Real-time Robot Control Using Leap Motion Technology

Trinadh Venkata Swamy Naidu Venna
RISC Laboratory, Computer Science and Engineering
University of Bridgeport , Bridgeport, CT

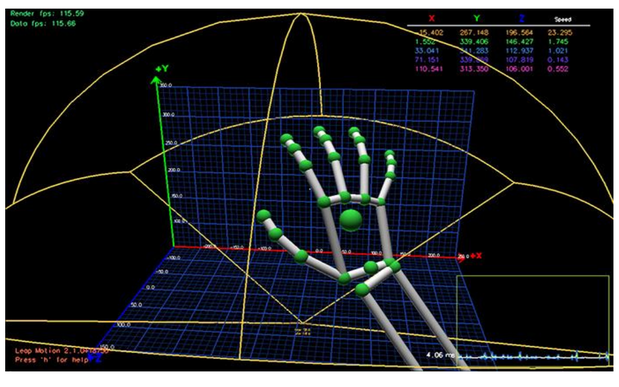
Abstract

Unlike pre-programmed controlling, real-time control of a 6-DOF manipulator is a complex task which involves physical interaction with a traditional manual controller clustered with numerous joysticks. This takes lot of joystick movements in order to bring the robotic arm to a position followed by an action with the end-effector. Using the latest technology in human-computer interaction i.e.., Leap Motion, we can reduce the complexity of this task to minimum effort by making the robotic arm mimic human hand in real time.

Introduction to Leap Motion

The Leap Motion controller is a small USB peripheral device which is designed to be placed on a physical desktop, facing upward to track human hands and fingers. Using two monochromatic IR cameras and three infrared LEDs, the device observes a roughly hemispherical area, to a distance of about 1 meter (3.28084 feet). The LEDs generate a 3D pattern of dots of IR light and the cameras generate almost 300 frames per second of reflected data, which is then sent through a USB cable to the host computer, where it is analyzed by the Leap Motion controller software using complex maths, **3D position data** in some way synthesizing by comparing the 2D frames generated by the two cameras.

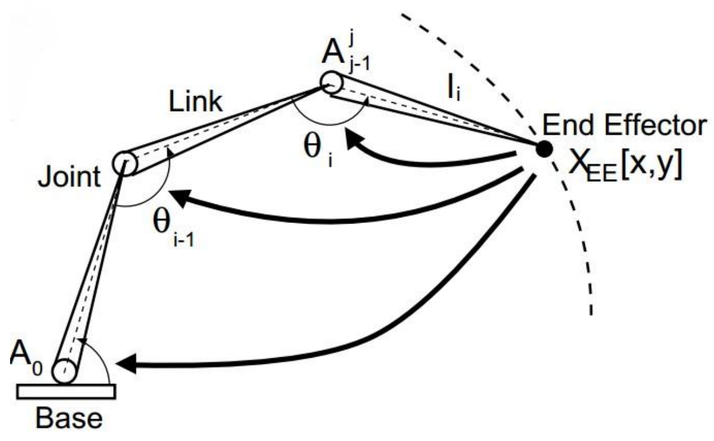
fig(1). Leap Motion Sensor



fig(2). Obtaining hand co-ordinates in 3D space

Inverse Kinematics

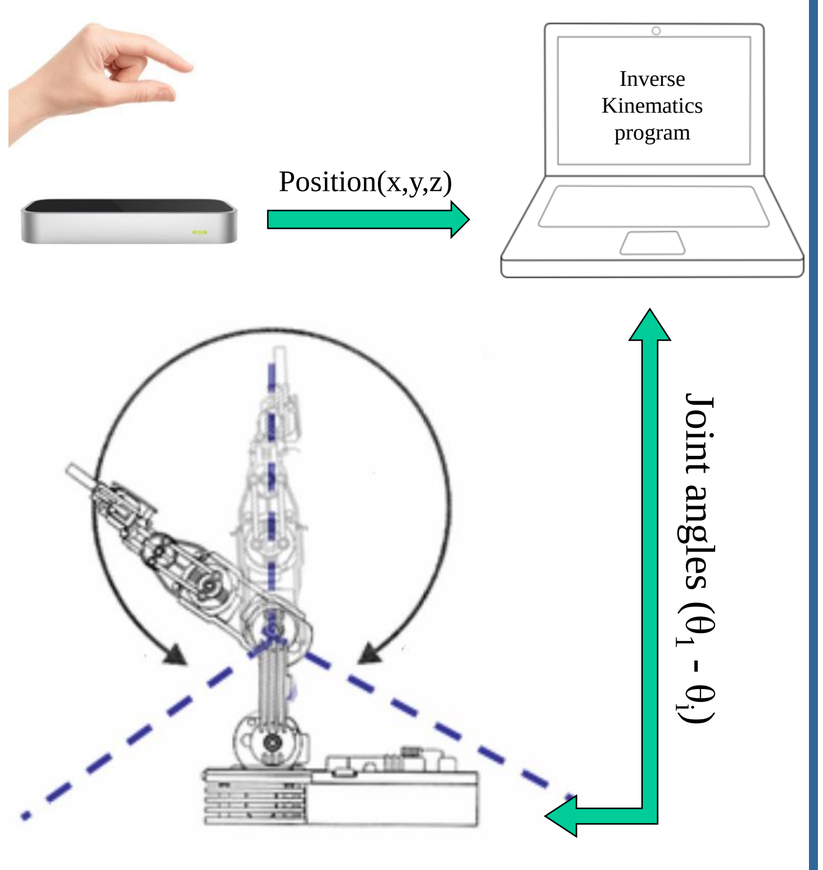
Inverse kinematics here refers to the use of the kinematics equations of a robot to determine the joint parameters that provide a desired position of the hand by leap motion. Specification of the movement of a robot so that its end-effector achieves a desired task is known as motion planning. Inverse kinematics transforms the motion plan into joint actuator trajectories for the robot.



fig(3). Calculating joint angles (θ_1-θ_i) using end- effector co-ordinates from leap motion

Architecture

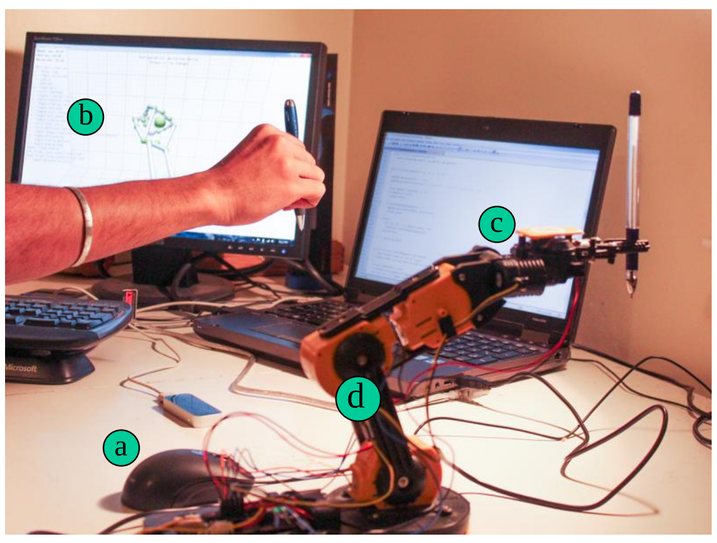
Since we have leap motion sensor to keep track of hand and inverse kinematics equations to solve the joint angles, we need an interface to bridge the gap between the sensor and the robotic arm. This can be done using a java program which takes input from the sensor and calculates the inverse kinematics followed by sending commands to the robotic arm in real time(every 100ms).



fig(4). Architecture of real-time robot control with leap motion

Experiment

Research was carried out on standard 4-DOF arm called “Owi Robotic arm” by Omnitron Electronics. The following is an image of Owi robot imitating the human hand movement in real time using leap motion technology and inverse kinematics program written in java.



fig(5). Experiment outcome

(a) Leap motion sensor, (b) Hand tracking data in real time, (c) Inverse Kinematics program, (d) 4 – DOF Robotic arm.

Areas of Application

This approach can be used at all the areas where typical robotic arms are manually controlled.

Areas such as:
1) Manufacturing industries
2) Military bots
3) Humanoid robotics
4) Rovers of space missions

are the most benefited areas of this latest way of interacting with robots.

Acknowledgements

1. Mark W.Spong, Seth Hutchinson, M.Vidyasagar, “Chapter 3. Forward and Inverse Kinematics”, “Robot Modeling and Control.

2. Dr. Andrew Davison, Dept. of Computer Engineering, Prince of Songkla University, “Inverse Kinematics program for OWI”, http://fivedots.coe.psu.ac.th/~ad/jg

3. Leap motion developer portal, “Leap motion Java API”, https://developer.leapmotion.com