In **School of Robotics** we will be delving into many different aspects of robotics, from history to hardware to maths – all the elements that are the basic building blocks of robotics.

In this first instalment we are going to introduce you to Prof. Peter Corke from the Queensland University of Technology and his Robot Academy. We caught up with Prof. Corke recently to learn more about him and his academy:

ROBOT ACADEMY

WITH PETER CORKE

I grew up in central Victoria, Australia, and was a geeky student—I guess I liked maths, I liked physics, and so went to university to study electrical engineering. For me there was nothing that was remotely as interesting as electrical engineering, so I did that. I got my first job at the university, and almost accidentally fell into robotics.

We were putting together a demonstration for an open day, and I bought this little tiny robot I think that had five stepper motors in it—a tiny armed robot. I wrote some code in FORTRAN running on a mini computer that allowed it to play checkers (draughts). And it was good, it was very popular at the open day, which I'm happy about. Then maybe a year later there was an advertisement in the newspaper: the CSIRO (Commonwealth Scientific and Industrial Research Organization) were looking for a roboticist.

I thought that was pretty cool—so I applied for the job and got it, and I worked for CSIRO for 25 years, in Melbourne initially, doing a lot of work for the manufacturing industry, which is the traditional user of robotics, and then we got into other things, like traffic monitoring systems and sewer inspection systems; we also made food-sorting machines. And then I moved to Brisbane to try and take some of the robotic technology that we'd developed for manufacturing and apply it to mining.

I was pretty naïve when I did this, because I thought, okay, a mine is like a factory—you're sort of moving stuff around, yeah, the machines are a bit bigger—how hard can it be? And so I built a really great team of people up here in Brisbane, and we did a lot of work in the mining industry. We automated big open-pit excavation machines, things called draglines, which can move 100 tons of dirt at a time.

We did some work with underground mining, with machines called load–haul–dump (LHD) units, which are like squat, front-end loader machines that drive around underground and carry ore. We automated those—basically self-driving cars, self-driving vehicles, in the late 90s, that traveled underground at 20 km an hour with hardly any clearance between the vehicle and the walls in the tunnel—and that technology got commercialized, which is pretty nice for a scientist/engineer.

It's nice when your stuff gets into a product. I think that to me personally, that's as good as it gets. Then we looked at other applications. We started doing flying robots in the very late 90s. This was before drones were a thing. So we automated small petrol-powered helicopters and worked on some very early 4-rotor flying robots in the early 2000s. Yeah, I'm amazed at how far drones have come . . . you can just buy them from the shop now and they just work! In the early days it was hard work to keep these machines in the air. The things that we built back then were heavy and didn't work all that well.

Then, in 2010, I got interested in a change of direction and came to university. I became a Professor of Robotics, which is a pretty cool title, and I've been involved in education, and setting up a really major research group here at the Queensland University of Technology (QUT), our particular focus being on robotic vision. We want to find out how to give robots the ability to see, like we have, and particularly how to do hand-like coordination, which is something we really underestimate.

If I want to pick something up off my desk, I just look down and pick it up. I don't know the coordinates of my glasses. I don't know the coordinates of my hand. I pick them up by effectively minimizing the error between where my hands are and where the glasses are. I do it in a very different way in which robots would do it. I'm really interested in "Can robots do it in a way that's more human-like?" Which would make it faster, more graceful, and more reliable—all of the things that robots are not at the moment.

I think the most exciting thing about robotics today is the awesome computing power we've got available to us. When I started doing robotics, a long time ago in the 80s, computers were pretty poor. The first vision system I built, and I was really always interested in very high speed computer vision, I had a 19-inch rack of equipment—all those VME bus cards in there doing image processing. Today my laptop computer can do better and faster! A huge amount of computation is available to us, and that's really important. Because, to me, the perceptual side of robotics is where the biggest challenges are. We can build pretty good robot machines, have been able to for decades. Whether it's a robot arm, whether it's a car, whether it's a flying drone-y thing, the machine's not the problem—it's the smarts, it's the intelligence that is the problem.

My focus is on perception—say looking at a table with a pair of glasses on it, how do I recognize where the pair of glasses are, and how should I pick them up? That's really hard for a computer, and we've struggled with that capability for a very, very long time. People have been doing computer vision research since the 60s. But in the last five years there's been this amazing breakthrough in deep learning. Deep learning has revolutionized computer vision. We can train the deep net with lots of images and I could take a

picture of this table and say, "Where are the glasses?" The computer would say, "There are the glasses" and draw a box around them. That's pretty nice. We say, "Where's the book?" It draws a box right around the book and says, "There's the book." That's pretty good. We have the computing power to be able to run these networks in real time. Training is not real time, however; training still takes days to weeks on racks full of GPUs, but we have that kind of computing power available to us.

People at Google have heaps of this sort of computing power. At the university we have got pretty decent GPU-based computing power as we have lots of students here all working on projects that involve deep learning. I think that the prodigious amount of computing that's available today, the amount of data that we've got for training, and then this new technique, deep learning, which requires you to have lots of computing and lots of data—these are the big exciting things for me.

There are a lot of misconceptions around robotics, and I think the biggest one is that robots "work". We have lots of robots in the lab, and students spend a lot of time here struggling to get the robot systems to work. When we have a demo we are all secretly crossing our fingers and hoping that the robot will work for this demo. Reliability is not nearly as good as it should be. We think we probably lack some sort of craft or discipline in creating robots.

There's all sorts of open-source code out there, which helps, but most robots are still bespoke creations, built by an individual for a particular task. They're brittle, and they don't generalize well to different tasks. We are good at building robots that can do just one thing, like maybe it finds crown-of-thorns starfish and eradicates them or finds weeds in a field—it can do that—but it can't cross over easily, even though the technologies involved are similar.

A robot like you see in the movies—C3PO, R2D2, or Rosie the robot in the Jetsons—can do many things competently and interact well with human beings, but this is still cutting edge and we're quite a long way away from that.

The thing that has surprised me the most in the field of robotics is the rise of self-driving cars. Self-driving cars is a really robotic technology; they're mobile robots that carry people around. That technology has been developing slowly over a long period of time.

I think all of us in the robotics community knew that this would become possible, but we didn't understand the huge public appetite for this technology. We researchers were probably worrying about things like the legalities and what happens if something goes wrong—who's to blame? Actually that doesn't seem to be a showstopper. People are cracking on at a huge pace. I'm sure the laws and the way we do insurance, and all of those issues, will be dealt with because the public really want self-driving cars. People are over traffic. People are over driving. Driving is not a delight anymore. People just want to get from point A to point B. That's what robots do. Robots move themselves or things from point A to point B.

The other surprising thing I'd say would be humanoid robots, which I have for a very long time thought to be rather gimmicky and I couldn't really see the point of them. It probably says more about me than it does about humanoid robots. But watching the way the general public interact with a humanoid robot is fascinating. People become deeply involved with them, deeply engaged, and I think as a roboticist, we can harness that propensity of human beings to connect with something else that looks like a human. If you have emotional cues, the robot can use facial gestures, emotional gestures, and the level of communications they're going to have with a person is just so much deeper. It's a little bit spooky. We've recently started a project in humanoid robots, not so much on the human–robot interaction side, which is very important—we don't have too much skill in that area—our focus right now is giving those robots traditional robotic capability, like being able to navigate around this floor here.

My dream is that within a year, when you come to my lab, you'll be greeted by a humanoid robot, and you say, "Where is Peter?" and it will

know where I am. It will say, "Follow me" and it will take you to me. It'll give lab tours and all sorts of things like that. I don't think that's so far away.

Advice for young people thinking about robotics

There seems to be a lot of activities for young people today that involve robotics, even from primary school, and young people are being encouraged to build robots to do sometimes quite complex tasks. If you look at some of the challenges within, say, the FIRST Robotics League (https://firstaustralia.org/programs/first-robotics-competition/), the students are building seriously sophisticated robots. But I think there comes a point where you have to transition from tinkering, fiddling, creating a robot by experimentation, trial and error, to start engineering. Engineering is a discipline. You have to know a lot about mathematics and physics, and this knowledge reduces the time to come up with solutions, because you use design principles and knowledge that humanity has collected over centuries. You can apply that knowledge to create a robot in a lot less time. So at some point, probably late high school, you're going to make a transition from fiddling and experimentation to using principles, mathematical principles, and principles of physics, to understand how robots work, how they operate, and how to design them.

My advice to young people is to study maths, applied maths, pure maths, physics, and learn how to code. Don't be scared of coding. You're probably going to be learning programming languages like C, C++, Python, the languages that are very, very commonly used in building robots.

I've been interested for a very long time in education and helping people get up to speed with robotics. It is a big field. There's a lot of theory that you need to understand or think about before you can start engineering robot systems. More than 20 years ago I started off with creating some open source code. Originally it was for my own benefit, and I shared it, and lots of people piled in and liked it and used it, so I've been maintaining it for a long period of time. But what I found with creating open source code is that people mostly just complain: "It doesn't do this. It doesn't do that." They haven't read the instructions, and they say it's broken. It's a bit frustrating when that happens.

I know that this code is used by other people for teaching, but the students are not hearing my voice. They're hearing the teacher's voice, and they're using or misusing my code to educate students. So when I came to university, I thought okay, I need to put this right. The students need to be hearing my voice, so I wrote a book, Robotics, Vision and Control: Fundamental Algorithms in Matlab® (Springer Tracts in Advanced Robotics, 2013), and it says everything that I think is important that the students should learn about robotics. It's written for undergraduate engineering level, but you don't have to have a PhD to understand this book. If you're a second to third year engineering student, or computer science student, you should be able to understand this book. It contains tons of code examples.

This is good. Now it's my voice . . . But it isn't exactly because other people use my book and then they teach a class. They pick and choose bits and they're still interpreting what's in my book and presenting it to a class. I thought okay, I'm going to nail this right. They've got to hear my voice and see my face and I'll teach them about robotics. The Robot Academy was born out of that, a big project a few years ago at QUT, where we created lots of video lessons—over 200 video lessons about robotics, particularly robot arms; nothing at the moment about mobile robots; and a lot about simple computer vision and how you tie it in together to create what I call "robotic vision".

They ran as online courses, starting in 2015, and then we reconfigured them recently so they still run as online courses, but an online course has some limitations—they start at the beginning, and take you through step-by-step to an end point. But if you just want to know a thing, just one thing, then you have to wait until the course comes around and starts. Perhaps it starts three times a year, so you have to wait till it comes around; scroll through the stuff to get to what you want to learn. It's not random access. It's not like a Wiki. I thought okay—what I'm going to do is take that same content and create another view of it, and that's what the Robot Academy

Image: Robot Academy

is. The Robot Academy is a gorgeous WordPress front-end that lets you search for the topics, but they're also organized in groups in different ways. It takes you right to the lesson.

If you want to know what a Jacobian is, you can just type Jacobian in the search box. Up will come a few lessons that are related to Jacobians, and away you go. You don't have to wait for the course to come around. I think people who want a refresher, people who've just heard a term and want to know what it means, you can just dive right in there and get it. If you don't understand the fundamental principles behind the Jacobian, well then you can go to an earlier lesson and pick that up and then come back to the Jacobian. You say, "Okay, now I know what a Jacobian is. How do I use it?" Well then you can go to successor lessons. That's what the Academy is. It's all this content that I've been working on for lots of years, told in the way I think it should be told, in a way I think that that's quite engaging and captivating. The content spans a range of technical difficulties. Around 20% requires no maths and would be accessible to the wider public.

We've got content on what robots are and why we need them, and the ethics of robots. Anyone can understand that. But some of the lessons require that you know something about imaginary numbers and some calculus, so they are rated from one to five, and based on what you know, you could say, well this lesson's for me. Or this lesson is not for me.

All the lessons in the Robot Academy are available for free to everybody, wherever you are in the world, and the URL is https://robotacademy.net.au/

Image: Robot Academy

The QUT Robot Academy provides free-to-use undergraduate-level learning resources for robotics and robotic vision.

The content was developed for two 6-week MOOCs that ran in 2015 and 2016, which in turn was based on courses taught at QUT. The MOOC content is now available as individual lessons (over 200 videos, each less than 10 minutes long) or a masterclass (a collection of videos, around 1 hour in duration, previously a MOOC lecture). Unlike a MOOC, all lessons are available all the time. Although targeted at undergraduate-level around 20% of the lessons require no more than general knowledge, and the required knowledge (on a 5-point scale) is indicated for each lesson.

robotacademy.net.au