Chapter 12

Vision-Based Manipulator

On successful completion of this course, students will be able to:

- Explain how the manipulator works.
- Describe types of manipulator.
- Develop a vision-based manipulator.

Introduction

One of the most common of manipulation tasks is grasping an object. Tasks performed by humans involve some form of grasping action. In the absence of feedback, the grasping action cannot be completed effectively. A human being grasps an object almost invariably with the aid of vision. We use visual information to identify and locate the object, and then decide how to grasp them.

Inverse Kinematics

In a two-joint arm robot, given the angles of the joints, the kinematics equations give the location of the tip of the arm. Inverse kinematics refers to the reverse process. Given a desired location for the tip of the robotic arm, what should the angles of the joints be so as to locate the tip of the arm at the desired location? There is usually more than one solution and can at times be a difficult problem to solve. In a 2-dimensional input space, with a two-joint robotic arm and given the desired co-ordinate, the problem reduces to finding the two angles involved. The first angle is between the first arm and the ground (or whatever it is attached to). The second angle is between the first arm and the second arm.

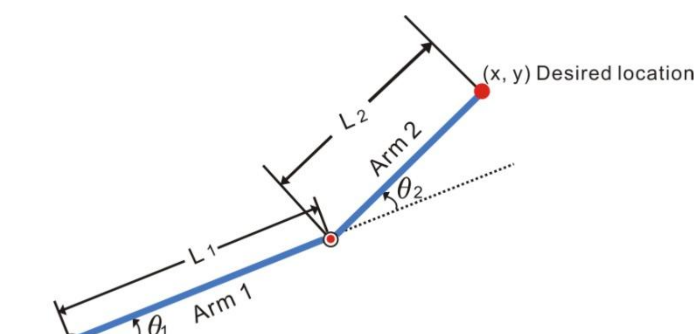

Figure 12.1 *inverse kinematics for 2 DOF arm robot[12].*

For simple structures like the two-joint robotic arm, it is possible to mathematically deduce the angles at the joints given the desired location of the tip of the arm. However with more complex structures (eg: n-joint robotic arms operating in a 3-dimensional input space) deducing a mathematical solution for the inverse kinematics may prove challenging. Using fuzzy logic, we can construct a Fuzzy Inference System that deduces the inverse kinematics if the forward kinematics of the problem is known, hence sidestepping the need to develop an analytical solution.

Vision-Based Manipulator

Most work in robotic manipulation assumes a known 3-D model of the object and the environment, and focuses on designing control and planning methods to achieve a successful and stable grasp in simulation environments. Grasping is usually preceded by a number of tasks that effect the final grasping action. The sequence of steps involved is:

1) The movement of the end-effector from a given position to within a reaching position from the object.
2) The estimation of grasp points and orientation of the end-effector to perform the grasp operation.
3) The grasping action, once the end effector is in the appropriate position.

Based on the previous literature (visual-servoing) is huge and largely unorganized. A variety of methods have been proposed to solve vision-based manipulation [1-5]. They use vision to aid just one of the above mentioned steps. In the past, most approaches to robotic grasping [6] [7] assume availability of a complete 3-D model of the object to be grasped. In practice, however, such a model is often not available—the 3D models obtained from a stereo system are often noisy with many points missing, and 3-D models obtained from a laser system are very sparse. This makes grasping a hard problem in practice. In more general grasping, Kamon et al. [8] used Q-learning to control the arm to reach towards a spherical object to grasp it using a parallel plate gripper.

For grasping 2D planar objects, most prior work focuses on finding the location of the fingers given the object contour, which one can find quite reliably for uniformly colored planar objects lying on a uniformly colored table top. Using local visual features (based on the 2-d contour) and other properties such as force and form closure, the methods discussed below decide the 2D location at which to place the fingertips (two or three) to grasp the object.

Edsinger and Kemp [9] grasped cylindrical objects using a power grasp by using visual servoing and do not apply to grasping for general shapes. An inverse kinematic solver is proposed in [10] to find all joint angles for given position of the effectors on the manipulator. The target object is recognized by color segmentation. The 3D position is computed by the stereo vision system after contour extraction.

Inverse kinematics of manipulator and object location are the key technology for arm robot. We study various visual feedback methods from previous literature and develop a new model for grasping a bottle. We know that Extraction of image information and control of a robot are two separate tasks where at first image processing is performed followed by the generation of a control sequence. A typical example is to recognize the object to be manipulated by matching image features to a model of the object and compute its pose relative to the camera (robot) coordinate system. In general method, first the target object is recognized by the vision system which than estimates the 3D pose of the object. Based on this information, the controller coordinates to move the arm robot to grasp the object/bottle. The framework proposed in this experiment shown in fig. 12.2 below, where the stereo camera for pose estimation attached about 50cm at the side of manipulator.

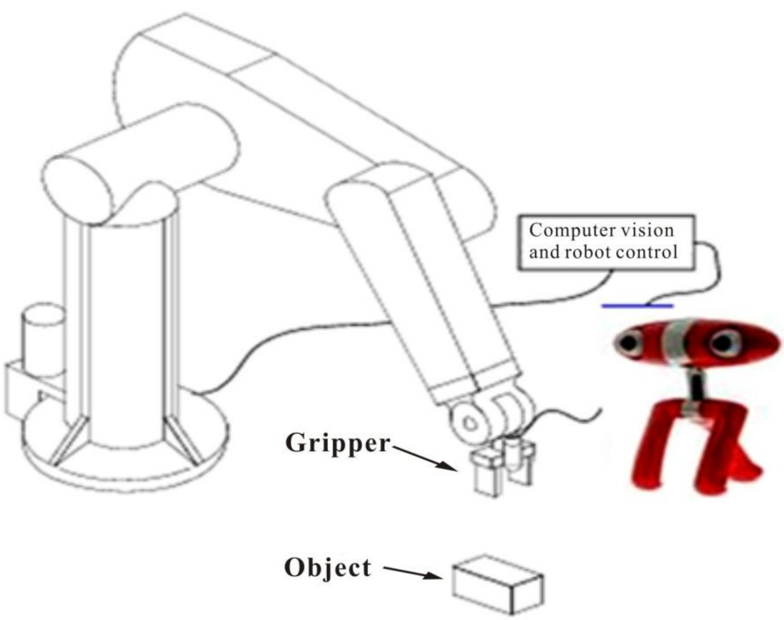

Figure 12.2 *example of vision-based grasping using stereo vision.*

We developed a framework of vision-based arm robot using 4 DOF (Degree of Freedom) arm robot from Lynxmotion that able to delivers fast, accurate, and repeatable movement. The robot features: base rotation, single plane shoulder, elbow, wrist motion, a functional gripper, and optional wrist rotate as shown in figure 12.3. This robotic arm is an affordable system with a time tested rock solid design that will last and last.

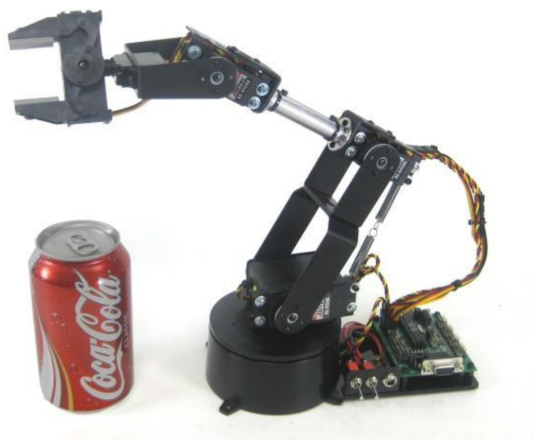

Figure 12.3 *4 DOF arm robot using stereo vision used in the experiment.*

The specification of this manipulator:

```
Base: Height = 6.9 cm
Hand/Grip: Max Length (No Wrist Rotate) = 8.7 cm
Hand/Grip: Max Length (With LW Rotate) = 11.3 cm
Hand/Grip: Max Length (With HD Rotate) = 10.0 cm
Length: Forearm = 12.7 cm
Length: Arm = 11.9 cm
```

Grasping Model

Grasp determination is probably one of the most important questions from manipulation point of view. Usually, the object is of unknown shape. In our model, we propose a simple way, if we know the distance of the bottle then move the arm to that position, then when the center point of a bottle exactly meet the center point of the gripper, then it means that is the time for grasping the bottle/object as shown in figure 12.4:

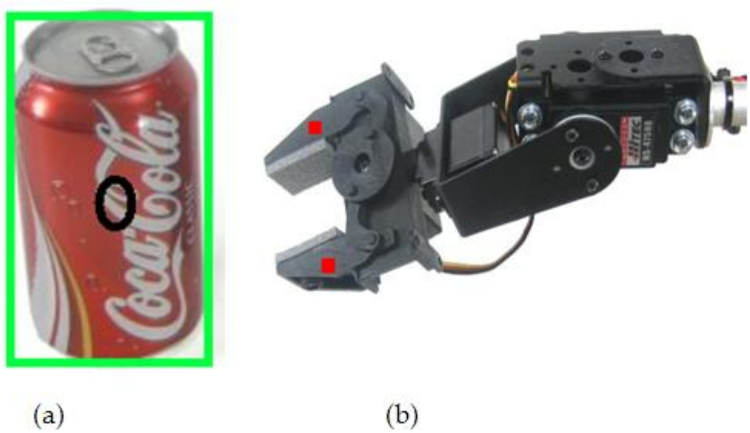

Figure 12.4 *Finding the center of the bottle using vision (a) and color marking for indicating the position of gripper with an object [11].*

The proposed algorithm for detect an object/bottle, grasp it and move to the destination is shown below:

```
do
detect the object/bottle
if object/bottle detected then
      begin
  find position of the object/bottle
  move  arm robot to the object/bottle
  move the gripper to the center of the object/bottle
  if the position equal
  grasp the object/bottle
  else
  move to gripper to the center of the object/bottle
     end
endif
move the object to destination
go to the initial position
loop
```

An environment consists of a variety of objects, such as the robot itself, walls, floor, tables, objects to be grasped, etc. In order to successfully move the arm without hitting an obstacle, we provide 1 distance sensor at the gripper. To determine the state of the object, the "good" grasp position should be first determined. The experiment conducted at our lab to grasp and move a bottle to the destination. For testing the connection of hardware, we use Lynx SSC-32 as shown below:

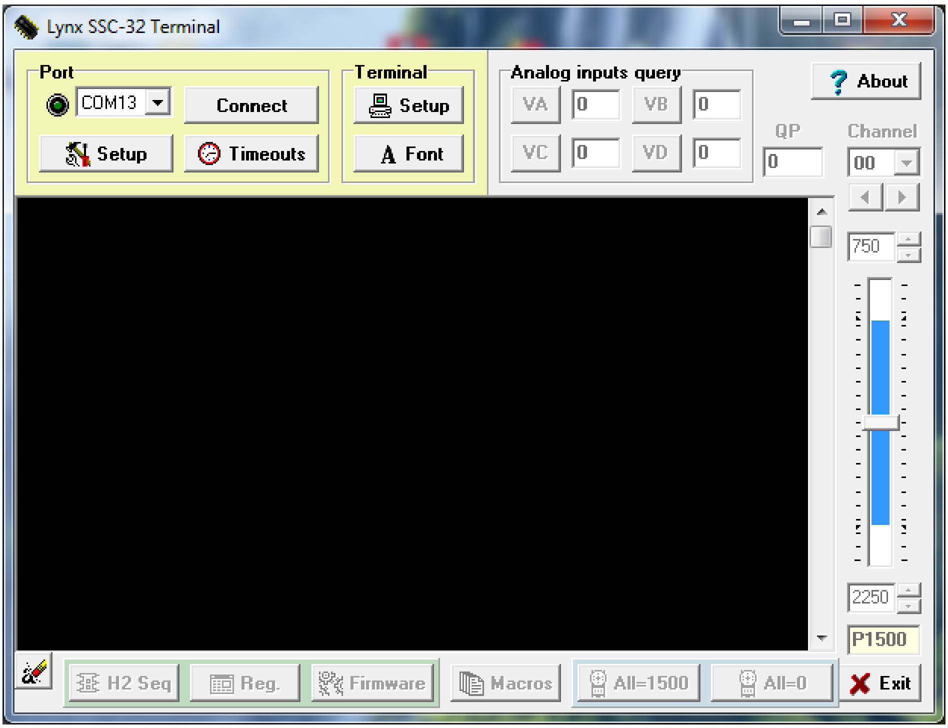

Figure 12.5 *using Lynx SSC-32 Terminal for testing the hardware [13].*

We use Lynxmotion RIOS (Robotic Arm Interactive Operating System) SSC-32 software to configure and control the arm robot as shown in figure 12.6:

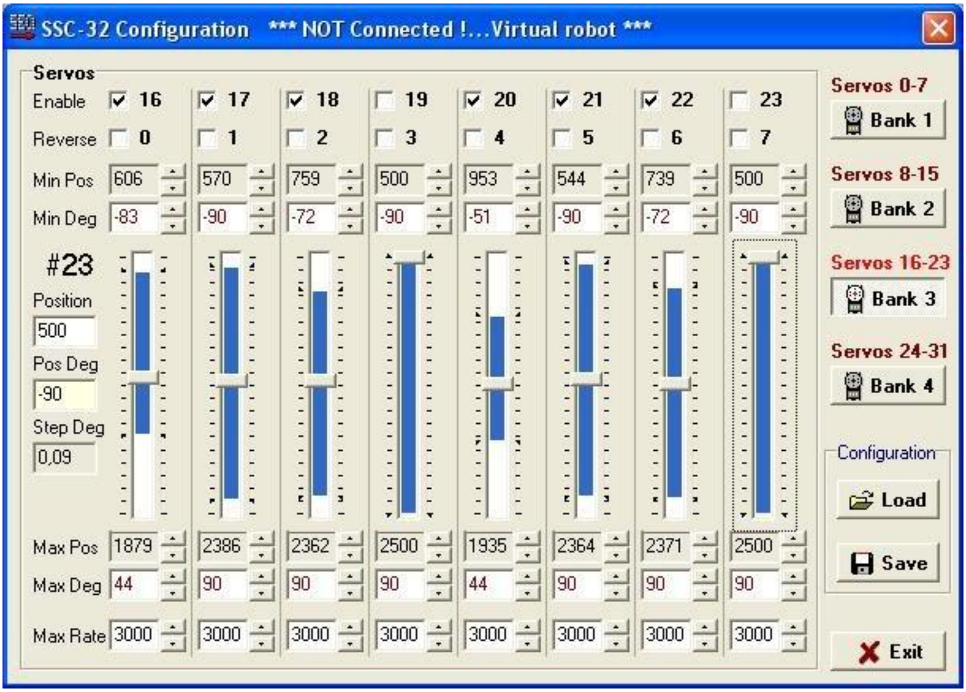

Figure 12.6 *Configuring and control the board of arm robot using RIOS SSC-32 Software[13].*

After configuring and calibrating the servos of arm robot. We put an object/bottle in front of the arm robot to be grasped and move to other position. Based on our experiment we get the expected result as shown in table 12.1.

Table 12.1 *Result of Detecting and Grasping an Object in 10x.*

No	Action	Success	failure
1	Identify the object as bottle	100%	0%
2	Grasping an object correctly	90%	10%
3	Estimate the distance of the bottle	90%	10%

The accuracy and robustness of the system and the algorithm were tested and proven to be effective in real scenarios.

Program for object detection and grasping successfully developed with OpenCV and the manipulator able to grasp it as shown in figure 12.7

Figure 12.7 *Object detection and grasping OpenCV.*

Exercise

1) Create a simulator program for inverse kinematics using function as shown below:

```
data = Convert.ToInt32(txtInput.Text);
    MessageBox.Show("value from sin " + data + " "
+Convert.ToString(Math.Sin(Radians2Degrees(data))));
    MessageBox.Show("value from cos " +data + " "
+Convert.ToString(Math.Cos(Radians2Degrees(data))));
    MessageBox.Show("value from tan " + data + " "+
Convert.ToString(Math.Tan(Radians2Degrees(data))));
    MessageBox.Show("value from Theta(Asin) : " +
Convert.ToString((Math.Asin(Math.Sin(Radians2Degrees(data))))*(18
0/Math.PI)));
    MessageBox.Show("value from Theta(Acos) : " +
Convert.ToString((Math.Acos(0.866))*(180/Math.PI) ));
```

```
    MessageBox.Show("value from Theta(Atan) : " +
Convert.ToString((Math.Atan(Math.Tan(Radians2Degrees(data)))) *
(180 / Math.PI)));
    MessageBox.Show("value from  Theta2(arccos) : " +
Convert.ToString((Math.Acos(Math.Cos(Radians2Degrees( (((x*x)+(y*
y))-(l1*l1)-(l2*l2))/(2*l1*l2) )))) * (180 / Math.PI)));
```

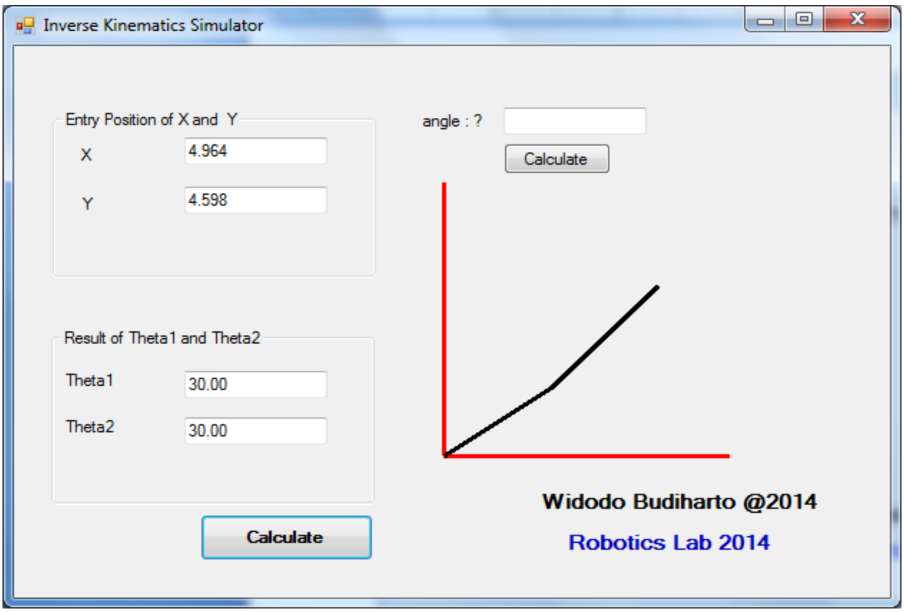

Figure 12.7 *Inverse Kinematics Simulator.*

2) Create a program to grasp an object using arm robot and stereo vision.

References

[1] A. Bendiksen and G. D. Hager. "A vision-based grasping system for unfamiliar planar objects". In ICRA, pages2844–2849, 1994.

[2] H. I. Christensen and P. L. Corke. "Visual servoing". I. J.Robotic Res., 22(10-11):779–780,2003.

[3] D. Kragic and H. I. Christensen. "A framework for visual servoing". In ICVS, pages 345–354, 2003

[4] A. Bicchi and V. Kumar, "Robotic grasping and contact: a review," in ICRA, 2000.

[5] J. Jiao, Embedded Vision-Based Autonomous Move-to-Grasp Approach for a Mobile Manipulator,International Journal of Advanced Robotics System, vol. 9, pp.1-6, 2012.

[6] K. Shimoga, “Robot grasp synthesis: a survey,” IJRR, vol. 15, pp. 230–266, 1996.

[7] D. Purwanto. Visual Feedback Control in Multi-Degrees-of-Freedom Motion System. PhD thesis at Graduate School of Science and Technology - Keio University, Japan, 2001.

[8] I., Kamon, T. Flash, and S. Edelman, “Learning to grasp using visual information,” in ICRA, 1996.

[9] A. Edsinger and C. C. Kemp, “Manipulation in human environments,” in IEEE/RAS Int’l Conf on Humanoid Robotics (Humanoids06), 2006.

[10] Y. Yang, “Binocular Stereo Vision based Humanoid Manipulator Control”, 30th International Conference on Control, pp. 3996 - 4001, China. 2011.

[11] Widodo Budiharto, Bayu Kanigoro and Anita Rahayu, Vision-Based Grasping for Manipulator using Object Features, Annual Conference on Engineering and Information Technology, pp. 228-235, Japan, March 28-30th, 2014.

[12] Matworks.com.

[13] www.lynxmotion.com.