

Baxter Research Robot: Solving a Rubik's Cube

Chris Dawes

Student # 10282558

30 Mar 2015

Supervisor: Dr Alexander Rassau

Abstract

Robotics is currently used to perform many tasks but many of these are simple repetition of a predefined method. By combining AI with robotics we can greatly increase the applications of robotics. An algorithm that combines the vision and servo systems of a Baxter Research Robot with a solving solution for a Rubik's cube will demonstrate that the use of even simple AI with robotics allows complex tasks to be completed. Further integration of object recognition will allow the task to be completed in a dynamic environment, and further increase the areas robots are capable of working within.

1. Introduction

1.1. Motivation

The Baxter Research Robot by Rethink Robotics is a dual arm robot, with seven degrees of freedom per arm, released in 2012. Developed to be affordable, flexible in its purpose, and above all else safe, Baxter includes three cameras, one on each wrist and the other on its head, and a screen for displaying information relating to Baxter's current task. The robot is designed to be a versatile research platform while containing the same hardware as its industry counterpart, allowing research to translate into industrial applications (Rethink Robotics, 2015).

In general robotics artificial intelligence (AI) has been developed separately to robotics, but is now starting to become integrated. Unfortunately current AI is fragmented as each application focuses on one area, as opposed to making a true AI that thinks like a human (Bogue, 2014). Current usable AI is more akin to 'smart' robotics where decisions are made and problems solved by the robot in very specific applications. In industry, robots are expanding into areas that require more flexibility allowing robots to fill many more positions in increasingly complex areas (Hajduk, Jenčík, Jezný, & Vargovčík, 2013). Mobile robots are even becoming more common place, allowing for dynamic and spread out workspaces. These are all due to adding sensing and analysis to robots allowing them to react to dynamic environments.

To further robotics in industry, multi robot work cells have been designed that combine several robots working on the same part while cooperatively performing either one task, such as welding and the required handling, or multiple tasks at the same time (Hajduk, Jenčík, Jezný, & Vargovčík, 2013). The number of activities these work cells can perform increases

dramatically, as the complexity of the task or tasks can be higher while the robots don't need to be capable of performing the whole task individually.

For performing more human tasks, dual arm robots have begun to emerge (Hajduk, Jenčík, Jezný, & Vargovčík, 2013). By having two arms on one robot, mobility can be enhanced while providing some benefits of multi robot systems, further extending the use of industrial robots. Dual arm robots have a greater operational efficiency as they have a centralized controller, whereas multiple robots are individual standalone systems, thus needing individual controllers and additional centralized controllers for communication and task assignment (Zhou, Ding, & Yu, 2011). An example of dual arm robotics can be found in space applications. Development of dual arm space robotic systems is currently achieved by relying on tele-operation from an external location. This requires the same sensory information that could be used by an AI to automate tasks without the need of constant external control. This could lead to adapting dual arm robots with AI to industrial applications, especially where the task or environment is hazardous to humans. This could lead to much safer workplaces along with decreasing running costs and increasing efficiency while maintaining high quality.

Current research with Baxter has explored many different research areas. For motion planning an approach was developed that performed a learned task while avoiding obstacles and reacting to changing task related objects (Bowen & Alterovitz, 2014). This is executed in a closed loop manner, where an automatic control system is regulated by a feedback loop. This is implemented by constantly updating information on task-related objects.

The area of object grasping has been explored with Baxter by research into using grippers that can adapt to the shape and size of an object (Jentoft, Wan, & Howe, 2014). By applying tactile sensing to each of the three 'fingers' of the gripper, information regarding the object shape can be received, and grips can be adapted. Additionally by analysing sensor data from occurrences where the object is ejected from the hand, these forces and grips can be avoided.

In addition to motion planning and object grasping, real-time tracking of articulated objects has been demonstrated with Baxter (Schmidt, Newcombe, & Fox, 2014). Using Dense Articulated Real-Time Tracking (DART), objects consisting of rigid articulated bodies can be tracked. These can include many object such as chairs, humans and doorknobs. As position can be determined, robots can more easily interact with objects.

A demonstration of the Baxter Research Robot solving a Rubik's cube has been developed previously (Coles-Abell & Pugh, 2014). This was undertaken to demonstrate the ease of integrating Baxter Research Robot into a research application. They developed a workflow consisting of detect, verify, solve, manipulate and complete. Baxter was taught to use a flatbed scanner to image each face of the cube. This information was fed into OpenCV, an open source computer vision library, and processed generating a mapping of the cube. This mapping was then verified using custom algorithms to ensure that the cube was a valid cube. If this failed the cube was rescanned. This mapping was sent to the Kociemba algorithm which solves the cube and returns a set of required manipulations. Using the standard grippers the manipulations are achieved by Inverse Kinematics moves.

AI and robotics have been developed separately and are only recently being combined. By combining these a higher level of automation for robotics can be achieved. As explored previously, robotics is currently going through a change where AI has the ability to improve automation and flexibility of robots greatly in many industries and environments. This project is aimed at developing an algorithm that combines a simple form of AI with a Baxter Research Robot to automate a task. By choosing the task as solving a Rubik's cube, vision recognition and analysis systems are combined with motion planning and manipulation of objects to demonstrate the added flexibility by implementing simple AI.

1.2. Objectives

The overall objective of this project is to develop an algorithm that controls the vision and servos of the Baxter robot to allow it to pick up a Rubik's cube, visually analyse the cube, find a solution, and allow the Baxter to manipulate the cube to solve it. As research has already been done on this, I will be aiming to use only the Baxter robot and no external hardware. This means that the project will require only the Baxter robot and a computer setup as a workstation to communicate to the robot.

The solving algorithm must allow any combination of a 3x3 Rubik's cube to be solved. This requires a set of rules for solving the Rubik's cube to be programmed, as opposed to using a database of solutions. The vision must be able to recognise and determine placement of a Rubik's cube placed in front of Baxter on a table, and to recognise the different colours on a Rubik's cube with accuracy.

Movements made by Baxter will be mostly programmed in, such as the movements for analysing the cube and for manipulating it. Motion planning will be needed for retrieving the cube and placing it back. Ideally the cube will be placed on any spot in reach of Baxter.

As an extension the algorithm could also include the ability to solve a 4x4 Rubik's cube. This will require additional research into a solving solution, and possible modification of the grippers. Additionally the vision software will also be required to identify the type of Rubik's cube, and additional movements to manipulate the 4x4 cube will need to be defined.

1.3. Significance

By demonstrating the use of an algorithm that solves a task using vision recognition systems, servo systems and motion planning system, along with a solving solution, it shows robots can be used in more dynamic and variable systems. This increases the applications of robots greatly and allows more complex tasks to be automated in industry. In turn safety in workplaces can be improved by replacing the worker, quality of the products can be ensured, and speed of tasks can be improved.

Currently robotics is used mainly in repeat tasks, where high repetition is needed, whereas developing intelligent automation will allow an increase in available environments and tasks that can be performed by robots. Applying this to dual arm robotics extends task further, allowing human like manipulations to be performed. This combination of intelligent automation and dual arm robotics will further the uses of robotics greatly in all applications, from space, to industry and manufacturing.

2. Proposed Approach

The project can be broken down into a number of task that need to be completed sequentially and successfully for the objective to be achieved. All the programming will be done in Python using the Integrated Development Environment (IDE) Visual Studios. Python was selected as a programming language as the Baxter robot has an Application Programming Interface (API) that allows direct python control of the Baxter robot. Additionally the Python programming language has been developed with emphasis on code readability allowing creation of well laid out code.

The majority of the work will be done on two computers, one of which I have set up for programming, and one with simulation software Gazebo (Open Source Robotics Foundation, 2014). This tool is designed for robot simulation capable of simulating detailed environments and multiple robots and allowing algorithm testing. The computers will be available at all times meaning work on the programming will not be limited by resource availability. For movement planning and manipulation MoveIt will be used (Sucan & Chitta, n.d.). This allows movement planning without having to interact with Baxter physically, reducing the impact of the robots unavailability.

When working with Baxter RViz will be used (Rethink Robotics, 2014). This tool allows the current configuration of Baxter to be displayed, along with camera vision and individual sensor values which will allow superior control and visualisation of Baxter's movements.

Only one Baxter robot is available but there are multiple projects running on Baxter. A division of time to be spent testing and working with Baxter will need to be established to allow all the projects ample use of Baxter. However, use of the simulation software should allow some work to be done without working with Baxter directly and thus allowing work to continue at a similar pace.

First vision recognition will be used to identify a Rubik's cube placed in front of Baxter. Then inverse kinematics will be needed to obtain the cube, given the relative positions of the gripper and the cube. The initial placement of the cube will need to be recorded for returning the cube. A vision system is already provided with Baxter but colour recognition will need to be added. This will require research on relevant software and to pick the optimal solution through a trade-off analysis.

Vision recognition will also serve the purpose of identifying the placement of the 6 different colours. As this system could have errors a check to ensure that there is 9 of each colour will be put in place to determine if any errors have occurred. If an error occurs, the colour will either be inferred or the cube will undergo a rescan. All of the colours will be saved either as strings or values in tables, each representing a face. The colour will be determined by comparing the pixel value to a range of values that represent each colour. By including a range of values for each colour the vision recognition should be more robust.

Before the cube can be scanned, a set of movements must be defined that will scan all 6 sides of the cube one at a time. This will require programming in a set of movements for each arm to perform, with the colour recognition being used between each movement to scan the cube.

Movelt will be used to plan the motions for Baxter's arms and then these motions will be converted into Python code.

The saved colour positions will serve as the inputs to a solving solution that will solve the Rubik's cube and output a list of face rotations that will need to be performed. These rotations will be translated to sets of movements that are predefined within the code and then performed in order. As many programs have been written to solve a Rubik's cube, research will be done into an efficient solution that requires the least time to solve, and the smallest number of movements. However the ability to integrate this program into python, and allow it to interact with the rest of the algorithm will be the most important criteria that will ultimately determine the solution used.

After the cube is solved, Baxter will then return the cube to its initial position and the algorithm will end. This will be achieved using reverse kinematics based on the saved initial position of the cube.

3. Timeline

A Gantt chart can be found as Attachment 2. This chart assumes that by week 12 of semester two, all project work should be completed for presentation in week 13. The chart also omits semester two assessments as dates cannot be determined at this time, although the assessments will be worked on in parallel to the project. The final report will require continuous work and as such will run for most of semester two. The semester one report and presentation have been given two weeks each, and are assigned time parallel to task work.

As seen in the chart each task has been further broken into subtasks. Each task is defined as a section of the project that can be developed and tested by itself which allows for more targeted debugging. Each task for the project are completed sequentially as they are designed to be modules that will be combined after all project work related tasks are completed. Even though time for testing each module has been assigned, rigorous testing has been allowed for once all the tasks have been combined. This is for unseen issues that may arise. Additionally two weeks has been given no project work as this is exam time for other units.

Given the generous time allowed, and the various testing that occurs, this timeline is realistic and should allow for any unseen setbacks.

4. Risk Assessment

Majority of the project work is done on a computer, with testing the algorithm done by using the Baxter robot. This is reflected in the overall low risk rating shown by the risk assessment matrix as Attachment 1. The risks have all been assigned a three letter reference that fits in the following categories:

- SUP - Supervisor
- PER - Personal
- ERG - Ergonomics
- EQU - Equipment
- CMP - Computer
- SFT - Software

Many of the risks are inherently low due to low probability of occurrence or low impact on the project if they were to occur. The two medium risks are eye strain due to computer work and loss of project data. These risks required minimisation as they couldn't be removed completely. For risk ERG_03 the approach to reduce the risk is to require regular breaks from the computer, and to ensure ergonomic eye angles and screen distance are abided by. Risk CMP_01 is medium because of its impact to the project. Losing project work can set back the project by weeks and as such must be minimised. By using cloud storage as an additional backup to having the files stored on multiple computers, this reduces the risk impact to low. Cloud storage also has the added benefit of allowing multiple computers to have access to the same version of project files.

5. Progress to Date

Current progress has been made to get the Baxter research robot ready to be used. A workstation installed with Ubuntu 14.04 with Robot Operating System (ROS) Indigo has been set up to allow a permanent connection to Baxter. The software on Baxter itself has also been updated to the latest version, 1.1. The arm configuration program has also been run to ensure correct sensor measurements and precise movements by Baxter. However the configuration program will need to be run every month as recommended by Rethink Robotics.

A personal Laptop has also been installed with Ubuntu 14.04 and ROS. This is for the purpose of simulating the Baxter robot when physical testing is unavailable. This laptop can also be setup to connect to Baxter, if the permanent workstation stops working.

6. Conclusion

Automation through the use of robotics is currently at a stage where many tasks are simple repeating a predetermined method. By applying the growing field of AI to automation, robots role can expand greatly, increasing the environments they can be used in alongside expansion of the tasks that can be performed. By developing a set of algorithms that harnesses a solution algorithm to solve a Rubik's cube, with vision recognition systems, manipulation and movement planning systems and a dual arm robot, a task that is relatively complex can be completed. This can change how robots are used in not only industry, but in applications as complex as space maintenance and exploration.

7. References

- Adorno, B. V. (2011). Two-arm Manipulation: From Manipulators to Enhanced Human-Robot Collaboration. Université Montpellier II - Sciences et Techniques du Languedoc. Retrieved from <https://tel.archives-ouvertes.fr/tel-00641678/>
- Bogue, R. (2014). The role of artificial intelligence in robotics. *The Industrial Robot*, 41(2), 119-123. doi:<http://dx.doi.org.ezproxy.ecu.edu.au/10.1108/IR-01-2014-0300>
- Bowen, C., & Alterovitz, R. (2014). Closed-Loop Global Motion Planning for Reactive Execution of Learned Tasks. In K. Lynch, & L. Parker (Ed.), *IROS* (pp. 1-7). Chicago: IROS.
- Chen, F., Sekiyama, K., Cannella, F., & Fukuda, T. (2014). Optimal Subtask Allocation for Human and Robot Collaboration Within Hybrid Assembly System. *Automation Science and Engineering, IEEE Transactions on*, 11(4), 1065-1075. doi:10.1109/TASE.2013.2274099
- Coles-Abell, H., & Pugh, V. (2014). *Baxter Solves Rubik's Cube*. Retrieved from Baxter Research Robot Wiki: http://sdk.rethinkrobotics.com/wiki/Baxter_Solves_Rubiks_Cube
- Hajduk, M., Jenčík, P., Jezný, J., & Vargovčík, L. (2013). Trends in industrial robotics development. *Applied Mechanics and Materials : Robotics in Theory and Practice*, 282, 1-6. doi:10.4028/www.scientific.net/AMM.282.1

- Jentoft, L. P., Wan, Q., & Howe, R. D. (2014). Limits to Compliance and the Role of Tactile Sensing in Grasping. *International Conference on Robotics and Automation* (pp. 1-6). Hong Kong: International Conference on Robotics and Automation.
- Open Source Robotics Foundation. (2014). *Gazebo*. Retrieved from Gazebo: <http://gazebo.org/>
- Rethink Robotics. (2014). *Baxter Research Robot Datasheet*. Retrieved from Rethink Robotics: http://cdn-staging.rethinkrobotics.com/wp-content/uploads/2014/08/BRR_009.09.13.pdf
- Rethink Robotics. (2014). *Baxter Research Robot Wiki*. Retrieved from Rethink Robotics: http://sdk.rethinkrobotics.com/wiki/Main_Page
- Rethink Robotics. (2014). *Rviz*. Retrieved from Baxter Research Robot Wiki: <http://sdk.rethinkrobotics.com/wiki/Rviz>
- Rethink Robotics. (2015). *Baxter Research Robot*. Retrieved from Rethink Robotics: <http://www.rethinkrobotics.com/baxter-research-robot/>
- Schmidt, T., Newcombe, R., & Fox, D. (2014). DART: Dense Articulated Real-Time Tracking. *Robotics: Science and Systems* (pp. 1-9). California: Robotics: Science and Systems.
- Sucan, I. A., & Chitta, S. (n.d.). *Movelt! Home*. Retrieved March 1, 2015, from Movelt!: <http://moveit.ros.org/>
- Zhang, T., & Ouyang, F. (2012). Offline motion planning and simulation of two-robot welding coordination. *Frontiers of Mechanical Engineering*, 7(1), 81-92. doi:10.1007/s11465-012-0309-4
- Zhou, J., Ding, X., & Yu, Y. Q. (2011). Automatic planning and coordinated control for redundant dual-arm space robot system. *The Industrial Robot*, 38(1), 27-37. doi:<http://dx.doi.org/10.1108/01439911111097823>

Attachment 1 – Risk Assessment Matrix

Risk Reference	Risks	Consequences	Current Risk Treatments	Current Level of Risk				Additional Risk Treatments	Residual Level of Risk			
				Likelihood	Consequence	Risk Level	Ranking		Likelihood	Consequence	Risk Level	Ranking
SUP_01	Supervisor is not contactable/unavailable for an extended period of time	Advice and guidance cannot be given, possibly allowing misinformation and a lack of help when needed.	The unit co-ordinator can be approached for general information.	1	2	2	L	None Required	1	2	2	L
PER_01	Sickness or unrelated injury	Reduced work progress or complete stoppage. Can set behind the whole project causing stress.	Computers available at residence has required software to work on much of the project, allowing work to continue, albeit at a slower pace.	2	2	4	L	None Required	2	2	4	L
PER_02	Project work causes too much stress	Reduced ability to work on both project and other units. Can fall behind further increasing stress.	Maintain health level of commitment to work, while allowing for time spent outside of project and unit work.	2	2	4	L	None Required	2	2	4	L
ERG_01	Bad posture or sitting position	Sore back, shoulders and neck. Will reduce productivity and act as a distraction preventing focused work.	Adjustable chairs with movable seat angle and back position allowing an ergonomic seating position.	2	2	4	L	None Required	2	2	4	L
ERG_02	Repetitive Strain Injuries in hands or wrists	Computer work becomes painful impacting on both project work and unit work.	Regular breaks from computer work.	2	2	4	L	Implement regular wrist exercises. Use correct hand position according to ergonomic standards.	1	2	2	L
ERG_03	Eye strain/soreness	Work must be stopped to rest eyes.	Have regular breaks from computer work.	3	2	6	M	Correct ergonomic eye level and distance from screen.	2	2	4	L

EQU_01	Baxter robot breaks completely	Baxter cannot be used to test or demonstrate project.	Simulation software Gazebo can be used in place of Baxter for proof of concept.	1	3	3	L	None Required	1	3	3	L
EQU_02	Collision of Baxter robot with person	Affected area can have slight soreness.	Baxter's limited force application (lifting limit of 2kg) and slow motion ensures all collisions are low impact.	2	1	2	L	Area near Baxter to be clear of persons before and during use.	1	1	1	L
EQU_03	Some systems of Baxter stop working	All the main functions are needed, and as such this could completely eliminate using Baxter.	Use of Gazebo simulation software in place of Baxter may be required but will be sufficient to perform tests and present the project.	1	3	3	L	None Required	1	3	3	L
CMP_01	Loss of data	Time and effort wasted by losing data due to hard drive crashes or data corruption.	Copies of all project files are on different computers and backed up on USB's	2	3	6	M	Use of cloud storage to add another backup of project data and allow syncing between devices.	1	1	1	L
CMP_02	Workstation computer that connects to Baxter fails	Communication through workstation to Baxter and thus testing cannot be done.	Laptop is set up with required software and can be used instead of lab computer. Software can be reinstalled on a different computer in the lab. This would require a couple hours of work.	1	3	3	L	None Required	1	3	3	L
CMP_03	Either personal computer used for programming work fails	Limit personal computer work and therefore set back or slow down project work.	Multiple computers set up to work on.	1	2	2	L	None Required	1	2	2	L
SFT_01	Communication software becomes unsupported	Cannot communicate and thus use the Baxter robot.	Gazebo Simulation Software can be used in place of Baxter robot.	1	3	3	L	None Required	1	3	3	L
SFT_02	MoveIT software becomes unsupported	Motion planning cannot be done using the software. Can cause a large setback.	Motion planning will require test with the Baxter robot, or use of Gazebo simulation software.	1	3	3	L	None Required	1	3	3	L

SFT_03	Rviz software becomes unsupported	Detailed information about Baxter cannot be displayed while testing.	Although less information can be obtained, tests can still be performed. Debugging of Baxter can be done by rqt_console software.	1	2	2	L	None Required	1	2	2	L
SFT_04	Visual Studios IED becomes unsupported	Programming cannot be done using Visual Studios, stopping relevant work.	Another IDE can be used in place to allow programming work to recommence.	1	2	2	L	None Required	1	2	2	L
SFT_05	Gazebo Simulation software becomes unsupported	Accurate simulation of Baxter robot in an environment cannot be done. Can slow down project development.	Physical testing with Baxter must replace simulation tests.	1	3	3	L	None Required	1	3	3	L

Activity Overall Risk Rating	Low
-------------------------------------	------------

Attachment 2 – Timeline Gantt chart

