Controlling Robots
Using Vision and Bio Sensors
Vacation Research Experience Scheme

Name: Hanlin Liang
Email: h2.liang@student.qut.edu.au
Supervisor: Juxi Leitner & Sareh Shirazi & Ben Upcroft
Executive Summary
This report summary the outcomes of the 7-week Vacation Research Experience Scheme at Queensland University of Technology during 2014 – 2015 Summer. The study is supervised by Associate Professor Ben Upcroft, Dr. Jürgen Leitner and Dr. Sareh Shirzi.

The goal of this research is to understand the fundamentals of controlling robots systems and explore the potential future usages of robots.

The main focus of the study is to investigate the use of visual feedback and bio-sensors for controlling a robot. Two practical experiments have been conducted to have a better understanding in this research. The results show both control approaches are feasible to operate a robot. However, the accuracy, speed and effectiveness of operations vary, depending on the external environment such as brightness. Possible future robot usages have also been discussed in the end of the paper.
Table of Contents
Executive Summary........................................................................................................................................... i
List of Figures................................................................................................................................................. iii
List of Tables .................................................................................................................................................. iv
1.0 Introduction .............................................................................................................................................. 5
  1.1 Robots in Society ................................................................................................................................. 5
  1.2 Robot Motion Controls ......................................................................................................................... 6
  1.3 Aims and Objectives ........................................................................................................................... 6
2.0 Setup and Experiments ............................................................................................................................ 7
  2.1 Setup ..................................................................................................................................................... 7
  2.2 Robot Motion Control by Using Visual Feedback ................................................................................. 8
  2.3 Tele-operation of a Robot by using Bio-signals .................................................................................... 15
3.0 Limitations and Conclusions ................................................................................................................... 20
Reference ......................................................................................................................................................... 21
List of Figures
Figure 1 Robot serve the meal at a restaurant in Harbin, China. Adapted from “Robot restaurant opens in China’s Harbin” by Xinhua News, 2012. ................................................. 5
Figure 2 Multiple Single-featured Objects Detection using Robot ............................................. 9
Figure 3 Top-down Manipulation Control Approach ........................................................................ 10
Figure 4 Plane-Scanning Manipulation Control Approach ................................................................. 11
Figure 5 Data Segmentation in Multiple Objects Detection ............................................................... 12
Figure 6 Measurement and Conversion between Pixels and Centimetres ........................................... 13
Figure 7 Result of Baxter Single Object Manipulation Controls ....................................................... 14
Figure 8 Result of Baxter Multiple Objects Manipulation Controls ............................................... 15
Figure 9 Turtle Simulation Controls by Using Myo Armband ............................................................ 15
Figure 10 Robot Motion Controls by Using Myo Armband ................................................................. 16
Figure 11 Multi-feastured Object Detection using Baxter Robot and Myo ........................................ 17
Figure 12 Control System using Myo Armband and Baxter Robot .................................................... 18
Figure 13 Baxter Robot Motion Controls System by Using Myo Armband ......................................... 19
Figure 14 Baxter Robot Multi-features Object Detection .................................................................... 19
List of Tables
Table 1 Four Reasons for choosing Rethink Baxter Robot........................................7
Table 2 Relationship between Pixels and Centimetres..................................................13
Table 3 Myo Gestures and Associated Commands in the Turtle Simulation....................16
1.0 Introduction

1.1 Robots in Society
Today, more and more robots are appearing in daily life and are increasingly applied in a variety of settings to serve people’s needs. Robots are now not only appearing to manufacturing, but also awaking a greater interest in the home usage and other service industries. Products such as vacuum cleaning robots or robotic toys can be widely seen in the market nowadays. According to International Federation of Robotics (2014), “about 4 million service robots for personal and domestic use were sold in 2013, 28% more than in 2012. The value of sales increased to US$1.7 billion.” At the same time, the capabilities of autonomous or semi-autonomous mobile robots also grew rapidly. Robots today can use a number of sensors to help themselves perform tasks in dynamic environments and data can also be computed much faster than before. It is believed that robots will “become an important technology in this century” and have much more future usages. (Corke, 2011)

The robotic restaurant in Harbin, China, is an interesting example of robots’ potential usage (See Figure 1). The restaurant owner employs 20 robots instead of humans that cook, serve and entertain its guests. Catering services are usually heterogeneous, in other words, operational inputs and outputs tend to vary more widely. By using robots, services can be standardized and risks of making mistakes can be reduced to minimum. Apart from that, the operating costs can be cut significantly as well. Hence, business may consider lowering the price and extending service hours to attract more customers.

Figure 1 Robot serve the meal at a restaurant in Harbin, China. Adapted from “Robot restaurant opens in China's Harbin” by Xinhua News, 2012.
1.2 Robot Motion Controls
Robot controls is one of the most critical technologies in the previous example. Robot need to know how to walk to the right customer’s table and what to do if there is an obstacle on the way. Over the years, robot motion controls has become a major research topic in robotics. Its goal is to enable robots to automatically compute their motions from high-level descriptions of tasks and models acquired through sensing (Choset, 2005). Today, robot operations have been expanded from structured environments to more dynamic environments. For instance, they are likely to work in the medical surgery, ocean or even space station, where brightness, humidity or pressure is quite different. As a result, sensing is affected, making robots perform tasks more challengeable. A remote-operated bomb disposal robot was also used in the 2014 Sydney hostage crisis, which requires a much more precise, accurate and flexible operation. As these needs increase, it is crucial to improve the motion controlling technologies to offer robots more capabilities. Human-operated robot need to understand operator’s each command precisely. Some robots are autonomous but they also need to ‘understand’ what is going on. This is supported by Argall (2009) who states that “robust motion control is fundamental to the successful, autonomous operation of mobile robots.”

1.3 Aims and Objectives
The aim of this explorative research is to understand the fundamentals of controlling robotic systems and explore the potential future usages of robots. Two different approaches are described and discussed in details in this paper. This research is conducted through QUT Vacation Research Experience Scheme program during 2014 – 2015 Summer. Five specific objectives are set at the beginning of this research, which are:

• Gain fundamental knowledge in robots and motion controls.
• Demonstrate and apply technical skills in robot control.
• Understand and apply computer vision techniques to control.
• Identify and compare different approaches for improving robot control.
• Identify potential usages of robots.
2.0 Setup and Experiments

2.1 Setup
Two practical experiments have been conducted to better understand the robot controls during the VRES research. Key instruments and tools include:

**Rethink Robotics: Baxter Research Robot**

In this research, Baxter will be mainly used for testing robot control in both experiments. As is shown in Table 1, there are four key reasons why Baxter has been chosen as the main experiment robot. Baxter is a safe, affordable robot which has a number of built-in sensors installed. Cameras on the robot can be used for controlling the motion, which fulfil the purpose of this experiment. Its LCD screen can also help improve the user experience, such as displaying the robot current status or the image of the detected object. Apart from that, Baxter’s force sensing arms ensure the safety of both operator and itself. But most importantly, Baxter is available for research student to do the experiment in the robotic lab at QUT Gardens Point.

<table>
<thead>
<tr>
<th>Components</th>
<th>Baxter Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Built-in Sensors</td>
<td>Left/Right/Head Camera + LCD Screen, etc.</td>
</tr>
<tr>
<td>Force Sensing Arms</td>
<td>Patented Series Elastic Actuator Sprint Technology</td>
</tr>
<tr>
<td>Cost / Availabilities</td>
<td>$25,000 / Available in GP Robotic Lab</td>
</tr>
<tr>
<td>Community / SDK</td>
<td>Rich Community Support and SDK for Python</td>
</tr>
</tbody>
</table>

*Table 1 Four Reasons for choosing Rethink Baxter Robot*

**Robot Operating System (ROS)**

Robot Operating System (also called ROS) is a collection of software frameworks that provides plenty of libraries and tools to help developers create robot applications. ROS is made of a set of nodes, running on one or more computers connected to the same network. Each node can offer services or broadcast messages consumed by other nodes that subscribe to the same communication channel, which is also called topic. Each topic has a certain type of message that can be delivered (Quigley, Conley, Gerkey, Faust, Foote, Leibs, … Ng, 2009). For instance, a topic will be used to deliver position from object detect node to motion node in the project. Therefore, program can move the end-effector to the right position of the detected object. In this research, ROS will be used as a middleware to develop all applications for the Baxter robot.
Thalmic Labs Myo Armband

Myo Armband is an innovative computer control device that detects the operator’s muscles electrical activity to allow control of computer. Myo consists of EMG muscle sensors and highly sensitive motion sensors. Unlike Microsoft Kinect, which usually requires camera, Myo Armband approach only detects gestures by monitoring the user’s muscle activities and sends the command back to a computer through Bluetooth. In this way, it can avoid many weaknesses of the traditional camera detection, such as brightness in the room or the resolution of the image. By using the Myo Armband, users will also have more flexibility as operators do not have to physically stand in front of camera. There have been many potential applications. For example, it has been shown that Myo can be used to move, stop and change robot’s directions.

2.2 Robot Motion Control by Using Visual Feedback

Vision-based control is one of the most popular approaches towards the improvement of robot motion controls. The robot has one or more camera installed and uses visual features extracted from the image to control the position of the robot’s end-effector. By repeating appropriate image processing and robot motion, robots can accomplish manipulation tasks more effectively under varying environments. According to Naoki, Masahide and Masaaki (2009), “the visual feedback loop is very effective for increasing the dexterity and flexibility of robot’s task.”

The first study is inspired by reactive reaching research (Leitner, 2014). An eye-in-hand configuration of the robot is used in the experiment, where the camera is attached to the moving hand and observing the relative position of the target. The goal of the first experiment is to use robot to find multiple objects placed on a table. To achieve this goal, robot will use its camera to monitor the whole table and estimate the objects’ positions (See Figure 2). Robot will move to each position achieved from the image afterwards. To keep it simple, each object will have the same feature (red circle in this case).
It was decided to start with a simple project with only one object to be detected. In order to identify the object placed on the table, Baxter’s right hand camera is used. To keep the experiment easy to implement, color of the object is selected as a parameter to recognize the object in the image scanning. Once the scanning of the table is finished, a built-in service from Baxter’s SDK is called, which can solve the inverse kinematic problem that determines the robot joint coordinates by using the estimated object position (Corke, 2011). Then another service is used to move the joints to the appropriate angles computed.

**Experiment Design**

The initial design (as is shown in Figure 3) is a top-down manipulation control approach. The right-hand camera will firstly move to the highest position where the entire table can be seen. If an object is found, the end-effector (gripper) will move along x-axis and y-axis with a fixed distance every time until the object is centred in the image. After that, the gripper will slowly lower its altitude and redo previous steps to centre the object. The program will work within a loop till the end-effector reaches the lowest point.

However, this simple approach failed in more than half of the cases due to a number of reasons. Firstly, the camera sensor is significantly affected by the brightness of the image. The picture looks darker as the camera is getting lower. Objects are sometimes wrongly, or not recognized in the image. Losing the object leads to an undefined state. Secondly, the movement distance along x-axis and y-axis is fixed, therefore the object is less likely to be centred perfectly. The end-effector went over the object and back to...
the start position several times. Thus, without a centred object, the application entered into an infinite loop. At the same time, due to the kinematic of Baxter, some given positions are unreachable. Other problems may also occur, leading to a new attempt and different approach.

![Flowchart](image)

**Figure 3 Top-down Manipulation Control Approach**

To avoid problems happening before, plane-scanning manipulation control approach was designed to achieve a more accurate position data and move the end-effector in a more flexible fashion. An idea is that computer will calculate the object distance (pixel) in the image and convert it to real-world distance (centimetre) for Baxter to execute. More concretely, the camera will firstly scan the table in a fixed altitude. The camera is meanwhile activated and used to recognize a red object. If a valid detection message is received from the object detection node, the given position will be
recorded in a list. When the scanning function ends, the average position will be calculated. The distance error between end-effector and the object is computed. The end-effector can be moved to the top of the object accurately. (See Figure 4)

![Figure 4 Plane-Scanning Manipulation Control Approach](image)

Similarly, multiple red objects detection can also be achieved by simply working on the previous idea. Most of steps can be kept in the same way except the position data processing. This is a challenging problem as the computer has to be ‘smart’ enough to differentiate objects. In this experiment, data will be segmented according to the distance between known positions in the list and predicted object positions.

Figure 5 shows the detailed idea of the segmentation in this project. Once an object is detected and a valid position is received, the program will check if the dataset is empty. If yes, data will be added to the list as a first element then call the Compute Average function, which calculates the averages of each segment. Otherwise, program will compare received data with each segment average data.
The distance between two points indicates the relationship between them. If the estimated distance to a segment is less than 0.10 m, it means the received position belongs to this segment. If no segment is suitable, the position will be regarded as a new object position and added to the end of the list.

Compute Average function will also be called to update the average list.

![Flowchart](image)

**Figure 5 Data Segmentation in Multiple Objects Detection**

**Experiment Implementation**

The first thing is to retrieve an image from Baxter’s right hand camera. In order to do that, a ROS service called `scan_publisher` is created, that subscribes to the topic called `/cameras/right_hand_camera/image`, where the images coming from that camera are published. The algorithm uses functions from the OpenCV library to process the image received. It filters color from a certain range in the HSV color space that corresponds to the color red, transforming it to a black and...
white image. White indicates the original red object. Then it searches for the largest contour on the image, which is assumed to be the object.

The algorithm calculates the distance between the end-effector to the centre of the contour. As discussed before, the program will convert the distance from the unit of pixels to the unit of centimetres at a fixed height. To find out the relationship between these two units, a small experiment has also been conducted (See Figure 6). Every time the object will be shifted one centimetre along the x or y-axis. The movement distance on the image is observed.

![Figure 6 Measurement and Conversion between Pixels and Centimetres](image)

Table 2 below records centimetres and pixel data in the experiment. The result indicates an estimated conversion formula that:

\[
1 \text{ pixel} = 0.061205 \text{ centimetres} \quad (Z \text{ Axis} = 25 \text{ centimetres})
\]

<table>
<thead>
<tr>
<th>Centimetres</th>
<th>Move Along X-axis</th>
<th>Move Along Y-axis</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 cm</td>
<td>(620, 504)</td>
<td>(637, 309)</td>
</tr>
<tr>
<td>2 cm</td>
<td>(620, 487)</td>
<td>(615, 309)</td>
</tr>
<tr>
<td>3 cm</td>
<td>(621, 470)</td>
<td>(603, 309)</td>
</tr>
<tr>
<td>4 cm</td>
<td>(621, 453)</td>
<td>(588, 308)</td>
</tr>
<tr>
<td>5 cm</td>
<td>(621, 436)</td>
<td>(570, 308)</td>
</tr>
<tr>
<td>6 cm</td>
<td>(621, 421)</td>
<td>(550, 308)</td>
</tr>
<tr>
<td>7 cm</td>
<td>(621, 405)</td>
<td>(539, 308)</td>
</tr>
<tr>
<td>8 cm</td>
<td>(621, 388)</td>
<td>(525, 308)</td>
</tr>
<tr>
<td>9 cm</td>
<td>(622, 373)</td>
<td>(508, 308)</td>
</tr>
<tr>
<td>10 cm</td>
<td>(622, 356)</td>
<td>(491, 308)</td>
</tr>
</tbody>
</table>

Table 2 Relationship between Pixels and Centimetres
The Program then estimates a relatively accurate object position in centimetres. It publishes the result to the ‘scan_simulation’ node, where all position data will be segmented and recorded.

To move Baxter’s arm to a specific point where the object is centred, the inverse kinematics service by the Baxter SDK is used. After the spatial and rotational coordinates are set, the SDK will return one possible set of joint angles. Baxter can then move its right arm and reach the x, y and z coordinates. Baxter then iterates through all the positions in the list.

**Experiment Results**

Baxter was able to find a single colored object and multiple same-colored objects on the table.

![Figure 7 Result of Baxter Single Object Manipulation Controls](image)

Figure 7 and Figure 8 show the project results. The program has been tested several times while objects were placed in different positions each time. It is mostly successful and accurate. However, a movement speed of only 0.05 m/s was set in order to achieve the best prediction result in the test. The scanning process takes fairly long to complete.
2.3 Tele-operation of a Robot by using Bio-signals

The purpose of this experiment is to control a robot’s behaviour by using bio-signals. The initial idea is to control the robot as the motion of the user’s arm in real-time. To achieve this, gyroscope, accelerometer and electromyography (EMG) data of user’s arm are required.

![Figure 8 Result of Baxter Multiple Objects Manipulation Controls](image1)

![Figure 9 Turtle Simulation Controls by Using Myo Armband](image2)
At the beginning, a simple application was created to get started with the Myo sensor. The aim is to control a turtle simulation available in Python (See Figure 9). Table 3 below shows the control gestures and associated commands used in turtle simulation. The application was written in Python and uses Python bindings from the Myo community. This attempt gave a very good introduction to the Myo and its functionalities. The result was fairly successful and turtle can be easily controlled. However, it seems that the sensor has some problems of detecting the “Wave Right” gesture and “Finger Spread” gesture. The motion of the operator’s arm also has to be significant, which can easily exhaust the users after a while.

<table>
<thead>
<tr>
<th>Gestures</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>Double Tap</td>
<td>Unlock the Myo Armband</td>
</tr>
<tr>
<td>Double Tap Twice</td>
<td>Change Pen Color and Draw a Circle</td>
</tr>
<tr>
<td>Wave Left</td>
<td>Turn Left 90 Degrees</td>
</tr>
<tr>
<td>Wave Right</td>
<td>Turn Right 90 Degrees</td>
</tr>
<tr>
<td>Fist and Rotate</td>
<td>Change the Speed of Turtle</td>
</tr>
<tr>
<td>Finger Spread</td>
<td>Start Moving or Pause Moving</td>
</tr>
</tbody>
</table>

Table 3 Myo Gestures and Associated Commands in the Turtle Simulation

Another similar experiment has also been conducted with the Baxter robot. Some joint positions of Baxter’s right arm have been pre-defined. A “Wave Left” gesture will move the right arm to the left and “Wave Right” will move it to the right. “Finger Spread” will allow Baxter to perform a waving hand action for three seconds. (See Figure 10)
Baxter can perform each task correctly in this experiment while same problems discussed above exist. At the same time, it was found out that the Python binding only includes a very limited number of key functions from the official SDK, which is written in C++. This brought a lot of inconvenience and problems for developing the initially planned application of directly controlling Baxter in Python.

Due to limited available time, an easier application that uses Myo as a controller for the behaviour of the robot was created. The application will use the program of the previous experiment to detect multiple-featured objects by using vision scanning. To keep it simple, colors will still be used as a parameter for feature detection. Instead of entering each command code in the command line, gestures will be detected to select and confirm the specific command. The behaviour selections and the current status are displayed on the screen of the robot.

**Experiment Design**

![Diagram](image)

**Figure 11 Multi-featuered Object Detection using Baxter Robot and Myo**

To achieve this goal, the same object detection approach used in the previous experiment is taken. Three red objects and two yellow objects are placed on the table. The workstation receives data from the Myo armband through Bluetooth. Once a gesture of the operator is detected, an associated command will be published to Baxter. Baxter will find the request colored object in the scanning and move to the position respectively. When the program finishes, Baxter will return to the starting position and wait for next command. (See Figure 11)
**Experiment Implementation**

The object detection approach from the previous experiment was used in this project. The only difference is that multiple features exist, addressed by changing the HSV color ranges to detect different objects. As before, Baxter moves its right arm to the predicted positions in the end.

In order to allow users to select and confirm their command, a number of selections will be displayed on Baxter’s screen. The system represents a menu of two levels and contains actions such as “Say Hi” or a color detection. Confirmation messages will also be displayed if the user makes a fist to confirm their selection. To achieve this, a ROS node called ‘screen_selection’ was created that publishes message to the topic called ‘/robot/xdisplay’. The ‘screen_selection’ node also subscribes to the ‘myo_command’ node that monitors the Myo gestures. The menu consists of a number of actions with specific image to display (See Figure 12). The system will publish the current selected behaviour.

![Image of Control System using Myo Armband and Baxter Robot](image-url)
Experiment Results

The results show that the control of Baxter using Myo is possible (See Figure 13). Baxter is performing the right task at the user’s request. The head screen also displayed the menu correctly.

![Image of Baxter Robot Motion Controls System by Using Myo Armband](image1)

**Figure 13 Baxter Robot Motion Controls System by Using Myo Armband**

Meanwhile, Baxter Robot can also successfully detect the different-colored objects and move to the right position in the end (See Figure 14).

![Image of Baxter Robot detecting objects](image2)
3.0 Limitations and Conclusions
Two experiments were successfully conducted in this study, even though some parts of experiments have been simplified due to limited time of the research. By doing these two experiments, I gained fundamental knowledge about robotics and motion control through practical experiments.

However, there are also some limitations about this research. The aim of this research experience scheme was to explore the robotic motion control approaches. A limited number of methods were tested. In the first experiment, only color feature was used to detect the object, but this approach might not work well in a more complex environment. Using object shape or object models should be considered in future experiments. Further investigation will be required to increase the speed and accuracy of the position predictions. In the second experiment, Myo was used to control the behaviour of the robot while the initial idea is to control the robotic motion. This is not feasible because only some features of Myo have been explored and used to control the robot due to limited functionalities in Python SDK binding. In addition to the five gesture detections developed by Thalmic Labs, more gestures should be developed to allow more actions of robots.

Lastly, a number of ideas about the potential robot usages also came out during this research. One of the future applications is to control a medical robot using bio-sensors. In recent years, the idea of using surgical robots has been increasingly popular and widely explored by many researchers. Surgical robots can overcome the limitations of minimally-invasive surgery and enhance the capabilities of surgeons performing open surgery. But at the same time, managing a robot in the operating room might also be challenging for medical doctors as they are not allowed to touch anything else during a surgery. Using bio-sensors such as Myo for robot control can be an ideal solution to allow effective collaboration between human and robot. In this way, it will not only increase the speed and safety in the surgery, but also reduce the cost of training surgeon. Another potential application is robotic dog with vision sensors for blind people. Rather than walking alone with a stick, the visually-impaired can walk a robot dog which can use the vision sensors to guide the directions and avoid the obstacles on the way. This can bring much more convenience for those who cannot see and people do not have to spend lots of money or efforts on raising a real dog.
Reference


