

Object Sorting Robotic Arm using openCV

Yayati Dandekar¹, Shubham Gawand², Deshmukh Kushal³
& Akshay Adepu⁴

^{1,2,3,4} Student, Department of Electronics & Telecommunication Engineering, KJSIEIT, Mumbai, India.

Abstract: Robotic Arms play an integral part in today's manufacturing and logistics operations to provide more speed and efficiency and reduce the possibility of errors. The objective of this project is to build a robotic arm with an emphasis on automating the assembly by object and color detection. The basic arm is made up of light materials and will have 4 DOF to provide maximum mobility. The arm is controlled by Atmega328p microcontroller with the aid of Arduino Development Environment for hassle free interfacing of sensors. Webcam is used for sensing the surroundings. Feed from the webcam is converted into matrix data by openCV libraries and is further processed in Visual Studio. The real time coordinates data and color information will be fed to MCU (Microcontroller Unit). MCU will control the arm servos accordingly to grip the appropriate object and will place it in a predetermined location. The use of easily available component reduces the cost but at the same time leaves much room for future improvements.

1. Introduction

In the constant evolving industrial environment, importance of machine automation is undisputable. Nowadays various industrial processes are being automated by using robotic arms to increase the productivity and efficiency of the manufacturing and logistics operations. Any robotic system will mainly consist of a sensory system which will sense the surroundings i.e. objects, and obstacles, a microcontroller which will process the signal and a mechanical assembly that will perform operation according instructions given by the microcontroller.

The proposed system will make use of a microcontroller, 5 servo motors, a gripper mechanism, an ultrasonic sensor and openCV for color sensing of the object and a mechanical arm assembly to support this components. The project is fully implemented to provide a complete prototype. The assembly is fully custom made and hence much cost effective. The arm will have 4 DOF and will be able to move freely with minimal human intervention.

2. Literature Survey

The development of robotic arm for industrial manufacturing and operations begun in 1960s with early generation of robotic arms bridging the gap between manual labor and machines. Best example of robotic arms used in industry today are 'SCARA' arm. SCARAs are generally faster and cleaner than comparable Cartesian robot systems. Their single pedestal mount requires a small footprint and provides an easy, unhindered form of mounting. On the other hand, SCARAs can be more expensive than comparable Cartesian systems. [1] Literature survey for this project can be divided in four categories as follows

A. Mechanical Assembly

Mechanical Assembly is the main part of any mechanical arm as it defines the arm characteristics such as mobility, load-bearing and efficiency. Commercial industry grade arm can use strong and light materials such as metal alloys, carbon fibre composite materials and sometimes plastic polymers. Experimental and prototype design use mostly wood, acrylic and light metals. Metals such as aluminium can be used in the form of brackets to give a lightweight yet sturdy arm mechanism. [2]

B. Joints

Servo motors are used extensively in building a robotic arm joint as they are sturdy and can be interfaced with almost any microcontroller. Motors at the base of the assembly require high torque and stability. Motor at the gripper requires least torque as it is located at the top of the assembly. The number of motors used will increase with increase in no. of joints and considerably with increase in no. of DOF. [3][4]

C. Sensors

Sensors are used in arms to automate the entire assembly to the extent that it requires minimal human intervention. Experimental designs mostly use ultrasonic, color and pressure sensors. [5]

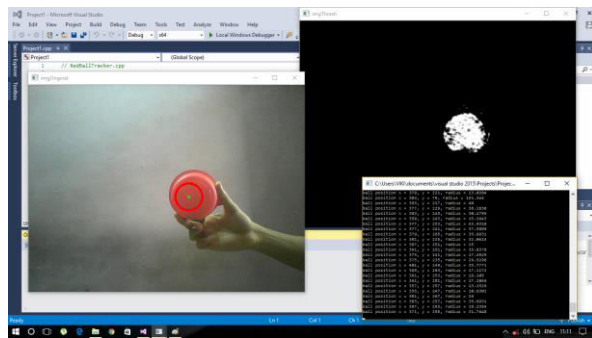


Fig. 1. Object Sensing & Recognition

D. Microcontroller

Experimental setups generally tend to use user-friendly and inexpensive systems such as arduino or Raspberry Pi. Arduino provides a good development environment with arduino IDE. [6]

E. Object Sensing and Recognition

Various software platforms are available for image processing required for object sensing and recognition. MATLAB and openCV are most widely used platforms for object sensing. openCV developed by Intel has an advantage that it is open source platform without any proprietary constraints. [7]

3. Proposed Model

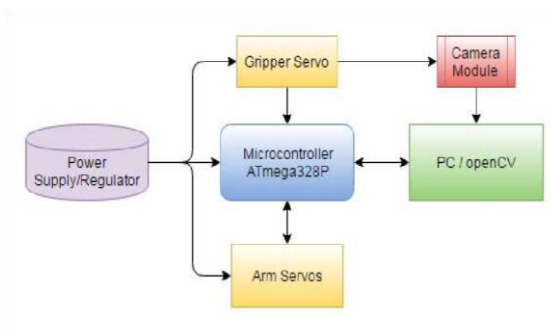


Fig. 2. Block Diagram of Proposed System

The basic design the whole system can be summarized as the given block diagram.

ATmega328p is at the heart of the system and controls the entire assembly. MCU (Microcontroller Unit) is the central processing unit, which controls all the functions of other blocks in this system. It controls all the functions of the whole system by manipulating these data. MCU provides signal to servos of both arm and griper which driven by external power supply. Arm move towards the desire object and pick that object (color box) and place it to desire location according to the color of the object. Its output is processed by the software environment which consist of openCV libraries for feed capture and visual studio for post processing of the feed.

OpenCV has inbuilt libraries for object detection base on size, shape and color. It takes the real time

image and converts it into the matrix of data and process the information using visual studio.

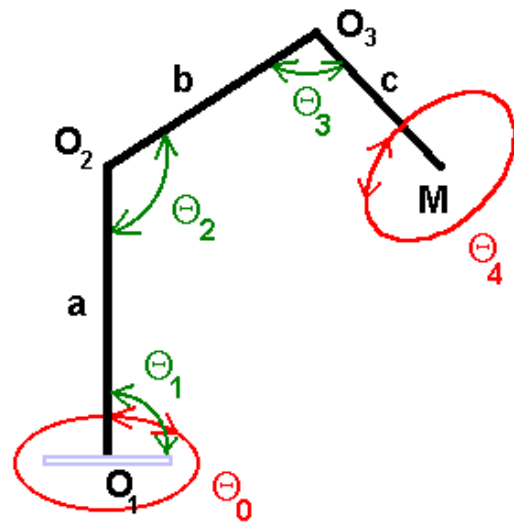


Fig. 3. Robotic Arm Basic Layout

4. Implementation

First arm is set in its default position i.e. 0 Deg. Arm will rotate between 0 to 180 deg. For object detection. With camera attached to the gripper of the arm. Depending upon the priority set by the user, arm will detect a particular object and determine its position by coordinates provided by processing using OpenCV. Microcontroller i.e.; Atmega328p will be provided by real time coordinates data by serial monitoring.

Algorithm used for Object Detection and Recognition is as follows,

1. Capture the video from web cam.
2. If not success, exit program.
3. Create a window called "Control".
4. Create track bars in "Control" window, Hue (0 - 179), Saturation (0 - 255), Value (0 - 255) for tracker.
5. Read a new frame from video.
6. If not success, break loop.
7. Convert the captured frame from BGR to HSV.
8. Morphological opening (remove small objects from the foreground).
9. Morphological closing (fill small holes in the foreground).
10. Show the thresholded image.
11. Show the original image.
12. Find contours in mask and initialize the current (x, y) center of the ball.
13. Only proceed if at least one contour was found.

14. Find the largest contour in the mask, then use it to compute the minimum enclosing circle and centroid.
15. Only proceed if the radius meets a minimum size.
16. Draw the circle and centroid on the frame, then update the list of tracked points, update the points queue.
17. Send the list of tracked points to the Microcontroller in the form of coordinate data.

5. Expected Results

The arm should be able to sense the predefined colored object from the given set of objects. Gripper should be able to grip the object by using the coordinates. The arm should be able to place the object on the particular box according to its color.

6. References

- [1] <http://mecademic.com/DexTAR.html>
- [2] <http://www.robotoid.com/howto/materials-for-robotbuilding-an-introduction.html>
- [3] http://www.societyofrobots.com/robot_arm_tutorial.shtml
- [4] <http://robotics.stackexchange.com/questions/869/building-robotic-arm-joint>
- [5] <http://blog.robotiq.com/bid/72633/7-Types-of-Industrial-Robot-Sensors>
- [6] http://www.atmel.com/images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet_Complete.pdf
- [7] <http://opencv-srf.blogspot.in/2010/09/object-detection-using-color-separation.html>