Design and Manufacturing of a Real Time Imitation Based Robotic Arm using Low Cost Microcontroller
AHSAN QAMAR, JAVAID IQBAL, IBRAHIM JAVED, ISHAQUE, HARIS REHAN
Department of Mechatronics Engineering
National University of Sciences & Technology
Peshawar road, Rawalpindi
PAKISTAN

Abstract: - The ultimate goal of research in robotics is to make robots able to replace humans for a certain set of tasks. Imitation is a powerful means to interact with robots, as it is imitation through which humans can transfer their skills to robots in a way that is quite natural to us. The proposed methodology provides a scheme for robot control programming by real time imitation of human demonstrator. Emphasis is placed on the end-to-end completeness and generality of the methodology. This paper addresses various aspects of control programming with reference to imitation based robotics, starting from problem selection to complete working robotic system.

Key-Words: - imitation, real-time, robotic arm, microcontroller, kinematics, biometrics, DOF, I2C

1 INTRODUCTION
As a capability of learning by observation, imitation can improve and speed up the learning of sensory-motor associations [1]. Using imitation based techniques; robots can learn more sophisticated actions with relatively greater precision and speed. Imitation has a communicative function [2], which simplifies interaction between humans and robots. The theoretical basis influencing overall perception about imitation’s contribution to development in robotics has been completely altered mainly by virtue of developmental studies carried out since the 1970s. Imitation can equip robots with abilities to perform efficient human-robot interaction, eventually helping humans in personal tasks [3]. It also seems that imitation could be a tool to acquire new behaviors and to adapt these within new contexts [4].

We have confined our work to a specific area in imitation based robotic control i.e. imitation based control of a robotic arm using human arm as a demonstrator in real time. The robotic arm constructed for this purpose is a two DOF structure that can imitate the human arm by capturing the trajectory defined by the shoulder and elbow joints in a planar workspace. A pair of Permanent Magnet DC (PMDC) geared motors with shaft mount encoders have been used as joint actuators and suitable control [5, 6] has been applied on them. The core of entire control system is a low cost RISC based Atmel Mega8535 microcontroller [7].

To quantify the motion demonstrated by the human arm, various vision-based and sensors based techniques have been developed by the “Biometrics” professionals, a field dedicated for measurement and quantification of body movements. We tend to go for a vision-based approach due to its cost-affectivity at a fairly justifiable degree of accuracy, precision, repeatability and robustness. A camera is used to build the trajectory information of the human arm by using markers as joint identifiers.

Since, the imitator’s physical possibilities do not match with that of the demonstrator, there has to be a way to correlate the two. For this purpose, we need to take a look at how humans can perform these actions by observing other humans. To achieve those actions humans use the information derived from two sources:

• The demonstrator’s body schematic: It contains information about position and velocity data of human arm at any instant.
• The imitator’s body schematic: It represents a person’s own body position at any instant of time. It also contains information about spatial relationship between various body parts and how they can be used to perform a certain task.

2 Platform
A two degree of freedom (DOF) robotic arm has been manufactured using aluminum due to ease of fabrication. Direct coupling is used for the shoulder and indirect chain-sprocket drive has been used for the elbow joint. Two PMDC motors have been used as joint actuators. Both motors have operating voltage of 12V and no load speed of 20 rpm for the output shaft. The gear ratio between motor shaft and the output shaft is 95.5:1. Figure-1 shows the fabricated mechanical structure.

2.1 The Controller and Motor Drive Modules
An “AVR ATmega8535” microcontroller serves as the master controller in the system. ATmega8535 has built in I²C functions that have interrupt-based operation. It enables the microcontroller to do other tasks while it communicates with the motor drive modules on I²C protocol. The I²C mode has been used because it is fast, requires only two I/O lines to interface up to 8 such modules with the microcontroller and the only external hardware required are 2 pull-up resistors.

3 Methodology
The anatomy of demonstrator (the human arm) is in sharp contrast to that of the imitator (the robotic arm). Therefore, we established joint-to-joint correspondence between the demonstrator and the imitator. The General layout of the system is shown in Figure-2. The mechanical structure that we used has 2 degree of freedom (DOF).
It has two joints representing the shoulder and the elbow joints of human arm. Our data flow is broken into two streams being processed simultaneously: one that collects motion information from the human arm and the other that collects information from the robotic arm. We need interface and control software that integrates the two streams, establishes fast as well as accurate data transfer between them and provides an efficient control of the robotic arm. These data streams are linked together through a wireless connection.

4 Algorithm
We used digital image processing for tracking the human arm currently on a fixed black background. Three white circles are used for identifying the shoulder, elbow and wrist joints. The distinction between the three joints is made on the basis of relative sizes of circles representing the joints. The circles are identified by thresholding the color values for the markers [8]. The locations of all joints are calculated in terms of pixel coordinates of their center points.

![Diagram of the Vision System](image)

**Figure 3 – The Vision System**
The ActiveX control “video.ocx” is used to capture frames from the live video of human arm and import them into the .NET environment. Every frame is treated as a single 320x240 bitmap image and is processed separately. After that the circles are arranged in descending order with respect to their diameters thereby, differentiating the shoulder, elbow and wrist joints. The joint angles are then finally computed using the trigonometric and kinematics principles as shown in Figure-3. Let (x1, y1), (x2, y2) and (x3, y3) represent the pixel coordinates of shoulder, elbow and wrist joints respectively. Equations (1), (2) and (3) are used to compute the joint angles.

\[ \theta_1 = \tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \]  
\[ \theta_2 = \tan^{-1}\left(\frac{y_3 - y_2}{x_3 - x_2}\right) \]  
\[ \theta_3 = |\theta_1 - \theta_2| \]  

Where \( \theta_1 \) and \( \theta_3 \) represent the shoulder and elbow joint angles for the robotic arm.

### 4.1 Catering for the relative movement in elbow joint

The indirect drive for the elbow joint induces a relative motion in elbow joint with respect to the shoulder joint. Considering Figure 4, (Xe, Ye) represents the reference axis for elbow joint in the software that remains fixed and (Xe/s, Ye/s) represents the reference axis for elbow joint on the actual mechanical structure that keeps on shifting relative to the shoulder movement. At the home position (figure 4-a), the two axes are aligned so no offset is required for elbow joint angle. In that case, a destination angle calculated by software as \( \theta_3 \) is sent to the controller without adding any offset. If shoulder moves through an angle \( \theta_1 \), it also causes the elbow joint to move by same amount causing a shift of \( \theta_1 \) degrees in the physical axis (Xe/s, Ye/s) of elbow joint (figure 4-b). For elbow joint adjustment, the same destination elbow joint angle \( \theta_3 \) is sent to the controller after adding an offset of \( \theta_1 \) degrees in it. Consequently, the shoulder joint angle is constantly added to the elbow joint angle in each frame to cater for the shift in reference position of elbow joint. After all the angles are computed, they are converted into a form readable by the control device for the robot.

![Diagram](image)

**Figure 4 – Movement in Elbow joint relative to Shoulder joint**

### 4.2 Control topology

Both motors used have quadrature encoders mounted on their rear shafts. Each encoder has 2-channel output. The phase shift between the output pulses of a particular encoder determines the direction of rotation. The following equation is used for conversion of joint angles determined by the algorithm into encoder steps:

\[ r(t) = \theta_d * \left[ \left( E_c \times gr \right) / 360 \right] \]

Where,

- \( r(t) \) = desired position in terms of total encoder counts
- \( \theta_d \) = destination angle
- \( E_c \) = encoder counts in one revolution of encoder shaft
- \( gr \) = gear ratio between rear and output shaft of motor

Using the above information, appropriate P control is applied to the robotic arm using the following equation:
Where, $Y_p(t) =$ output of the proportional controller  
$K_p =$ proportional gain  
$y(t) =$ actual position of the robotic arm  
$[r(t) - y(t)] =$ error between actual and desired position of robotic arm

\[ Y_p(t) = K_p \cdot [r(t) - y(t)] \quad (5) \]

Figure 5 – Behavior of P-control with respect to destination angle

The P-control applied selects a suitable speed for each motor according to the rate at which the location of demonstrator changes between two consecutive frames. It selects a maximum duty ratio of 1 for the motors at a destination angle of 32 degrees or more and reduces it proportionally for any destination less than that. In that way, the robotic arm tracks the demonstrator in terms of its position as well as speed.

5 Experimental results

To assess the working of our system, we tested the behavior of robotic arm when imitating complex trajectories followed by the human arm in real time. The algorithm also has the capability to draw the shape followed by end-effector of the human arm. This shape is drawn on a 320 x 240 sized bitmap panel, which can be saved and can be used later on for analysis purposes. The results obtained are as shown in the Figure-6. Figure-7 helps to compare the movements of demonstrator and imitator. These results however, can be much enhanced by increasing the degree of freedom of the robotic arm and the number of human arm joints used to build the trajectory.

Figure 6 – (a) & (b) are two paths traced by the human arm while (c) & (d) respectively show how the robot imitated these paths

Figure 7 – (a) The path traced by human arm (b) The same path imitated by the robot. Both paths have been scaled down to equal size for comparison. Each grid box in both figures measures 20 x 20 pixels

6 Conclusion

The control of robots based upon imitation is an emerging and yet developing field. Its key applications include imitation based learning and providing efficient control of a robotic arm placed in a remote area virtually inaccessible by humans. We accomplished the imitation task by tracking the human arm in real time. We also propose the same approach for implementing imitation based learning in our system. This trajectory information can be saved and later on used by the robot to repeat the same task.
References:


