

---

# Generation of Baxter Joint position value from human arm coordinate

```
%rosinit('169.254.9.127') % ip address of baxter

%   ROSTOPIC INFO
% rostopic info /robot/limb/left/joint_command
% rostopic info/robot/joint_states
% rostopic info/Robot_1/pose
% rostopic info/Robot_2/pose
% rostopic info/Robot_3/pose
% rostopic info/Robot_4/pose

% SUBSCRIBER MOCAP AND BAXTER
%we take data from the topic of joint state position for error
  checking and
%the four topic of motion capture, each one for any rigid body:
  shoulder
%elbow,wrist and hand.
sub = rossubscriber('/robot/joint_states');
sub1 = rossubscriber('/Robot_1/pose');
sub2 = rossubscriber('/Robot_2/pose');
sub3 = rossubscriber('/Robot_3/pose');
sub4 = rossubscriber('/Robot_4/pose');

% PUBLISHER BAXTER COMMAND
%We publish command data on joint_command topic for control robotic
  arm
%given a real time series of position
pub = rospublisher('/robot/limb/left/joint_command');

% MESSAGES

msg = rosmessage('baxter_core_msgs/JointCommand'); %publish joint arm
msg1 = rosmessage('sensor_msgs/JointState'); %sub error checking
msg2 = rosmessage('geometry_msgs/Pose'); %shouder
msg3 = rosmessage('geometry_msgs/Pose');%elbow
msg4 = rosmessage('geometry_msgs/Pose');%wrist
msg5 = rosmessage('geometry_msgs/Pose'); %hand

%setting control position Baxter Arm on position
msg.Mode = 1;
```

---

# STARTING WHILE LOOP FOR ON-LINE ROBOT TELEOPERATION

```
while 1

    %message received from Mocap topic and Joint states Topic
    msg1= receive(sub);
    msg2= receive(sub1);
    msg3= receive(sub2);
    msg4= receive(sub3);
    msg5= receive(sub4);

    % HUMAN ARM COORDINATE X Y Z
    shoulder_coo =msg2.Pose.Position;
    elbow_coo = msg3.Pose.Position;
    wrist_coo = msg4.Pose.Position;
    hand_coo = msg5.Pose.Position;

    % TAKE SHOULDER AS REFERRING FRAME
    shoulder_coo_X =shoulder_coo.X;
    shoulder_coo_Y= shoulder_coo.Y;
    shoulder_coo_Z= shoulder_coo.Z;

    elbow_coo_X =elbow_coo.X -shoulder_coo_X;
    elbow_coo_Y= elbow_coo.Y -shoulder_coo_Y;
    elbow_coo_Z= elbow_coo.Z - shoulder_coo_Z;

    wrist_coo_X =wrist_coo.X-shoulder_coo_X;
    wrist_coo_Y= wrist_coo.Y - shoulder_coo_Y;
    wrist_coo_Z= wrist_coo.Z - shoulder_coo_Z;

    hand_coo_X =hand_coo.X-shoulder_coo_X;
    hand_coo_Y= hand_coo.Y - shoulder_coo_Y;
    hand_coo_Z= hand_coo.Z - shoulder_coo_Z;

    % CREATE ARRAY OF HUMAN COORDINATE
    Human_coo
    =[shoulder_coo_X ;shoulder_coo_Y;shoulder_coo_Z;elbow_coo_X;elbow_coo_Y;elbow_coo_Z;wrist_coo_X;wrist_coo_Y;wrist_coo_Z;hand_coo_X;hand_coo_Y;hand_coo_Z];

    % NEURAL NETWORK
    % This is a neural network that we have create and trained from matlab
    %script. It is used in order to obtain an array of baxter joint
    % desired posture as an
    % output given human coordinate took on line by Mocap node.

    Joint_position = myNeuralNetworkFunction(Human_coo)
    Joint_position = Joint_position';
```

---

```

% PUBLISHER POSITION CONTROL MSG
%the output of neural network is used as position control of Baxter's
  arm.

msg.POSITIONMODE;
msg.Command=[Joint_position(1,1),Joint_position(1,2),Joint_position(1,3),Joint_pos
msg.Names={'left_e0','left_e1','left_s0','left_s1','left_w0','left_w1','left_w2'};

%send message to Baxter
send(pub,msg)

%ERROR CHECKING
%when the position of the joints thate we control given by Joint_State
  Topic
%is similar we check the error
flag = true;
while flag
baxter_joint_states = msg1.Position;
baxter_joint_states = baxter_joint_states';

[p,v]=size(baxter_joint_states);

if v == 17
position_joint = baxter_joint_states(1,3:9);
error_check = abs(position_joint - Joint_position)
  if norm(error_check) < 1.06
    display('position reached correctly')
    flag = false;
  end
end

end

% loop rate
  pause(1);

end

% rosshutdown

```

*Published with MATLAB® R2017b*