinates. So we will get θ and Φ, which is nothing but the angle θ1 and θ2. Same goes for the Elbow joint θ3and θ4 will be θ and Φ of elbow with respect to hand.

Fig. 7 shows the methodology overview, we get the skeletal data from the Kinect of a user and the data will be computed by LabVIEW. Angles made by the joints is calculated and sent to Arduino, Arduino will later transfer these values to respective servo motor connected to it.

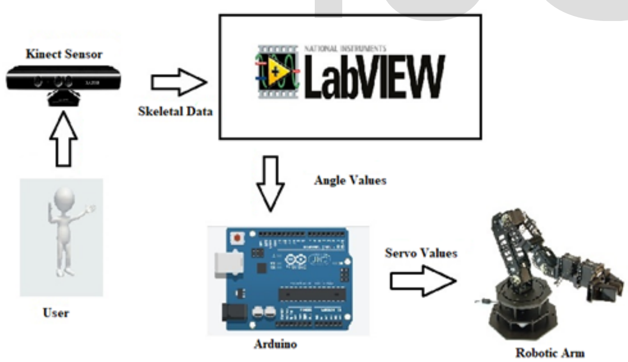

Fig. 7: Methodology Overview

6 Comparison with Existing Methods & Systems

The ability to track a person's movements and determine what gestures they may be performing can be achieved through various tools. Although there is a large amount of research done in image/video based gesture recognition, there is some variation within the tools and environments used between implementations.

Wired Gloves is one of the currently existing gesture recognition method, where it can provide input to the computer about the position and rotation of the hands using magnetic or inertial tracking devices. Furthermore, some gloves can detect finger bending with a high degree of accuracy. Other Gesture-based controllers act as an extension of the body so that when gestures are performed, some of their motion can be conveniently captured by software.

There are many challenges associated with the accuracy and usefulness of gesture recognition. For image-based gesture recognition there are limitations on the equipment used and image noise. Images or video may not be under consistent lighting, or in the same location. Items in the background or distinct features of the users may make recognition more difficult. Futhermore, the amount of background noise also causes tracking and recognition difficulties, especially when occlusions (partial and full) occur.

Instead of using intensive processing of the 3D models and dealing with a lot of parameters like in the case of '3D model-based algorithms', one can just use a simplified version of joint angle parameters along with segment lengths using 'Skeletal Based Algorithms'. This is known as a skeletal representation of the body, where a virtual skeleton of the person is computed and parts of the body are mapped to certain segments. The analysis here is done using the position and orientation of these segments and the relation between each one of them. These types of Algorithms are used by Kinect Sensor.

Advantages of using skeletal models:

- Algorithms are faster because only key parameters are analyzed.
- Pattern matching against a template database is possible.
- Using key points allows the detection program to focus on the significant parts of the body.

7 Experimental Results

Here we have accomplished the task of making the robotic arm imitate human hand movements in three-dimensional space. Calculation of torque of the servo motors placed at various joints. There was noise in the movement and the movement was not smooth. This can be corrected by adding filter to the angles or the skeleton data. Fig 8 shows us the user and the skeleton view. Fig 9 photograph series depicting gesture control process execution. When the user applies the desired gesture with a variable velocity, our algorithem can take 30 angles per seond and if our velocity is more the algorithem cannot track the arm, because the Knect only supports 30 frames persecond. The control strategy should be improved if we hope to use in medicine applications, this can be used in industrial application.

Fig. 8: User and Skeletal View

Fig 9: User and Robotic Arm

8 Conclusion

This robotic arm can be used in wide ranging applications such as entertainment, industrial, healthcare and consumer electronics. They can replace human workers in harmful, potentially dangerous environments such as mining. The elderly can use the arm to perform daily tasks without having to get up or move about. They can simply sit at one location and be able to control the arm through gestures. In the medical industry mechanical arms may be used to perform complex surgeries in the absence of availability of a doctor or in case of an extreme emergency.

Acknowledgment

It is my proud privilege and duty to acknowledge the kind of help and guidance received from several people in preparation of this paper. It would not have been possible to prepare this report in this form without their valuable help, cooperation and guidance.

I would like to express my sincere gratitude to my guide **Mr. Sandesh Kumar**, Assistant Professor, Department of Electronics and Communication Engineering, SMVITM, Bantakal for his guidance, encouragement and inspiration.

References

[1] Martin Tykal, Alberto Montebelli and Ville Kyrki, “Incrementally Assisted Kinesthetic Teaching for Programming by Demonstration,” 978-1-4673-8370-7/16, IEEE, 2016.

[2] Edwin Basil Mathew, Dushyant Khanduja, Bhavya Sapra and Bharat Bhushan, “Robotic Arm Control through Human Arm Movement Detection using Potentiometers,” 978-1-4799-7247-0/15, IEEE, 2015.

[3] Mohd Aliff, Shujiro Dohta and Tetsuya Akagi, “Trajectory controls and its analysis for robot arm using flexible pneumatic cylinders”, 78-1-4673-7124-7, IEEE, 2016.

[4] Toshika Fegade, Yogesh Kurl, Sagar Nikale and Praful Kalpund, “Wireless Gesture Controlled Semi-Humanoid Robot,” 978-1-5090-2807-8/16, IEEE, 2016.

[5] Genival Alves dos Anjos, Gabriel Almeida Santos, Jose A. Diaz Amado and João Erivando Soares Marques, “Haptic Glove Development for Passive Control of a Robotic Arm,” 978-1-5090-1341-8/16, IEEE, 2016.