UNSYS digital

International Journal of Robotics and Mechatronics

Control of Robot Arm Based on Hand Gesture using Leap Motion Sensor Technology

P.D.S.H. Gunawardane, Nimali T. Medagedara, B.G.D.A. Madhusanka

Department of Mechanical Engineering, Open University of Sri Lanka, Nawala, Nugegoda, Sri Lanka

Abstract- This paper presents a mathematical model, design and implementation of the hand gesture controlled robot manipulator. Nowadays, various types of robots are used in medical industries. For the case of surgeries, surgeon uses either direct or through the computer control methods to control instruments required. For electro mechanical designing of a bilateral robot manipulators, controlling is identified as one of the key issues. Hand gesture controlling is one of the major developments emerged in recent past. A technique of controlling robot manipulators in slave side of the bilateral system using Leap Motion Contoller (LMC) based gesture detections is developed with the help of this research. The mathematical model for inverse kinematics, dynamics and trajectory generation is developed and simulated in the "Wolfram Mathematica 10". This paper shows the possibilities of optimizing the joint actuator by improving the trajectory planning. The model is verified using MATLAB simulation in real-time. The simulation is developed using MATLAB Robotics Toolbox 9.10 to imitate the fingertip. A 3 DOF (Degree of Freedom) customized robot model was tested with the developed mathematical model. The prototype was developed using MG995 servo motors and Processing IDE. The development was succeeded in real-time but due to the use of low cost servos, the precession of the system was not very high. In order to apply this system in surgical type applications, the system should be implemented with high robot servos; such as Dynamixel robot actuators.

Keywords—Bilateral Systems, Fingertip Imitating, Robot Manipulator, MATLAB Robotics Toolbox, Kinematic Model, Hand Gesture.

DOI: 10.21535/ijrm.v3i1.930

I. INTRODUCTION

THE technological advancement of this era is in rapid development. In industrial applications bilateral systems are often found and many ongoing researches are available for both master and slave designs. The main features of these systems are convenient HMI (Human Machine Interface) in master side, reduced time delay in tele-presence and goal oriented design of slave side.

To measure the effectiveness, the following specific master designs from the research papers were selected and evaluated using parameters shown in **TABLE 3**. The main master side designs were Joystick Control [1] as shown in **Figure 1.a**, Haptic Control [2] (mechanical or touch sensitive) as in **Figure 1.b**, Exoskeleton Control [2] as in **Figure 1.c**, Hand Gesture Control [3] as in **Figure 1.d**, BCI (Brain Control Interface) [4] as in **Figure 1.e**, Electromyography Control [5] [6] as in **Figure 1.f,** Eye Gaze Control [7] as shown in **Figure 1.g** and Human Body Gesture Control [8] [9] as shown in **Figure 1.h**, respectively [15].

Figure 1 Availability of master designs: [a] Joystick Control, [b] Haptic Control, [c] Exoskeleton Control, [d] Hand Gesture Control, [e] Brain Control Interface, [f] Electromyography Control, [g] Eye Gaze Control, [h] Human Body Gesture Control

TABLE 1 CALCULATED EFFECTIVENESS

Methodology	Effectiveness
Joystick/Remote Control (RC)	35
Electromyography	3
Voice Control (VC)	30
Haptic (mechanical/touch/glove)	30
Hand Gesture Vision Attentive	60
Hand Gesture Glove Based	30
Hand Gesture Exoskeleton	30
Body Gesture (Stereo Vision Based)	50
Eye Gaze (Stereo Vision Based)/ Face Recognition (EG)	45
Brain Control Interface (BCI)	20

TABLE 1 shows the effectiveness of the master designs in bilateral systems. Contactless is a direct parameter and it

Corresponding author: H. Gunawardane (e-mail: pdgun@ou.ac.lk).
This paper was submitted on November 26, 2016; revised on March 29, 2017; and accepted on May 30, 2017.

identifies the operator independence of the methodology, level of the wear, response and high repeatability. To decide the controllability, requirement of mapping (in brain) to operator to control (e.g.:- for joystick control more brain mapping is required with the real environment to make control smooth), ease of feedback and making control decisions (e.g.:- understanding the feedback and capability of taking control decisions fast) and supportiveness for the reaction (speed of the overall reaction, e.g.:- operator might get feedback in an easy way to react fast) are considered. The complexity of the device is defined by requirement of computation (Hardware, e.g.:- maximum processing speed required to smooth run), complexity of the algorithm required (e.g.:- in order to control the slave to solve the equations for velocity, acceleration, position & orientation), complexity of the electromechanical interface requirement (e.g.:- special mechanisms/ servos might need to interface). For the analysing of response time delay in communication and processing delays (e.g.:- expectance of the communication delay in tele-presence), intermediate device requirement (e.g.:- some system required special software to run in a PC to work with it) and feedback generation and operator awareness to the feedback (e.g.:- as a feedback, slave side might have vibrations and therefore vision might not be enough to operator to control smoothly) are considered. For the reliability and safety factor, robustness to face emergency and critical situations (e.g.:- ease of moving from here to there in a critical situation) and how system behave if operator get seriously damaged (e.g.:- a sudden damage to operator and the response for it) are considered [10]. Based on above factors (three for each) the effectiveness was analysed and the values were calculated. **Figure 2** shows the graphical representation of the effectiveness of master design methodologies. It shows that the highest effectiveness is in Hand Gesture vision attentive systems and lowest is in Electromyography. Other systems averagely vary with their relative effectiveness values and each of these methodologies are application oriented. For example Electromyography is not suitable for epidemic environment but suits for a rehabilitation type robot system.

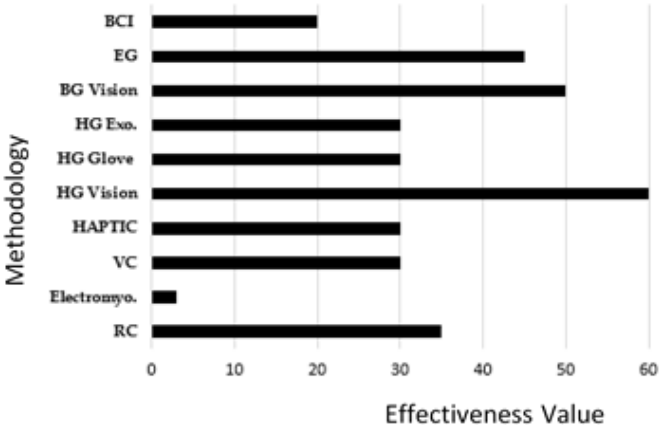

Figure 2 Methodology vs Calculated Effectiveness

Therefore it is useful to develop a mechanism which detects the gesture and control the robot. Basically fingertip imitating robot was the first step of the development. The main reasons for choosing Leap Motion Controller as the 3D sensor for hand gesture detection is low cost with respect to other available 3D sensor modules, range of the sensor (1m) fits for the application, sensor module is specially designed for the hand gesture tracking, free SDK (Software Development Kit) and library availability, availability of the free support such as blogs, forums and free testing software and applications, MATLAB compatibility (matleap integration between MATLAB and LMC is possible), application novelty for manipulator robot integration, accuracy fits for the application, lightweight, portable and plug and play device. The design orientation of both hardware and software of the Leap Motion is to hand gesture application development. Also above mentioned features are readily available with the system and causes for the selection of this device. Especially a survey is carried out among other available 3D cameras.

Hand gesture controller design is based on stereo vision technology. Stereo vision is meant to be the extraction of 3D information from the digital images. It is artificial stereopsis which is followed by two cameras to obtain depth and 3D structural information. Literature survey [10] reveals that 3D data acquisition vastly increases with the potential industrial applications within last few years. Accuracy and the robustness of 3D sensors were increased while dropping the price. Mainly 3D sensors are used for object tracking, motion analysis, 3D scene reconstruction and gesture based user interfaces.

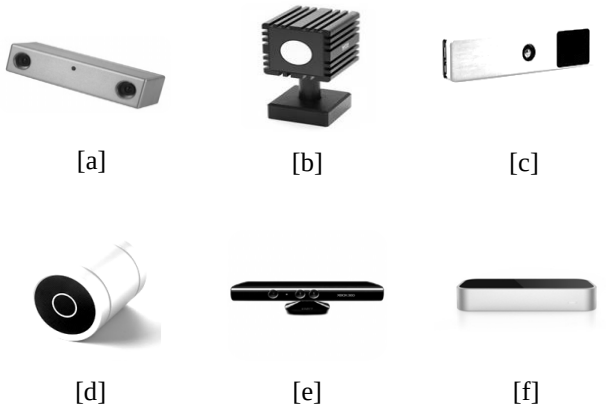

Figure 3 Available Types of 3D Sensors [a] Bumble bee 2 Sensor [b] Mesa Swissranger 4000 [c] PMD Camboard PICO [d] FHOTONIC B70 [e] Microsoft Kinect Sensor [f] Leap Motion Controller

Mainly optical 3D sensors can be categorized by its design mechanism. Structured light analyses the deformation of a known pattern in to unknown surface to determine the 3D shape and Microsoft Kinect sensor [11] shown in **Figure 3.e** is the well-known example. The ToF (Time of Flight) 3D cameras are based on the well-known time of flight principle which is based on phase shift and MESA Swissranger 4000 [12] shown in **Figure 3.b**, PMD (Photonic Mixer Device) [13] Camboard Pico shown in **Figure 3.c** and FHOTONIC B70 shown in **Figure 3.d** are well known examples. Stereo vision cameras are based on two optical 2D cameras with known extrinsic parameters. The concept of determining the depth in the scene

is based on searching correspondence points in both 2D images and Bumble Bee 2 sensor [14] shown in **Figure 3.a** and LMC [15] shown in **Figure 3.f** are famous examples.

In LMC both stereoscopic vision technology is integrated with structured light technology.

TABLE 2 COST ANALYSIS

Name of the Sensor	Approx. Price in USD
Microsoft Kinect	200
Mesa Swissranger 4000	2199
Camboard PICO	800
Leap Motion Controller	99

Table 2 shows the cost analysis of the 3D sensors with approximate selling prices without shipping costs. Above table shows the affordability of the Leap Motion Controller for low cost applications and developments. In-depth analysis of above sensors shows that the ranges were varied from centimetres to few meters. Mainly the Leap Motion Controller is working in 1m range. The specialty of the Leap Motion is the design orientation for hand gestural tracking.

II. MATHEMATICAL MODEL

The design of the Fingertip Imitating Robot Manipulator is a bilateral system and the main design comes as master side design, slave side design and communication in between. Mainly the master side is the *LMC (Leap Motion Controller)* which detects the fingertip and the slave side is the *Robot Manipulator Mechanism*. So the system was designed individually for two systems and then finally both systems has been integrated.

Leap Motion Controller is a conjunction with its API (Application Programming Interface) drive positions in Cartesian space with predefined objects like finger tips, pen tips etc. This is designed as a USB (Universal Serial Bus) peripheral device. Two monochromatic IR cameras and three infrared LEDs (Light Emitting Diodes) are assembled as shown in Figure 4. The LEDs generate a 3D pattern of dots of IR light and cameras were assembled in such a fashion to generate 300 frames per second of reflected data.

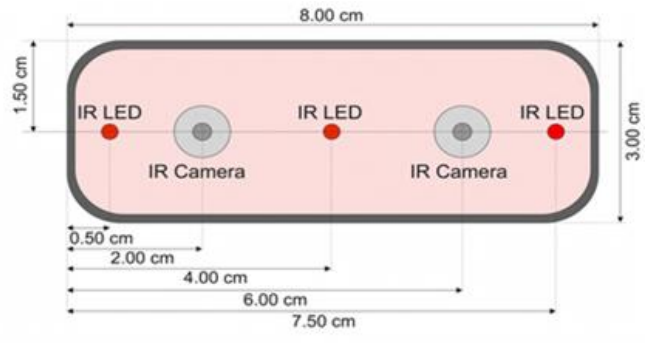

Figure 4 Construction of the LMC

Then the data was sent to the V2 Tracking software and 2D frames generated by two cameras were used to analyse furthermore to generate 3D position data. As shown in the **Figure 6**, LMC consists of a right-handed Cartesian coordinate system. The origin lies on the top of the LMC. X and Z axes lie in the horizontal plane and Y axis is vertical. **TABLE 3** shows the measurement details of the LMC. The measurements can vary according to the external conditions.

TABLE 3 LMC I/O SENSITIVITY

Parameter	Measurement
Distance	Millimetres
Time	Microseconds
Speed	Millimetres/ Seconds
Angle	Radians

As shown in the **Figure 6** the frame {0} is assigned to the LMC and the frame {1} is assigned to the finger tip. LMC detects the position, orientation and the velocity of the finger tip. LMC provides the X, Y, Z, α, β, γ, V_x, V_y, and V_z values with respect to the coordinate frame {0}. In order to relate these two frames basically a homogeneous transformation matrix between frame {0} and {1} was derived. If the rotation is given by α, β, γ which is $\theta_1, \theta_2, \theta_3$ and the translation is X, Y and Z. Then the composite transformation matrix can be derived as,

$$ {}^{0}_{1}T = \begin{bmatrix} C_2C_3 & S_1S_2S_3 - C_1S_3 & C_1S_2C_3 + S_1S_3 & X \\ C_2S_3 & S_1S_2S_3 + C_1C_3 & C_1S_2S_3 - S_1C_3 & Y \\ -S_2 & S_1C_2 & C_1C_2 & Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1) $$

where $C_i = \cos(\theta_i)$ and $S_i = \sin\theta_i$. Equation (1) should be operated with the inverse kinematics of the robot mechanism to find the joint angles. **Figure 8** shows the mechanism of the robot. It consists three R (Revolute) type joints, three links and end-effector. Each joint is an actuator which capable of performing the revolve action (e.g.:- DC servo motors). The main task is to track the fingertip with the aid of the LMC and imitate by controlled end-effector motion. It is required to develop a mathematical model between robot base and the end-effector through joint angles. Therefore coordinate frames were assigned to the system. The world coordinate frame was placed in the base of the robot and end-effector was mapped with respect to the base frame. **Figure 8** shows how the coordinate frames are assigned to the robot frame work/mechanism. The frames are assigned to satisfy the Denavit-Hartenberg notation. **Figure 7** shows the frame assignment to the robot by considering the home position. By inspecting the frame assignment in **Figure 8** and by referring to the DH notation, and the DH table can be developed accordingly.

Figure 5 Available Classes in LMC [a] Hand Class API Model [b] Finger Class API Model [d] Tool Class API Model *starting from left (LMC website)

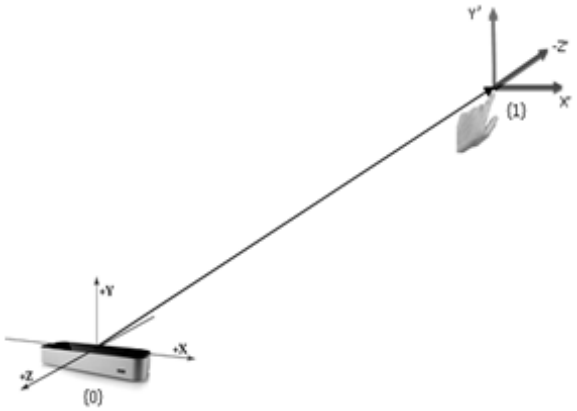

Figure 6 Frame assignment of the gesture workspace

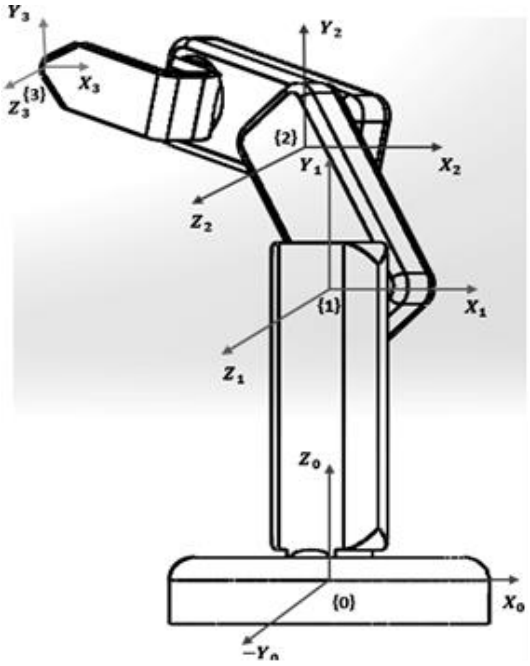

Figure 7 Frame assignment in joint space

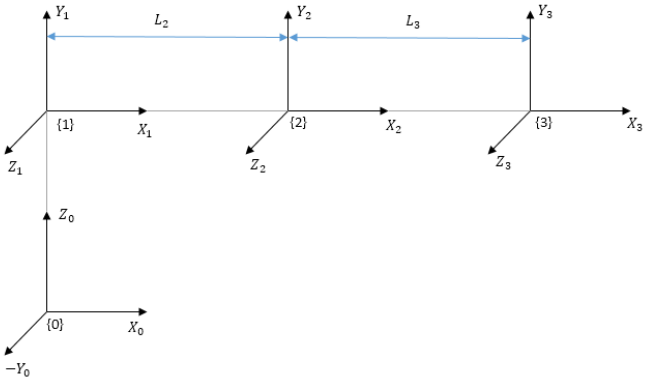

Figure 8 Frame assignment by considering the home position

The resultant forward kinematics equation for robot mechanism can be shown as in Equation (2),

$$ {}^{0}_{3}T = \begin{bmatrix} C_1C_{23} & -C_1S_{23} & S_1 & C_1[L_3C_{23} + L_2C_2] \\ S_1C_{23} & -S_1S_{23} & -C_1 & S_2[L_3C_{23} + L_2C_2] \\ S_{23} & C_{23} & 0 & L_3S_{23} + L_2S_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2) $$

where $C_i = \cos\theta_i$, $S_i = \sin\theta_i$, $C_{ij} = \cos(\theta_i + \theta_j)$ and $S_{ij} = \sin(\theta_i + \theta_j)$. Equation (2) shows the forward kinematics solution which relates frame {0} (base frame) to the end-effector frame {3}. For the known joint angles then equation (2) will provide the end-effector configuration (position and orientation). To find required joint angles for a known end-effector configuration geometrical solutions were obtained as follows, say the known end-effector configuration is,

$$ T = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \quad (3) $$

Then with respect to the forward kinematic model developed above, following joint angle values can be obtained analytically,

$$ \theta_1 = Atan2\,(a_{24}, a_{14}) \quad (4) $$

$$ \theta_2 = atan2[\,(-L_3a_{31} + a_{34}), \pm\sqrt{(-L_3a_{11} + a_{14})^2 + (-L_3a_{21} + a_{24})^2}\,] \quad (5) $$

$$ \theta_3 = Atan2[a_{31}, a_{32}] - \theta_2 $$

For a known particular end-effector configuration by using equations (4), (5) and (6) the joint space values (θ_1, θ_2 and θ_3) related to that particular end-effector configuration can be obtained. Also LMC provides us the fingertip velocity in terms of V_x, V_y, and V_z and the data is used to develop a mathematical relation to imitate fingertip velocities. The Jacobian matrix for the 3DOF robot was derived as,

$$ J = \begin{bmatrix} -S_1(L_3C_{23} + L_2C_2) & -C_1(L_3S_{23} + L_2S_2) & -L_3C_1S_{23} \\ C_1(L_3C_{23} + L_2C_2) & -S_1(L_3S_{23} + L_2S_2) & -L_3S_1S_{23} \\ 0 & L_3C_{23} + L_2C_2 & L_3C_{23} \\ 0 & S_1 & S_1 \\ 0 & -C_1 & -C_1 \\ 1 & 0 & 0 \end{bmatrix} \quad (7) $$

Then the inverse Jacobian is calculated as follows,

$$ [J^{-1}] = \frac{adj(J)}{\det(J)} = \frac{F}{K} \quad (8) $$

The joint space velocities are as follows,

$$ \dot{q} = J^{-1}(q)V_e \quad (9) $$

$$ \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} = \left[\frac{F}{K}\right] \cdot \begin{bmatrix} V_x \\ V_y \\ V_z \end{bmatrix} \quad (10) $$

Equations (9) and (10) lead to the joint space velocity results for known Cartesian space velocities. Therefore detected fingertip position, orientation and velocities are used to develop the joint angles values and joint space velocities of the robot to imitate the fingertip by the end-effector of the robot. Then the

trajectory was generated by considering the 5 degree polynomial shown in the Equation (11).

$$\theta(t) = a_o + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \quad (11)$$
$$\dot{\theta}(t) = a_1 + a_2 t + 3a_3 t^2 + 4a_4 t^3 + 5a_5 t^4 \quad (12)$$
$$\ddot{\theta}(t) = 2a_2 + 6a_3 t + 12a_4 t^2 + 20a_5 t^3 \quad (13)$$

Equations (12) and (13) show the equations for the velocity and acceleration, which are the first and second derivatives of the trajectory polynomial. **TABLE 4** shows the start and end data for the trajectory solution and therefore by solving the matrix shown in (14) the trajectory of the robot was developed. **Figure 9** shows the behavior of the MATLAB generated trajectory for a particular case. In this case beginning and end points were considered to be taken between 0 and 1 with 50 steps. The velocity and acceleration are taken to be zero in two extreme points.

$$\begin{bmatrix}\theta_s\\ \theta_f\\ \dot{\theta}_s\\ \dot{\theta}_f\\ \ddot{\theta}_s\\ \ddot{\theta}_f\end{bmatrix} = \begin{bmatrix}1 & t_s & t_s^2 & t_s^3 & t_s^4 & t_s^5\\ 1 & t_f & t_f^2 & t_f^3 & t_f^4 & t_f^5\\ 0 & 1 & 2t_s & 3t_s^2 & 4t_s^3 & 5t_s^4\\ 0 & 1 & 2t_f & 3t_f^2 & 4t_f^2 & 5t_f^4\\ 0 & 0 & 2 & 6t_s & 12t_s^2 & 20t_s^3\\ 0 & 0 & 2 & 6t_f & 12t_f^2 & 20t_f^3\end{bmatrix}\begin{bmatrix}a_0\\ a_1\\ a_2\\ a_3\\ a_4\\ a_5\end{bmatrix} \quad (14)$$

TABLE 2 START AND END POINT PARAMETERS

	Time	θ	$\dot{\theta}$	$\ddot{\theta}$
Start	t_s	θ_s	$\dot{\theta}_s$	$\ddot{\theta}_s$
End	t_f	θ_f	$\dot{\theta}_f$	$\ddot{\theta}_f$

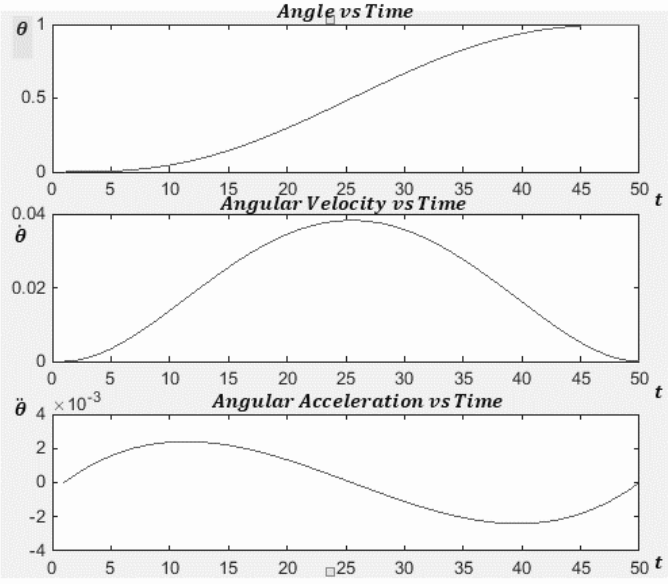

Figure 9 MATLAb generated Traj. Plot for a single joint

$$Q = M(q)\ddot{q} + C(q,\dot{q}) + G(q) \quad (15)$$

The Dynamic model of the robot can be derived as shown in the equation (15). Where Q= $Joint\ Torque\ Matrix$ $M(q) = Inertia\ Matrix$, $C(q,\dot{q}) = Corriolis\ Matrix$ and $G(q) = Gravity\ Vector$. For the known joint angular displacement, angular velocity and angular Equation (15) gives the joint torques. Using the Lagrange-Euler formalization Equation (15) can be rewritten as shown in (16).

$$\begin{bmatrix}\tau_1\\ \tau_2\\ \tau_3\end{bmatrix} = \begin{bmatrix}M_{11} & M_{12} & M_{13}\\ M_{21} & M_{22} & M_{23}\\ M_{31} & M_{32} & M_{33}\end{bmatrix}\begin{bmatrix}\ddot{\theta}_1\\ \ddot{\theta}_2\\ \ddot{\theta}_3\end{bmatrix} + \begin{bmatrix}C_{11}\\ C_{12}\\ C_{13}\end{bmatrix} + \begin{bmatrix}G_{11}\\ G_{12}\\ G_{13}\end{bmatrix} \quad (16)$$

Equation (17) to (24) is the Inertia Matrix, Equation (25) to (27) is the Coriolis Matrix and Equation (28) to (30) is the Gravity vector. In this model, friction in considered to be zero. The Equations shown above are the dynamic model of the robot and this model can be used to implement precise joint control such as Fuzzy PID.

$$M_{11} = \frac{1}{2}m_1 l_1^2 + \frac{1}{3}m_2 l_2^2 cos^2(\theta_2) + \frac{1}{3}m_3 l_3^2 cos^2(\theta_2+\theta_3) + m_3 l_2^2 cos^2(\theta_2) + m_3 l_2 l_3 \cos(\theta_2+\theta_3)\cos(\theta_2) \quad (17)$$
$$M_{12} = 0 \quad (18)$$
$$M_{13} = 0 \quad (19)$$
$$M_{21} = 0 \quad (20)$$
$$M_{12} = \frac{1}{3}m_3 l_3^2 \quad (21)$$
$$M_{22} = \frac{1}{3}m_2 l_2^2 + \frac{1}{3}m_3 l_3^2 + m_2 l_2^2 + m_3 l_2 l_3 \cos(\theta_3) \quad (22)$$
$$M_{23} = \frac{1}{3}m_3 l_3^2 + m_2 l_2^2 + \frac{1}{3}m_3 l_2 l_3 \cos(\theta_3) \quad (23)$$
$$M_{31} = 0 \quad (24)$$
$$M_{32} = \frac{1}{2}m_3 l_3^2 + m_3 l_2^2 + \frac{1}{3}m_3 l_2 l_3 \cos(\theta_3) \quad (25)$$

$$C_{11} = [-\frac{4}{3}m_2 l_2^2 \sin(2\theta_2) - \frac{1}{3}m_3 l_3^2 sin2(\theta_2+\theta_3) - m_3 l_2 l_2 \sin(2\theta_1+\theta_3)]\dot{\theta}_2\dot{\theta}_1 [-\frac{1}{3}m_3 l_3^2 sin2(\theta_2+\theta_3) - m_3 l_2 l_3 \cos(\theta_2)\sin(\theta_2+\theta_3)]\dot{\theta}_3\dot{\theta}_1 \quad (26)$$

$$C_{21} = [-m_3 l_2 l_3 \sin(\theta_3)]\dot{\theta}_2\dot{\theta}_3 + \left[-\frac{1}{2}m_3 l_2 l_3 \sin(\theta_3)\right]\dot{\theta}_3^2 + [\frac{1}{6}m_2 l_2^2 sin2(\theta_2) + \frac{1}{6}m_3 l_3^2 sin2(\theta_2+\theta_3) + \frac{1}{2}m_3 l_2^2 \sin(2\theta_2) + \frac{1}{2}m_3 l_2 l_3 \sin(2\theta_2+\theta_3)]\dot{\theta}_1^2 \quad (27)$$

$$C_{31} = \left[\frac{1}{2}m_2 l_2 l_3 \sin(\theta_3)\right]\dot{\theta}_2^2 + [\frac{1}{6}m_3 l_3^2 sin2(\theta_1+\theta_3) + \frac{1}{2}m_2 l_2 l_3 \cos(\theta_2)\sin(\theta_2+\theta_3)]\dot{\theta}_1^2 \quad (28)$$
$$G_{11} = 0 \quad (29)$$
$$G_{21} = \frac{1}{2}m_2 g l_2 \cos(\theta_2) + \frac{1}{2}m_3 g l_3 \cos(\theta_2+\theta_3) + m_3 g l_2 \cos(\theta_2) \quad (30)$$
$$G_{31} = \frac{1}{2}m_3 g l_3 \cos(\theta_2+\theta_3) \quad (31)$$

III. COMPUTER SIMULATION

The mathematical model for 3DOF robot manipulator is developed in the "Mathematica 10" and generated the angle, velocity and the acceleration plots for a particular case as shown in the **Figure 10**. Cartesian displacement of the fingertip is taken as 1 unit along x, y, z axes and –Pi/6, Pi/12, -Pi/12 rotation in respective axes with 15% reduction in velocity in all three axes. Initial and final joint accelerations were taken to be zero. Then the joint torques were calculated for a particular case.

This model was tested with real fingertip data. Robotics Tool Box for MATLAB (RTB 9.10 by Peter Corke) was used to develop the 3DOF robot model for the simulation. Link objects and the SerialLink constructer was used to develop the model. LMC detects the position, orientation and the velocity parameters of the fingertip and then the MATLAB matleap library reads the parameters. Then from the position and orientation data, the composite transformation matrix was generated and then it was equated with inverse kinematics to find the joint angle values.

Also velocity was used to find the required joint space velocities. Based on these parameters robot trajectory was generated and then the simulation was imitated using the fingertip motion. **Figure 11** shows the 3DOF robot model used for the simulation.

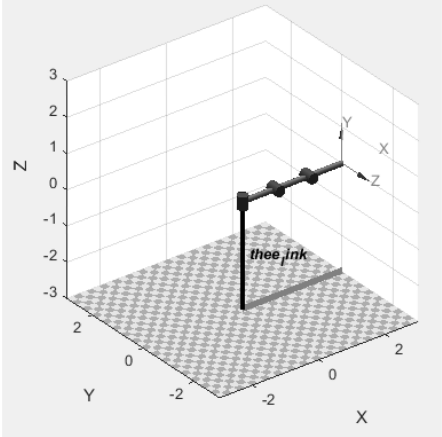

Figure 9 3DOF robot model

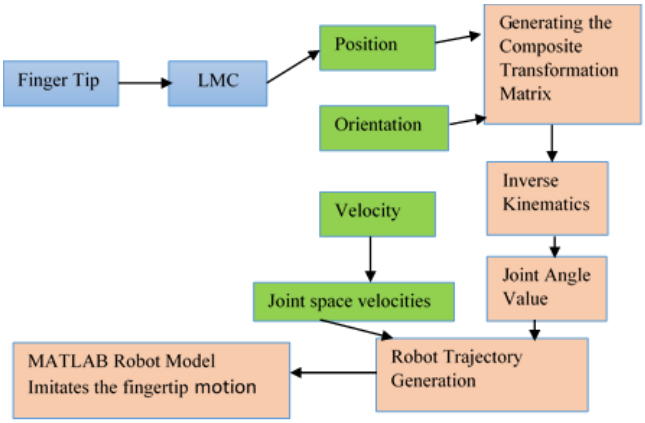

Figure 11 Block diagram of the gesture controlled robot simulation

Figure 11 shows the block diagram of the MATLAB simulation. It is a crucial issue to design and develop the robots due to the singularities. This program is designed by considering both interior and boundary singularities. To avoid interior singularities, after performing inverse kinematics values are checked and excluded abnormal. Then with the aid of this algorithm, proposed system tested.

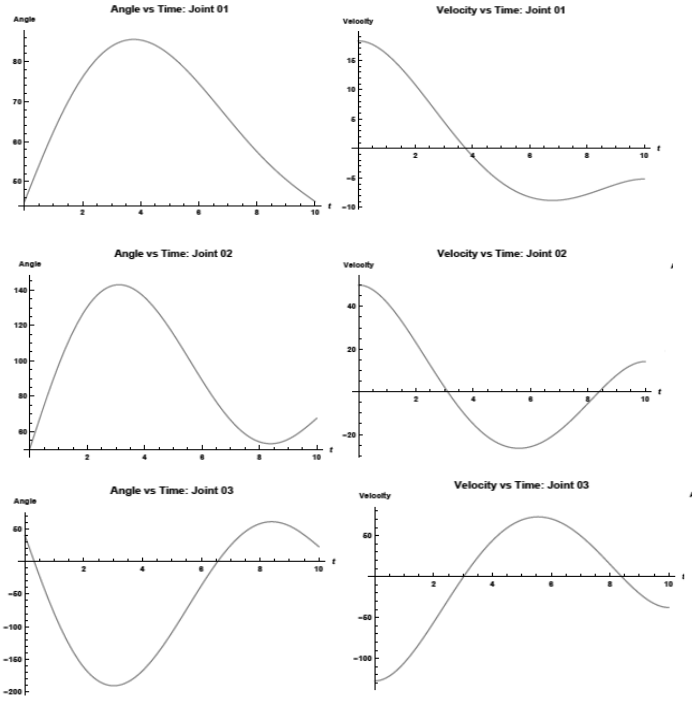

Figure 10 Generated angle and velocity diagrams using Mathematica software tool.

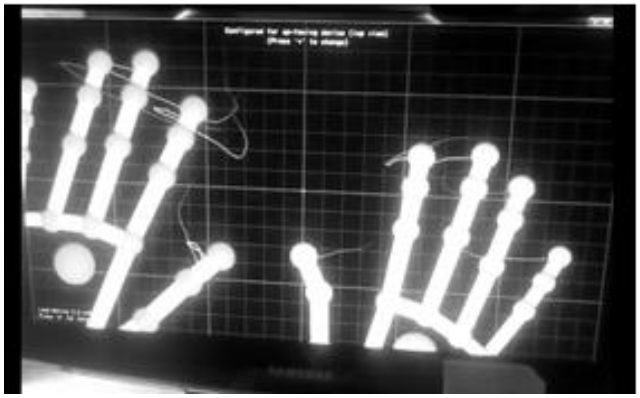

Figure 13 Skelton of the tracked two hands

Figure 14 AR development

In this gesture controlled robot application, captured LMC data were merged with a video to facilitate AR . **Figure 14** shows the setup for the AR development in the Processing window. The blue dot represents the fingertip and it moves in 3D space with the aid of the OPENGL library. A web cam was used for the tests and therefore AR in 3D cannot be visualized effectively. In future developments, merging this with a 3D camera will create a more interactive environment to the user. Once fingertip data is extracted, inverse kinematic and dynamic solutions are obtained and data is sent to the servo motors. **Figure 15** shows the block diagram of the gesture controlled robot with processing.

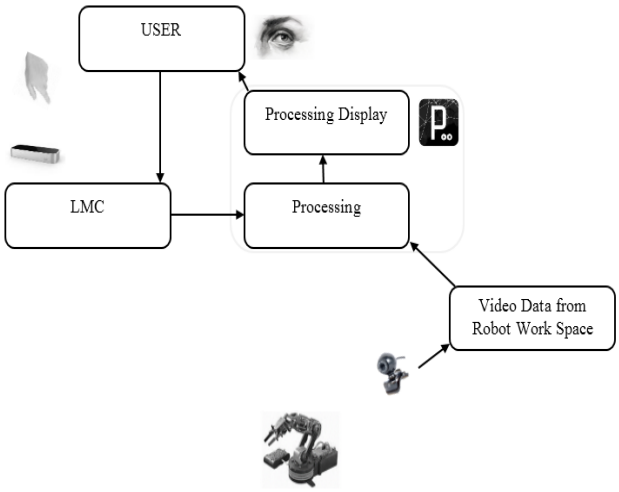

Figure 15 Block Diagram of the Gesture Control robot with Processing

Figure 16 shows 3DOF robot manipulator prototype. It consists MG 995 TowerPro servo motors. Arduino Uno is used as the interface to connect Processing with servo motors. Once fingertip data are extracted and inverse kinematic solutions are obtained, the Processing algorithm is developed to vary the pulse output of the Arduino UNO digital pin to control the position of the servo motor. (View the development in the Youtube Channel:

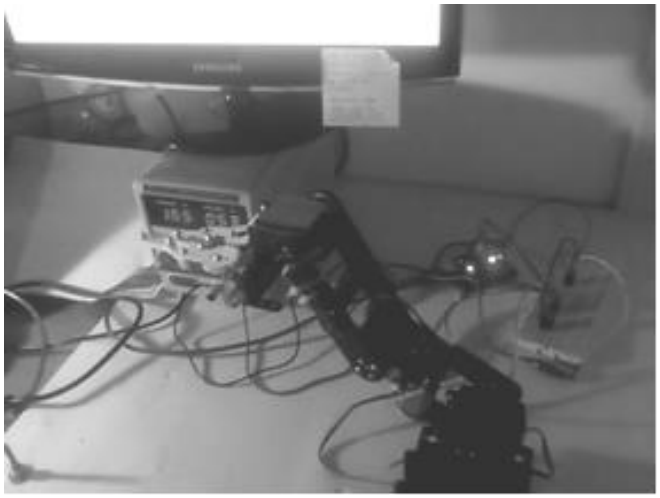

Figure 16 3 DOF robot manipulator mechanism

IV. RESULTS AND DISCUSSION

It is known that Robotics and Automation Technologies are developing very fast, research and developments promises more reliable and sophisticated innovations to make things easy. The recent incidents like EBOLA showed the requirement of developing much more promising technologies to reduce death count in such disastrous situation. The recent study by PDSH Gunawardane [10] discussed about the competitiveness of the Master Designs and illustrates the effectiveness of gesture controlling. Among so many available gesture technologies, 3D spectroscopic vision systems are popular today and emerging of devices like Microsoft Kinect and Leap Motion Controller shows the advancement of technology. These novel technologies have opened new doorways to many applications in control and mechatronics.

It is very important to assure the health of the workers in medical sector from epidemic environments. Therefore this Hand Gesture Controlled Robot Manipulator for Medical Industry was developed to play that role in an epidemic environment. In order to fulfill the partial requirement in Bachelor of Technology in Engineering (Mechatronics) this has been implemented for the last nine months. Throughout this project verification of the potential of Gesture Controlling in 3D IR Spectroscopic vision technology in Leap Motion to controlling a robot was simulated using MATLAB. Then a real time prototype was developed. Due to the limited time period available, the system was scaled down by considering the feasibility. Once the concept was verified using MATLAB, the system was developed using Processing interactive programming IDE which is a major step towards implementing a standalone version of this system. Due to the limited time availability the project was scaled down. However, this project compromises the capacity to undergo future developments and implementations,

- This can be implemented as 6DOF robot manipulator which is more aplicable as an industrial robot.
- The end-effector is designed to be a servo actuator based gripper which can be replaced by any other actuator which is application dependent.
- Due to the low budget of the project MG 995 Tower Pro Servos (Hobby Servos) are used hence introducing the robot servos like Dynamixel 12A will enable feedback control.
- The introducing of a 3D camera to Augmented Reality based control window will promises as more reliable controlling interface.
- Enabling IoT (Internet of Things) in this project can take this project to another dimension.
- This project has considered the slave side of the bilateral system to be a manipulator robot, therofre by replacing it with Soft Robots and Smart Materials the flexibility of the bilateral system would be increased.
- Developing a better library (eg. a Robotics Library for Processing) for these systems will enable repaid prototyping of new ideas.

REFERENCES

[1] K. M. Tsui, "Design and evaluation of a visual control interface of a wheelchair mounted robotic arm for users with cognitive impairments," 2008.

[2] N. Popescu, M. Poboroniuc, D. Popescu and C. D. Popescu, "Intelligent haptic roboticglove for patients diagnosed with cerebrovascular accidents," in *IEEE International Conference on Rehabilitation Robotics*, Iasi, Romania, 2013. CrossRef

[3] M. T. Wolf, C. Assad, M. T. Vernacchia, J. Fromm and H. L. Jethani, "Gesture-Based Robot Control with Variable Autonomy from the JPL BioSleeve," in 2013 IEEE International Conference on Robotics and Automation (ICRA), Karlsruhe, Germany, 2013. CrossRef

[4] K. Yanagisawa, H. Sawai and H. Tsunashima, "Development of NIRS-BCI system using perception," in 2012 12th International Conference on Control, Automation and Systems, Jeju Island, Korea, 2012.

[5] A. Harada, T. Nakakuki, M. Hikita and C. Ishii, "Robot finger design for myoelectric prosthetic hand and recognition of finger motions via surface EMG," in Proceedings of the International Conference on Automation and Logistics, Hong Kong and Macau, 2010. CrossRef

[6] M. V. Liarokapis, P. K. Artemiadis, P. T. Katsiaris and K. J. Kyriakopoulos, "Learning Task-Specific Models for Reach to Grasp Movements: Towards EMG-based Teleoperation of Robotic Arm-Hand Systems," in The Fourth IEEE RAS/EMBS International Conference, Roma, Italy, 2012. CrossRef

[7] H.-P. Huang, J.-H. Chen and H.-J. Jian, "Development of the Joint Attention with a New Face Tracking Method for Multiple People," in IEEE International Conference on Advanced Robotics and its Social Impacts, Taipei, Taiwan, 2008. CrossRef

[8] D. Guo, X. M. Yin, Y. Jin and M. Xie, "Efficient gesture interpretation for gesture-based human-service robot interaction". [Online: Accessed 7 Jan. 2015] VIEW

[9] M. G. Jacob, Y.-T. Li and J. P. Wachs, "Surgical Instrument Handling and Retrieval in the Operating Room with a Multimodal Robotic Assistant," in 2013 IEEE International Conference on Robotics and Automation (ICRA), Karlsruhe, Germany, 2013. CrossRef

[10] P. D. S. H. Gunawardane, N. T. Medagedara and B. G. D. A. Madhusanka, "Studying the Competitiveness of Master Designs in Bilateral Systems for an Epidemic Environment," on 2015 IEEE International Conference on Ubi-Media Computing), Colombo, Sri Lanka, 2015. CrossRef

[11] "Kinect for Windows Sensor Components and Specifications(Library, Microsoft MSDN)," 2014. [Online: Accessed 7 Jan. 2015] VIEW

[12] "SR400 (Imaging, MESA)," 2014. [Online: Accessed 7 Jan. 2015]. VIEW

[13] "Reference Design Brief CamBoard (pmdtec.com, Leading 3D chip technology provider)," 2014. [Online: Accessed 7 Jan. 2015]. VIEW

[14] "Bumblebee XB3 1.3 MP Color FireWire 1394b 3.8mm (Sony ICX445) Imaging, POINT GREY," 2014. [Online: Accessed 7 Jan. 2015]. VIEW

[15] Leap Motion Product," 2014. [Online: Accessed 7 Jan 2015]. VIEW