From Robotics to 'Tesco Technology': An Evolving Engineering Curriculum

Presenting Author: Philip Tranter
ptranter@uclan.ac.uk
Coauthors: Geoff Hall, Martin R Varley, Mahesh M Patel
ghall5@uclan.ac.uk, mrvarley@uclan.ac.uk, mmpatel@uclan.ac.uk
University of Central Lancashire, Preston, UK

Abstract

At the University of Central Lancashire (UCLan), School of Computing, Engineering and Physical Sciences (CEPS), we run a successful suite of courses in the area of Electronic and Computer Engineering (ECE). This paper tells the story of how Robotics took over the delivery of several aspects of the ECE curriculum. The story isn't over yet, because it is starting to influence other modules on other courses. It starts with a number of largely out of date robot systems, fortuitously bequeathed to a small group of interested staff, who then went on to build a complete Robotics stream, which is beginning to cause ripples throughout the ECE subject area and beyond. It has generated great interest and enthusiasm amongst staff and students alike. It was not formally planned, and its success cannot be attributed to any individual. It literally evolved, and continues to gather a momentum all of its own. It has challenged not only the way we teach, but <u>what</u> we teach, because it has forced us out of the modular approach of theoretical lectures accompanied by directed practical work, into the 'real world', where knowledge and skills addressed in various taught modules are integrated to address a particular problem In effect, we recognised that we needed to remove artificial dividing lines between disciplines, because what we teach our students, and how we expect them to learn, is fundamental to their future careers and employability.

Introduction

The University of Central Lancashire, School of Computing, Engineering and Physical Sciences offers a range of traditional degree programmes under the collective title 'Electronic and Computer Engineering' (ECE). The content of these programmes is regularly updated to reflect current trends in the Electronics Industry, and includes a range of specialist 'titled' awards like *Digital Signal and Image Processing*, *Digital Communications, Robotics and Mechatronics, Electronic Design Automation,* alongside the traditional title of *Electronic Engineering*. The specialist titles reflect staff expertise and the links with local industry. Being situated at the heart of the North West Aerospace industry, many of these specialisms are aligned to meet the needs of this industry; for instance, the image processing work is centred on imaging related to non-destructive testing. The application of Robotics to the Advanced Manufacturing techniques of the Aerospace industry is one that is gaining considerable interest [1], and so for a number of years the ECE team has been developing an interest in Robotics and its application to Industry.

The other main driver for the interest in Robotics has been the recent availability of cheap, powerful technology that has obvious (and not so obvious) applications to the development of Robot Systems for a whole range of domestic, entertainment, industrial, and even health and personal care applications. The heart of these applications often consists of software running on embedded systems; these systems must necessarily 'interact' with the outside world via electronic communication systems connected to a whole range of sensors/actuators. The availability of cheap, sophisticated processor and sensor systems has allowed our curriculum to 'evolve' in directions that we could not have predicted nearly three years ago when we started out on our journey. The term 'Tesco Technology' relates to the fact that digital systems like webcams, Bluetooth and infra-red (IR) communication systems, and radio frequency identification (RFID) technology are no longer the preserve of Research Centres, but are available and in use just round the corner at the local supermarket, and that very often, our students (and even our children!) understand the use of these technologies better than we do.

Supported by our colleagues in the Motor Sport/Computer Aided Engineering part of the School, we already had all of the ‘support subjects’ available to review and extend our expertise and course provision in Robotics.

Beginnings

The provision of Robotics courses at UCLan had been in existence for some time, but had never really taken off in terms of staff or student interest. As a result of the end of a research contract, we were bequeathed the following resources:

2 Neuronics Katana Robot arms
2 P3-DX mobile robot platforms
5 ‘Surveyor’ ARM-based mobile robots
A collection of Lego MINDStorms kits

(a) Katana Robot Arm *(b) Pioneer P3-DX mobile robot* *(c) Surveyor mobile robot*

Figure 1: Our first Robots [2,3,4]

All the documentation and drivers for these robots was quickly downloaded from the web. We soon discovered that the hardware for all these robots had subsequently been updated, but that they were all usable once we had overcome the initial difficulties of nursing them back to life. We spent some time using Linux to drive the Katana arm, before the software wizard of the team wrote wrappers which allowed us to drive it from the Windows .NET platform. The P3-DX had its own processor which allowed serial communication, and the Surveyors had a wireless XBee-USB communication system which showed up as a virtual serial port on a PC.

Things began to move . . . . literally. From our office, into the corridor, around the lab, and back into the office. Despite the availability of multi-£k cameras in the School, we fitted a £10 ‘Tesco-technology’ webcam to the Katana robot arm, and had the arm programmed to make a number of sweeping, diving gestures as if it was looking for something – the pictures were displayed on a PC screen. The psychological effect of a robot arm appearing to point a webcam at someone and display their picture on-screen in itself produces some interesting reactions. The robots became a point of interest – in class, in the office (people used to drop in just to watch), and at open days and publicity events.

The history of Robotics is littered with early enthusiastic promise followed by a more leisurely reflection on what went wrong – especially in the USA [5]. We were anxious not to allow a repeat of this. On the back of a lot of early enthusiasm, we started putting together an undergraduate year two (level 5) 'Robotic Systems' module, and assembled a small team of staff to deliver it. Our 'vision' at that stage was that it would include:

Sensors, Actuators and Control
Kinematics (including inverse kinematics)
Software Development
Robot simulation
Robot Learning

We wanted it to be available to our second and final year undergraduate degree students, but initially we wanted to restrict access to around 15-20 because of equipment constraints and the possibility of unanticipated 'teething problems'.

Delivering the Robotics Module

We made Robotic Systems an option at the start of the academic year and got 16 carefully selected year two students. We started the module with lots of demonstrations of robots, complemented by each student making a short presentation of a 'youtube' type of video on some aspect of modern robotics. We all learned a lot from those videos – the ingenuity of students at finding things on the web never ceases to amaze us. Staff and students were inspired by visionaries like Rodney Brooks, who confirmed our expectations that the future of Robotics really is just around the corner, and we were impressed by the antics of 'Big Dog' [6], as it slid around on the ice, but never quite fell over. We looked at the early autonomous robots, like 'Shakey' [7] and the 'Stanford Cart' [8] (which, in 1979, took a mere 6 hours to cross a room littered with obstacles), and we just 'knew' we that could do much better with the technology we now have.

The students were great; without exception, they were enthusiastic about robotics, and there was a sense of excitement each time we met as a group. It soon became clear (without it being intentional) that this subject was not like any other in terms of what we put into it, and what we got out of it. I have never felt so comfortable saying things like, *'I don't know – does anyone have any ideas?*'. I found myself giving 'introductory' talks on things, only to find that during the week, the students had gone away and read up on the subject, and came back knowing more than I did. This was surely how teaching should be . . .

There was one problem, however, that came back to haunt us over and over again – in fact this problem came to define our approach to Robotics. The problem was simple (and I'm sure very familiar) – the students just didn't *know* enough to be able to solve 'real world' problems beyond the confines of individual modules. In some ways they were like our robots without sensors – they had powerful enough brains, but no knowledge of the world around them or how to interact with it. The problem was not that they were weak students; the problem was that what they had been taught until then just didn't prepare them to consider practical interpretations of questions such as: *How powerful should a motor be? How do we choose a power source? What is the difference between an a.c. and a d.c. motor? How long does a battery last? What is torque? What is power? How strong should a gripper be? When does the strength of a Robot become dangerous to humans?*

It got worse. The students had not been taught control theory in year one, and they had only met feedback in the context of operational amplifiers. They could program microcontrollers in assembler language, and PCs in C (and even C++), but the idea of a processor interacting with the outside world in real time was beyond their experience. They (almost) all hated programming, because it was perceived as too difficult, and too boring to be of any use. As lecturers who had taught all these subjects, we take collective responsibility and it made us question what had gone wrong with Engineering Education. We also questioned whether it was 'just us' in our School, but realised that there are enough of us with external links and contacts within academia and industry, including several members of the team acting as external examiners, to know that we are probably not alone in having these problems. One theme that arose out of team discussions was '*How can we send our students in to Industry like this?*' We started asking them what they thought an Electronic Engineer did in their daily work.

There were successes, however. The most surprising of which was when we introduced .NET, C#, object-oriented programming (OOP) and Visual Studio. Of all the things we covered, this was perhaps the one that seemed like it might go completely wrong. We started by looking at traditional 'procedural' programming, and noted that a graphical user interface (GUI) presents serious problems to this approach. We discussed the difficulties of making a Window appear on a screen, with all its functionality and properties. We looked at the distinction between 'interface' and 'program logic'. We started up a C# template 'project' and noted that there were four main files (plus others that we needn't concern ourselves with), and we wrote that first program where you press a button and a dialogue box pops up and says *'Hello World'*.

And we never looked back . . .

Within a few weeks, we had Surveyor robots scurrying around the laboratory, radio controlled by fully-featured Windows GUI programs with buttons, sliders, text boxes reporting progress, and beeps of satisfaction when robots reached their goals. Looking back at the lecture notes, we taught them about 20% of what they knew by the end of that part of the course – students who previously wanted explicit instructions on how to . . . . just went away and found out how to do things by themselves. Not only that, they came back the following week to show us how clever they'd been, and what they were capable of. In the space of six weeks, students who didn't like programming were writing multi-threaded, real-time (almost) software, that handled asynchronous radio communications with mobile robots. We realised that we had tapped into their enthusiasm.

We recognised gaps in the students' prior knowledge and experience as we went along, and introduced aspects of mechanics, motors, control (including a little Laplace). In the longer term, we resolved to put this right in a more structured way. We very soon realised that in future we needed an undergraduate year one (level 4) module, because there was just no way to fit everything that we wanted to cover in a single module. We used spreadsheets to model 'forward-backward neural networks' which solved inverse kinematic problems, and we came to understand why babies have difficulties taking hold of things, even when you place them right in front of their eyes. We looked at a range of sensors, and how they don't always tell you what you think they are telling you (try using an infra-red sensor on a sunny day). We started to try to think about how it is that humans are so good at things like catching a bus, and yet Robots that can beat Grand-Masters at Chess [9] can't even find the end of the corridor. We looked at Robots in Films – C3P0 and R2D2 [10], Marvin the Paranoid Android [11], and 'Maria' from the film *Metropolis* [12] – and wondered why after nearly 100 years of Robots in films, we still don't have anything like them in reality. We looked at Artificial Intelligence (AI), and again resolved to spend more time on it in future. We started to consider the idea of the final year undergraduate (level 6) module . . .

By the end of that first year of delivering the level 5 module, we had very clear ideas of what had gone well, and what needed reviewing. The module got good reviews by the students, and we started to think about Final Year Robotics projects for those students. Then, something else came along.

Parallel to the delivery of the Robotics module, a new development was taking place centred around 'Advanced Manufacturing'. It consists of a facility for showcasing our School's expertise in 'digital technologies' [1] as applied to the manufacturing, inspection and test processes of high-tech manufacturing companies like those in the North West Aerospace Industry. Part of the facility uses Robots for automated assembly and non-destructive ultrasonic scanning. This resulted in the availability of two Fanuc LR Mate 200iC Industrial Robots (see figure 2), and re-ignited our interest in the Industrial application of Robotics.

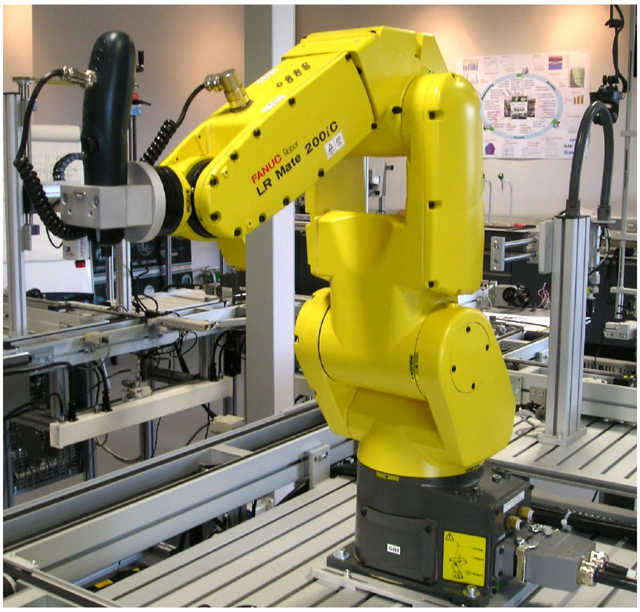

Figure 2 FANUC Robot at the Advanced Digital Manufacturing Facility, armed with screwdriver

The assembly robot was initially configured to tighten up screws in a partially assembled piece – simple enough, until you try to do it. The Robot holding the screwdriver never knew if there was a part there (although it knew exactly when there ought to be a part there). When there was a part there, it couldn't tell if there was a screw present. If the screw didn't line up with the screwdriver, the robot tightened it anyway (or at least, tried to).

Ideas for solutions came fast – in fact everyone knew how to solve this particular problem, until it came time to actually implementing the solution! There would be image processing, infra-red sensors, LabVIEW, MATLAB, laser beams and a whole range of other technologies brought to bear on this 'trivial' problem.

The existing work on Robotics plus this new facility allowed a whole range of final year project ideas, many of which are coming to a conclusion at the time of writing. Students work on 'real' problems (for example, how to determine which way round a screw is, and how far down a thread it is), and consider how to make Robots move around safely using a whole range of sensors and image processing techniques. It turns out that none of the problems are trivial, but all of them are interesting.

Moving Forwards

We now have a year one module, which ran for the first time in 2010-11. The first year students, just 8 weeks into a C programming course, took to C# and .NET with the same enthusiasm as year two students did last year. We continued to look for new ways of controlling robots. The first year assignment was to remotely steer a robot around an obstacle course in the minimum time, using any means. I had thought they might use Windows buttons and slider controls, and maybe even the mouse - I was completely wrong! Being keen 'gamers' (and used to remembering and using complex key combinations like 'Ctrl-Alt-Shift-A'), they all knew that by far the quickest way to communicate with a computer is via the keyboard. They went away and learned how to receive input from the keyboard – the result was a flurry of hands and fingers as the robots raced around the track – apart from one student, who brought in what looked like a TV remote from Tesco, and used that!

We now have demos of robots being controlled with even more 'Tesco Technology', including a Wii steering wheel and an Android mobile phone. The Wii controller also has an infra-red camera and built-in image processing to locate and report the position of up to 4 IR LED light sources – this will be used as part of a robot navigation system. The 'Nunchuck' Wii controller uses a two-axis position-sensed joystick for control purposes – this is to be offered as a Robotics project for next year. We are about to order an Xbox 'Kinect' in order to use its 3D vision system. All available at a supermarket near you!

There is now a UCLan radio-controlled/autonomous mobile robot, loosely based on the Surveyor Robot, using the same software interface, and costing about a quarter of the price. We aim to have a set of 20 for practical work.

The content of the three modules is now starting to settle down, and is detailed below:

Year One:
Brief history of robotics, up to current state of the art
Mechanics – force, torque, power, velocity, acceleration
Software – C# and .NET, simple Windows-based control applications, single threaded
Forward Kinematics
Mechanical drawing using SolidWorks

Year Two:
Degrees of freedom, forward and inverse kinematics
Sensors and actuators/motors
Software – multithreaded PC based applications responding to asynchronous input
– multithreaded, mobile ARM-based applications using the .NET micro framework
Basic control theory
Application of Robotics to Industry
Introduction to AI, development of Robotic paradigms
Introduction to image processing

Final Year:
Industrial Robotics using the FANUC robots
Artificial Intelligence, comparison of the three paradigms
Learning systems
Multi-agent systems
Web-based remote control
Robot navigation systems
Vision systems

The final year work will be coupled to Final Year Project work on Robotics. Typical topics proposed are:

Autonomous mobile robots with collision detection
Robot navigation systems (various)
Task decomposition into simple behaviours
Robotic learning/behavioural systems
Vision systems for Industrial Robotics
Instrumentation using mobile robots to gather data
Remote process/factory supervision using semi-autonomous robots
Collaborative Robots
Use of intelligent robots in assembly/disassembly
Human-Robot interfaces

One interesting development is the availability of cheap ARM-based processor boards that carry the .NET micro framework and are small enough to be built into the UCLan mobile robot (the specification reads like a 1990 PC: 640K memory, 48 MHz processor). These use the same development tools as those for writing Windows GUI software on a PC, so experience in one feeds into the other; the undergraduate year 2 students taught themselves to use them! There are a range of sensors and add-on boards for communications, etc., making them very flexible for quickly building up quite complex systems. The strength of the .NET micro framework is that interfaces like serial, I^2C, SPI, Ethernet, and SD cards are built-in, and require just a few lines of code to implement.

The Future

Rodney Brooks [13] told us that ‘Robots will Invade our Lives’. He’s right – they have invaded ours already! There is so much interesting, cheap technology out there just waiting to be used to make better robots. There are so many good reasons to want to build better robots – you can already buy robots that will sweep your floors and cut your grass, but there are developments over in Japan [14] where ‘robot pets’ are being developed as companions for older single people, and they are even experimenting with robot ‘carers’ [16]. As an illustration of how socially engaging robots might be, an estimated 38 million of the furry ‘Furby’ robots were sold worldwide [13]. Brooks [15] also makes the point that future demographics in developed countries will *necessitate* the move to robotics as a means of improving the productivity of an aging population. Put simply, there won’t be enough workers to pay for our increasingly long retirements unless we can increase the effective productivity of our dwindling active workforce. Japan is acutely aware of the demographic transition and has invested over many decades not only in production robots but also assistive robots including for health care [16].

Concluding Remarks

The future is bright for Robotics in Engineering Education. The subject is interesting, challenging, uses all the modern technology that we Engineers like to play with, and most importantly, it captures the imagination of our students. Studying Robotics encompasses Electronics, Mechanics and Software, all in a ‘problem-solving’ context. Our students didn’t learn C# because they wanted to be programmers – they learned it because they needed it to solve a problem. Most importantly, they learned *how* to learn by themselves, and once they realised they could do this, they had the confidence to tackle much more complex problems. Importantly, the students appear to perceive problem solving as the focus of the challenge. For example, one of our best students just finished his Final Year Project on a vision/sonar based navigation and collision avoidance system [17]. This was all implemented in software, yet he hardly mentioned the software techniques in his report – software was not <u>what</u> he did, it was just the tool he used to solve the problem.

Figure 3 Collision Avoidance project [17]

Robots have invaded our curriculum, and they have forced us to change and adapt to their presence. In short, helped by cheap, readily available technology, they have caused our Engineering Curriculum to evolve.

References

1] Tranter et al, 2011. 'Factories of the Future: Advanced Digital Manufacturing in Undergraduate Education'. Paper submitted to ICEE 2011 conference

[2] Katana Robot Arm: http://www.livescience.com/4286-nimble-robot-safe-humans.html (accessed 10/05/11)

[3] Pioneer P3-DX mobile robot: http://www.mobilerobots.com/researchrobots/researchrobots/pioneerp3dx.aspx (accessed 10/05/11)

[4] Surveyor SRV1: http://www.surveyor.com/SRV_setup.html (accessed 10/05/11)

[5] History of Industrial Robots: http://www.used-robots.com/robot-education.php?page=benefits+of+industrial+robots (accessed 10/05/11)

[6] Boston Dynamics 'Big dog': http://www.bostondynamics.com/robot_bigdog.html (accessed 10/05/11)

[7] Shakey Nilsson, 1984] N.J.Nilsson, ed., *Shakey the Robot*, SRI Technical Note no. 323 (1984), SRI, Menlo Park, California.

[8] Stanford Cart, http://www.stanford.edu/~learnest/cart.htm (accessed 10/05/11)

[9] Elephants don't Chess: http://people.csail.mit.edu/brooks/papers/elephants.pdf

[10] [11] [12] Robot Hall of Fame: http://www.robothalloffame.org/r2d2.html (accessed 10/05/11)

[13] Robots will Invade our Lives: http://www.ted.com/talks/rodney_brooks_on_robots.html, (accessed 10/05/11)

[14] No, robot: Japan's elderly fail to welcome their robot overlords http://www.bbc.co.uk/news/business-12347219 (accessed 10/05/11)

[15] Remaking manufacturing with Robots http://fora.tv/2009/05/30/Rodney_Brooks_Remaking_Manufacturing_With_Robotics (accessed 10/05/11)

[16] Robots and the Declining Japanese Population http://www.csa.com/discoveryguides/robots/review.pdf (accessed 10/05/11)

[17] Saed Hussain, 'Autonomous Robot Navigation & Collision Avoidance', submitted as a Final Year Project at University of Central Lancashire