

Teleoperation of a robotic arm made with Lego Mindstorm using an accelerometer sensor HiTechnic

Pérula Martínez, Raúl «raul.perula@alumnos.uc3m.es »

Teleoperation and Telepresence in Robotic

Carlos III University of Madrid

Abstract – Teleoperation in robotics is one of the most important topics today because there are not still autonomous systems with intelligence so great as to be very precise in the work that have to play. Therefore, research is much in this field to allow operators to have a precise robot management and in the easiest way possible. In the case developed, I has made the programming of a Lego Mindstorm robot formed by the shape of a robotic arm that can move in the same way that the operator moves.

Index term – Lego Mindstorm, robotic arm, teleoperating control, accelerometer sensor.

I. INTRODUCTION

From the beginning, robotics has looking for getting robots to perform a task the same way as human beings to. This is not easy to get, and sometimes impossible with the means that exist today. Thus, for tasks where there is great difficulty to perform, which requires a large enough precision or when intelligence needs to be high when there is any danger to do the task, usually makes use of the teleoperation.

Teleoperation of the robot can be very useful in many cases because it is a practical way for a robot perform a task, either supervised or controlled by an operator. These tasks may be similar movements made by the operator without any intermediate logic, or tasks in which the operator give instructions and the robot performs the action through a logic.

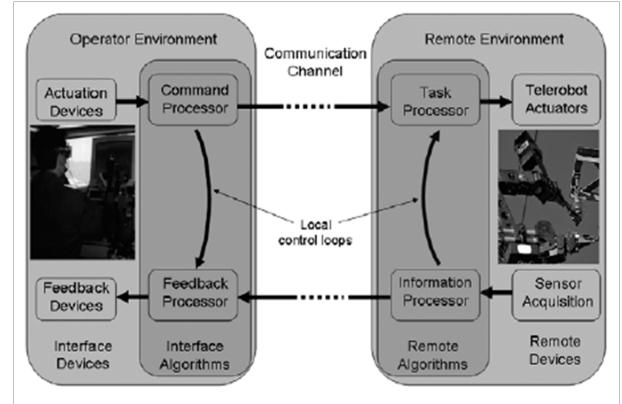
In this work I have done a robotic arm made from Lego Mindstorms NXT system. Although it is not very complex system, you can see the features of teleoperation have and the utility

it can offer to perform tasks more difficult in more complex systems.

The following sections will study the key points to be treated by a teleoperated system and show in detail the programming and the structure of the robot.

II. TELEOPERATION SYSTEM ELEMENTS

This section describe what are the elements that the teleoperated system has, being this system a manipulator guided by an operator operator for several sensor and actuator elements.



Environment teleoperated.

A. Operator

The operator is the person responsible for managing or controlling the robot remotely. He will handle by hand controller (joystick) which shall consist on a couple of sensors that move the device.

B. Teleoperated device

The teleoperated device is the device in the remote area which is controlled by the operator, this will be the robotic arm. It will consist

of a mechanism in which the movement will be equal to that made by the operator. The robotic arm has two degrees of freedom to move around the three coordinate axes (3D). In the end, the robotic device will have a gripper to manipulate objects.

C. Sensors

One of the most important components of any robotic system are the sensors. These are a device that measures a physical quantity and converts it into a signal which can be read by an observer or by an instrument.

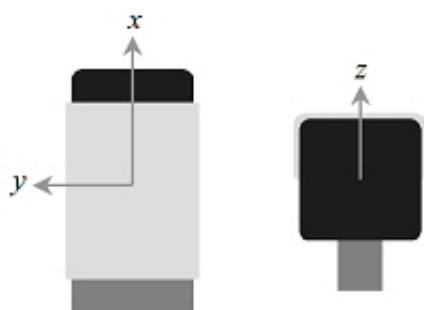
The sensors used for this project have been:

- Acceleration sensor
- Touch sensor

Below are the characteristics of the sensors to have a little more information about it.

Acceleration sensor characteristics

NXT Acceleration Sensor contains a 3-axis accelerometer that measures the acceleration in 3 axes, X, Y and Z. The acceleration is measured in the range of -2g to +2 g with a resolution of approximately 200 units per g.



Acceleration Sensor

The acceleration sensor can also be used to measure the tilt in all 3 axes.

The acceleration sensor connects to an NXT sensor port using a NXT standard cable via I2C digital communications protocol.

The measurement of acceleration of each axis is updated 100 times per second.

This sensor has been encapsulated in a similar way to other MINDSTORMS® elements to maintain a coherent presentation.

The value shown on the NXT represents the acceleration or inclination value for the X axis and will be in a range between 0 and 254.

Touch sensor characteristics

The NXT touch sensor consists of a piece that makes contact when it is pressed and sends a signal to recognize a keystroke.



Touch Sensor

As shown in the figure, the sensor may have two positions, normal, in which there is no touch and we can say it is the resting position. And down position, which may be continuous or only one pulse.

The value shown on the NXT represents if exist or not pulsation.

D. Interface

The interface to control the robotic device is provided by Lego Mindstorm. Based on block-structured programming that gives easy access to a variety of sensors and actuators that can operate the system. This way you can program the system how to behave in response to movements of the operator.

E. Control and communication channels

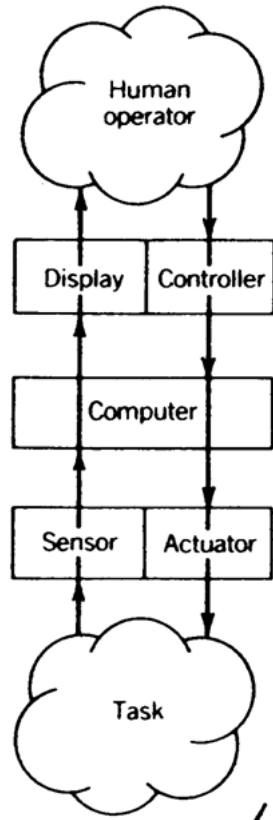
The control and communication channels are a collection of devices that modulate, transmit and adapt a set of signals which are transmitted between the local and the remote area. For this system, the communication channels

are made using a data transmission cables standard NXT working via I2C digital communications protocol.

III. TELEOPERATION ARCHITECTURE

The architecture followed to implement the project has been a control supervised architecture.

The figure shows the overall architecture is based on a system like this.



Supervised control. Manual control architecture.

In our case, the operator is the person responsible for teleoperate the device as discussed above.

The display to be used is the human eye, which can directly observe the movements of the device.

The controller is the joystick that has been created which contains the acceleration sensor

to capture the positions and directions that will have to replicate the device.



Left perfil, joystick (sensors).



Frontal perfil, joystick (sensors).



Right perfil, joystick (sensors).

Sensors capture data necessary to perform their functions the actuators, causing the movement of the device and tasks are executed correctly.



Robotic device right (actuators).



Robotic device back (actuators).

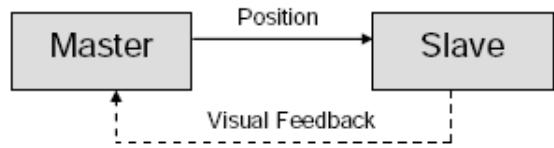


Robotic device front (actuators).

IV. CONTROL MODE

In the master-slave control methodology, the slave robot (teleoperator) exactly replicates the movements of the master. Methods for

controlling master-slave robot systems may be divided into two categories - unilateral control system and a bilateral control system. In a unilateral control system, shown in the figure, no force feedback is available from the slave unit. The only form of feedback to the master unit operator is in the form of vision data. Such a system has the merit of having a simple controller and mechanism; however dexterous manipulation is difficult.



Unilateral master-slave control.

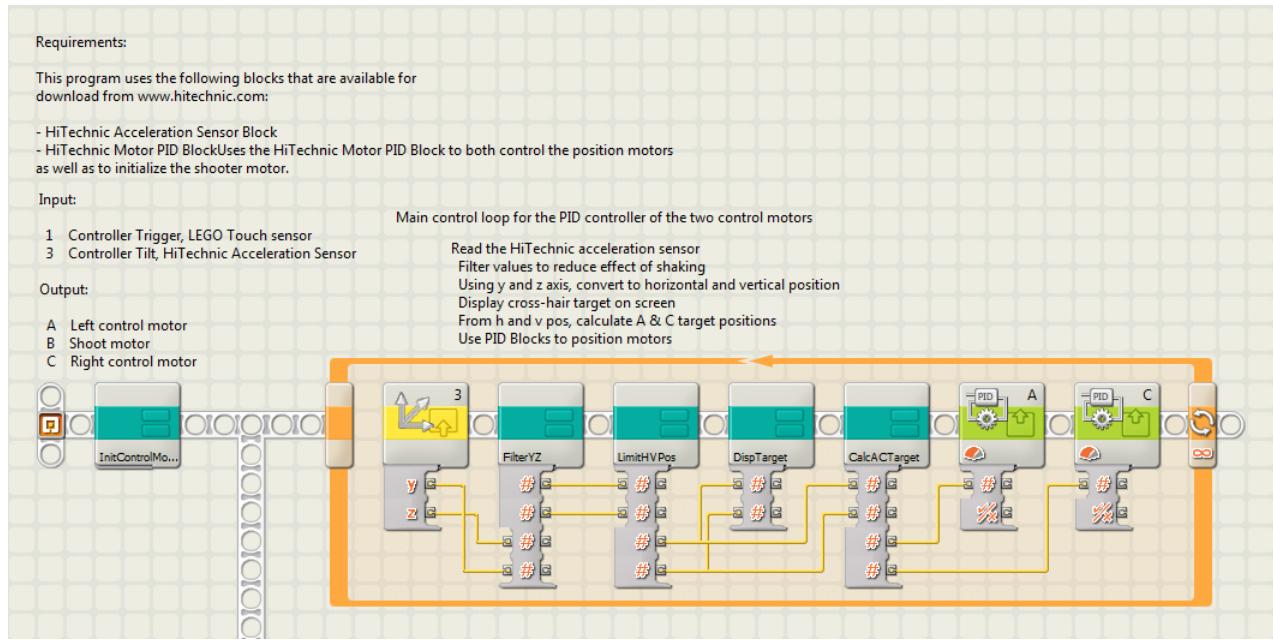
Most people find incredibly easy to use their arms as they have had so much practice. This natural ability of most human beings can be exploited to give a human operator an easy to use tool to control a robot.

This system allows the user to move his hand in a natural way and the robot moves in the same way. In this manner the user is able to effectively and precisely manipulate the robot with very little training [3].

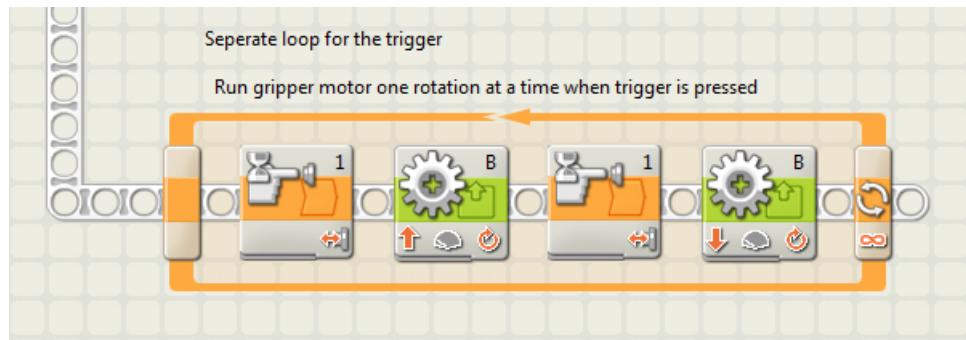
V. SOFTWARE SYSTEM

This section will explain the program which has been created to control and manipulate the robotic device, revealing each one of the blocks used.

The figure shows the structure of the main program, you can observe two distinct parts. The first is responsible for performing control for the arm to move in the same way in which the operator moves the handle. And the second part controls the grip to catch things.



Main program (1).

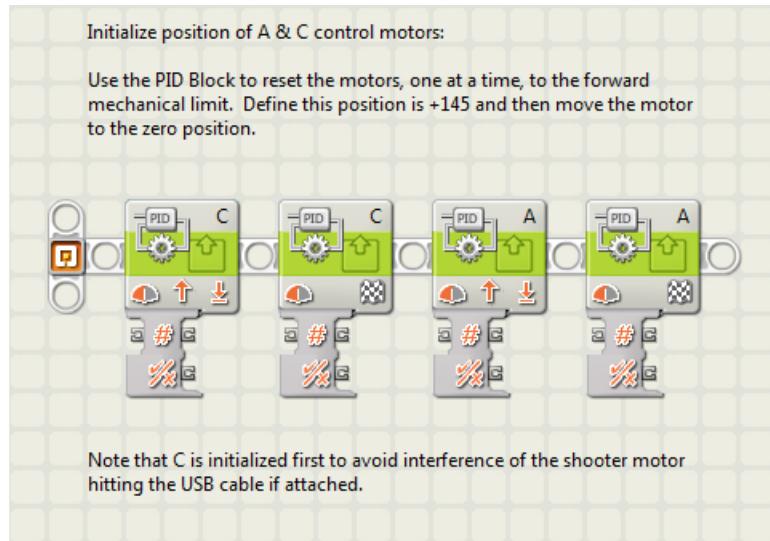


Main program (2).

Inside the main structure of the program there are several blocks which are encapsulations of other subprograms to control the robotic arm. These are:

- Init Control Motors

Set the initial positions of the two motors of the robotic arm. This subblock uses PID block to establish the limits of the motors and in this way controlling the latter movements of the operator replicated will not be out of the established ranges.

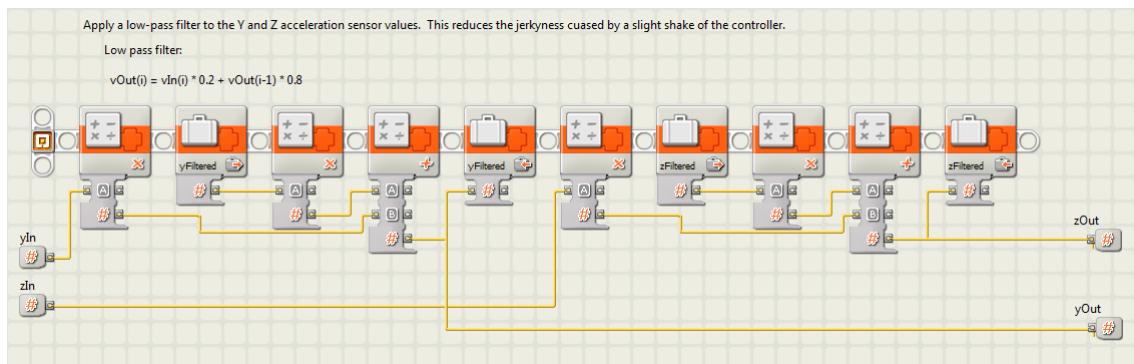


Init Control Motors.

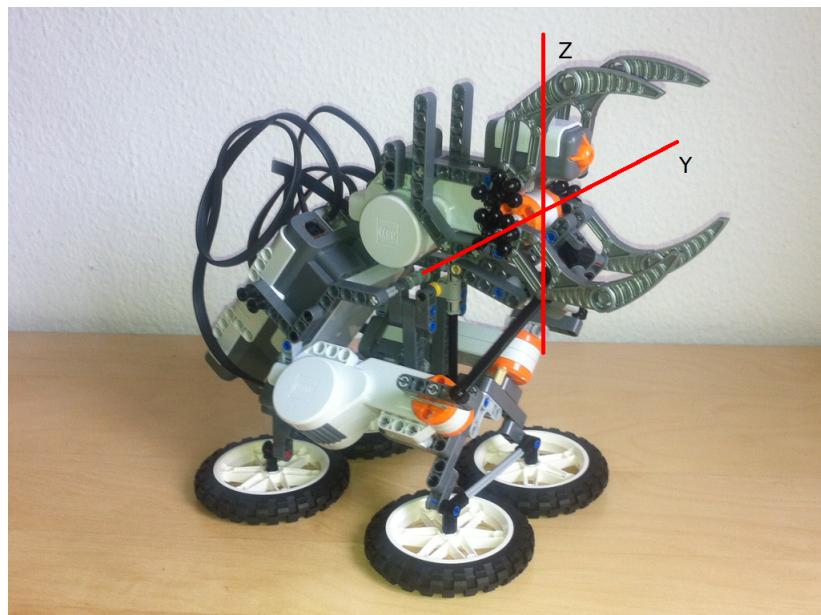
■ Filter Y-Z

This block performs a low-pass filter to the acceleration values of Y and Z. This is

done because it reduces jerkiness caused by the slight shake of the controller. The equation used, as shown in the image, is $vOut(i) = vIn(i) * 0,2 + vOut(i - 1) * 0,8$.



Filter Y-Z.

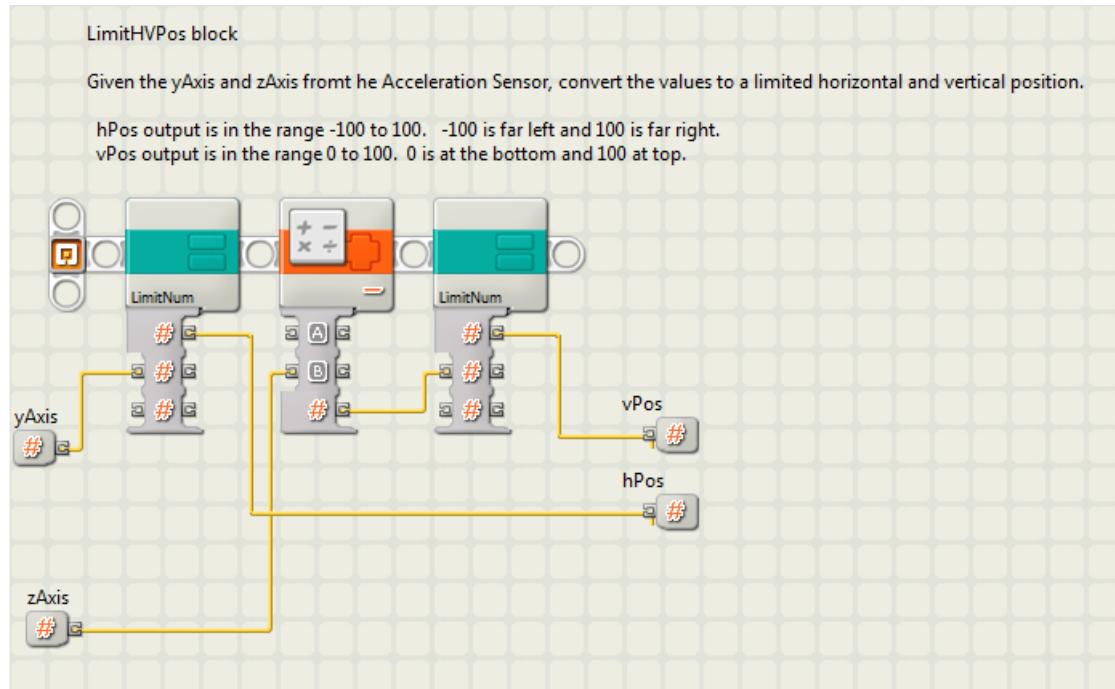


Axes in robotic device.

- Limit H/V Position

The robotic arm movement will be replicated in two directions, horizontally and vertically. Therefore, to use the axes Y (horizontal) and Z (vertical). In this case, this block performs the task of limiting the values of the axes so that the movement

can be controlled. The ranges of values which are managed, horizontally, from -100 to 100 where the first is the left end and the second the right end which can reach the arm, vertically, from 0 to 100 being the first the lower end and the second the top. These values were chosen because they are quite intuitive in a traditional coordinate system.

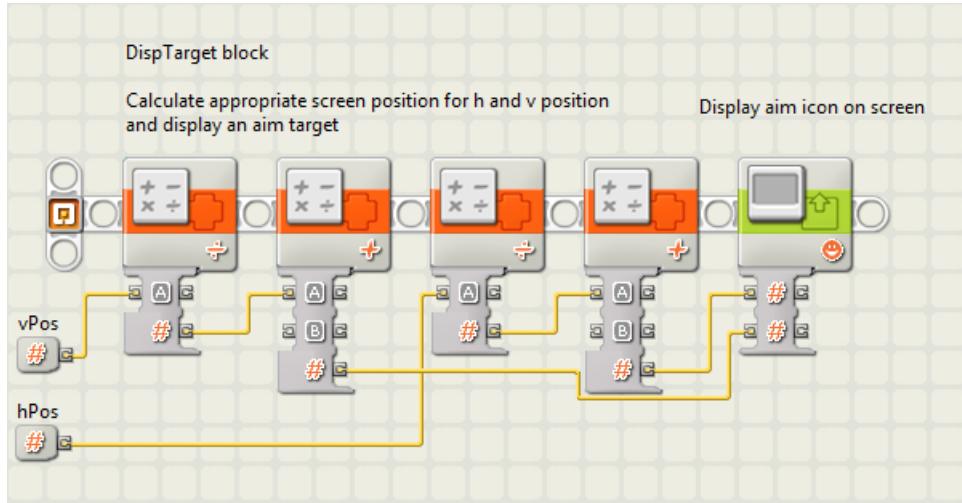


Limit H/V Position.

- Display Target

As in any system teleoperated is required to have a feedback that shows how the system is in every moment. In the case of this block which has the function to display on the screen of the NXT brick which the position where is the end of

the robotic arm to control at any moment which is within the limits set in which the robot can move. It is a graphical and easy way to make a feedback for the operator who can see the position captured by the system besides to have the option of viewing directly the result in their own robotic arm.

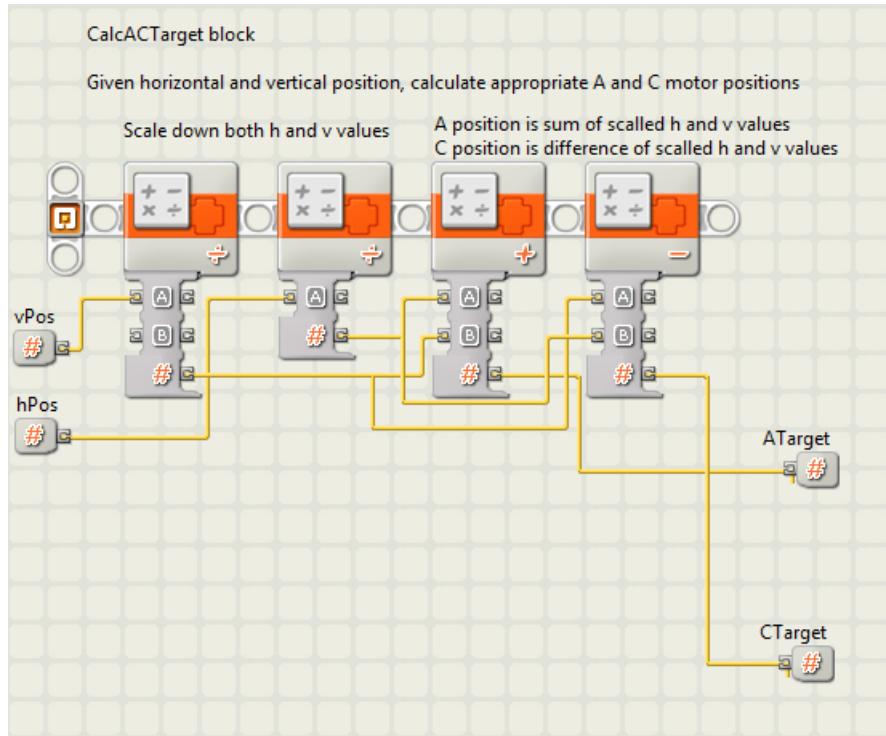


Display Target.

- Calc AC Target

This block calculates the relative positions that will take the robot, these positions will be transformed into movements for each from motors. Once captured the

horizontal and vertical positions where the sensor is, is performed a conversion of values and calculate the position of the motor, being the position of the first motor the sum of the scaled values h and v, and the other motor the difference between these values.



Calc AC Target.

VI. CONCLUSIONS

This section discussed the conclusions obtained from this project.

One of the first conclusions that can be obtained is a teleoperated system is not easy at all, it is necessary that the system has a lot of precision in control and it comes great complexity. However, there are several methods for

control, from a PID to intelligent systems using neural networks and evolutive algorithms.

Of course, there are a number of future improvements which could be used to improve the system. Since adding artificial vision to recognize where to find the object and calculate the relative position , or add more motors to have more degrees of freedom and the movement of the robotic arm covers a larger space, or include a motorized system for not just a robotic arm, but a moving vehicle with the robotic arm. There are many possible improvements that would make the system increases its complexity and functionality.

References

- [1] Ro-botica. Educative and personal robotic shop. Last visit: Dec-2011.
Link: <http://goo.gl/8Yv0H>

- [2] HiTechnic. Last visit: Dec-2011.
Link: <http://goo.gl/62Vv6>
- [3] G. Sen Gupta, S.C. Mukhopadhyay, C. H. Messom and S. Demidenko. "Master-Slave Control of a Teleoperated Anthropomorphic Robotic Arm with Gripping Force Sensing". Last visit: Jan-2012.
Link: <http://goo.gl/4ZtDe>
- [4] Lesson 3. Teleoperation and Telepresence in Robotic subject.
- [5] Lesson 4. Teleoperation and Telepresence in Robotic subject.
- [6] Vertut, Jean. Ed. Hermes. "Teleoperation and robotics. Evolution and development". 1985.
- [7] Marc C. Bequet. Ed. Kluwer Academic. "Teleoperation, numerical simulation and experimental validation". 1992.



This work by Raúl Pérrula Martínez is under a license [Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported](#).