Using open-source hardware to investigate biological behaviours

A DIY approach aimed at educational institutions

A Thesis submitted to the University of Sussex for
the degree of Master of Philosophy

Harry Richard Kent

August 2019

Declaration

I hereby declare that this thesis has not been and will not be, submitted in whole or in part to another University for the award of any other degree. However, the thesis incorporates to the extent indicated below, material already submitted as part of required coursework for the degree of Bachelor of Science in Biomedical Science which was awarded by the University of Sussex.

The initial DIY microscope was produced as part of the 3rd Year Research Project, and some ideas and images have been reused.

Summary

This MPhil thesis is the result of some time working with the Raspberry Pi, a cheap, credit-card sized computer. This has been used in this project to offer teachers new opportunities and methods to study a curious microscopic organism, the tardigrade, and a more common animal, found everywhere- the ant. Through these short, education based experiments, I have attempted to identify how this hardware could be used in schools to boost cross-curricular teaching and to enhance the work that can be carried out in a school setting, while sticking to a very tight budget. The entire project has been accomplished for less than £200 and has incorporated a wide range of external sensors that provide real, useful data for studying the effect of environment on the behaviour of animals.

This thesis takes the reader through a history of the technology and gives an introduction to the equipment and programmes used throughout, before applying this to two very practical studies which attempt to utilise the hardware to advance our knowledge of animal behaviour.

It is, to my knowledge, the first time a Raspberry Pi has been used in ant behavioural studies, and provides schools and colleges ideas on how to utilise this equipment to look at animals in new ways.

Paper arising from this thesis

Kent, H.R and Bacon, J.P. (2016). Microsco-pi: a novel and inexpensive way of merging biology and IT. *School Science Review* 98 (363). pp. 75-82. ISSN 0036-6811 (Attached as *Appendix 1*)

Acknowledgements

This MPhil thesis is the culmination of over 3 years of work. While we initially thought that this project would be completed in one year, several factors meant that it took (just a little bit) longer than planned. During this time, I have been diagnosed with an autoimmune condition, depression and OCD, which has led to many delays in getting this work done. However, I am glad to be finally submitting this work and grateful to everyone who has made this possible.

Firstly I'd like to thank all the staff at the University of Sussex, particularly those in RSAO and the Alumni relations office. Their support during my intermission has been so appreciated and has made it possible for me to be able to submit this thesis. My thanks also go to the Friends of Sussex University for their incredible financial support that made the first year of this MPhil project possible.

My thanks also go out to the Technology Enhanced Learning department based at Sussex. Without your Innovation Challenge, I'd never have been able to carry on with my undergraduate work during the summer after my graduation, which really kicked off this whole thing.

I also have to thank all the amazing staff and students based in LASI, one of the leading research institutions on the subject of insect behavioural biology, where I was lucky enough to be based during this time. Despite my flaky attendance, you all welcomed me into the lab and were always on hand whenever I had a question about the project. A special thanks goes out to Professor Francis Ratnieks, the UK's only Professor of Apiculture, who kindly agreed to be my co-supervisor and a constant source of great ideas for the project and how to extend it further. Thanks to Tom Butterfield and Alan Gallagher for letting me share an office with you and for all your help whenever I needed it. Thanks to Anthony Bracutti for helping out on data collection, species identification and for pointing out critical flaws with the hardware set up. I'd also like to thank the Life Sci teaching lab technicians for providing me with a near endless supply of tardigrades and for putting up with all my questions. You all do an incredible job to help so many people.

Thanks also to the Widening Participation and Public Outreach teams, both within the school of Life Science and for the University as a whole for giving me opportunities to work with people of all ages and to share a passion for science and higher education.

There's also a special group of graduates who really made final year a truly enjoyable experience working together on our 3rd Year Research Projects: Andy, Dubi, Ella, Katie and Vivek. Thank you for your friendship.

I'd also like to thank everyone who has kindly given me permission to reuse their images, particularly Tomer Czaczkes, Martin and Sabine Mach, and Alasdair Davies for their generous contributions of original images.

I wish to extend my deepest gratitude to Professor Jonathan Bacon, my supervisor, who has been unbelievably patient with me and has never given up on me, even when I wanted to give up on myself. There aren't many Professors who will let a student throw water over them in a lecture (let alone 6 times to date!) but Jonathan's approach to teaching has made him a firm favourite of many students in Life Sciences. His insight into a wide range of biological processes and his genuine interest in emergent technology has made this whole process possible. Jonathan, I truly couldn't have done this without you. Thank you.

Of course, I couldn't get through this without thanking my family. My parents constantly encouraged me to do my best at anything and to push myself through the hard times. I cannot thank them enough for their sacrifices while I was growing up which made it possible for me to attend University, being amongst the first in my family to ever do so.

My biggest thanks of all though has to go to my amazing wife, Gemma, who has put up with the worst of my symptoms, and has been there to gently encourage me to finish this. Without her, there's no way I'd be considering carrying on with this project. Her support has been unending and I can't thank her enough, somehow keeping me going while achieving a First Class degree in Charity Development (one of the first in the world) alongside fundraising for 2 charities and organising our wedding. She is the most dependable person in the world and without her, there would be no thesis to submit. I love you.

Preface

This thesis has evolved over time, initially being an extension of my undergraduate project looking at tardigrade behaviour in changing conditions, into what is really a look at how cheap technology is applicable to studying biological processes as a whole, partly as a result of my emerging interest in the social insects.

As a result, the writing style is quite different to that of a traditional MPhil. It has been written in a similar style to that found in *School Science Review*, an educational journal aimed at school science teachers. My hope is that the material within this thesis will be written in a way that is easily accessible to people who are not specialists in biology or computer science and makes it possible for school staff and students to be able to follow along.

Furthermore, due to the time I had off, some of the material within is no longer up-to-date. As is the case with science and technology, everything is constantly changing, and so some of the software and hardware discussed in this thesis has since been updated. Where possible, I have made a note of this, but having said that, the methods discussed in this paper are still usable. My hope is that someone will be able to use newer versions of the software to improve upon the project and make it even more useful for the teaching of biology and computer science to students.

Finally then, I'd just like to say a huge thank you once again to everyone who has been involved in this project, and I hope you find some way to apply what you have read about within to your own study or work.

Contents

Chapter 1 Introduction and Background

Interdisciplinary research has led to breakthroughs in all sorts of different issues facing the world today. From problems surrounding climate change to international health crises, teams made up of people with skills from a wide range of backgrounds are leading the way to solving some of the biggest challenges that we currently face.

However, while we know that interdisciplinary work is vital, it is not a skill that is really taught in the British schooling system. Subjects are generally taught in isolation, with very little overlap, outside of the statutory requirements to focus on core skills in English, Maths, Science and ICT. Even in the Higher Education system, there has recently been a decrease in the number of students studying for a joint degree course (HESA, 2018). Some Universities are taking steps to enhance the opportunities for students to study modules outside of their chosen subject, such as the “Sussex Choice” programme at the University of Sussex allowing students to pick electives from a wide range of subjects (University of Sussex, 2018).

There are ample opportunities to bridge the gap between subjects in schools and colleges, but so often this does not happen due to demands on teachers time, lack of confidence from teaching staff and the lack of a sharing culture in the school (Savage, 2011). There of course are also budget constraints to consider, and this is becoming a more pressing issue all the time (BBC 2018a, BBC 2018b). This thesis aims to suggest how schools and colleges can use simple, off-the-shelf hardware to increase cross-curricular teaching, to give staff practical advice in the production and development of suitable practical experiments and to see how the same hardware can be adapted to investigate a wide range of living organisms around us in a variety of situations.

When setting out on this journey, I had 2 overarching aims:

1. To bring together accessible hardware and technology to allow interdisciplinary teaching in a school environment and to produce new ideas for a classroom environment that was delivered in a way that is accessible to someone with a non-technical background, like myself.
2. To develop hands-on practical sessions that:
    a. Introduced the Raspberry Pi based microscope into a classroom using possibly unfamiliar, yet extremely interesting, organisms and further develop the microscopy aspects of this project.
    b. Used the hardware to develop a new tool, namely an automated insect tracker which could be used to teach aspects of behavioural biology in a classroom setting; showing the versatility of the hardware purchased for this project.

I have come into this project as a technology enthusiast, with very limited computer science experience, having done very little coding since my GCSEs. However, I have managed to develop the software in here teaching myself using guides available on the internet.

With this in mind, this thesis has been written in an easily accessible style, with the aim of being simple to understand for many people, from students through to teaching staff to allow them to put this into practice at home or in the classroom.

This introductory chapter will briefly describe the history of the Raspberry Pi, a cheap and accessible computer, and the brains behind the project. It will then look at how the Pi has been used in scientific research around the world, across a range of disciplines. After that, I will give a snapshot overview of how scientific equipment is changing with new developments in technology, before finally looking at the animals involved in this study.

Part 1: The Raspberry Pi

The Raspberry Pi was born out of conversations about how few students applying to study computer science at university actually had any experience with coding. A group of computer scientists at the University of Cambridge identified several barriers to young people coding, including the cost of computers and a fear of breaking the computer. In 2006, they started to design a small, cheap computer system that would remove some of these barriers and encourage a new generation of enthusiastic programmers. This computer became the Raspberry Pi (Figure 1.1). The designers then went on to set up the Raspberry Pi Foundation. The Raspberry Pi Foundation is the UK-based charity responsible for the design, production and marketing of the Raspberry Pi computer, and the Foundation aims to see cheap, programmable systems revolutionise how young people interact with computers.

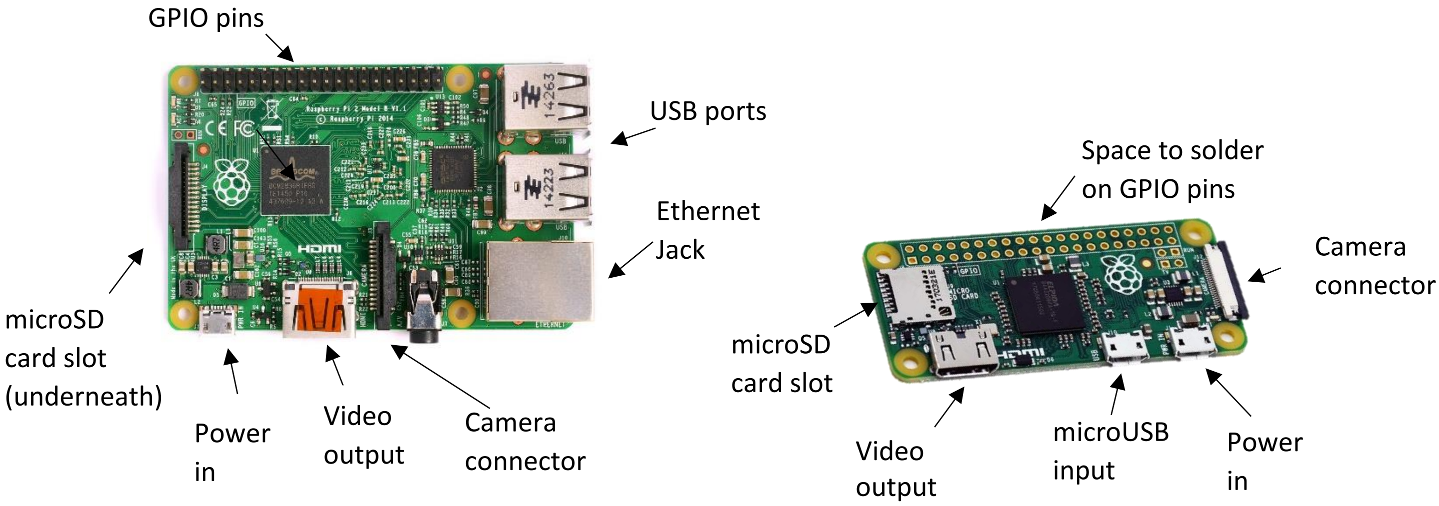


Figure 1.1: The Raspberry Pi 2 Model B board on the left (Multicherry, 2015) and the Raspberry Pi Zero board on the right (SimonWaldherr, 2018) with some key components highlighted.
Images licensed under the Creative Commons Attribution-Share Alike 4.0 International license.

Since 2008, the Raspberry Pi Foundation has produced a range of computers aimed at encouraging people of all ages to get actively involved with coding and electrics. The Raspberry Pi was first released to the public on “Raspberry Pi Day” (the 29th February 2012) and in the 6 years since the initial release, a number of different models have been released. There are 2 key models used throughout this project; the Raspberry Pi 2 Model B and the Raspberry Pi Zero Version 1.3 (shown above in Figure 1.1).

The Pi Zero is about a third of the size of the original Raspberry Pi. This size allows the Raspberry Pi to be integrated into even smaller projects. Figure 1.2 gives a sense of scale of the hardware.

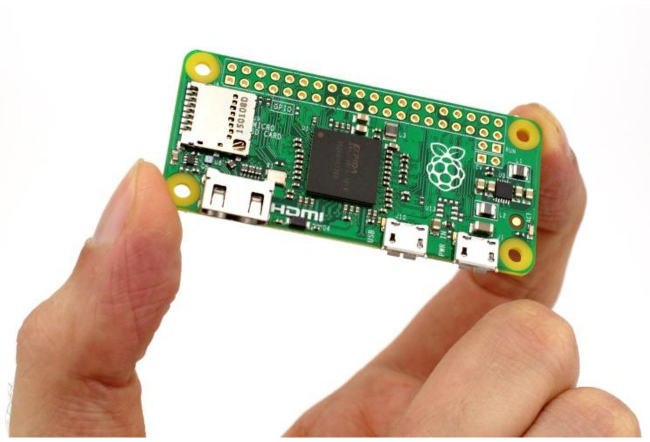


Figure 1.2: The Raspberry Pi Zero measures 6.5cm across and 3.5cm tall (Halfacree, 2015)
Image licensed under the Creative Commons Attribution-Share Alike 2.0 Generic license.

Unlike conventional computer hardware where everything is sealed in a plastic case, the Raspberry Pi board comes completely exposed, and the open nature of the Pi really encourages the user to just have a go, knowing that if anything goes badly wrong, they simply have to format the memory card to get the Pi back to its original state.

The Raspberry Pi is intended to be cheap enough for young people to be able to afford. The Pi Model 2 costs around £30, whereas the Pi Zero can be purchased for £4. Additional equipment is required, including microSD cards, microUSB power supplies, HDMI cables and USB mice and keyboards, but these are things that are easy to obtain and that most homes and schools will likely have some available, or starter kits can be purchased containing everything needed.

Unlike more common personal computers, the Raspberry Pi does not run the Windows operating system. Instead, the Raspberry Pi runs on Linux, an operating system that has been around since the early 1990’s (Torvalds and Diamond, 2001). One of Linux’s defining features is that it is a free, open-source package, which means anyone is able to modify and improve upon its features, unlike the locked down and proprietary Windows and Mac OS systems from Microsoft and Apple Inc. respectively. As a result, there are lots of 'variants' of Linux, including Ubuntu, Debian and Fedora, amongst many others.

Although not widely used on home computers, Linux is used for industrial and research purposes across many sectors. It is also very popular on mobile devices, as the Linux Kernel is the backbone of Google's Android operating system (Hoffman, 2014). Linux is also used on many set-top boxes, in smart TVs, in micro-controllers and a range of other hardware. In recent years Linux has become better known, with companies such as Ubuntu working hard to promote Linux and even large computer manufacturers such as Dell are now starting to offer desktop computers and laptops with Linux pre-installed as opposed to Windows (Dell, 2016). In 2017, Microsoft allowed a version of Ubuntu to be released as an app on its store for Windows 10 devices, meaning you can now run Ubuntu within Windows like any other programme (Microsoft, 2017).

The Raspberry Pi Foundation released a specifically produced distribution, called Raspbian, based on Debian, one of the most popular versions of Linux (Raspbian, 2018). However, the Pi can run a range of other Linux distributions, such as Ubuntu, XMBC, RiscOS and more, and even a special version of Windows 10 made especially for the Raspberry Pi (Microsoft, 2018).

Due to its open nature and easily available hardware, the Raspberry Pi is often used in projects that link the physical and digital worlds, and is the ideal tool to be used to teach young students about how to programme and interact with computers.

The Pi is able to interface with all sorts of external hardware. The Raspberry Pi boards have some exposed pins on their surface, called the General Purposes Input/Output (GPIO) pins (highlighted in Figure 1.1). These pins allow for a huge variety of external hardware to be wired up to the Pi, such as thermometers, buttons, buzzers, LEDs, motors, servos and many more inputs and outputs. The GPIO pins come pre-attached to most Raspberry Pi models (excluding the Pi Zero/Zero W, where the user can easily add them onto the board if they require them).

The GPIO pins can also be used to interface with 'HATs'. A 'HAT' (Hardware Attached on Top) is a small board (generally the same size as the Raspberry Pi) which contains all the necessary hardware to carry out a specific function. For example, a Display Hat will have all the hardware to power a small screen and send the video feed. The HAT boards simply connect to the Pi by being placed onto the GPIO pins (see Figure 1.3).

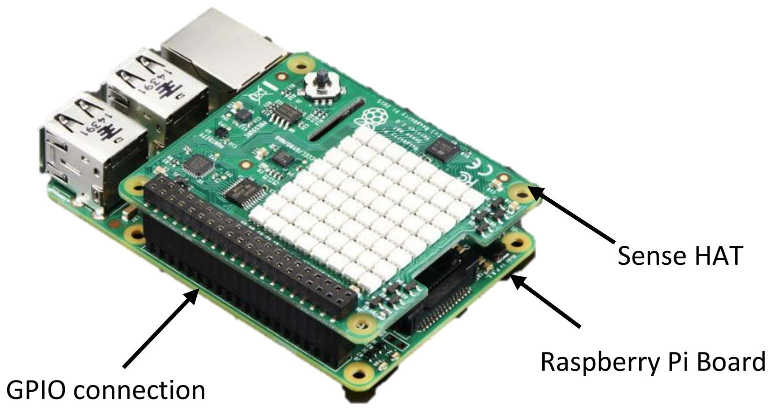


Figure 1.3: The Sense HAT (described later in this chapter) is seen sitting on top of the Raspberry Pi board (Adafruit, 2018a).

Also important for the project described here, the Raspberry Pi boards can all be connected to camera modules. The Pi Foundation (and third parties) have produced cameras specifically designed for the Raspberry Pi; these are discussed further in Chapter 2.

Although the Raspberry Pi was initially intended to be used to promote computer science to students, it is also extremely popular amongst hobbyists. In just over 18 months, a million units were manufactured in the UK, with the Raspberry Pi Foundation estimating about a quarter were in the hands of children, with the vast majority being purchased by adult hobbyists, many of whom were experiencing computer science for the very first time themselves.

4 years after the Pi was launched, the Foundation announced 8 million units had been sold (Raspberry Pi Foundation, 2016), and just over 6 months later, the ten-millionth unit was sold. In 2017, the fourteenth million board was sold (Beta News, 2017), with over ten million of

these having been produced in the UK (Wales Online, 2017). The boards have been used in a wide variety of projects, from DIY video game arcade cabinets, to high altitude ballooning.

Raspberry Pi in Education

The Pi is a brilliant tool for teaching coding. The Pi supports Python, a cross-platform coding language which is really quick to pick up and is well suited to beginners. This is the language that has been used to create the microscope user interface and tracking applications in this project, which are detailed in Chapter 2.

The Pi also runs Scratch (MIT, 2002), an interactive drag and drop programming environment which is commonly used by younger students. It is very simple to use, and the user slots instructions together like a jigsaw to give the programme a function (Figure 1.4). Crucially, Scratch is available for free, and projects can be viewed directly from the Scratch website.

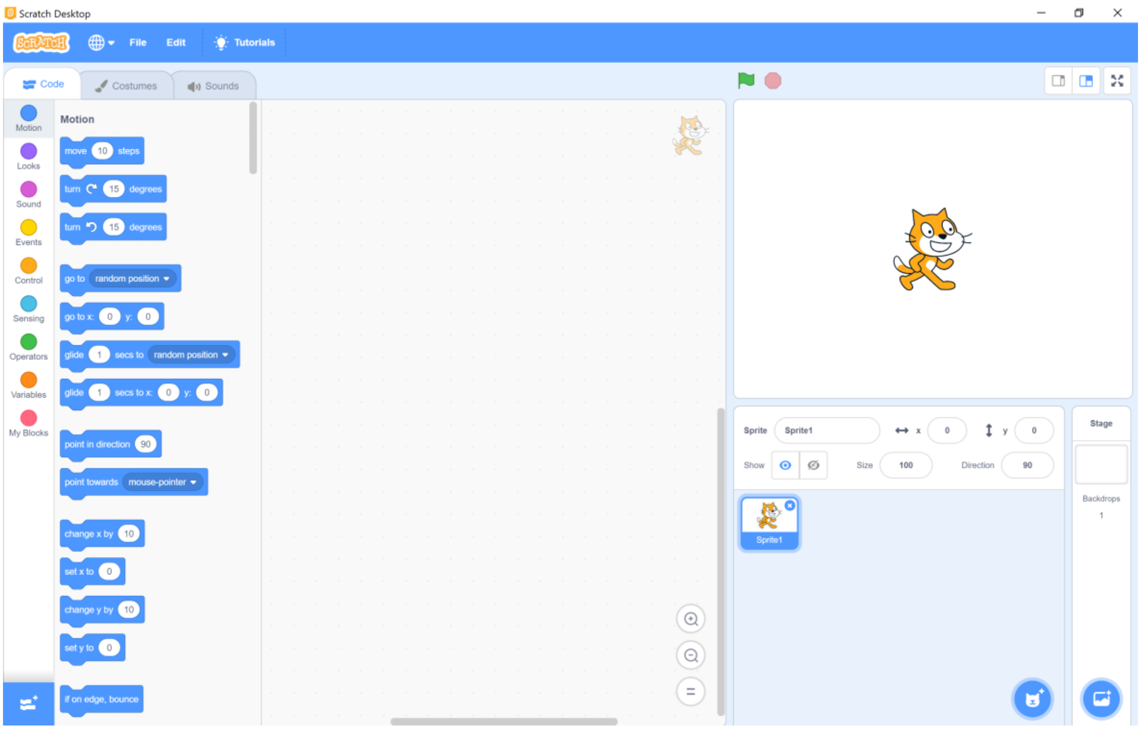


Figure 1.4: Scratch user interface. The blocks in the left pane are the instructions given to the programme, and the window in the upper right shows a live image of the current programme.

The BlueJ Java development environment (BlueJ, 2018) also comes preinstalled, another cross-platform programme for developing Java applications. BlueJ is used to teach first-year Computer Science undergraduates at the University of Sussex.

In summary the Rapsberry Pi is an ideal tool for use in the classroom (and at home for keen students) for a whole range of ages, right from primary school through to university.

In 2015/2016, the Raspberry Pi Foundation ran competitions in schools across Britain to develop programmes for the Raspberry Pi (Astro Pi, 2018). UK astronaut, Major Tim Peake, took some Raspberry Pis to the International Space Station (ISS) with him to gather data. The winning students received data back from the ISS to interpret. These programmes took advantage of the Sense HAT (Figure 1.3 and 1.5), a HAT produced for this competition. The Sense HAT contains an LED matrix to give visual feedback to the user, buttons, a joystick, a gyroscope, a magnetometer, a barometer and a thermometer, meaning a wide range of data could be collected quickly, and there were plenty of inputs and visual feedback for the astronauts, allowing them to interact with the software.

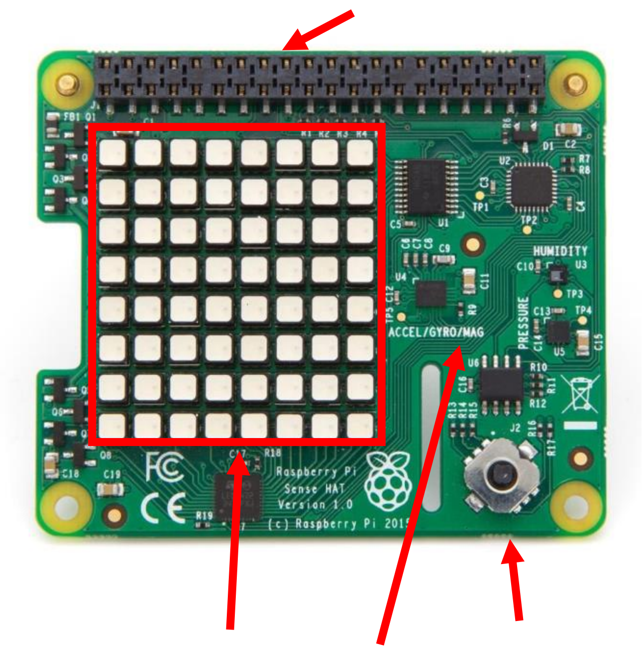


Figure 1.5: The Raspberry PI Sense HAT. The 8x8 LED grid can be programmed to flash all sorts of colours and the joystick can be seen in the bottom right. The other sensors can be seen to the right of the LED matrix, above the joystick. The connection to the board is provided by the GPIO connector at the top (The Pi Hut 2018a).

One of the winning entries was a programme that attempted to detect the presence of crew by monitoring the humidity in the air of the room. Another programme logged orientation, temperature and pressure data from the ISS and visualised it in Minecraft (a popular video game). Some of the other winners came up with some software that took pictures from the ISS, which were later be run through some more software to monitor the health of Earth's vegetation. All of these programmes have been released free to the public so others can see how they work.

There was such a wide range of entries for this competition because the Raspberry Pi has become very popular with schools. They are cheap to purchase, easy to backup and restore and simple for school IT technicians to work with. Google donated 15,000 units to schools across the UK in 2013 in a bid to increase coding lessons on the Raspberry Pi in classrooms (Raspberry Pi Foundation, 2013a).

It has been noted that Britain faces a digital skills shortage (Ecorys UK, 2016), and the new computer science curricula aim to address this (AQA, 2016). Coding is becoming more prominent in the new syllabus and there is a real push for developing these digital skills, from the primary level through to the Higher Education sector. Even the BBC have been involved with increasing digital literacy and released the micro:bit in 2016, a small computer that was given to all Year 7s in the country (and made available to the public to buy), with the aim to encourage more students to take up computer science and develop these key skills (BBC, 2016).

However, it is not just at primary and secondary schools that this skills shortage is being addressed. Recognising the shortfall in students training in engineering and computer science fields, the government provided funding to plug the gap. In September 2016 the University of Sussex launched a new Masters degree, MSc Robotics and Autonomous Systems, which is teaching students from a non-computer science background about small computers, such as the Raspberry Pi, and how they can be used to interface with the real world (University of Sussex 2016a; University of Sussex, 2016b). This new course has been supported by funding from the Higher Education Funding Council for England.

Other universities have also appreciated the importance of the Raspberry Pi. In 2013, the University of Southampton announced that they had produced a supercomputer using 64 Raspberry Pi boards connected to each other, in an attempt to teach students about the principles of supercomputer design and computation (University of Southampton, 2018).

The Pi is also being used in training the next generation of life scientists. In 2013, Barker and colleagues wrote a paper which described the "*4273π Bioinformatics for Biologists*" course run at the University of St. Andrews (Barker *et al.*, 2013). This highly innovative course used a specially produced version of the Raspberry Pi operating system (which has been released freely by the authors) to teach basic principles of bioinformatics to students. By using the Raspberry Pi and giving each student administrator access, students are able to learn about software installation, basic coding and networking principles. These are essential skills for anyone considering a career in bioinformatics, and the University benefits from the low outlay and limited risk of damage to expensive hardware.

Raspberry Pi as a Research Tool

The popularity of the Raspberry Pi has also led to its use in a wide range of published research across multiple disciplines, either as the focal point of the project, or used to innovate current methodology.

The Raspberry Pi has been used to help study life in the oceans, for example as part of a system to record acoustics underwater in Brazil (Caldas-Morgan, Alvarez-Rosario and Rodrigues Padovese, 2015). While not as power efficient as commercially available underwater recorders, the Pi showed promise with its low entry price, ease of use and the way that it could be customised to suit the researchers' needs.

This is not the only published paper which utilised the Raspberry Pi to study marine biology. In 2014, the Raspberry Pi was used as the controller for an underwater stereo camera trap (Williams *et al.*, 2014). This paper described how the Raspberry Pi was used to monitor the environment and to automatically trigger image capture when animals were present in the field of view. Two cameras were synchronised up to allow the researchers to study 3D spatial positioning and gain other details from the subjects' behaviour. This project was designed to be open source so that other researchers could use the design to study marine populations. This method allowed the researchers to obtain good images of the local wildlife without scaring them away by triggering the flash unnecessarily, or by using a remote control vehicle, which also often lead to animals fleeing. In addition, this set-up could be deployed in a range of locations easily and with limited interference to the local environment.

The Raspberry Pi is very popular with researchers studying animals in environments where traditional hardware would make it either very expensive to run studies, or where it would not be feasible to consider such research. The Zoological Society of London (ZSL), a leading charity dedicated to the research and conservation of wildlife with a strong outreach and educational focus, has been involved heavily with the Raspberry Pi and has utilised the small computer for a range of purposes, specifically studying animals in extreme environmental conditions.

In a report published on the Raspberry Pi Blog in 2013, ZSL described a novel set-up which utilised a Raspberry Pi as a processing unit for a satellite connected camera which could monitor the activity of deep-water animals in marine protected areas (Raspberry Pi Foundation, 2013b). This project was a collaboration with DIY science enthusiasts from around the UK, in which the Pi was part of a system which captured images underwater (at a depth of around 50m) and sent them via satellite to the researchers. This system was relatively inexpensive, customisable and utilised parts that could easily be replaced were anything to fail (Fletcher, 2013).

ZSL were also involved in another project in which the cameras were placed in the Artic and used to monitor penguin populations (Figure 1.6a). In this project, images were uploaded to a website, and members of the public were invited to participate in the research and identify eggs and chicks in the images sent via satellite. This project shows the versatility of the Raspberry Pi, and the range of environments it can be used in. Again, the Pi's low power consumption and cost made it a suitable research tool (Raspberry Pi Foundation, 2014).

Finally, the Raspberry Pi Zero was used in a project with ZSL and The Arribada Initiative in 2017 (Raspberry Pi, 2017). In this project, a Raspberry Pi Zero and camera were used to create a camera tag that could be placed on Green Sea turtles and the footage could later be viewed (Figure 1.6b).

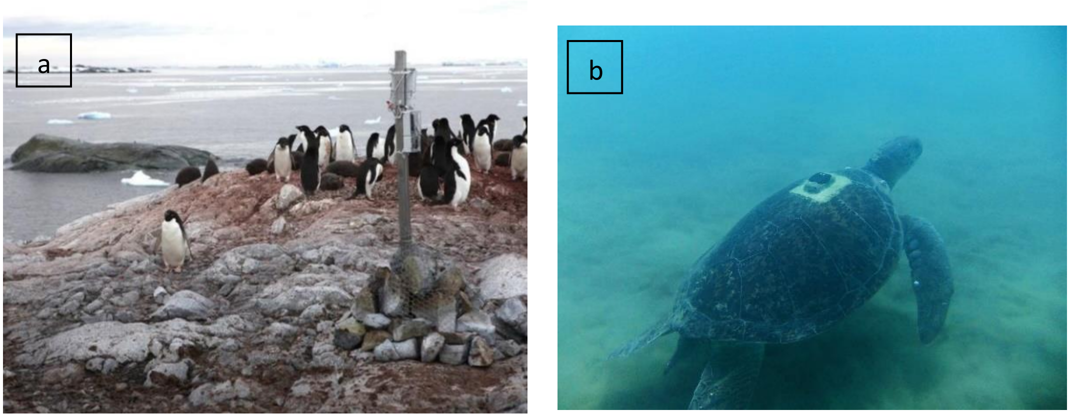


Figure 1.6: Usage of the Raspberry Pi in research and conservation projects. (a) The Penguin observation set up; the Pi is on the pole on the right of the image; (b) Raspberry Pi camera tag on sea turtle (images provided by Alasdair Davies).

Raspberry Pi computers have also been used to study bioinformatics. Generally, when people think about studying and comparing gene sequences, they assume expensive supercomputers are used to provide the computing power necessary for this type of work. However in 2015 a Raspberry Pi was used to compare the protein-coding gene content of *Chlamydia trachomatis* (the bacteria that causes chlamydia infections in humans and can lead to trachoma) and *Protochlamydia amoebophila* (an endosymbiont of amoeba), showing that these cheap computers are perfectly capable of running the required software for such a comparison (Robson and Barker, 2015). This paper also highlights the possibility of using the Raspberry Pi as a tool in rural and under developed areas to study or teach bioinformatics.

OpenSesame (Mathôt, Schreij and Theeuwes, 2012) is an open-source tool used to produce graphical experiments. OpenSesame has been used in a wide range of experiments across multiple disciplines, from psychology to economics. In 2013, Mathôt delivered a talk in which he described getting OpenSesame to run on a Raspberry Pi (Mathôt, 2013). This would allow students and researchers to very quickly and easily design and run experiments using cheap and accessible hardware, which is great for educational purposes and for encouraging more people to study these subjects at all levels.

The above examples show how widely the Raspberry Pi is being used, from primary to tertiary education and from individual hobbyists, right through to some of the world's leading charities and multi-disciplinary research projects. The Raspberry Pi has been an unprecedented success, and is the perfect tool to encourage a fusion of traditional life science and computer science in a school environment.

Part 2: The Microscope

Microscopes are some of the simplest tools that biologists use to investigate the world around us. Since their (disputed) invention by Janssen in the late 1500s, microscopes have led to the discovery of cells, microorganisms and details of cellular function that were just not possible to discover before their invention.

Microscopy is one of the simpler techniques available to schools and colleges, but it is often underutilised (if ever used at all). In a survey I conducted it was found that out of 100 respondents, only about two-thirds had actually ever used a microscope in school, and of those, the vast majority had only used a microscope to observe pre-prepared slides of dead plant material, and had not seen living organisms. Another survey I conducted of 130 first-year life science students at the University of Sussex, showed a similar trend. Only 28% of students had ever seen living organisms under a microscope before coming to university (Kent and Bacon, 2016).

This led to conversations about why this was the case, and more curiously, why were so many people applying to study the Life Sciences at Sussex (including Biology, Zoology, Neuroscience, Medicine, Ecology and Conservation), yet had never used one of the most well established historical techniques to observe living organisms?

In 2013, a report was released which examined how well resourced primary and secondary schools are for effectively teaching science practicals (SCORE, 2013). Only 60% of respondents said that they had enough light microscopes in working order to do group work. Not only that, over half of teachers reported that they did not feel they had adequate funding to effectively teach practical science.

This project aimed to address these points. I wanted to produce a high quality, but low cost digital microscope which teachers could easily use to study living organisms in a classroom or field environment. The low cost could allow for multiple microscopes to be produced at a fraction of the price of conventional light microscopes, and remove some of the costs of maintenance and upkeep, as all parts could be produced or repaired in a school's workshop or classroom.

The original idea for this microscope came from a DIY guide from the online community, Instructables (Yoshinok, 2013). This guide described how to build a simple microscope (Figure 1.7) which uses a smartphone and a lens from a laser pointer to magnify samples.

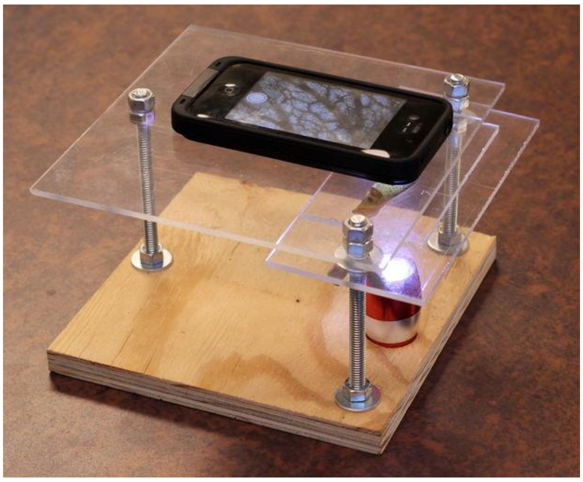


Figure 1.7: Yoshino's DIY microscope (Yoshinok, 2013).

Although this microscope could not replace a light microscope, the magnification achievable is sufficient to clearly see onion cells and multicellular microscopic animals such as tardigrades and nematode worms, while providing students the chance to get hands on with the production of their own lab equipment. The portability of this equipment also opens up brand new opportunities to spend time outside in the field, looking at nature in a new way, without risk to expensive equipment.

When looking at the smartphone microscope, it was clear that there was the potential to also utilise other popular hardware (such as the Raspberry Pi) to open up a range of educational opportunities, and offered a great chance to explore interdisciplinary lessons.

The Raspberry Pi can be programmed by the user to capture footage over a defined time period, as a response to an external stimulus or wirelessly over a network. Furthermore, footage and images can be uploaded quickly and easily to a remote storage site, such as Dropbox, for distribution and back up.

The low cost is also a factor to consider. The Raspberry Pi and all the equipment needed to produce a digital microscope costs far less than most smartphones but can be more versatile. The whole set up also costs far less than the standard teaching microscope, and by following simple online instructions, the hardware can be reutilised for a wide range of purposes, so it can be a very worthwhile investment for a forward thinking school or college.

Currently we are at the start of a revolution in how research groups and teaching institutions are producing their own equipment, using technology like the Raspberry Pi and 3D printing. This whole area is known as 'Open Hardware', where people are able to use modern technology to easily produce their own equipment (examples shown in Figure 1.8), such as desktop centrifuges (CopabX, 2013) DIY pipettes (BadenLab, 2014), realistic anatomy teaching tools (Kanagasuntheram *et al*., *2019*) and more. For a more comprehensive overview of this new area of exciting possibilities, see Chagas and Baden's article in the PLoS One collection, Open Source Toolkit: Hardware (2015) or Pearce's book, Open-Source Lab: How to Build Your Own Hardware and Reduce Research Costs (2013).

Open hardware has the advantage of being more flexible than the commercially purchased option and can easily be repaired or improved upon in-house, but has the drawback of taking time and research to build, as well as the need for the required tools.

Open hardware, like the microscope described in subsequent chapters in this thesis, provides opportunities not only for schools, but also for people in more remote or developing locations. Freely available designs on the internet mean that important tools for teaching and for healthcare can be made *in situ*, as opposed to relying on expensive hardware which may be prohibitively expensive to buy. A more thorough introduction to the hardware of the Raspberry Pi, and the software used in this project will be covered in Chapter 2.

Figure 1.8: Teaching and research tools developed using modern production methods; (a) DIY Centrifuge (CopabX, 2013), (b) DIY pipette (BadenLab, 2014) and (c) 3D printed anatomy teaching equipment (Kanagasuntheram *et al*., 2019).

Part 3: The Animals

The digital microscope was originally designed to be used to study tardigrades, microscopic ecdyzosoans that are well known for their ability to survive extreme conditions. However, due to the flexibility of the hardware, it was later adapted to study ants and their behaviour in response to environmental stimuli.

Tardigrades

Tardigrades are fascinating microscopic animals. There is a long history of research into tardigrades, stretching back to the 1700's. There has been some disagreement (referred to in Kinchin, 1994) as to who originally discovered the tardigrade, but their discovery is most frequently attributed to Goeze, a German pastor who translated and revised Bonnet's book, *Traite de'Insectology* (Bonnet, 1773). While producing this translation, Goeze saw a small 8 legged creature, which he named "*kleiner Wasserbär*" (little water bear). The first illustration (Figure 1.9) of the tardigrade comes from Goeze's translation of Bonnet's book.

Figure 1.9: Goeze's illustration of "*kleiner Wasserbär*".

A few years later, Spallanzani coined the term '*Tardigrada*' (slow stepper) (Bordenstein, 2017). Since then, over 1150 species have been identified (Degma, Bertolani and Guidetti, 2018; Degma, Bertolani and Guidetti, *no date*). Species can be identified through a range of morphological clues and keys (Morgan and King, 1976) or now, more commonly through DNA sequencing (Cesari *et al.*, 2009). The tardigrades are generally grouped into three classes; the Eutardigrada, the Heterotardigrada and the Mesotardigrada (although it is worth pointing out that there are no known extant members of the Mesotardigrada).

Morphologically tardigrades are very distinctive animals (Figure 1.10). They have 4 pairs of limbs, with distal claws. The number, shape, length and width of claws are useful for identifying the species. The rear limbs seem to move less, but are used to grip onto substrates. The head of the organism contains the eyespots (although this varies by species) and the feeding apparatus, which can also be used to help identify species.

Figure 1.10: A photograph of a tardigrade (*Hypsibius dujardini*) captured on a DLSR down a microscope at 200x magnification. In this image, the limbs, eyespots, feeding apparatus and digested food are all clearly visible.

Tardigrades, like insects, crustaceans and nematode worms, are members of the Ecdysozoa, a superphylum which incorporates many phyla including the arthropods (Telford *et al.*, 2008). Like other ecdysozoans, tardigrades grow by shedding their cuticle, a process called ecdysis (Figure 1.11). Tardigrades deposit their eggs into these cuticles, which act as protective environments (Figure 1.12). Some species of tardigrades have 'plates' on their dorsal cuticle, while others do not (Figure 1.13).

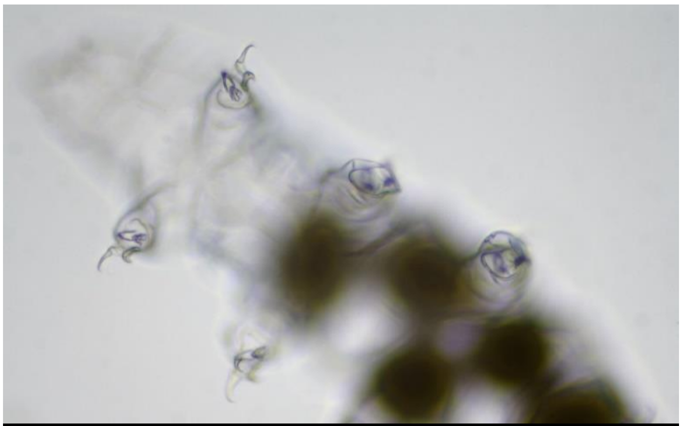


Figure 1.11: A close up of a cuticle showing the details of the claws. (Specious Reasons, 2012). *Image licensed under Creative Commons Attribution-Non-commercial 2.0 license.*

Figure 1.12: Darkfield image showing tardigrade egg clutch within the shed cuticle of a tardigrade. This individual laid 14 eggs in one clutch. (Image provided by Mach, 2019).

Figure 1.13: On the left, an “armored” or “plated” tardigrade of the genus *Echiniscus* (Mach, 2002). On the right, *Acutuncus antarcticus*, the Antarctic tardigrade (Tsujimoto, Imura and Kanda, 2016).

Not only morphologically, but genetically, tardigrades are very interesting organisms. A recent paper (Smith *et al.*, 2016) compares molecular identities of the segments of a range of arthropods, and concludes that tardigrades appear to be homologous to just the head region of arthropods, and suggests that tardigrades lost several hox genes early on in their development.

Furthermore, tardigrades are eutelic organisms (Kinchin, 1994). This means that mature tardigrades have a fixed number of cells (dependant on species). At one point, Sydney Brenner considered using tardigrades as a model organism for his research into cell division/death, before choosing the nematode worm instead (Gabriel *et al.*, 2007).

There has been some controversy about the genetic makeup of tardigrades. A paper was published which suggested that *Hypsibius dujardini*, one of the most well studied tardigrades, showed a very large percentage of foreign DNA, derived from a range of organisms, including plants, fungi and bacteria, taken up via a process known as horizontal gene transfer. This foreign acquired DNA made up approximately one-sixth of the tardigrades genome (Boothby *et al.*, 2015).

However, this was soon refuted, when another paper suggested that the effects of horizontal gene transfer in tardigrades were very minimal, and in fact, the results of Boothby *et al*. were probably due to contamination from ingested organisms. Their draft genome of *H. dujardini* suggested that only 1-2% of the genes were incorporated as a result of horizontal gene transfer (Koutsovoulos *et al.*, 2016).

Research into the DNA repair mechanisms of tardigrades have also revealed some unexpected discoveries. Tardigrades show rapid DNA repair in response to UVC radiation and that when in the anhydrobiotic state, DNA damage was effectively prevented from accumulating (Horikawa *et al.*, 2013). Furthermore, Hashimoto *et al*. (2016) showed that a protein unique to tardigrades (Dsup) is actually able to protect cultured human cells from X-Ray induced DNA damage. It is thought that Dsup could be the first of many discoveries yet to be made from studying the genome of the tardigrade that could have a role in protecting DNA. As a result of their rapid and reliable DNA repair mechanism and their unique proteins that confer additional protection against all sorts of damage, tardigrades are some of the hardiest creatures known to exist; they are able to endure almost any environmental condition.

The tardigrades' most well-known ability is being able to survive dehydration, via anhydrobiosis. Rebecchi et al. published a review about how anhydriobiosis impacts tardigrades with respect to extreme conditions (2007). Subsequently, Welnicz *et al.* (2011) produced a review of the last decade of research into anhydrobiosis.

When in this ametabolic state, known as the tun form, tardigrades are able to survive extreme conditions. There are differing theories about how the tardigrade is able to survive such conditions, and there appears to be some divergence within the species. Some tardigrades produce a disaccharide, trehalose, which is a protective substance that helps prevent dehydration damage when they move into their tun form in a controlled manner. However, trehalose was not found to be produced in experiments on several other species of tardigrade (Hengherr *et al.*, 2008).

Pigon and Weglarska (1955) showed that a tardigrade in an anhydrobiotic state showed dramatically reduced oxygen consumption. Beisser *et al.* (2012) used Gas Chromatography–Mass Spectrometry (GC-MS) to identify some of the changes in metabolites in the active and anhydrobiotic states of the tardigrade (shown in Figure 1.14).

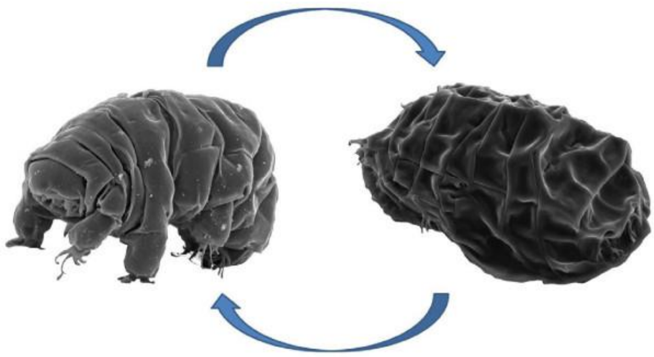


Figure 1.14: A Scanning Electron Micrograph of the active (left) and tun (right) state of the tardigrade *Milnesium tardigradum.* These two states are not shown to scale (Beisser *et al.*, 2012).

Tardigrades are able to survive high levels of radiation (Horikawa *et al.*, 2006). It was shown that they could survive high doses of gamma and ionising radiation, although doses above 1000Gy made the animals sterile. To put this in human context, 1Gy would cause radiation sickness, whereas 1000Gy would lead to death within the hour (MedLine Plus, 2018).

Another extreme survival technique possessed by tardigrades is their ability to survive high pressure. Seki and Toyoshima found that, in their tun state, two different species of tardigrades (one from the Heterotardigrada and one from the Eutardigrada) were able to survive being subjected to 600MPa of pressure, about 6 times that found at the Mariana Trench, the deepest point of the ocean, around 7 miles down (Seki and Toyoshima, 1998).

In addition, tardigrades are able to survive extreme temperatures. Many tardigrades species survive when exposed to temperatures above 80°C, and one species was shown to even survive temperatures in excess of 100°C (Hengherr *et al.*, 2009). Freezing also does not seem to affect them. Sømme and Meier (1995) showed that frozen tardigrades could be successfully rehydrated after 8 years of being frozen. In 2015, a team from Japan thawed a sample of Antarctic moss that had been kept in storage for 30 years, and were able to get a tardigrade to successfully reproduce after recovering from the freezing and thawing process (Tsujimoto, Imura and Kanda, 2016).

One of the most amazing observations however is that tardigrades are able to survive being exposed to the vacuum of space and on return to Earth they were healthy and able to reproduce (Jönsson *et al.*, 2008). Until recently, tardigrades were the only living thing known to be able to survive this extreme environment, but some species of algae have subsequently been shown to survive exposure to space on the outside of the International Space Station (Fraunhofer, 2017).

Tardigrades have a very interesting life cycle, especially because of their ability to utilize cryptobiosis. Suzuki (2003) reared a culture of *Milnesium tardigradum*, and monitored their growth and reproductive rates. The longest living individual had a life span of 58 days. However, in non-ideal conditions, with the ability to enter diapause, tardigrade life spans can be substantially increased. Because tardigrades in the wild generally have intermittent periods of activity, it is important for them to be able to reproduce quickly and to have large numbers of offspring per cycle, with a mean clutch size of 6.9 eggs in Suzuki’s study.

These unique abilities make tardigrades truly fascinating, and a brilliant animal to study as they bring together many different aspects of the school science curriculum.

One of the most remarkable (and useful) things about tardigrades, considering their very unique biology, is the fact that they can be found almost anywhere. Although many papers make comments about finding tardigrades in extreme environments, they can be found very easily in pretty much any piece of moss, making them ideal for schools and colleges as they can be collected from the school grounds.

After identifying a clear need to produce an inexpensive, DIY approach on limited classroom resources, and finding an ideal test species to study the effectiveness of the microscope, I then set about designing a practical investigation that could be carried out in a school classroom, and some possible extensions to this that would stretch and challenge learners. This work is described in Chapter 3.

Ants

After seeing how flexible the Raspberry Pi could be, I wondered if there were other animals that could be used to highlight its effectiveness in the classroom. While writing this thesis, I was based in the Laboratory of Apiculture and Social Insects, and decided ants could be an ideal test species. Ants are members of the Hymenoptera, an order containing the wasps, bees, sawflies and ants, within the phylum Arthropoda. Within the Hymenoptera, a wide variety of lifestyles are found, from solitary wasps to massive ant colonies. Some Hymenoptera are parasitic, such as the parasitoid wasp, *Ampulex compressa* (Herzner *et al.*, 2013) while others are not. The wide range of lifestyles lead to some very interesting behavioural adaptations.

Their physical strength relative to their size is well known by the general public, but probably vastly underestimated by many (Nguyen, Lilly and Castro, 2014). Students may be aware of the fact that ants live in large colonies, and may have some idea about the reproductive role of the queen ant and the role of the worker ants (Figure 1.15).

Figure 1.15: *Lasius niger* queen (centre), workers, larvae (left) and eggs (centre). (Pan Weterynarz, 2010).

Within the colony usually only one individual will reproduce, the queen ant. However, this one ant can produce thousands (sometimes even millions) of offspring, and workers will care for the queen's offspring and the queen herself. Workers will rear the brood by bringing food and water to the queen, and feed the larvae as they develop, but can be prevented from reproducing via the production of pheromones by the queen ant. This reproductive regulating pheromone was recently discovered in *Lasius niger* (Holman *et al.*, 2010). The queen (and other workers) may also conduct policing (Kikuta and Tsuji, 1999; D'Ettorre, Heinze and Ratnieks, 2004; Trettin *et al.*, 2011) and destroy eggs laid by worker ants.

While in general only one ant in a colony can reproduce (a monogynous colony), this is not always the case. Some species, such as *Pachycondyla inversa* and *Pachycondyla villosa* are polygynous, which means more than one mated queens found a nest collectively and are able to lay eggs (Kellner *et al.*, 2007).

Some colonies do not have a queen ant in a traditional sense. In some queenless species, such as *Dinoponera quadriceps*, multiple individuals are capable of reproducing at the same time but the most dominant individual, the alpha (or gamergate), prevents the other workers from reproducing by displays of aggression, physically preventing egg laying or by eating eggs produced by other members of the colony (Monnin and Ratnieks, 2001).

Because of this reproductive division of labour, ant colonies can be very large indeed. While some species will have colonies consisting of just tens of members, other species, like the highly invasive Argentine ant have “supercolonies” composed of millions of workers (Giraud, Pedersen and Keller, 2002).

Ants have shown all sorts of interesting behaviours across a range of species. Ants have been shown to self-medicate (Bos *et al.*, 2015), farm effectively (Ariniello, 1999), rear animals as a food source (Schneider *et al.*, 2013), teach individuals the best route to a new nest site (Franks and Richardson, 2006) and take slaves (Gladstone, 1981). This range of behaviours make ants an ideal study subject to show the wide range of diversity, even in animals that many people often view as just being some sort of pest.

One of the keys to ants’ ubiquity is their ability to navigate effectively. It is common knowledge that ants will follow each other to a food source. This ‘march’ has served as the inspiration for a range of scenes in movies or popular culture. Different ants utilise different methods for navigating effectively. Some ants rely on mental maps to navigate successfully, utilising a strategy called landmark guidance (Collett *et al.*, 1992; Wehner, Michel and Antonsen, 1996; Narendra, Gourmaud and Zeil, 2013; Graham and Mangan, 2015). However, this visual memory is not perfect; it has been shown that some ants struggle to navigate at night, as visual guides are obviously harder to see in the dark (Narendra, Reid and Raderschall, 2013).

Some ants (and other hymenoptera) utilise path integration (Müller and Wehner, 1988), another clever method used to allow foragers to go out and explore an area, yet return directly to the nest. Path integration relies on internal clues to the ant, such as number of steps, and turns made to allow an ant to return directly to its starting point, regardless of the outward route. Path integration has been studied in the desert ant (Wehner, 2003), and also in the honeybee (Srinivasan, 2015), and a chapter summarising path integration in insects has been produced by Wehner and Srinivasan (2003).

Finally, some ants, such as *Themnothorax albipennis*, have been shown to learn routes via a process known as tandem running (shown in Figure 1.16) in which a forager will return to its old nest and guide a nest mate to a new nest site (Franks and Richardson, 2006*)*. This process is highly effective and is quickly reinforced by repeated trips.

Research by Franklin *et al.* (2011) suggests that tandem running relies on visual, tactile and chemical cues, but that the ants are capable of both leading and following a run, even with visual impairments. Tandem running is thought to be the first time that a non-human animal has been observed formally teaching another member of its species.

Figure 1.16: A ‘teacher’ ant and ‘student’ ant engaging in tandem running. The student (white paint) keeps in physical contact with the teacher ant (red paint) via the antennae (Franks and Richardson, 2006).

However, ants’ navigational expertise is not based solely on their visual abilities or being taught a route. Ants also rely on chemical cues to help them navigate. Ants are known to deposit pheromones in response to their environment. Pheromones are generally fairly volatile chemicals that are released by an animal which cause or influence some sort of behaviour in another animal (generally of the same species). For example, ants may leave a deposition of pheromone having found a good food source or as a reaction to a threatening situation.

Ants deposit pheromones by dropping their gaster (the rear section of the abdomen) to the surface that they are walking on. In some species, such as *Lasius niger*, this is a very deliberate action, and by close observation, the ant can be seen to stop walking, and touch its gaster to the substrate (shown in Figure 1.17). In other species, particularly those with a stinger, the ant may leave a deposition by dragging the stinger along the floor.

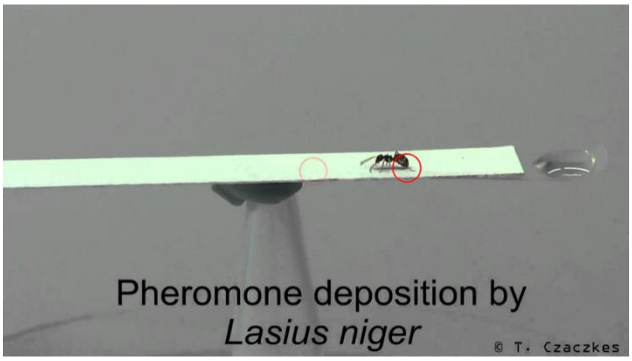


Figure 1.17: A *Lasius niger* ant drops its gaster to the surface to deposit pheromone. This image is a still from a Youtube video showing the distinctive physical action carried out by *L. niger* to deposit pheromone after finding a food source (Czaczkes, 2015).

Many species of ants are known to deposit a pheromone trail on their return to the nest after finding food so that other foragers can be quickly recruited to the food source. Trail pheromones can act as a form of social memory, and subsequent foragers can utilise this social memory to reinforce behaviour, or switch their behaviour to act in a more effective and profitable way for the colony as a whole.

However, while these trails can help guide individuals to a good food source, they can cause potential issues. The trail can be reinforced by repeated visits, and this could lead to a positive loop, producing a very strong attractive signal, regardless of the presence or absence of the originally attractive food source. This could runaway and become unsustainable, with ants not examining other potential options (this is known as an information cascade) but as trail pheromones are generally volatile chemicals (and therefore fairly short lived), this usually is not a problem (Giraldeau, Valone and Templeton, 2002).

Navigational behaviours based on trail pheromones are varied and have been studied in many papers. A concise review of trail pheromone and how it integrates into social colony behaviour was published by Czaczkes, Gruter and Ratnieks (2015).

I was particularly curious about how ants could be used as a study species in secondary education to teach students about behavioural biology, and how to integrate some computer science skills into a biological context. For many children, ants are some of the first animals they encounter in their natural environment. Children are often curious about ants, and they are easily accessible to schools or colleges, with common species such as *Lasius flavus* (yellow meadow ants) and *Lasius niger* (black pavement ant) found in many habitats.

When thinking about how best to demonstrate the use of this hardware with ants as a test species, a range of ideas, including Quick Response (QR) codes and microchipping ants, were initially considered.

Previous insect tracking studies have revealed interesting trends in animal behaviour. For example, a paper by Mersch, Crespi and Keller (2013) used QR codes attached to individual ants (Figure 1.18a) to monitor the spatial and temporal distribution of a huge number of individuals and recorded over 9 million interactions over 41 days. These QR codes were detected by a camera attached to a computer able to decode the interactions in near real time. This study showed patterns in the behavioural roles of ants, and how these changed as they aged.

Another study by Robinson *et al.* (2009) used miniature radio frequency identification (RFID) tags attached to ants (Figure 1.18b) to monitor the behaviour of the ants and their interactions with each other and potential nest sites. This study found that ants can select a new nest site without having to sample all of the potential new locations. As a result of choices made by individual ants, behaviour emerges at colony level.

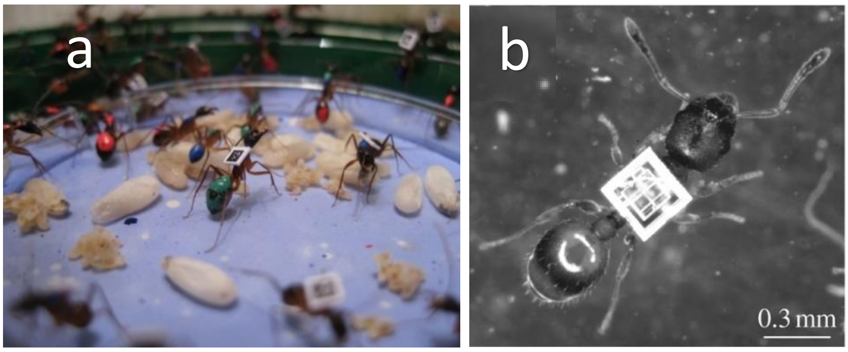


Figure 1.18: (a) QR codes attached to the ants in the study by Mersch, Cerspi and Keller, 2013 (Yong, 2013); and (b) *T. albinpennis* with a small RFID chip attached to its back (Robinson *et al*, 2009).

However, as interesting as replicating these studies would be, they are not feasible in a school setting. I did make some progress on getting the Raspberry Pi to identify QR codes, and designed a theoretical approach to studying ant behaviour using camera traps. The initial trial produced some positive indication that this could work, but this lost the educational focus somewhat (and relied on large tropical ants, such as *Dinoponera quadriceps*, shown in Figure 1.19 below). While very interesting to study, *D. quadriceps* both bites and stings, and needs to be kept in warm and humid conditions, making it an unsuitable choice for use in schools. This work could be picked up again at a future time.

Figure 1.19: *Dinoponera quadriceps*, a queenless species of ant native to Brazil, is about 5 – 10 times the size of most British ants. (Specimen FMNHINS0000050367 from antweb.org; Westrich, 2015)

Instead, thinking about a straight forward approach that could be used in schools, I decided to focus on ant decision making in response to chemicals artificially applied to a controlled environment. The Raspberry Pi is used to observe an individual ant and track its motion when exposed to a chemical stimulus. I was curious to see if individual responses were similar across a number of workers taken from a colony, and if I could establish a clean choice paradigm that could easily and reliably be used in the classroom. A more detailed description of the background research and the methodology, as well as the results, of this study can be found in Chapter 4.

Finally, Chapter 5 contains a summary of the work completed, and highlights some possible directions to take to continue the work produced in this thesis.

Chapter 2 The Raspberry Pi and the DIY Microscope

This project revolves around the Raspberry Pi, and how the flexibility of the hardware and the software can be applied to investigating animal behaviour. The Raspberry Pi is designed to be easily accessible to beginners (with no background in computer science), making it ideal for use in a teaching environment.

To ensure that this would be as helpful as possible, during the drafting of these instructions, I approached a range of people from different backgrounds and asked for their feedback. Importantly, until my undergraduate project I had never used a Raspberry Pi for anything, and even in that project I only used the Raspberry Pi to take images over a 10 minute period and upload them to an online storage site. Therefore, I approached this project as an enthusiastic novice. I have used some of the feedback received to modify the steps within this chapter.

User Group	Prior Experience	Observations
Life Science Graduate	- No computer science experience - Previously involved in animal observation studies	- Was really interested in the prospect of using a small device that could be automated for monitoring subjects. - Felt instructions were at a suitable level.
Charity Development Graduate	- No computer science experience - No animal observation experience	- Found initial draft of some of the instructions to be confusing; assumed some basic knowledge that wasn't made clear.
Computer Science Graduate	- Degree in Computing for Digital Media - 2 years experience in software development - Never studied biological science beyond GCSE.	- Felt that the original descriptions of some computing principles went too deep for the average user and didn't make clear their benefits. - Found the overlap between computer science and biology interesting, especially from a marketing aspect.
School Science Teacher	Limited computer coding experience from GCSEs	- Felt the instructions were at a suitable level in general. - Was unsure if students would follow them correctly. - Liked the format and style.

Table 2.1: A table showing user groups consulted for the development of these instructions.

The Raspberry Pi Foundation has produced many different models of the Raspberry Pi board over the years. All of the boards run on the same operating system and keep ease of access at the core of their design, but there are differences in processing power, connectivity and the memory available to the user. The costs of the board also vary, from £4 for the Pi Zero up to around £30 for the more powerful models. The differences between the models available (as of August 2018) are shown in Table 2.2.

This chapter will be split into 4 parts. The first will describe the general set up of the Raspberry Pi, showing the reader how to install the Operating System and the connected components. The second part will give an overview of some of the software which comes preinstalled on the Raspberry Pi and explains how to install additional programmes which have been used as part of this project. Some additional hardware has been used for the animal studies, and this will be discussed in the third part, before the final part briefly discusses the design and examples of use of the microscope itself.

Model	A	A+	B	B+	2	3B	3B+	Zero	Zero W	Zero WH
Release Date	February 2013	November 2014	February 2012	July 2014	February 2015	February 2016	February 2018	November 2015 / June 2016 (Revision 1.3)	February 2016	January 2018
CPU	700 MHz ARM11 ARM1176JZF-S	700 MHz ARM11 ARM1176JZF-S	700 MHz ARM11 ARM1176JZF-S	700 MHz ARM11 ARM1176JZF-S	900 MHz 32-bit quad-core ARM Cortex-A7	1.2 GHz 64-bit quad-core ARM Cortex-A53	1.4GHz 64-bit quad-core ARM Cortex-A53 CPU	1 GHz ARM1176JZF-S single-core	1 GHz ARM1176JZF-S single-core	1 GHz ARM1176JZF-S single-core
RAM	256MB	512MB	256 / 512MB	512MB	1 GB	1 GB	1 GB	512 MB	512 MB	512 MB
Connections	– 1 x USB 2.0 – 26 GPIO – HDMI – Composite Video – Camera (CSI) – Display (DSI) – Analog Audio 3.5mm jack – SD card slot	– 1 x USB 2.0 – 40 GPIO – HDMI – Composite Video – Camera (CSI) – Display (DSI) – Analog Audio 3.5mm jack – microSD card slot	– 2x USB 2.0 – 26 GPIO – HDMI – Composite Video – Camera (CSI) – Display (DSI) Analog Audio 3.5mm jack – SD card slot	– 4x USB 2.0 – 40 GPIO – HDMI – Composite Video – Camera (CSI) – Display (DSI) – Analog Audio 3.5mm jack – microSD card slot	– 4 x USB 2.0 – 40 GPIO – HDMI – Composite Video – Camera (CSI) – Display (DSI) – Analog Audio 3.5mm jack – microSD card slot	– 4 x USB 2.0 – 40 GPIO – HDMI – Composite Video – Camera (CSI) – Display (DSI) – Analog Audio 3.5mm jack – microSD card slot	– 4 x USB 2.0 – 40 GPIO – HDMI – Composite Video – Camera (CSI) – Display (DSI) – Analog Audio 3.5mm jack – microSD card slot	– 1x microUSB – Mini-HDMI microSD card slot – Camera Connector (only present on Revision 1.3)	– 1x microUSB – Mini-HDMI microSD card slot – Camera Connector (only present on Revision 1.3)	– 1x microUSB – 40 GPIO – Mini-HDMI microSD card slot – Camera Connector
On board Internet Connectivity	– None	– None	– Ethernet Port	– Ethernet Port	– Ethernet	– Ethernet – 802.11n Wireless – Bluetooth 4.1	– Ethernet – 802.11ac Wireless – Bluetooth 4.1	– None	– 802.11n Wireless – Bluetooth 4.0	– 802.11n Wireless – Bluetooth 4.0
Cost at Launch (USD)	– 25	– 20	– 35	– 35	– 35	– 35	– 35	– 5	– 10	– 18.25

Table 2.2: A table detailing the specifications of the key consumer variants of the Raspberry Pi computer as of August 2018.

Part 1 – Initial Set Up

For the project, a Raspberry Pi Model 2 B was used, but the software was also tested successfully on a Raspberry Pi Zero. While the steps are generally the same for all versions of the Raspberry Pi, there may be some variations in positions of hardware connections or steps to use various peripherals.

When a user buys a Raspberry Pi, they receive the board itself. The user needs to supply some additional hardware to get the Raspberry Pi functioning. These are detailed below in Box One.

Box One: Equipment List

- Raspberry Pi Model 2 B
- microUSB power cable (5v, 2.1a)
- HDMI cable
- HDMI capable monitor (or HDMI -> VGA adaptor / HDMI -> DVI adaptor)
- USB Keyboard
- USB mouse
- USB Wi-Fi dongle (or Ethernet cable, not tested in this project)
- microSD card (minimum 8GB, 32GB used here)
- Official Raspberry Pi Camera Board

Step 1 – Installing the Operating System

The first step is to get the Operating System (OS). As mentioned in Chapter 1, the Raspberry Pi can run several variants of Linux. For this study, I chose to use Raspbian, the official operating system of the Raspberry Pi Foundation (Raspbian, 2018). This is based upon Debian, one of the most well known Linux variants, and carries across some of its conventions and software. I used Raspbian Jessie (2015-09-28) for this project.

Please note that since the project was started a substantial update to Raspbian has been released, completely revamping the desktop's graphics and appearance. Although nothing has changed too drastically in how the OS works, there may be some changes between steps given in this Chapter and how it now works. No software used in this project has been tested to work on the new OS version.

On most computers or laptops a consumer buys, the OS comes pre-installed and the user just has to go through a set-up process. However, with the Raspberry Pi, the user must download the OS from the Raspberry Pi website on another device before 'flashing' it to the memory card using a programme such as 'Win32DiskImager' (see Appendix 2). When an Operating System is 'flashed' to an SD card, it is written onto the card in a way that the hardware recognizes it to contain the instructions necessary for the hardware to work. The Raspberry Pi Foundation has produced guides for users of all the common platforms to flash the OS to a microSD card (Raspberry Pi Foundation, 2018a).

If the user prefers, pre-flashed microSD cards can be purchased at additional cost with the OS all ready to go, saving the end user time. Once the OS is on the microSD card, it can be inserted into the Raspberry Pi.

Step 2 – Attach Peripherals

Once the user has the OS ready, they can begin to hook up the cables to their ports on the board; the HDMI cable, the USB keyboard and mouse and the Wi-Fi dongle (or ethernet cable) all use standard connectors. The HDMI cable should be plugged into the monitor. If there is no HDMI connection on the user's monitor, a HDMI to VGA or HDMI to DVI adaptor could be used.

It is important to note that the Raspberry Pi can be used without a monitor attached (known as being 'headless') and the board can be controlled over the internet, opening up new possibilities for projects where having a monitor present is not possible or desirable (such as a security camera or as a motion detector), or to calibrate external sensors.

Before the user connects the power cable, they should attach the camera to the Raspberry Pi. The camera port is labelled on the face of the board (on some models it is labelled CSI port). On the Pi that is used in this project, the camera port is located near to the Ethernet cable connection point (shown in Figure 2.1). The plastic can be gently prised upwards and the flexible ribbon cable can be inserted. The silver connections on the camera cable should be inserted facing away from the Ethernet port and toward the HDMI port, with the blue indicator facing the 3.5mm audio out port.

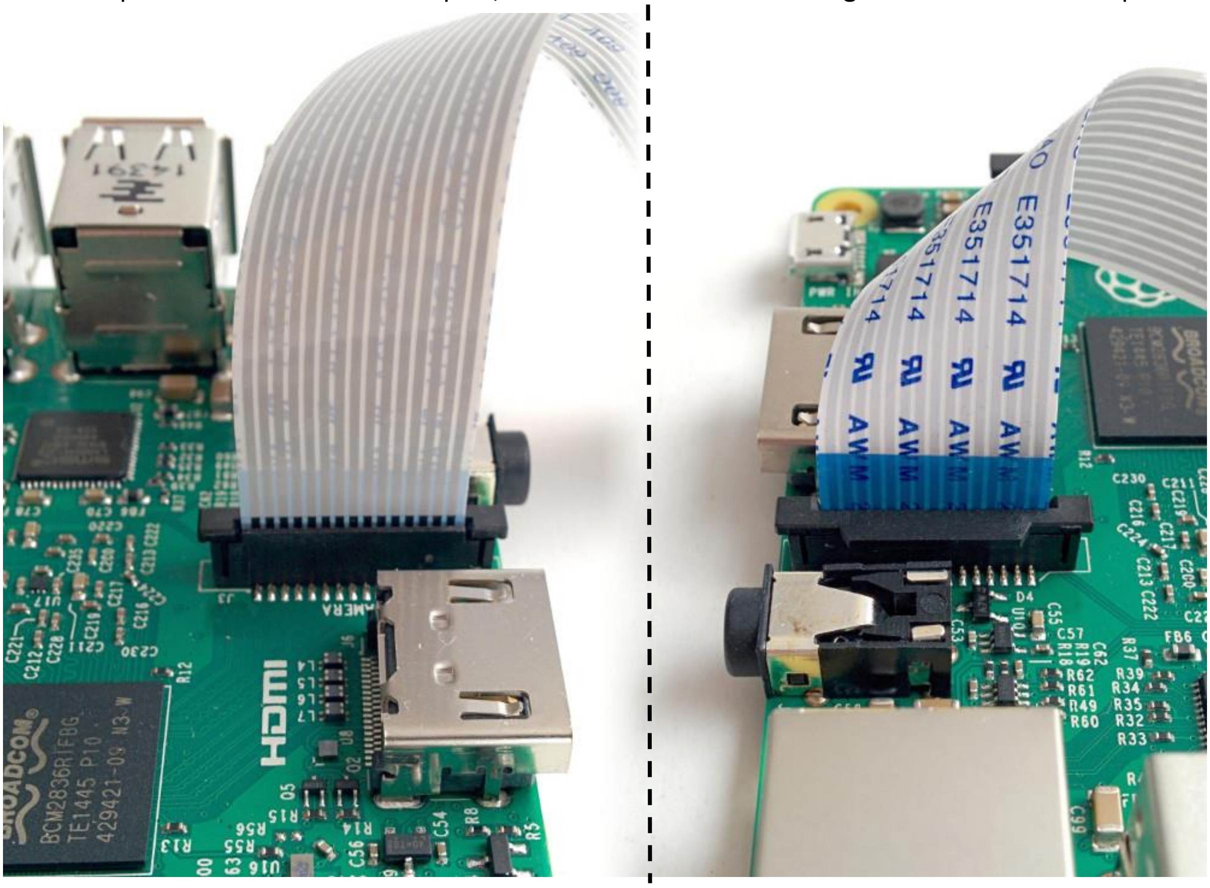


Figure 2.1: The Raspberry Pi with the camera connected, viewed from both sides (Raspberry Pi Foundation, 2018b).

Step 3 – First Boot

Once all the cables have been set in place and the microSD card has been inserted, the user can then attach the microUSB power cable, which will instantly cause the Pi to boot to its set-up window. After a few minutes of setting up, the Pi will show a desktop environment (Figure 2.2). This will look familiar to users of a Windows PC or an Apple Mac. Older versions of Raspbian do not default to a graphical environment, but instead boot straight into a terminal window, where the user interacts with the computer purely by code.

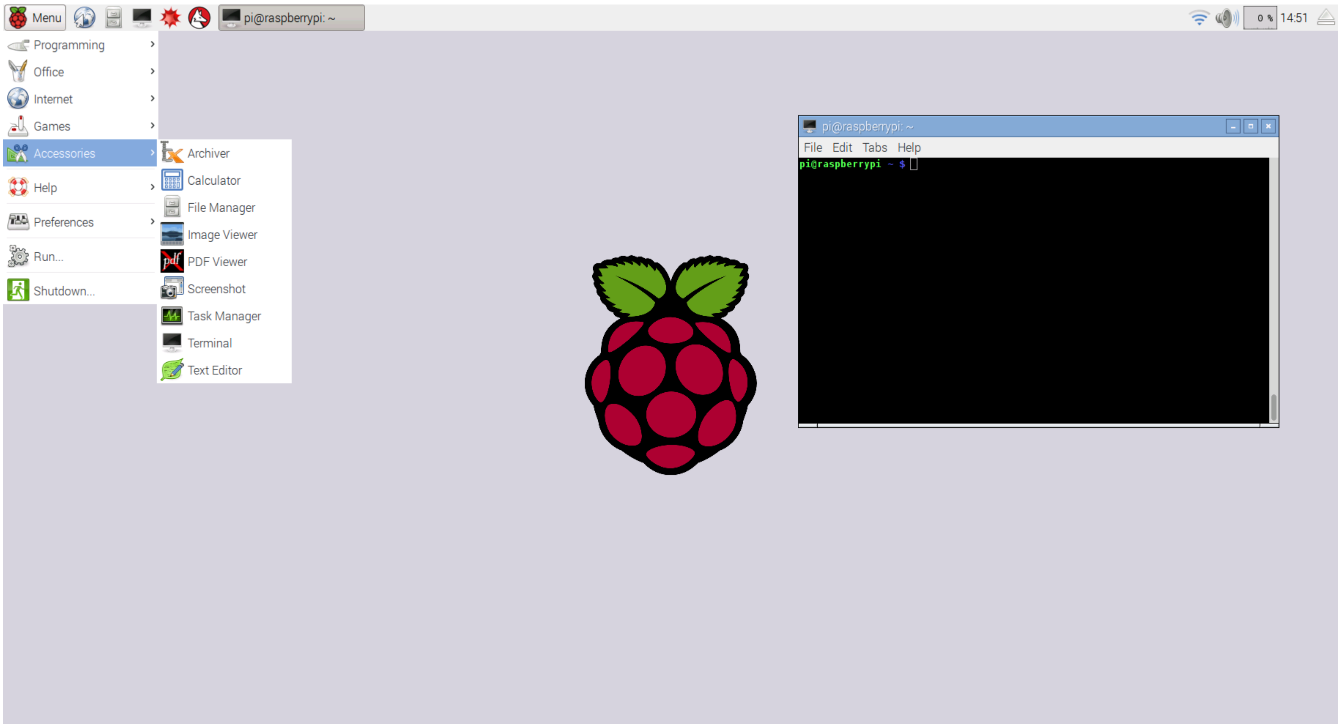


Figure 2.2: The Raspberry Pi desktop environment, showing the top taskbar, a cascading menu system on the left and an open Terminal Window on the right.[1]

All the normal computer desktop features are present. Along the top of the interface is the task bar, featuring a drop-down menu bar, some shortcuts to installed programmes, windows to show currently opened programmes (in this example, the Terminal programme is opened) and at the far right there are some status symbols, including wireless connectivity, volume, current processor load, the time and finally a button to eject external media, such as USB drives.

Step 4 – Getting Online

There are several ways to connect the Pi to the internet depending on what the user has connected and the networks available. Connecting an Ethernet cable to the Raspberry Pi is the easiest way of connecting to a network. Alternatively, there is a Wi-Fi tool in the menu to connect the Pi to a local Wi-Fi network. On most home networks, it is as simple as connecting any device onto the Wi-Fi network.

---

[1] *Newer versions of the Operating System's user interface (known as PIXEL) will look substantially different, but operate in the same way.*

However, as this thesis is focused on using the Pi in an educational setting, there may be some difficulties. Sensibly, educational networks should be protected against unauthorised access, and often do not have publicly available Wi-Fi. Connecting the Pi to a school or college's Wi-Fi may require the support of the institution's network administrator to allow the correct authorisations to ensure the safety of the students and staff (NEN, 2015). As an example, the University of Sussex has a Wi-Fi network which is available to students based on their University IT details. Helpfully, the university published a step-by-step guide to setting a Raspberry Pi up on their protected network (University Of Sussex, 2015).

Step 5 – User Interaction

One of the aims of this project was to give the user an easy way to interact with the computer which does not require using the keyboard or mouse. For this, the official Raspberry Pi 7 inch touchscreen monitor was used to give the user an interactive display, which comes in a DIY kit (shown in Figure 2.3). The screen is compatible with all models of the Raspberry Pi (except for the Zero boards, as they lack the connector needed). It supports up to ten points of touch, and does not require a stylus for use. The software required for using the Pi touch screen comes is built into recent versions of the Raspbian OS, so no additional software is required to be installed. For this project, I also purchased a stand and case for the screen, but this is not essential.

The board connects to the Raspberry Pi via a short flexible ribbon cable, similar to one used to connect a camera to the Raspberry Pi. This cable connects at the Display port (labelled DSI on older hardware), toward the bottom of the board, next to the microUSB power input. The DSI cable also connects to the LCD driver board on the touchscreen through the same connector. This cable carries both the video feed and the signal from the touch input. No HDMI cable is required in this set-up.

Figure 2.3: The Raspberry Pi Touchscreen and the hardware required to connect it to the Raspberry Pi (The Pi Hut, 2018b). The figure shows the touchscreen, the LCD driver board, connection ribbon, screws and stand-offs for mounting the Raspberry Pi and 4 jumper cables to provide power.

The screen itself can be powered several ways, depending on the user's needs. The first option is to use 'jumper cables' (shown in Figure 2.3 above). These come included with the touchscreen kit and connect the Pi to the screen by making contacts with the GPIO pins from the Pi. Alternatively, the user can simply power the screen over microUSB from a second plug, or connect the screen and the Pi via a USB to microUSB cable and power both the screen and the Pi from the same power socket.

In conclusion, although it may initially seem intimidating, setting up a Raspberry Pi is no harder than setting up any computer or tablet for the first time, and provides learning opportunities with regards to learning about how to install an Operating System.

Part 2 – Software

As previously mentioned, the Raspberry Pi runs on the Linux OS, specifically the Raspbian distribution. Many useful programmes come preinstalled as part of the OS, but some additional programmes are used in this project. The method of installing programmes on a Linux machine can be quite different to how a user may install programmes on a Windows PC or on a Mac.

More recently, some of the more mainstream Linux distributions have included an App Store that functions just like the Windows App store or the Mac App store, allowing for simple downloads of new programmes. However, some programmes need to be installed by manually entering code into the terminal window, which may seem daunting to a new user. Although it sounds complex, it is fairly simple and gives the user a greater understanding of how the machine is working as well as an opportunity to try coding.

Full documentation on the programmes described in this thesis can be found at their respective web pages; please see *Appendix 2: Software Guides* for more information.

Key Vocabulary

Before we look at the programmes that have been used as part of this project, we briefly need to look at the concept of 'Directories', 'Root', the '*Pi* account' and the 'Sudo' command, and apply this to how the Raspberry Pi responds to instructions. Then we will briefly look at how Raspbian installs programmes in general (and how updates work) as these will be necessary for installing the software required for this project, before discussing the software used as part of this project.

Directories

In a Linux based OS, folders are called directories. They are able to contain other directories (called a sub-directory) or files. The Root Directory houses all other directories on the system. The Root Directory is the only directory that is *not* a sub-directory, similar to the C Drive on a Windows PC.

The Root Directory is represented by '/' when looking at a file's location. A sub-directory follows the forward slash to describe the location of a file. For example, a file called 'thesis.doc' is in the Documents Directory of User's main Directory. The User Directory is a sub-directory of the Root Directory. This would be represented by: `/User/Documents/thesis.doc`

A user can freely move between directories using the 'change directory' command, `cd`, followed by a space. For example, if User was in their Documents Directory, they could move into their Photos Directory using the command:

```
cd /User/Photos
```

Root

Root is the name of the account on a Linux-based machine that has access to all files and folders on a Linux machine and can make any edits that it chooses to (Raspberry Pi, 2018c). In that aspect, it is somewhat similar to a Windows administrator, which can make changes that other, less-priveledged users cannot. It is also known as the Superuser.

Root is able to edit any file in the Operating System, including those that are absolutely critical for the hardware to work correctly. If a superuser accidentally edits the wrong file, they could completely destroy the system, which is why most user accounts should *not* be given Root privileges.

Pi

When the Raspbian OS installs, it creates a user account called *Pi*. *Pi* is the default account that is used in this project. If the user so chooses, they can create their own additional, personal user accounts for different users on the Raspberry Pi, as the Pi can be accessed by multiple users.

Pi is not the *Root* account so has limited access to files on the computer, and can only edit files within its own user area, and therefore is protected from damaging the machine. However, this does limit the ability of the user *Pi* to install additional software or move files that may be needed. This can be overcome through the use of *Sudo*.

Sudo

'Sudo' is a term that stands for 'Super User Do'. By entering Sudo before a command, the account *Pi* can access files or make changes that it would not normally be able to.

Entering Sudo effectively gives the user *Pi* all the privileges of the user *Root,* including the ability to install software, edit files outside of its own directory, and the ability to completely destroy the system by editing the wrong files.

The user *Pi* is automatically added to the 'sudoers' list on Raspbian when the OS sets up for the first time. Only users added to this list are able to access *Root* privileges by entering Sudo, otherwise any user could just type 'sudo' and then access anything on the machine. This is a neat balance between giving certain users necessary powers that have to be intentionally invoked, and protecting the system from users who may not know exactly what they are doing (very helpful in a school setting). If the Raspberry Pi owner wishes for other users to also be able to use Sudo, they can add additional users to the 'Sudoers' list.

Installations

As has been mentioned, some programmes need to be installed for the projects in this thesis. There are two different methods used to install different programmes used here, downloading software through the Advanced Package Tool and Git Cloning.

The Advanced Package Tool (APT)

As Raspbian is based on Debian it utilises the same conventions for installing software. Raspbian can use the Advanced Package Tool (APT) to install packages (software) (Raspberry Pi Foundation, 2018d). In simple terms, Linux software packages are kept on servers, and these servers contain information about the software, the current version and how to install it. The Raspberry Pi will check these central lists and compare the information held on the server against the local information it holds about the software in its own memory. The user can instruct a machine to update its local information with up-to-date information from the server by opening a terminal window and entering the command:

```
sudo apt-get update
```

Sometimes, this update command just finds out new information about where the programme is being hosted from. However, if the programme has been updated, the user can force the machine to download the newest version and upgrade the installed software. To carry out this operation, the user must enter the following command into the terminal:

```
sudo apt-get upgrade
```

In general, installing new programmes is very similar process. In the terminal window, the user types:

`sudo apt-get install` *`name of programme`*

In this case, the machine will search the database for a programme that matches the name entered, and if it is located, it will download and install the software.

For example, to install ImageJ (a programme which will be discussed later) onto the Raspberry Pi, the user would enter:

```
sudo apt-get install imagej
```

The Raspberry Pi would search the list of packages, find the one called '*imagej'* and begin the installation process. If the Raspberry Pi could not find a programme it would inform the user of an error. This is often the result of a simple spelling error or may be due to the programme having a different name on the database.

After entering these commands, the programmes generally install themselves with little input from the user, but some programmes will need the user to confirm the installation or to press a key during the installation process (for example when a programme might use a large amount of memory).

Git Cloning

This method clones a folder from a remote source and makes a local copy on your own Raspberry Pi. Some programmes are stored on GitHub, an online repository of programmes, and by using the command below, you can make local versions on the computer you are accessing. A terminal window is opened up, and the user enters the git clone command, followed by the location of the repository that the user wants to make a copy of:

```
git clone https://(insert_the_website_that_you_are_cloning_from_here)
```

This method is used to install the Dropbox Uploader programme used to transfer data from the Pi for analysis.

Additional Software

This project utilised four key programmes; 'ImageJ', 'Dropbox Uploader', 'OpenCV' and 'Motion Track'. In addition to this, I developed a specific user interface for the Raspberry Pi microscope application. This section will walk through their installation and usage, with references to other guides which have been used to help set up the software.

Image J

ImageJ (Schneider, Rasband and Eliceiri, 2012) is an open source, free image analysis programme that is widely used in scientific research. It is written in Java, a cross-platform programming language that is available across Windows, Mac and Linux machines.

As the programme is open-source, other people are able to develop additional features for the programme. These are known as 'plugins'. These do not come bundled with the application by default, but can be downloaded and installed separately. The MTrackJ and Image Science plugins were developed by Meijering, Dzyubachyk and Smal (2012). These plugins are used to track motion across a sequence of images.

The MTrackJ plugin has been used in a wide range of research, including fluorescence microscopy (Downey *et al.*, 2011), cytometry (De Vylder *et al.*, 2011), and studying cell migration in ovarian cancer models (Moran-Jones, Brown and Samimi, 2015). It has also been used in undergraduate projects at the University of Sussex, where it has been used to study tardigrade leg coordination (Hall, personal communication) and nematode movement behaviour (Ray, personal communication).

In this project I intended to use MTrackJ to track the motion of tardigrades under the DIY microscope, but due to time constraints and personal health issues, this part of the project was changed. It was however, shown to be successfully used to track tardigrades in changing temperatures as part of our School Science Review paper (Kent and Bacon, 2016).

ImageJ has been compiled for Linux, so is available by opening a Terminal window on the Raspberry Pi and inserting the command:

```
sudo apt-get install imagej
```

Once the programme has installed, it should be found in the main taskbar menu under the ‘Graphics’ drop-down option.

Once ImageJ is installed, the plugins also need to be downloaded from the Image Science website (Meijering, 2018). For MtrackJ to work correctly, the two plugin files (*MTrackJ_.jar* and *imagescience.jar*) need to be downloaded and saved to the ImageJ plugins folder (found at `/home/pi/ImageJ/plugins`). MTrackJ can be accessed under the ‘Plugins’ heading on the ImageJ menu bar. Selecting this from the menu bar will open a new pop-up box with a range of options for tracking subjects, as shown in Figure 2.4.

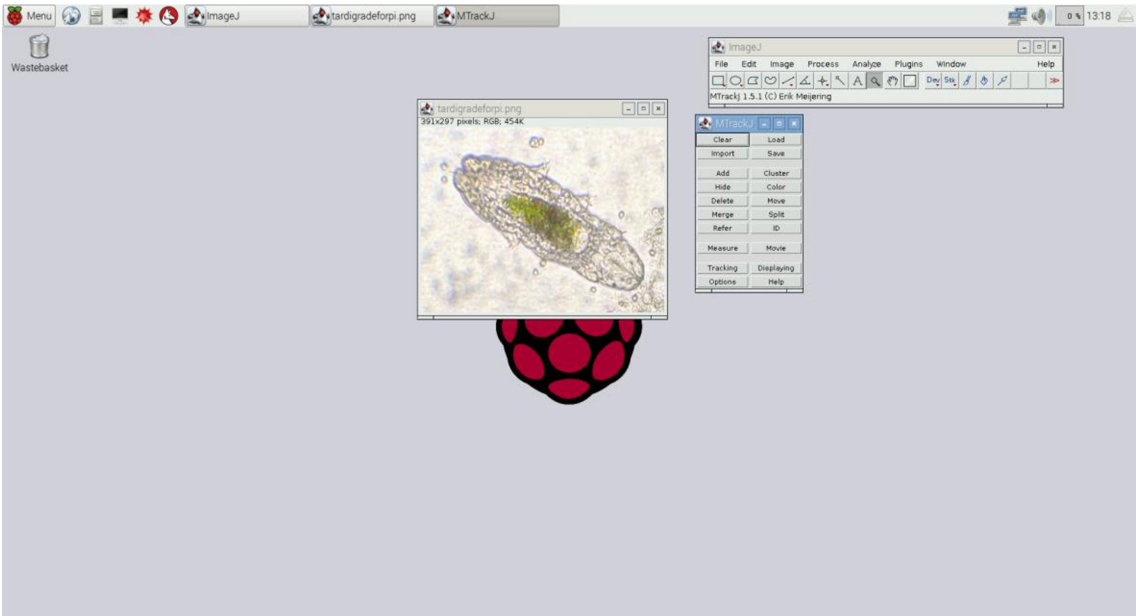


Figure 2.4: The ImageJ menu, with the MTrackJ plugin menu pop-up.

Dropbox Uploader

Dropbox is a popular online platform for storing and sharing work. Dropbox is an ideal tool to be used to collect, back up and share images and data collected by the Raspberry Pi. Andrea Fabrizi (2016) has developed the Dropbox Uploader tool. This allows a user to enter some commands into the terminal and upload files from the Pi directly to a Dropbox account which can be accessed from any device. This makes later analysis of data much easier to carry out, and saves the user from having to transfer data manually via USB memory sticks.

Firstly, it is important to have a Dropbox account that is to be used for this purpose. Dropbox accounts are free to make and come with a generous amount of free storage. Additional storage space can be purchased if the user requires it.

To install this programme, the user should open up the terminal window and enter the change directory command:

```
cd
```

When entered by itself, it brings the user back to the *home* directory, ensuring that the user is not still accessing another folder or file before making any changes.

Then the user should enter the following into the terminal window:

```
sudo git clone https://github.com/andreafabrizi/Dropbox-Uploader.git
```

Before hitting the 'Enter' Key, and then typing the 'List command':

```
ls
```

The List Command shows all the files or subdirectories contained within the user's current directory. By entering the List Command here, the User should be able to see the files in this directory, one of which should be a subdirectory called 'Dropbox-Uploader'. To enter this directory the user should enter:

```
cd Dropbox-Uploader
```

The user should now be in the subfolder called 'Dropbox-Uploader'. To see all the folders and files in the Dropbox-Uploader file, the user should enter the List Command again:

```
ls
```

There should be a file called 'dropbox_uploader./sh'. This can be run by entering the following command:

```
sudo ./dropbox_uploader.sh
```

The user then needs to follow the on-screen instructions to set up a new Dropbox App and enter the details requested. When this is all done, the user will have successfully installed the programme and will be able to upload files from the Raspberry Pi directly into a Dropbox account which can be accessed from anywhere.

To upload a file from the Raspberry Pi to Dropbox, the user must open the terminal and enter the following command:

```
cd home/pi/Dropbox-Uploader
```

Then the user must press 'Enter' before entering the following command:

```
sudo ./dropbox_uploader.sh upload /home/pi/name_of_upload_file name_of_upload_file
```

The first 'name_of_upload_file' gives the location of the file the user wants to upload into their Dropbox. After a space, the user then enter the name that they want the uploaded file to have. These can be the same, or the file can be renamed upon uploading. For example, if the user wanted to upload 'Picture1.jpg' from their Pi Directory (itself a sub-directory of their Home Directory) to the Dropbox, and rename it to 'Tardigrade Photo 1.jpg' they would enter:

```
sudo ./dropbox_uploader.sh upload /home/pi/Picture1.jpg TardigradePhoto1.jpg
```

Upon opening Dropbox on another device, the user should be able to see the TardigradePhoto1.jpg file in their account. While using the browser to upload files could be done on the Raspberry Pi, using Dropbox Uploader has a few benefits. Firstly, it is arguably quicker to enter some simple commands than to open the webpage, log in and select the files required. More importantly, using Dropbox Uploader means that a graphical environment does not need to be active (that is, you could upload or download files by the command line alone, or when logged in remotely). You can also incorporate the commands into another programme to have data uploaded automatically.

In this project, Dropbox-Uploader was used in the following way:

1. A series of images were taken, each saved to a folder (`/home/pi/Test1`) with a unique filename (image_001.jpg, image_002.jpg … image_060.jpg).
2. The Dropbox Uploader programme was opened, and the folder `Test1` was uploaded to the Dropbox account using the following commands:

   ```
   cd home/pi/Dropbox-Uploader

   sudo ./dropbox_uploader.sh upload /home/pi/Test1/ Test1
   ```

3. When the Dropbox was accessed on another computer, the Folder Test 1 contained all 60 images from the image sequence.

The programme in use can be seen in Figure 2.5.

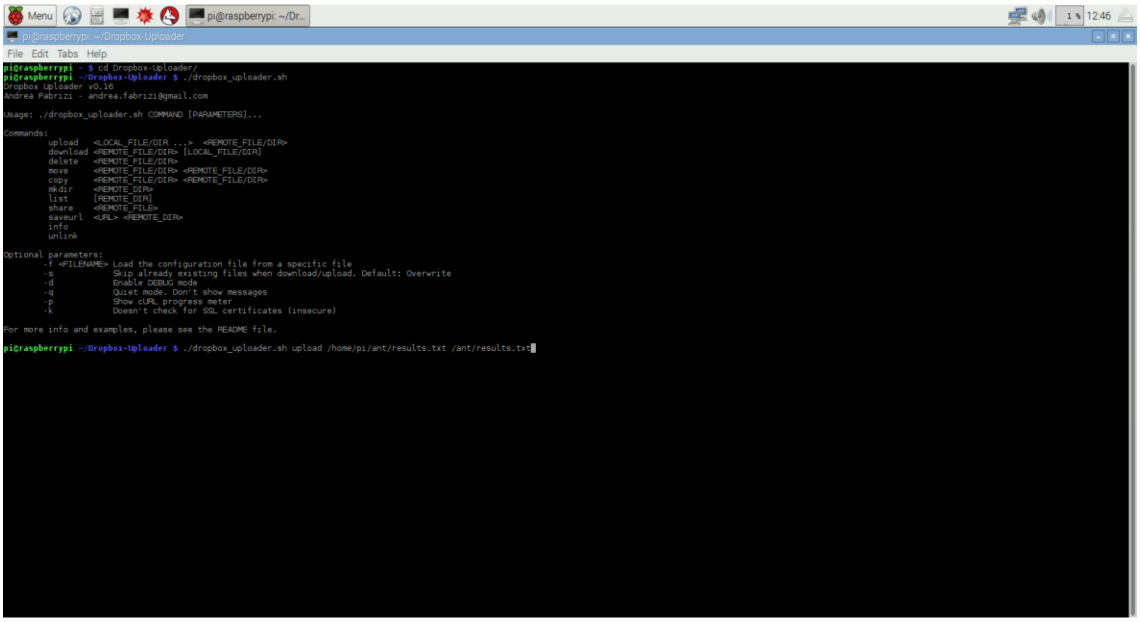


Figure 2.5: The Dropbox Uploader programme is controlled by entering text in the terminal.

Open CV

OpenCV (Bardski, 2000) is an open source computer vision and machine learning software library, available across different platforms and programming languages, which is used for real-time video analysis. Computer vision is a huge and growing field, but OpenCV is a very popular system, as it is open source, allowing anyone to build upon and improve its existing feature set. It was originally designed by Intel, a huge player in the computer market, so it had a strong initial backing leading to OpenCV's popularity. OpenCV has been used in a range of projects, from driver assistive technologies in cars (Garcia-Sierra, 2013) to popular camera apps for iPhone, such as HappyShutter (Sadun, 2012).

There are many guides available online to help install OpenCV. This project used the guide by Adrian Rosebrock (2015) to get OpenCV working on the Raspberry Pi.

Installing OpenCV is one of the hardest and longest parts of this project. On less powerful Raspberry Pi boards (Model A, B and Zero), this step can take up to 12 hours (providing it installs correctly first time). On the Pi 2 or 3, this is much quicker, due to the quad-core processor.

Motion Track

Motion Track is a programme developed by Claude Pageau (2018). This piece of software analyses a live video feed and detects if motion has occurred between two adjacent frames. If the programme detects motion, it draws a green circle onto the output to highlight the location of movement (Figure 2.6). Motion Track relies on OpenCV to compare frames from a video stream. This programme was used to track ants and observe their choice making behaviour and will be described in Chapter 4.

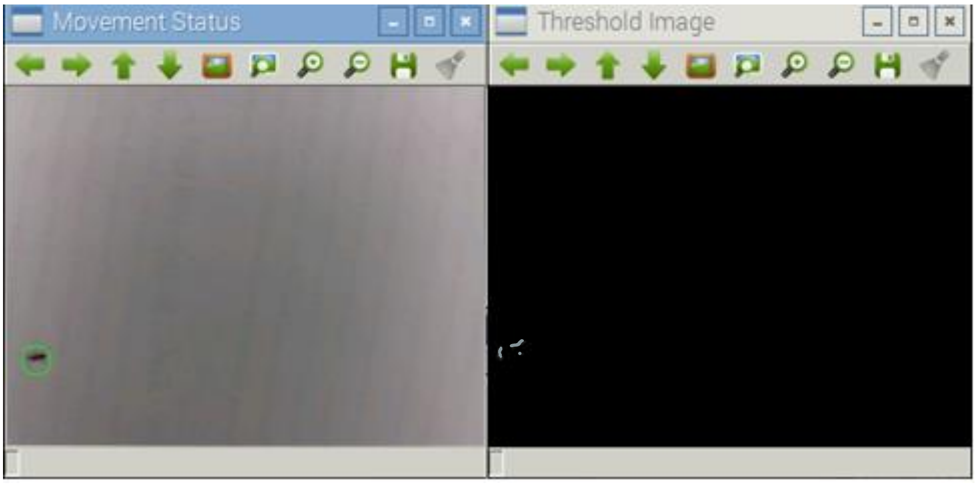


Figure 2.6: The Motion Track programme opens several windows. In the window on the left, an ant has been detected moving, and is highlighted by a green circle in the bottom left hand side. The window on the right shows where the programme has detected a change between frames (faint grey pixels). [2]

This programme was modified to be more useful in the project. While the programme displayed movement and highlights where movement is being detected, it did not record these data anywhere. Due to the way the experiment was carried out, the X co-ordinate of the detected motion was the key piece of data needed. As a result, the programme was adapted to create a text file on opening, and it would modify this text file anytime movement was detected, adding the X co-ordinate to a list. This file was then uploaded to the Dropbox and used to determine where in an arena an ant was present at any given time.

[2] *Please note, the threshold image has been edited for clarity; the pixels have been lightened to be more visible when printed; the threshold image is easier to see in motion.*

I also changed some of the parameters of the file to change the threshold required for movement to be detectable. The modified python code can be found in *Appendix 3*. To install this programme, the user must enter the following command into terminal:

```
curl -L https://raw.github.com/pageauc/motion-track/master/motion-track-install.sh | bash
```

This same software could be applied in all sorts of situations, for example looking at when animals enter or leave a trap or nest, or studying decision making in other animals, as long as the programme is edited to make sure the necessary data are saved. Additional data, for example a timestamp or image could also be collected when motion is detected, allowing other observations to also be made.

Please note, a newer version of this programme has been produced which utilises OpenCV3 and Python 3, neither of which were used in this project.

Microscope Application

A key aim of this study was to develop the microscope application further in such a way that a user could access the microscope without any need to interact with the computer via the keyboard or mouse. Previous versions of the microscope relied on the user entering commands into the terminal window to collect images or video, which is not ideal for someone who has never used such a system before. To improve upon this, I developed a graphical user interface (GUI) which allows the user to press buttons on the display to interact with the hardware (Figure 2.7).

When I created this programme, I had the following aims in mind:

1) It had to be accessible for people who had never used Linux before.
2) There had to be a clear guide and help menu.
3) All of the key camera operations (a still image, a time-lapse recording and a movie file) had to be available to the user on the main menu, ideally with a single button click.
4) The user had to be able to interact with or receive data from other connected devices (such as thermometers or lights) from within the programme.

5) It had to fit onto the official Touch Screen and make the most of the (relatively low) resolution.

I chose to produce the programme in Python. Python is a relatively simple language that is suitable for beginners. Python also has a very good GUI toolkit built in which allowed me to easily design the GUI. Additionally, Python is cross-platform which allowed me to develop the programme on a Windows PC and test it out at different times.

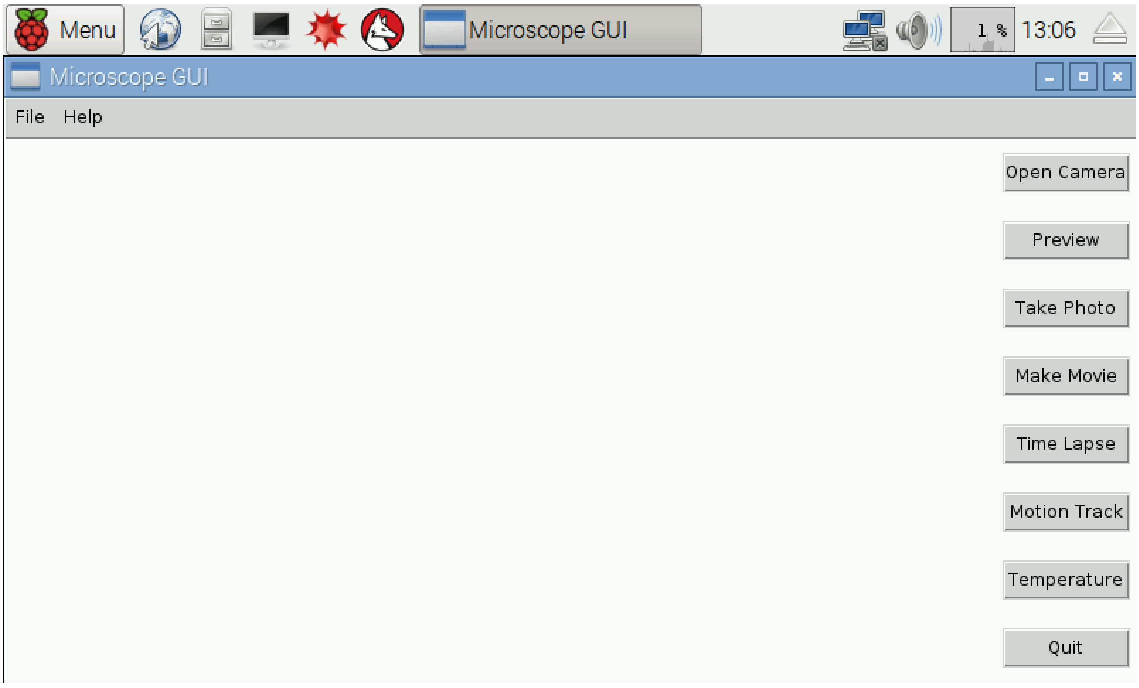


Figure 2.7: The main screen of the custom made Microscope GUI. The top section features the menu bar and down the right hand side are the buttons which activate features of the application.

On the right hand side, there are 8 buttons which can be used to activate a range of functions.

(1) *Open Camera* causes a preview of the camera to appear in the empty space on the screen for 10 seconds. This allows the user to position the camera in its desired location.

(2) *Preview* brings up a full screen 10 second preview; the image quality is better in this view, but it takes up the entire screen, overlapping the GUI.

(3) *Take Photo* begins a 3 second countdown and then takes a still image and saves it to the directory of the microscope programme.

(4) *Make Movie* creates a ten second movie file and saves it to the directory of the microscope programme.

(5) *Time Lapse* causes the Raspberry Pi to take 10 photos, with a one second pause between each image. These images are saved with an ascending 4-digit identifier.

(6) *Motion Track* causes the Motion Track programme to run, shown in Figure 2.8.

(7) *Temperature* causes a pop-up box to appear with the current external temperature, measured by external sensors. This window can be dismissed by clicking 'Okay' in the pop-up box. Dependent on the sensors attached, this button could also be labelled 'Humidity'.

(8) *Quit* causes the programme to quit.

Across the top menu of the GUI is a detailed help section which provides the user with more information about how each button functions.

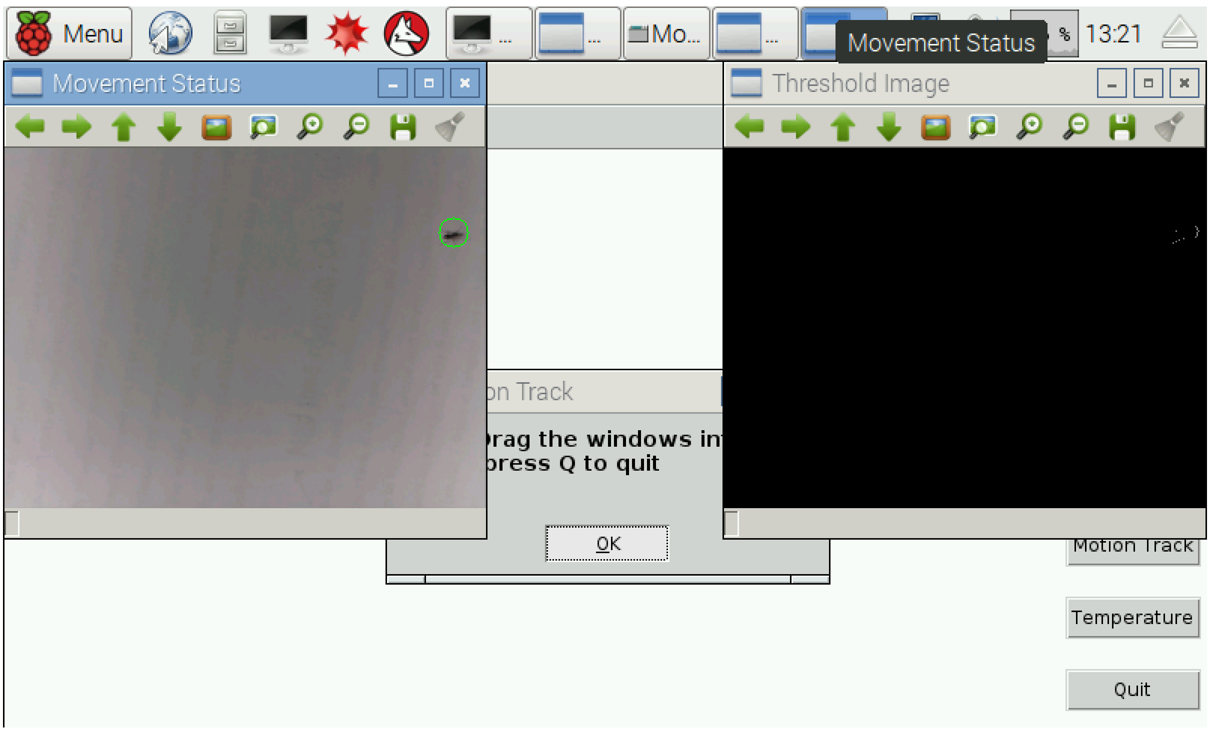


Figure 2.8: The Motion Track Programme running within the Microscope Application.

One of the advantages to creating this in Python is that other users can very quickly and easily change settings. If a user wanted to make minute long movies as opposed to ten seconds, this would be a simple matter of changing a few numbers in the code, and restarting the programme. Its flexibility makes it perfect for beginners in computer science.

The full code for the Microscope GUI application can be found in *Appendix 4*.

In conclusion, through a range of programmes, the Pi can easily be given new features and abilities. These programmes are all free or custom produced for this project, so no additional money was spent, and they are all useful for a range of purposes and could provide easy extensions for this project.

Part 3 - Additional Hardware

One of the most exciting things about the Raspberry Pi is the huge range of external hardware that can be purchased for the computer and the number of potential projects this opens up. As we have already seen from the introductory chapter, a range of HATs can be purchased to provide additional sensors and outputs. Other hardware, such as lights, buzzers, motors, touch pads, servos and more can all be purchased and incorporated into projects. Some more specific and scientific equipment can also be made to interface with the Raspberry Pi, such as soil humidity monitors, light gates and dissolved oxygen probes. As part of this project there are a few key pieces of hardware that have been used, which will be detailed below.

USB Power Bank

As the Pi uses industry standard connections, the Pi can be used with a variety of hardware. One of the most helpful connectors is the microUSB power cable. Portable USB power banks (like those used to charge phones on the go) can be used to power the Raspberry Pi away from the mains power supply. The length of time a Pi can be operated in this way depends on the capacity of the battery pack and on how much power is being used by the Raspberry Pi. Some models, such as the Pi

Zero or Pi A+ require much less power than the Raspberry Pi 2, so can be run from a USB power bank for a considerable length of time.

Raspberry Pi Camera

For this whole project, the addition of a camera has been essential. The Raspberry Pi Foundation has produced several camera boards that can be used with the Raspberry Pi. A brief outline of these models is provided below in Table 2.3. This project mostly used the standard Camera Module (Figure 2.9), but the other models have all been used at varying times. The cameras connect to the Raspberry Pi via a flexible ribbon cable. The included cable is about 15cm in length, but the camera board can be used with alternative cable lengths; cables up to 2m long are readily available.

	Camera Module	NoIR Camera Module	Camera Module V2	NoIR Camera Module V2
Resolution	5 MP	5 MP	8MP	8MP
Sensor	OmniVision OV5647	OmniVision OV5647	Sony IMX219	Sony IMX219
Infrared Capable	No	Yes	No	Yes
Cost at launch	$25	$25	$25	$25
Launch Date	14/5/2013	28/10/2013	25/4/2016	25/4/2016

Table 2.3: Key information about the different official Raspberry Pi Camera Modules.

Figure 2.9: The 5 MP Raspberry Pi Camera Module (Raspberry Pi, 2019a).

The official Pi cameras are sold as fixed focus cameras (although it is possible to break the glue holding the lens in place and adjust the focus manually by twisting the camera lens, voiding any warranty in the process).

The Raspberry Pi Zero (v1.3 and newer) boards do feature a camera port, but it is different to the ones found on other sized Raspberry Pis. To connect a camera to the Pi Zero, a specific adaptor cable is required (Figure 2.10).

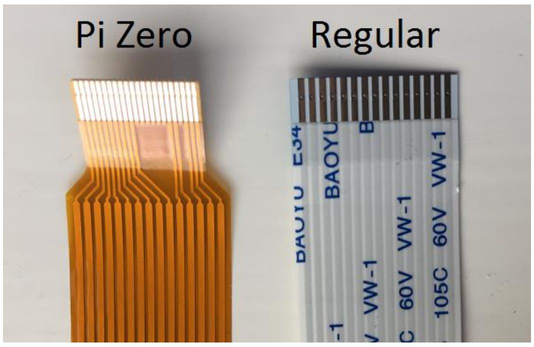


Figure 2.10: The Raspberry Pi Zero camera cable is a different size to the standard camera cable (Raspberry Pi, 2019b).

In addition to the official cameras, third-party cameras are available from other vendors. One of the most popular is the WaveShare camera (shown below in Figure 2.11). This camera uses the OmniVision OV5647 sensor (the same sensor found in the official camera board) and all of the commands for the camera work as normal. However, this camera has adjustable focus, allowing it to take excellent photos across a room, but also down to the near microscopic level. This camera has been used to take pictures of hydra (approximately 1mm long) and even individual tardigrades (approximately 500µm in length) (images shown in Figure 2.12).

Unfortunately, the flexible cable that comes with this camera is very sensitive and swapping it for another cable can break the camera board. The units that have been tested do not work with the extension cables and have caused power issues on the Raspberry Pi board when this was attempted. As a result, the Waveshare camera was not used in this project, and only official Raspberry Pi hardware was used instead.

Figure 2.11: The Waveshare Camera has an adjustable lens which means the image focus can be changed (The Pi Hut, 2019)

Figure 2.12: Image taken using the Waveshare Camera board. In the same image, both a hydra (*Hydra viridis)* and a tardigrade (*Dactylobiotus dispar)* can be seen. The tardigrade is highlighted by the white circle.
In the movie file, a ciliate can be seen moving away from the hydra. It is quite hard to see in this still, but is pointed out by the arrow.

Environmental Sensors

Thermocouple

A thermocouple is a tool used to accurately measure temperatures, and can be used to monitor the temperatures of solids, liquids and gases. The probe is made up of two metals, each with slightly different conductive properties. When the probe comes into contact with a surface, a liquid or the air, the difference in the voltage across the two metals can be used to determine the temperature to a high level of accuracy. There are several types of thermocouple, and these are classified based on the types of metals used.

The Raspberry Pi is able to interface with thermocouples. Adafruit, a leading producer of Raspberry Pi compatible hardware made the MAX31855 thermocouple board (shown in Figure 2.13). This allows a thermocouple to be attached to the Raspberry Pi via the GPIO interface. A python library produced by Adafruit was used to interpret the data from the thermocouple board. As the library was produced in Python, I was able to incorporate it into the Microscope GUI programme.

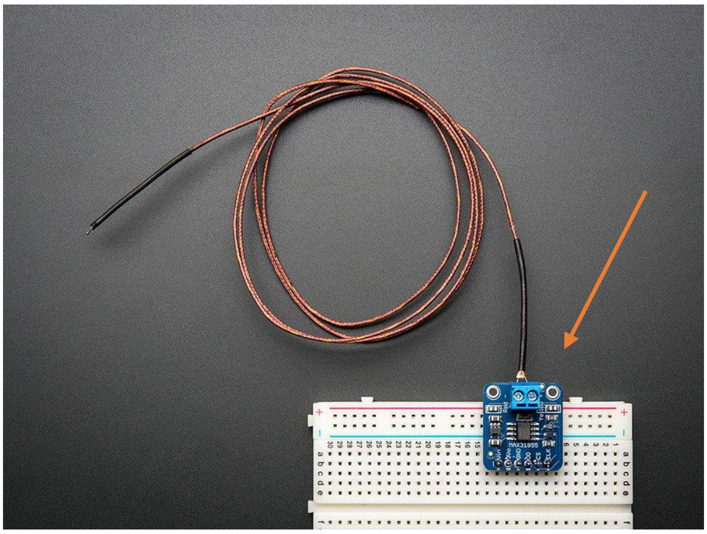


Figure 2.13: The MAX31855 Thermocouple breakout board (the blue square to the bottom right marked with the arrow) attached to a K Type thermocouple (Adafruit, 2018b).

DHT Sensors

In addition to the thermocouple, I also wanted to know about the humidity in the environment, as this could affect how animals behave. Initially two DHT11 sensors were purchased (Figure 2.14a), a temperature and humidity sensor which also has a python library developed by Adafruit. We later also purchased a DHT22 sensor (Figure 2.14b). There are some slight differences between the DHT11 and DHT22 which are described in Table 2.3.

	DHT11	**DHT22**
Temperature Readings	0-50°C temperature readings ±2°C accuracy	-40 to 80°C temperature readings ±0.5°C accuracy
Humidity Readings	20-80% humidity readings with 5% accuracy	0-100% humidity readings with 2-5% accuracy
Dimensions	15.5mm x 12mm x 5.5mm	27mm x 59mm x 13.5mm
Cost	£6.00	£10.00

Table 2.3: The differences between the DHT11 and DHT22, temperature and humidity sensors produced by Adafruit.

a b

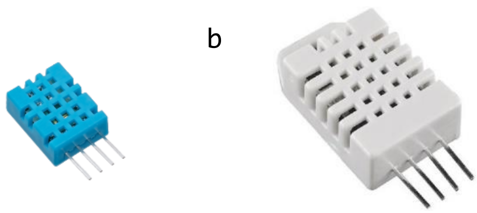


Figure 2.14: The (a) DHT11 temperature sensor and (b) the DHT22 temperature sensor (Adapted from Adafruit Industries (2018c) and Adafruit Industries (2018d).

Evaluation of Temperature Sensors

During testing of the DHT11 sensor, the humidity reading seemed to be low based on previous studies based in the same laboratory. The humidity readings produced by the hygrometer in the lab were consistently higher than the output from the DHT11, so I ran some tests to check the accuracy of the sensor.

Initially we compared the temperature reading of two DHT11 sensors (referred to in this test as DHT1 and DHT2) to the hygrometer (which also records temperature) from the laboratory over a 90 minute period, the results are shown in Figure 2.15 below.

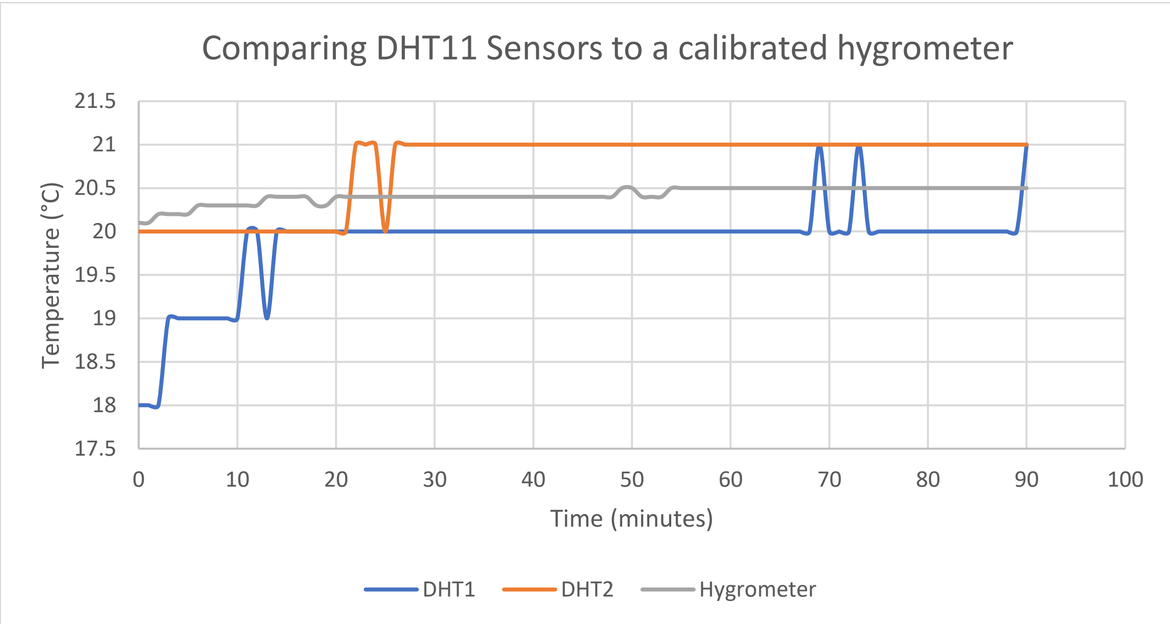


Figure 2.15: The results of a 90 minute test comparing two DHT11 sensors (referred to as DHT1 and DHT2) and a hygrometer. The temperature readings were fairly consistent over the 90 minutes, generally within 0.5°C of each other.

However, the humidity readings varied by about 20% over the same period, so using a DIY method recommended by many animal keeping and cigar enthusiasts' websites, I tested the accuracy of the sensors (McLeod, 2018; Normand, 2018; wikiHow, 2018).

A plastic milk bottle lid was filled level with salt. To this, 5 drops of tap water were added. This bottle lid, the calibrated hygrometer and a Raspberry Pi connected to a DHT11 sensor and powered by a

USB battery pack were placed into a sealable plastic bag. This should produce an environment within the sealed bag with a relative humidity of 75%.

The Raspberry Pi in the bag logged the temperature and the relative humidity every minute over a 6 hour period. These readings were then uploaded to the Dropbox, and copied into an Excel file on a Windows computer.

Another Raspberry Pi (positioned above the bag) ran a programme that took a picture of the hygrometers display every minute over the same time course, and was started simultaneously. These pictures were manually checked, and the temperature and humidity readings were placed into the same Excel file.

The temperatures were again within 0.5-1°C of each other and I believe the temperature sensor to be accurate. However, the humidity readings again varied. This test was repeated using the second DHT11 (DHT2) sensor and this produced the same pattern. The average of the humidity reading from both tests are plotted in Figure 2.16.

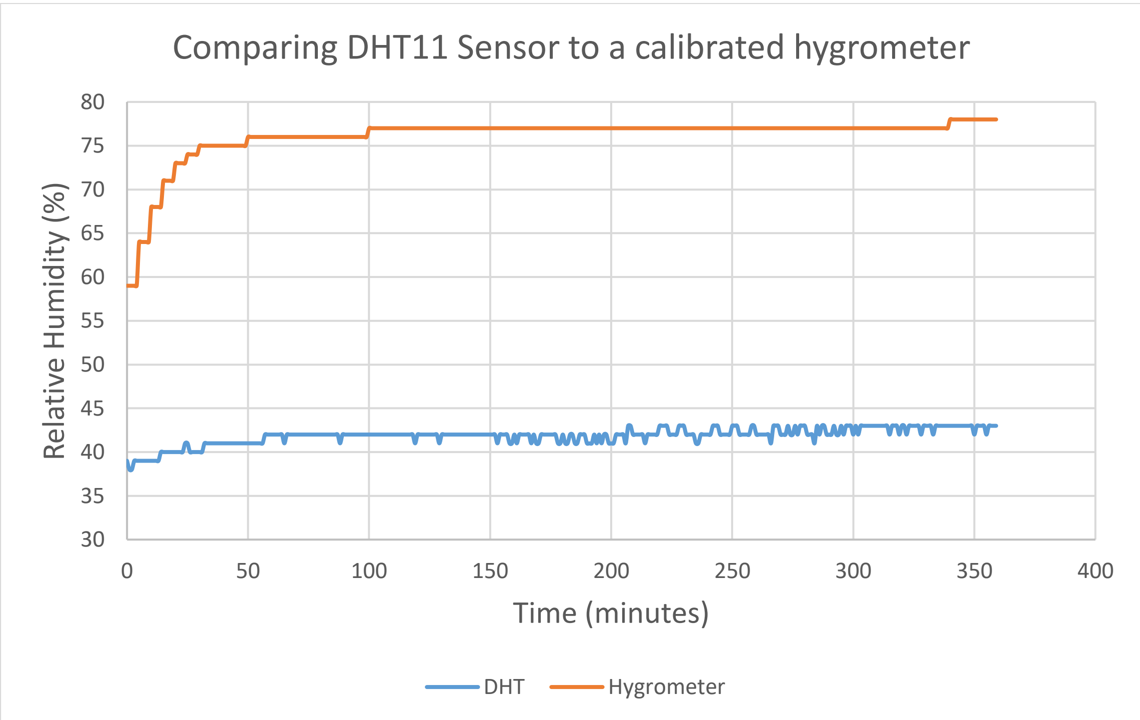


Figure 2.16: The DHT11 and hygrometer humidity readings varied by around 35% humidity, despite being in the same sealed environment.

As a result, I purchased a DHT22 to check the accuracy of this sensor. This sensor was subjected to the same test, and both temperature (shown in Figure 2.17) and humidity (Figure 2.18) were recorded.

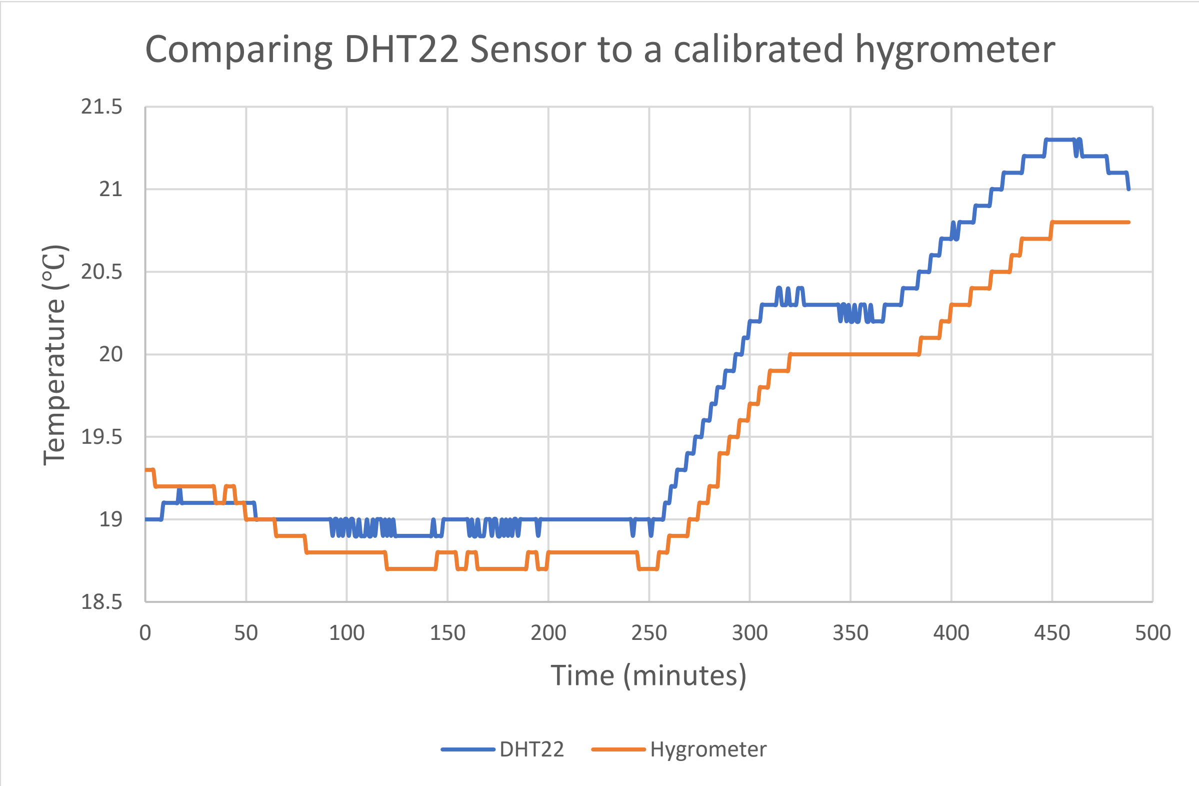


Figure 2.17: The DHT22 and hygrometer temperature readings matched very closely, never more than 0.5°C out from each other.

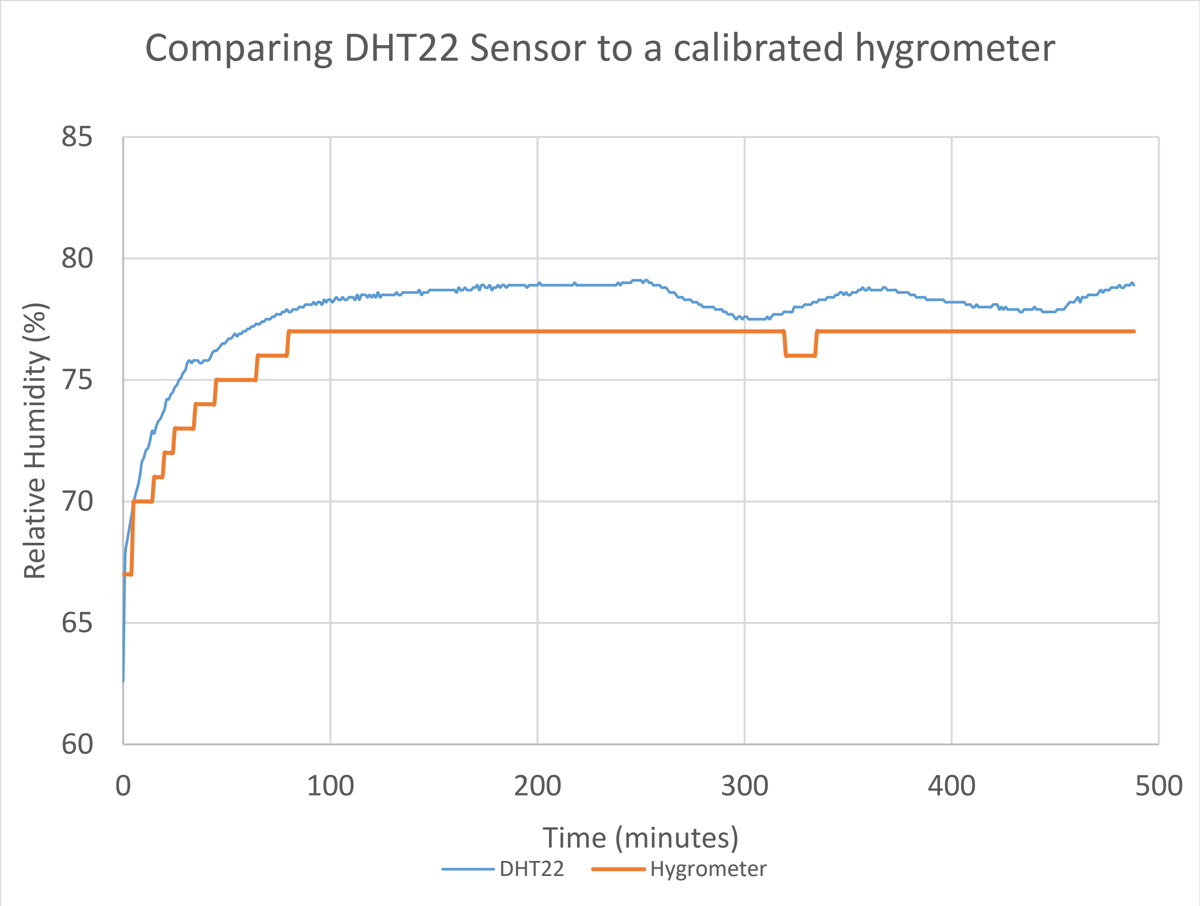


Figure 2.18: The DHT22 and hygrometer readings for relative humidity were also very close and showed the same trend. This is well within the manufacturer's stated error range.

Any of the above solutions would be ideal for a cheap and easy digital thermometer system for investigating the effects of changing external conditions on animal behaviour. The high accuracy and small probe made the thermocouple the most appealing choice for examining tardigrade behaviour.

Thermocouples also have the advantage that they can get wet and survive more extreme temperatures.

However, in general, the DHT22 would be appropriate for most experiments, especially at school level and are cheaper and arguably easier to set up and use. Plus, they can sense moisture in the air, unlike the thermocouple, making them ideal to study plant and animal behaviour.

In conclusion, the Raspberry Pi is an ideal platform to base school experiments and scientific research around. It is inexpensive, well supported by industry, and additional components, such as cameras and sensors, can easily be purchased. Most crucially, the Raspberry Pi is adaptable to the users' specific needs.

Part 4 - The Microscope

The microscope used in this project is developed from an initial guide from Yoshinok (2013) who produced a DIY guide on a popular website, "Instructables". This guide shows how to use cheap materials, such as wood, Perspex® and cheap lenses to produce a smartphone microscope that can capture high quality images.

Some modifications were made to this design to allow the Raspberry Pi camera module to be used instead of a smartphone. The actual design for the microscope will vary dependant on the users' needs and available hardware.

The microscope has many advantages over conventional classroom microscopes, which make it ideal for use in a classroom setting. As with anything, there are of course some downsides to using it over the typical hardware used in schools, but there are also novel learning opportunities to using this hardware in lessons.

Firstly, it is important to remember that this is something that young people could make for themselves. Hardware that can easily be produced by students and then used in lessons can help young people take ownership of their learning. In comparison to conventional microscopes, the materials used in this hardware are inexpensive, easy to obtain (and therefore replace) and hardwearing. The wood and Perspex® are able to take a fair amount of damage before breaking. The cost of replacement parts are also far lower than a conventional microscope, and can often be produced on site, as opposed to waiting for help from the manufacturer and the costs associated with this.

The portability of the hardware also allows for new ways to teach practical science. It would not normally be possible to take a class outside to study organisms in their natural environment and use a microscope. However, the DIY microscope could easily be flat packed and taken outside of the classroom for field trips and lessons outside, without risk of damaging expensive hardware (or having to carry it around). In addition, the students would be able to use their own phones to capture and store the images and films, allowing them to take ownership of their results, and easily incorporate these into classwork or homework presentations.

Of course, although there are many benefits to the DIY microscope, there are some issues. Firstly, the level of magnification that can be achieved is much lower than that of a traditional light microscope, and there is not (at present) a set of lenses with different magnifications that could be swapped out to look at the specimen in more detail.

It’s also important to note that the working distance on this microscope is quite limited. In general, this is not a problem in a school practical (it is a similar issue on school light microscopes anyway). However, tasks that include manual labour (for example dissections or isolating tardigrades) would be difficult. For such tasks, a dissecting microscope would be more suitable. This is not to say that you cannot get larger specimens under the microscope though. For example, spiders have been placed under the microscope to get images like the ones shown below in Figure 2.19, captured in daylight and under infrared light.

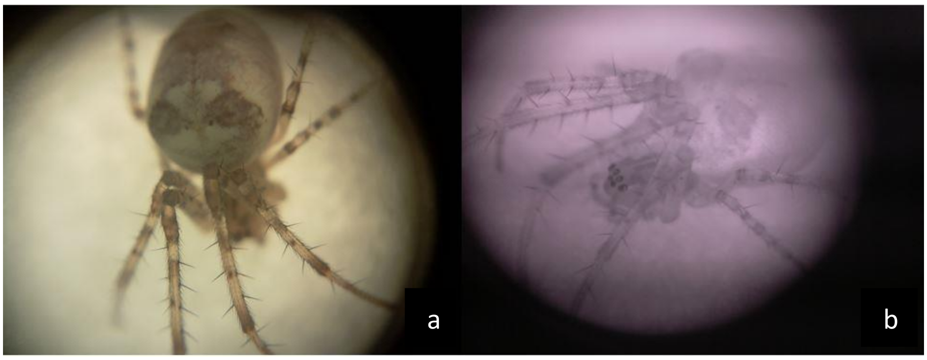


Figure 2.19: Photos taken on the Raspberry Pi microscope of a spider under (a) daylight and (b) infrared light.

I did consider attempting to produce the lenses in-house using plastic polymers. This unfortunately never happened due to lack of the chemical required and time constraints but there are methods to produce a high quality lens without relying on glass or expensive manufacturing processes (Lee *et* *al.*, 2014). These lenses may be producible in school, depending on the school's chemical policies and budgets.

Overall, this tool would be suitable to be used alongside current light microscopes in a cost and space effective manner, with many advantages and novel teaching opportunities presented to students. Combining this practical, hands-on approach to hardware, and introducing students to setting up their own software for classroom led research projects gives a whole new way to look at how science can be taught in a truly cross-curricular way.

Chapter 3 Hunting for Microscopic Animals: A School Science Practical

Abstract

When we think of microscopic organisms, most people will imagine bacteria and 'germs', yet few will be aware of the incredible multicellular animals that live solely in the microscopic world. Even within a clump of moss in the pavement, there is a whole ecosystem, full of animals that exhibit some of natures' most interesting characteristics. This chapter will look at how schools can search for animals in their natural environment, how they can easily be identified and how inexpensive hardware, such as the Raspberry Pi, can be used to produce an interactive, portable digital microscope which can be used to study the behaviour of these animals.

Introduction

Tardigrades (Figure 3.1) are microscopic animals, most notably known for their ability to endure environmental extremes, including radiation (Horikawa *et al.*, 2006), extremes of temperatures (Hengherr *et al.*, 2009; Tsujimoto, Imura and Kanda, 2016), pressure (Seki and Toyoshima, 1998) and exposure to the vacuum of space (Jönsson *et al.*, 2008).

Figure 3.1: A scanning electron micrograph of a tardigrade of the species *Hypsibius dujardini* produced by Madden and Goldstein of the University of North Carolina (2012).

In more recent years, tardigrades have started making their way into popular culture, for example, tardigrades have featured in *Star Trek: Discovery*, the popular video game *Tardigrades*, and also in

Marvel's *Ant Man and the Wasp*.

However, what might be less well known is the fact that tardigrades (or moss piglets as they are also known) are cosmopolitan animals that live in all sorts of environments. Tardigrades have been found in the Artic (Gąsiorek *et al.*, 2017) and Antarctic (Tsujimoto, Imura and Kanda, 2016), in rainforests (California Academy of Sciences, 2017) and open water (Cavalcanti Da Rocha *et al.*, 2013). There are already over 1150 known species of tardigrades (Degma, Bertolani and Guidetti, 2018; Degma, Bertolani and Guidetti, *no date*) with new species being regularly described (for example Stec, Arakawa and Michalczyk, 2018) and countless more waiting to be discovered.

Tardigrades are the ideal test animals to be used in a school setting to examine all sorts of life processes. Their incredible abilities make them exciting for students to study and the fact that they can be gathered from just about any location makes them a popular choice. One of the key abilities is the tardigrade's survival when dehydrated (anhydrobiosis) (Rebecchi, Altiero and Guidetti, 2007) which can be beautifully demonstrated in the school practical I describe in this chapter.

Tardigrades can enter the tun form and slow down their metabolic processes dramatically in order to survive. When the dehydrated organisms are exposed to water, the tardigrades can reanimate. This process is shown brilliantly in a YouTube video by Daiki D. Horikawa (2010).

This chapter will describe the process of 'tardigrade hunting', showing techniques used in undergraduate practicals and school visits at the University of Sussex, as well as at the British Science Festival in 2017, and give guidance on how to use existing school light microscopes to identify the organisms found. It will then look at how schools could use the DIY microscope to locate and identify tardigrades, and capture footage or images for study and discussion. The same techniques will also allow other microscopic animals, such as rotifer and nematodes, to be studied.

One of the big fears when carrying out a practical in a classroom is that the practical won't work. However, Table 3.1 shows that this set up has been successfully used with well over 700 people over the last 3 years, and so far, it has never failed; in every session some microscopic animals have been successfully found in moss.

Setting	Number of Participants	Group of participants	Aim of practical
University of Sussex	>500 (over 3 years)	Undergraduate Life Science students	Identify organisms found in fresh moss using microscopes
University of Sussex	200 (approx.) (over 3 years)	School students and teachers on campus visits	Collect moss from campus Identify organisms found in moss using microscopes
University of Sussex	80 (approx.) over 4 days	Potential Life Science students on applicant visit day	Look for organisms in moss Exposure to university style teaching
Brighton Library	80 (approx.) over 1 day	Members of the public as part of British Science Festival	Hunt for tardigrades and other organisms in moss supplied using university microscopes
King's School, Hove	13	Year 8 students on Enrichment Day	Collect moss from school playground and look for and identify animals in it using the school's light microscopes.

Table 3.1: Table showing the groups that have taken part in practical sessions I have been involved in leading during the course of this project. These groups vary in greatly in age and previous experience of science knowledge.

In each of these sessions I have delivered a very short talk before the practical, describing the key safety aspects of the practical, and outlining some of the things that might be found.

It's also really important to highlight that some students may not see everything. By setting this expectation at the start, you can reduce the sense of frustration some participants may experience, and you also get the opportunity to encourage participants to work together and share their finds with each other. It's also good to show participants that science doesn't always go the way you want to, and you may need to repeat an experiment or observation before finding the thing you're looking for.

Part One: Tardigrade Hunting

Equipment

This project relies on very little expensive equipment beyond what would reasonably be expected to be in a school's science laboratory. All of the equipment can be reused after the practical (except potentially the glass slides, depending on the school's glassware policy) and the samples can be obtained from the natural environment.

Box One provides a list of the necessary equipment for observing tardigrades using a school's light microscope. Box Two provides a list of the hardware required to use the DIY digital microscope to observe and capture images or movies of the tardigrade and for the use of the infrared capabilities of the Raspberry Pi's camera module.

<u>Box One</u>

Light microscope

Light source

Glass slide

Coverslips

Petridish or plastic container

Plastic pipette

Distilled or mineral water

<u>Box Two</u>

Raspberry Pi

Raspberry Pi camera module

Raspberry Pi touch screen (or monitor)

Power supply (USB power bank or access to plug)

Internet connection

Optional:

Raspberry Pi NoIR camera module

Infrared light source

Box to enclose microscope

DHT22 sensor

Step 1: Collecting samples

Tardigrades can be found in many locations. Almost any sample of moss (regardless of its moisture level) will generally contain some number of tardigrades within it. I have successfully located tardigrades in moss collected from the pavement, from trees, from concrete statues and from guttering. A small sample is all that is needed, no more than a clump that can be lifted between the thumb and forefinger and should be placed straight into a petri dish after collection. Teachers could go into their school's grounds and find different sources of moss growing in varied environments. Furthermore, the cost involved (i.e. nothing) is crucial as school's spending on pupils is decreasing (Weale, 2018).

Step 2: Preparing the sample

A small amount of the moss sample is placed in a petri dish and distilled water is added to soak the sample. It is left for around 20 minutes to allow the water to saturate the sample.

For this practical, the slides are prepared in a special way to ensure that the animals are not injured or affected by downward pressure applied during the observations. This slide production method also has the advantage of creating a limited area for the animals to be located within. The steps for producing this slide are detailed below and shown in Figure 3.2 and Figure 3.3.

A glass slide is placed on the work surface. To this, two drops of distilled (or mineral) water are added (approximately 2cm apart). Onto each drop, one cover slip is placed. The water acts to adhere the cover slip to the slide. Into the gap between these two cover slips, some of the moss sample is added. This is a tiny amount, a few leaves and some of the stem are all that is needed. Then a fresh pipette is pushed into the moss sample and some 'moss juice' is extracted. This liquid contains a mixture of particles from the moss and any animals that have been released from the surface of the moss when the water was added to the moss. This 'moss juice' is then dripped onto the moss leaves in the gap on the slide. A thin, even covering of the gap is ideal.

Finally a third coverslip is added to the slide, held up between the first two coverslips. This creates a walled environment for the organisms to be contained within. It also allows for additional liquid to be applied to the slide if the sample begins to dry out.

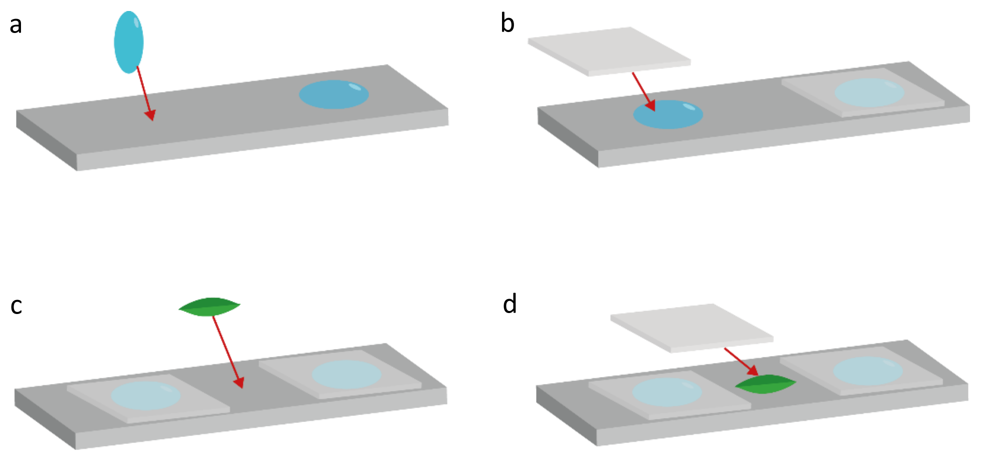


Figure 3.2: The stages involved in producing the moss observation slide. (a) Two drops of water are applied to the slide; (b) coverslips are placed directly onto drops; (c) some of the leaf material is placed onto the slide and moss juice added onto it; (d) the third coverslip is placed onto the slide, overlapping both the other coverslips.

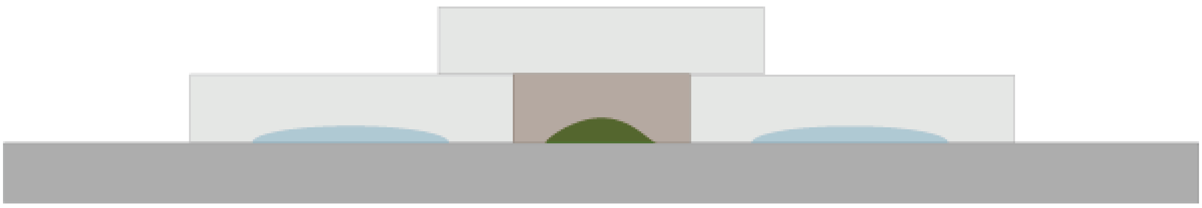


Figure 3.3: An illustration of the completed slide, drawn side on. The moss juice also helps to reduce the air interface.

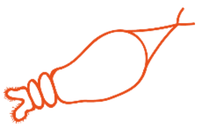
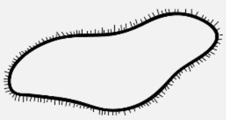
Step 3: Locating animals under the microscope

Tardigrades are generally between 400-600µm in length, although they can vary between 0.1 and 1.2mm in length (Seckbach and Rampelotto, 2015). This makes them easily visible under 10x objective magnification (the sort of magnification you would expect to find on a school science microscope). The University's microscopes have a 10x eyepiece lens and variable objective lenses on a rotating turret system, with 4x to 100x magnification achievable (for a total magnification of 40-1000x).

The method that I have found to be most reliable is to use the edge of a coverslip as the initial point to focus the microscope and then to slowly pan around the central gap, focusing especially on the edges of the coverslips (where animals are frequently observed) and the edge of the moss leaves. The focus should be changed slowly as the animals can sometimes be obscured by the moss. It takes a while to get used to looking for fine movements, but the user may see claws appearing from behind obstacles or see the edge of a body wiggling along. Over time, the observer will be able to differentiate between animal movement, and movement of air bubbles and particulates in the water.

Step 4: Identifying the organisms

It is likely that several different microscopic organisms will be seen in the moss sample, and they each have a fascinating lifestyle and set of adaptations for their environment. Just like tardigrades, some other microscopic organisms can survive extreme conditions, and have provided science with some key findings. Table 3.2 shows a list of some organisms that may be observed and some of their defining characteristics.

Organism	**Illustration**	**Photo**	**Description**
Tardigrade			4 pairs of limbs, each ending in claws. Moves by walking around. Generally around 0.5-1mm in size
Rotifer			May be freely swimming or anchored to surface or moss. Contracts and extends body Cilia on mouth piece may be visible.
Protozoa			Single celled organisms May be able to see cilia around surface. Wide range of shapes and colours

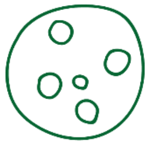
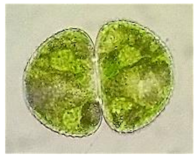
Algae			Single cellular or multicellular organisms Generally green May group together
Worms			Lots of different species Vary in colour

Table 3.2: An identification chart for some of the organisms which may be located in a moss sample. This is a brief and generic description aimed at a wide audience, some specific details are therefore not included. Illustrations by Gemma Kent. (Tardigrade: Goldstein Lab, 2007; Rotifer: Loarie, 2014; Protozoa: Picturepest, 2014; Algae: Atriplex82, 2015; Worm: Schley, 2010).

There are some distinctive movements which can help differentiate different microscopic animals from each other.

Firstly, the tardigrade body is made up of 5 segments; 4 segments which each have two legs and the head segment. They tend to crawl or swim through the water, their legs showing little co-ordination. Other movements that may be seen include the protrusion of their mouth piece from the head and the tardigrade curling up. When interacting with other things, the tardigrade often uses its hind legs to grip onto something, like a moss leaf, and use its other limbs to propel itself or gather food. They can also be seen dragging material along, gripped by their hind limbs. Due to the transparent nature of some species, their gut contents may be visible.

Rotifers have two distinct methods of movement. They will either be freely swimming (and move at quite a fast rate) or they will 'inch-worm', where they move in a concertina like fashion along a surface, contracting up their bodies before extending their anterior outward. They will anchor on and contract again, bringing their posterior closer to the front and retracting their rear-half. By repeating this pattern, the rotifer moves along a surface like a caterpillar. They also show another distinguishing behaviour, where they anchor onto something (such as the slide's surface, or a piece of moss) and will then stretch out and use the cilia on their head (or "crown") to produce a current and force particles into their mouth. Rotifers can be observed stretching out in multiple directions from one anchoring point to collect food from their environment.

Nematode worms move in either a thrashing manner or move their body in a sinusoidal like manner. This ‘S-wave’ pattern can be easily identified compared to the motions used by other microscopic animals. Other worms may also be seen.

Other organisms such as algae and protozoa may also be observed. These organisms are much simpler and tend to show almost random movement.

Using a smartphone or DSLR lined up against the eyepiece of a light microscope, images of microscopic creatures have been captured, shown in Figure 3.4.

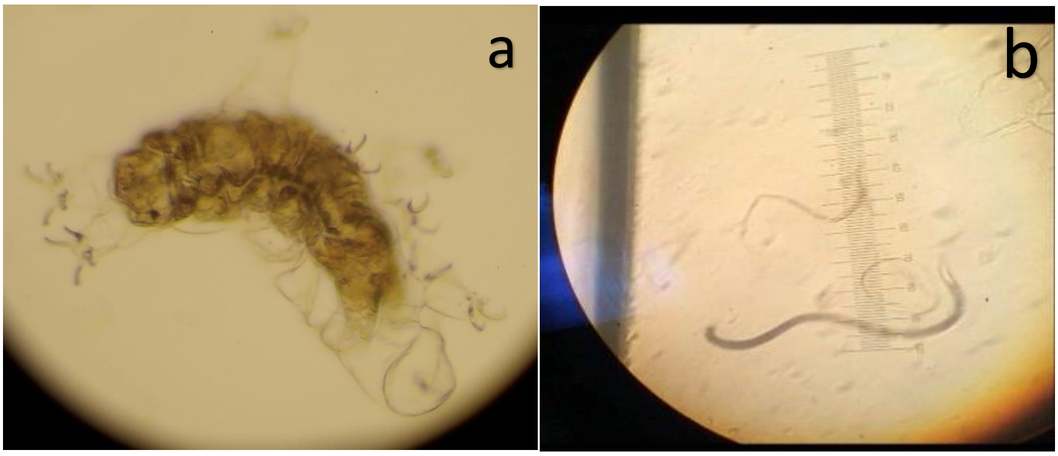


Figure 3.4: Microscopic creatures captured through eyepiece of light microscope. (a) A tardigrade seen in shed cuticle x200) captured on a Sony A58 DSLR; (b) nematode worms captured on an iPhone 5s.

Part Two: Utilising DIY hardware

While using existing technology to study microscopic animals is a great practical in itself, and opens up new areas of study for staff and students alike, the ability to be able to see this microscopic world using hardware built from scratch in a cross-curricula way makes the practical an even more educationally powerful project.

By using the hardware described in Chapter 2, these microscopic animals are easily visible using either a mobile phone or the Raspberry Pi based microscope (Figure 3.5).

Figure 3.5: Microscopic organisms seen under the DIY microscope, images captured from movie recorded on the Raspberry Pi. (a) A tardigrade seen moving on moss. (b) Nematode worms moving on slide. (c) A rotifer can be seen with its mouth piece extended toward the bottom of the image.

The distinctive movement patterns of the animals can also be seen using this DIY digital microscope, and still images or video footage can be produced easily in real time, allowing for students and staff to easily share their discoveries and discuss them in a biological context.

While the magnification is not as high as that available from the light microscope set up, it is possible to clearly see and identify microscopic organisms and to monitor their behaviour using equipment that can be built and maintained by students themselves.

To enhance the contrast between the animals and the leaf material, an infrared lightsource and the Pi's NoIR camera module can be used. This allows the capture of images such as the ones below in Figure 3.6. In these images, the animals can really clearly be seen, due to the high contrast between them and the background.

Figure 3.6: Microscopic organisms seen under infrared light, images taken from videos captured by DIY microscope utilising infrared camera. Images are blurry as stills are taken from movies. (a) Tardigrade, (b) Nematode worm, (c) Rotifer.

Possible Extensions

While this practical project is fairly simple in how it is carried out, there are many opportunities to extend it, based on the skills and needs of the group carrying out the work.

Some ideas that would be possible in a school setting include carrying out an investigation into how the number of tardigrades present in a sample varies dependent on where the sample is collected from (eg. shaded vs non-shaded environment, different altitudes, prevailing wind conditions, exposure to rain, etc).

As has been previously mentioned, tardigrades can survive long periods when dehydrated. Tardigrades enter the tun form and slow down their metabolic processes dramatically in order to survive. When the dehydrated organisms are exposed to water, the tardigrades (and to a degree both rotifers and nematodes too) can reanimate (as described earlier). This can be observed down the microscope in real time if the moss is sufficiently dry, and the observer can correctly identify a tun. The environment in which the tardigrade dehydrates has an impact on the likelihood of the survival of the tardigrade (Lundstrom and Stvensson, 2006). The tardigrades are far more likely to survive when dehydrated within the moss, as the process occurs slowly and gives the animal time to adapt to its changing environment and synthesise any necessary metabolic compounds.

Tardigrade hunting sessions have taken place in practical sessions at the University of Sussex using both fresh moss collected that day, and also using moss samples that have been stored in a dry plastic bag in the relatively warm laboratory for months, dehydrated to the point that the moss crumbles upon being touched. Even within these samples, tardigrades have been successfully seen to reanimate after being rehydrated by the addition of water.

A project could investigate how long the tardigrades can survive in dehydrated moss, looking at samples taken over a school year and see how the distribution of tardigrades changes in the older, more dehydrated moss compared to fresh moss, and how many individuals can be seen successfully coming out of the tun form.

An extension on this project could be to investigate how the humidity of the environment the moss is dehydrated in or stored in affects the survival rate of the tardigrades upon reanimation. This could be carried out using the DHT22 humidity sensor discussed in Chapter 2 to log the humidity and temperature in an environment.

Conclusions

The hardware developed for this project can open up interesting new avenues for the way that practicals could be carried out within the school environment, utilising DIY hardware to maximise the value for students and enhancing their knowledge base.

There are obviously things to consider, such as time constraints in a school day, but the core tardigrade hunting exercise could easily be completed within a typical school lesson.

The practical described in this chapter and the extension tasks suggested could be run as an after-school project or as a small part of a normal lesson and spread out over several weeks.

I hope that the open source nature of this equipment will act as an example to enable schools on tight budgets to think creatively about what practicals can be developed and enhanced by using open source hardware and software and how on-site cross-curricula teaching could give students a chance to put into practice skills from multiple subjects in one practical project.

YouTube Videos

The stills in these figures have been taken from movies that are available on the Microsco-Pi YouTube channel (https://www.youtube.com/channel/UCpicdd5Ppa-lDq-Z9PX40ew).

Chapter 4: A DIY Approach to Monitoring Ant Decision Making in Response to Chemicals in their Environment

Abstract

Despite their small size, ants are some of the most successful animals on the planet, estimated to make up around 15-20% of the world's terrestrial biomass (Schultz, 2000). Discoveries in ants have led to all sorts of scientific advancements and ants are very prominent in the public conscious.

This chapter uses the idea that ants are a well known, easily locatable study subject and creates a simple classroom practical session that looks at ant decision making using equipment that a school might have to hand. It then looks at how cheap, easily accessible hardware (such as the Raspberry Pi) could be used to set up an automated insect tracker that could record and analyse ant decision making in a school setting, and highlights some possible improvements to this set up.

Introduction

Ants are members of the Hymenoptera, the second largest order of insects. Ants have been referenced in the literature in almost every civilization; from ancient Hebrew literature (Proverbs 6:6-8, Holy Bible) through to comparatively recent authors such as Mark Twain (Twain, 1880). More recently, several movies have made ants their focus, from Dreamwork's *Antz,* Pixar's *Bugs Life* and Marvel's *Ant Man* movies.

Ants have adapted to successfully fill a range of ecological niches. This is partially due to their successful division of labour, and their ability to effectively communicate through a variety of methods, including audible cues (Hickling and Brown, 2000), physical interactions (Holldobler and Wilson, 1990; Franks and Richardson, 2006) and by chemical communication, for example to mediate nestmate discrimination (Lahav *et al*., 1999; Wagner *et al.,* 2000), mediate aggression (Liang, Blomquist and Silverman, 2001) and play a role in queen-worker interactions interactions (Dietemann *et al*., 2003; Vásquez, Schal and Silverman, 2008).

Ants show some of the most fascinating behaviour among the insects, for example they have been shown to farm other animals as a food source (Schneider *et al.,* 2013), self-medicate (Bos *et al*., 2015), farm crops (Ariniello, 1999), teach each other routes (Franks and Richardson, 2006) and take slaves (Gladstone, 1981). However, for the general public, ants are often seen as a pest that invade the home and leave a mess everywhere.

As part of this project, I wanted to see if everyday household products that have been claimed to be natural ant repellents are effective, and design a simple practical project that could be carried out in a school setting to test these products. There are many natural methods reported on websites to try and prevent ants from entering households, so as part of this project I tested some of the most popular suggestions; lemon juice and vinegar (The Frugal Life, 2014; Pest Kill, 2018; Reader's Digest, 2018). Through this practical, students (and staff) would explore to see if myths online could be trusted, and it would open up opportunities to think about the ideas of repeats, limitations and experimental design. Insects are easy to obtain and have very interesting behavioural adaptations, so they are an ideal subject for schools to study.

I also wanted to show that open source hardware and software could be used as a powerful, yet inexpensive, teaching tool. There have been many interesting ant studies that use very inventive methods to monitor ant behaviour, for example, the QR work by Mersch, Crespi and Keller (2013), and the RFID studies by Robinson *et al.* (2009). However, these experiments required expensive hardware and were very time consuming, two things to avoid in the average school environment.

By using simple, inexpensive technology and freely available software (described in Chapter 2), I wanted to encourage staff and students to produce their own equipment that could be used to direct their own learning. To this end, I wanted to show that the Raspberry Pi, with a camera attached, could be used to track an ant’s position, and allow students to use the recorded data to look at decision making in ants.

In addition to this, there is very little about behaviour in the school curriculum. Further, despite looking at animals in exotic environments (like the artic and the desert), students often do not look at local animals and their own interesting adaptations. I felt that this simple set up would allow for students to make a very visual, hands on link between animal behaviour and responses to their environment.

Therefore, to sum up, my two aims were:

1. To develop a simple ant experiment, using inexpensive, everyday products that could teach students about ant decision making.
2. To show that the Raspberry Pi, originally purchased for a microscopy project, could easily be used in a completely different way as an automated insect tracker that could monitor ant decision making.

Part One: Developing a simple experimental assay

Study Species

This project focuses on two common ants found in Britain; *Lasius flavus* and *Lasius niger*, shown in Figure 4.1. These ants were chosen as they are readily available for schools to collect and they are some of the more common species in the literature.

Figure 4.1: *Lasius flavus*, the yellow meadow ant (left) (Specimen CASENT0179923 from antweb.org; Padro, 2011a) and *Lasius niger*, the common black ant (right) (Specimen CASENT0179929 from antweb.org; Padro, 2011b).

L. flavus is known as the yellow meadow ant and is often found in grassy areas. This species builds distinctive mounds in grass, and are found all across the Northern Hemisphere. *L. flavus* has a yellow-orange colouration, although this colour tends toward brown with age, and can often be mistaken for red ants. Workers range in size from around 2-4mm in length. As underground foragers, *L. flavus* are not often seen on the surface. As a member of the Formicinae (a subfamily of the Formicidae), *Lasius flavus* produces formic acid as a defensive substance, and fires it at enemies through its acidophore.

L. flavus were collected from a field bordering the University of Sussex campus. The top layer of their mounds were cut into at right angles using a shovel, and this top layer of soil was lightly shaken into a fluon lined plastic box (measuring 30x30x10cm). Then, additional clumps of soil were picked up and shaken into the box. The aim was to collect as many workers and brood as possible, but without putting too much soil into the enclosure. A lid (with mesh-covered air flow holes) was placed on the box, and the ants were kept in a shelfing unit. All experimental colonies were queenless.

L. niger is known as the common black ant, or the pavement ant and is found across most of the Northern Hemisphere. It has a very closely related species, *Lasius japonicas*, in Asia. *L. niger* workers are bigger than *L. flavus*, around 3-5mm and have a jet black gaster, with some brown bands on their legs. Like *L. flavus*, *L. niger* are also Formicinae ants, and use formic acid as a defence against threats. However, they have been more intensely studied; in particular, the profile of the chemicals on their cuticle and their trail pheromones are well established in the literature (Lenoir et al., 2009).

L. niger colonies were collected from underneath paving slabs across the University of Sussex campus. These were carefully lifted and a handheld vacuum cleaner was used to collect as many workers (and some brood) as possible. These were then ejected into a fluon-lined plastic box, with a plaster base and plaster nest inside and covered with a lid containing mesh-covered air flow holes.

Both species of ants were given water tubes (capped with cotton wool) and fed protein jelly regularly.

The test arena (shown in Figure 4.2a) was built using three clear plastic petri dishes, each 9cm in diameter and 1cm in height. Dish A and B had their bases carefully removed using a sharp knife, and were then internally coated with fluon. Fluon is a readily available liquid polymer which acts as a barrier that insects find very difficult (although not impossible) to cross. Dish C acts as the base and was also lined with fluon. Then, the rims of Dish A and B were stacked on top of Dish C and the entire structure was covered externally with silver duct tape. This held Dish A and Dish B in place, removed visual information from the ant and also reduced glare from the plastic refracting incoming light.

A simple paper disc (the exact diameter of the arena) was inserted into the arena using forceps. This paper disc was divided in half with a black line, which acts as the key reference point for the decision making (Figure 4.2b).

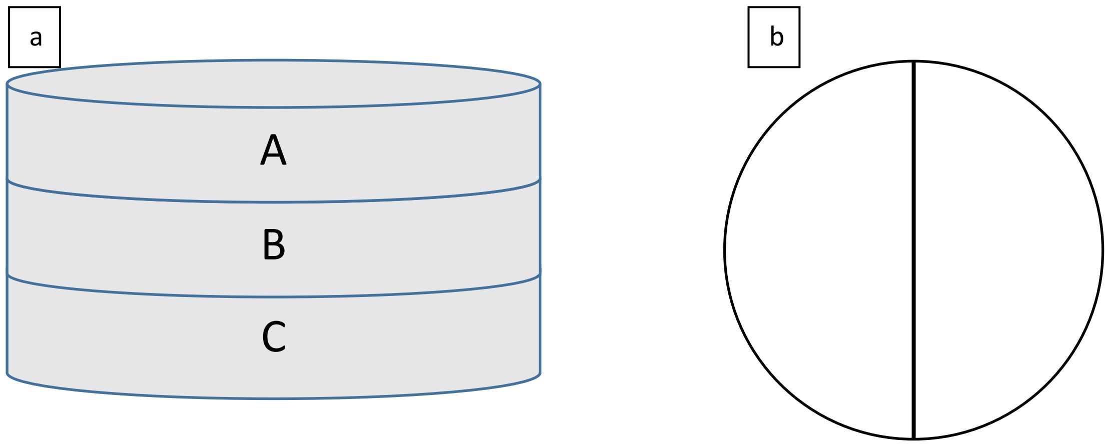


Figure 4.2: The resources required to build the test arena. (a) 2 petri dishes (A and B) have their bases removed and stacked upon a third dish (C), held together by duct tape and coated internally with fluon. (b) Paper discs were printed that could fit inside the arena, with the centre of the circle marked with a black line.

Trial Substances

In this practical, lemon juice and vinegar were chosen as the test materials (Table 4.1). Both are easily available and non-harmful, which makes them useable in a school setting. Water was chosen as the control substance, to ensure that the ants were not just responding to the presence of liquid in their environment.

Treatment Number	Household Substance	Species Tested
1	Control - None	*L. niger*
2	Control - None	*L. flavus*
3	Control – Water	*L. niger*
4	Control – Water	*L. flavus*
5	Lemon juice	*L. niger*
6	Lemon juice	*L. flavus*
7	Vinegar	*L. niger*

Table 4.1: Household substance trial tests.

Trial Procedure

An everyday substance that is supposed to act as a DIY ant repellent was applied to half of the paper disc using a soft paintbrush in a repeated way for each test. The paper disc was placed into the petri dish arena. A foraging ant was selected from the colony box and transferred to a fluon lined tube using soft forceps. This tube was placed open-end down in the centre of the arena and the ant was left for 1 minute to acclimatise to the apparatus. Then, the tube was removed and the ant was monitored for 5 minutes before being removed from the arena and placed in a holding box until all the trials were completed. The paper disc was removed after each test, and between different substances the fluon was reapplied.

While the ant is exploring the arena, a stop watch is used to record the amount of time the ant spends on each side of the arena and this can be collated across the repeats.

Part Two: a DIY insect tracker

Once I had established a repeatable, simple ant decision making assay that was suitable for use in a school setting, I then wanted to see if I could develop a novel insect tracker that would be suitable for a school setting in terms of cost, ease of use and reliability.

While there have been lots of successful insect trackers used in a range of contexts before, these generally rely on expensive hardware and labour-intensive setting up. For example, the RFID work by Robinson *et al*. (2009) and the QR study by Mersch, Cerspi and Keller (2013) have uncovered very interesting patterns in ant behaviour, but unfortunately are very time consuming and expensive, both issues to be avoided for an educational based project. Instead, with budget constraints and severe pressures on teachers' time in mind, this project aimed to identify something inexpensive, well supported and easy to use, like the Raspberry Pi.

By using simple, inexpensive technology, I wanted to encourage staff and students to produce their own equipment to direct their own learning. Everything used in this study was purchased from large name online retailers, and all software was free to download, making it accessible for schools on a tight budget.

Hardware and Software

The tests were captured on a Raspberry Pi 2 Model B running Raspbian Jessie (2015-09-28). The official Raspberry Pi camera (Version 1) was used (5-megapixel OmniVision OV5647 sensor). Despite its low cost, the Raspberry Pi is a very capable computer, able to utilise OpenCV to analyse video in real-time. The official Raspberry Pi 7 inch touch screen was used as the main user input. A slightly modified version of the Motion-Track programme (Pageau, 2018; released under LICENSE) was used to collect the location of the ant in real time at approximately 10 frames per second, capturing the video stream in a 320 x 240 pixel window. Co-ordinate files were saved in a text file, subsequently uploaded to a Dropbox account using Dropbox Uploader (Fabrizi, 2016) and analysed on a Windows 10 computer in Microsoft Excel. Statistical analysis was carried out in IBM SPSS Statistics 23.

Automated Ant Tracking

In this study, the ants are tracked using an automated process. The Motion Track programme compares two frames (e.g. Frame T and Frame T-1) and, if the two frames are sufficiently different (i.e. they meet the threshold criteria set by the user) a green circle marks the target. This happens at about 10 frames per second, and the X co-ordinate of the ant is logged in a data file. This data can be displayed visually, as in Figure 4.3, where the far left of the image is given the value 0, the midpoint of the test arena is given the value 160 and the far right of the image is given the value 320. In this example, the ant was most frequently detected near the centre of the arena.

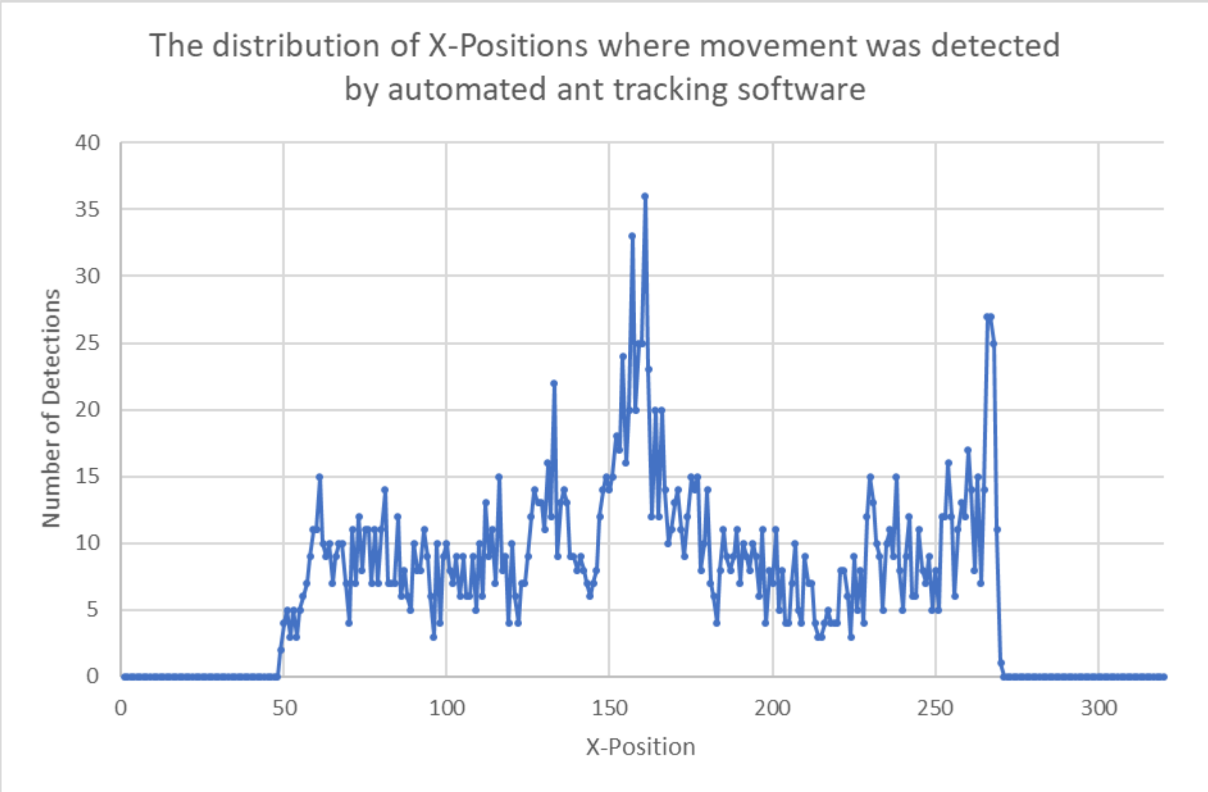


Figure 4.3: A graph showing the position of the ant within an arena. It was most frequently detected moving near the centre of the arena, with a high amount of motion detected near to the right hand of the arena too.

Calibration of hardware

If an ant is completely stationary between frames, then the programme is unable to detect a position for the ant itself. This is generally not an issue, as the ants spent very little time completely still. Even the movement of antennae is generally (though not always) enough for the ant to be detected.

Just to ensure that the issue of an ant being missed wouldn't affect the results too much, a calibration experiment was carried out. Blank paper discs with no substances applied to them were placed into the arenas, and the test was carried out as per the experimental procedure. However, while the experiment was running, stopwatches were used to record the percentage of time the ant spent on the right hand side of the arena, and this was compared to the output from the Raspberry Pi. This test was repeated 15 times and the results of this test are presented in Figure 4.4.

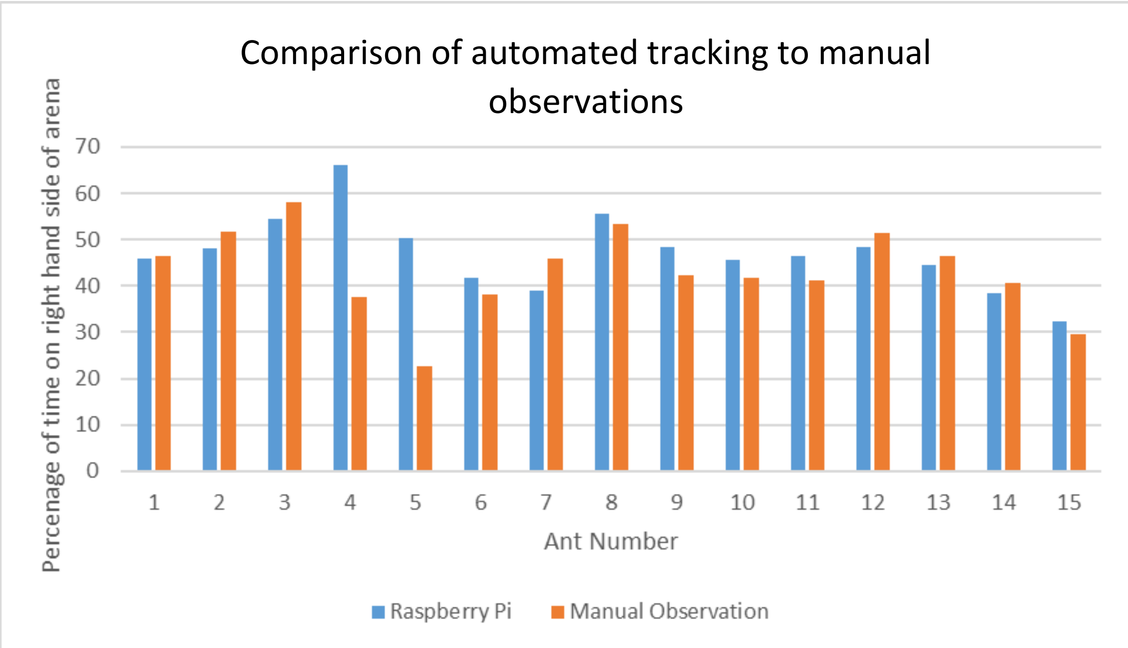


Figure 4.4: On the whole, the Pi is fairly accurate, with differences between manual observation and computational analysis falling within a few percent across all 15 trials. The mean difference is 3.88% across all 15 calibration tests. The results from the Raspberry Pi are not significantly different to the results measured by eye (T-test, p=0.2274).

Generally, the percentage of time (measured by eye and stop watch, shown in orange) or frames (detected on the Raspberry Pi and shown in blue) matched up fairly closely, with a few percent difference between the two.

For ant number 4 and ant number 5, there is quite a large difference. This is because the ants in those two tests spent a long time stationary on the left hand side, but spent more time moving on the right hand side. It is important to note that this was only an issue in 2 out of 15 trials in the calibration, and across the experimental trials was rarely seen, but it is a limitation which could be discussed in an educational setting.

Results

Statistical Analysis

Statistics were carried out in SPSS 23. T-tests were chosen because they are (a) fairly robust in terms of variance, (b) the simplest way of analysing the change between the control and the treatment is to compare the means and (c) it's a simple statistical test, suitable for school students.

Controls

Treatment Number 1

L. niger workers were placed in an untreated arena. Results shown in Figure 4.5.

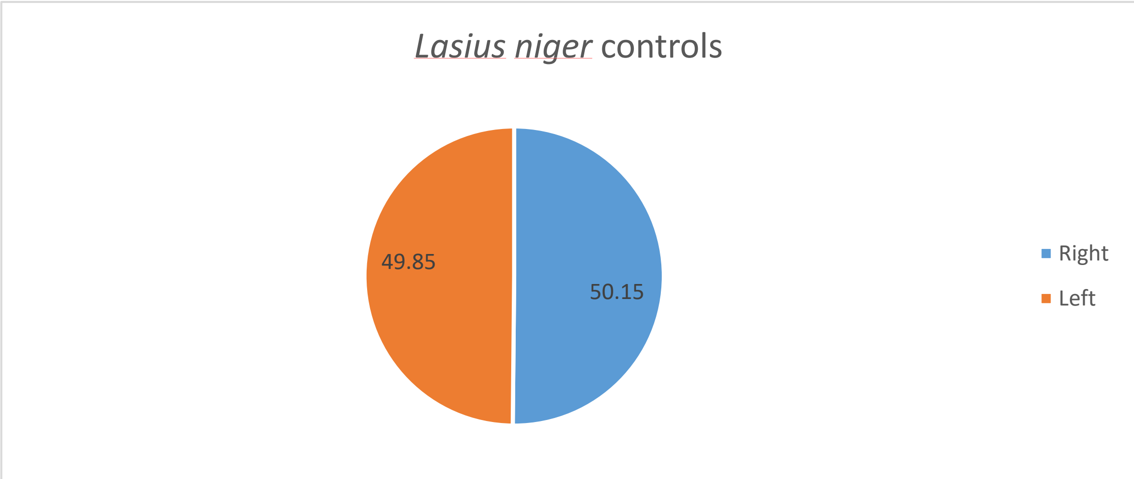


Figure 4.5: *L. niger* workers were placed in an untreated arena. Graph shows mean number of frames spent on either side of the arena (n=20).

Treatment Number 2

L. flavus workers were also placed into untreated arenas, and the results of this control are shown in Figure 4.6.

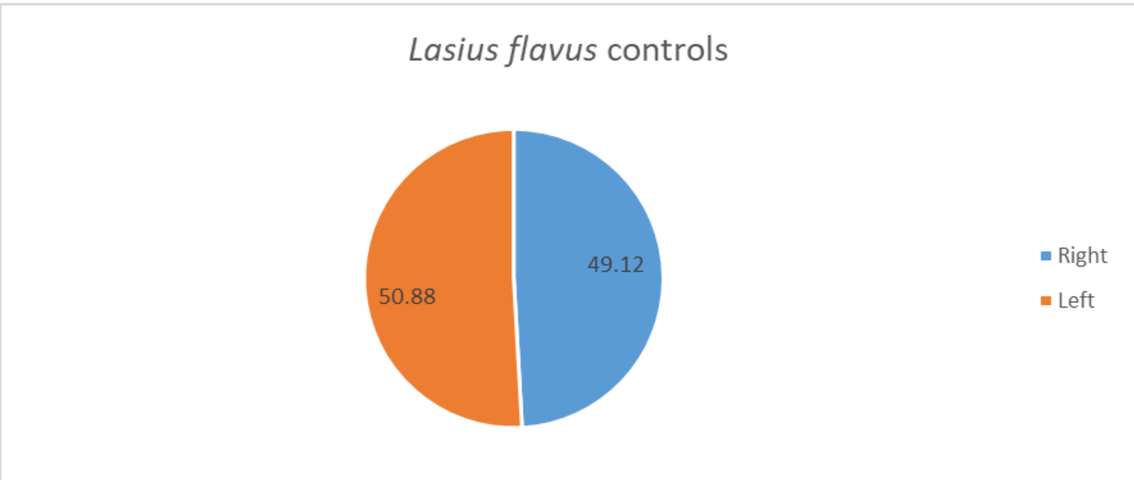


Figure 4.6: *L. flavus* workers were placed in an untreated arena. Graph shows mean number of frames spent on either side of the arena (n=20).

The results of Treatment 1 and Treatment 2 show that there is no inherent bias in the set up. When in an arena that has not been exposed to any substances, the ants spend roughly 50% of their time on either side of the paper disc. There was no internal or external cue that caused the ants to spend more time on a particular side.

Treatment Number 3

To test whether just the wetness of the substrate affected the behaviour of *Lasius* spp., half of the paper disc was treated with mineral water. Initially I compared the time on the water compared to the time on the dry surface. Later, when analysing the results of the different substances, I compared the percentage of frames on the trial substance to those on the water. This ensured I accounted for the effect of surface moisture.

Lasius niger showed a clear preference to being on the dry side of the paper, shown in Figure 4.7.

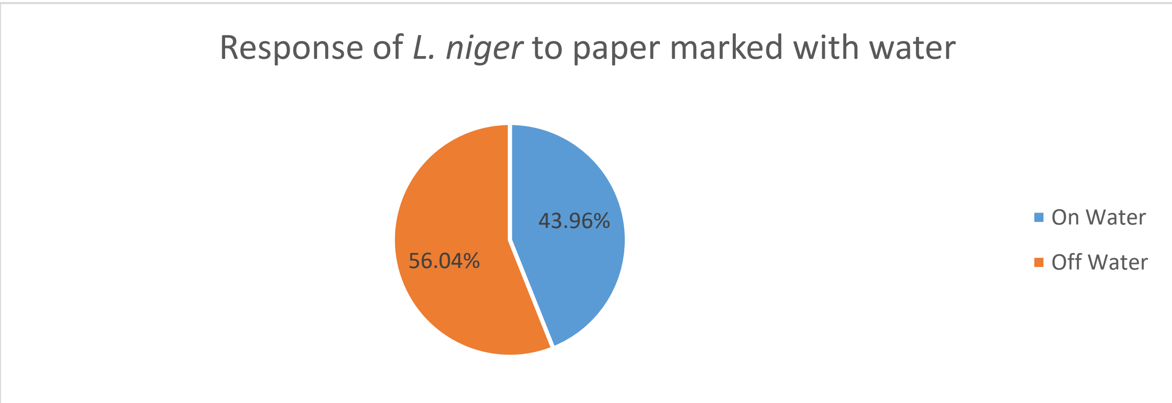


Figure 4.7 *L. niger* workers show a statistically significant preference for the dry region, compared to the wet half of a paper dish that has 50% of the surface lightly brushed with water (n=20, T-test p= 0.016).

Treatment 4

Lasius flavus showed a slight preference to being on the wet side of the dish (shown in Figure 4.8).

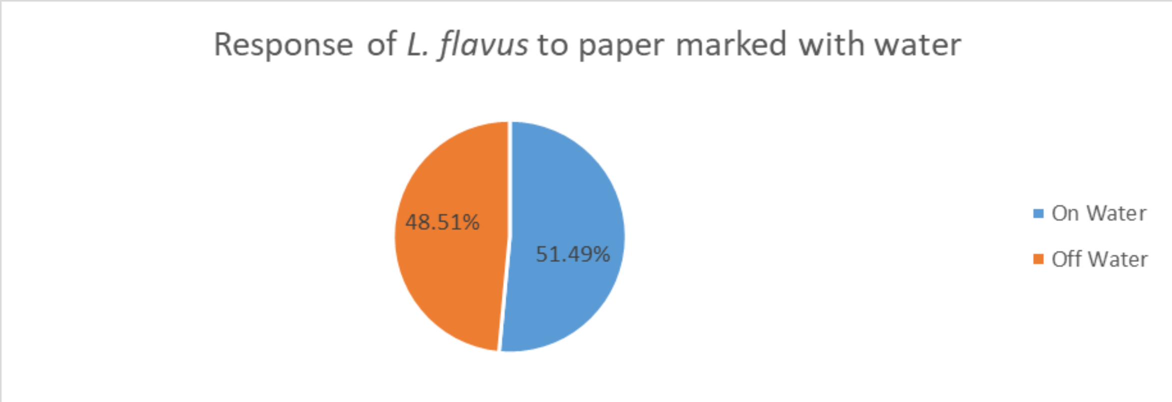


Figure 4.8: *L. flavus* workers were placed in an arena in which half the arena had been lightly brushed with water. The workers showed a slight preference to the wet side of the dish, but comparing wet to dry, this showed to be non-significant (n=20, T-test p=0.459).

Treatment 5 – Lemon Juice

L. niger workers were exposed to lemon juice, and showed a preference to the side not treated with lemon juice.

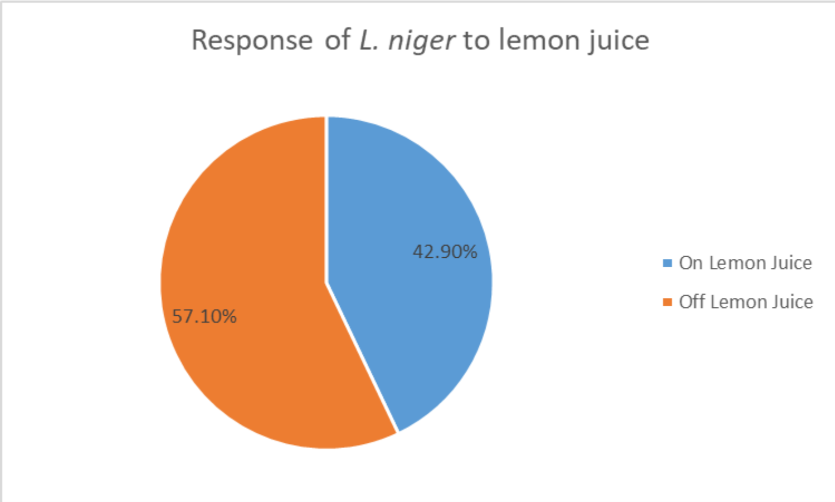


Figure 4.9: *L. niger* workers were exposed to a paper disc that had been brushed with lemon juice. The ants showed a preference to the dry side, but this was not statistically significant to the wet/dry control, Treatment 3, (n=20, T-test, p=0.777).

Treatment 6 – Lemon Juice

When *L. flavus* workers were exposed to lemon juice there was no statistically significant difference compared to the wet/dry control test. The ants showed slight preference to the dry (non-lemon) side, shown in Figure 4.10.

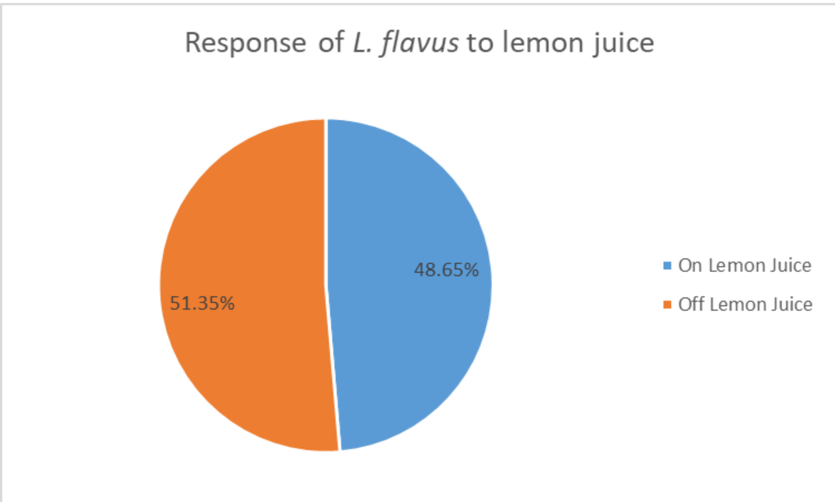


Figure 4.10: *L. flavus* workers were exposed to lemon juice. The ants showed a slight preference to the dry side, but this was not significant compared to the wet/dry control, Treatment 4 (n=20, T-test, p= 0.555).

Treatment 7 – Vinegar

In treatment 7, *L. niger* workers were exposed to vinegar. While the ants appeared to show avoidance of the vinegar (this was the most definitive response to a substance), there was no significant difference compared to the wet/dry control (Treatment 3) (Figure 4.11).

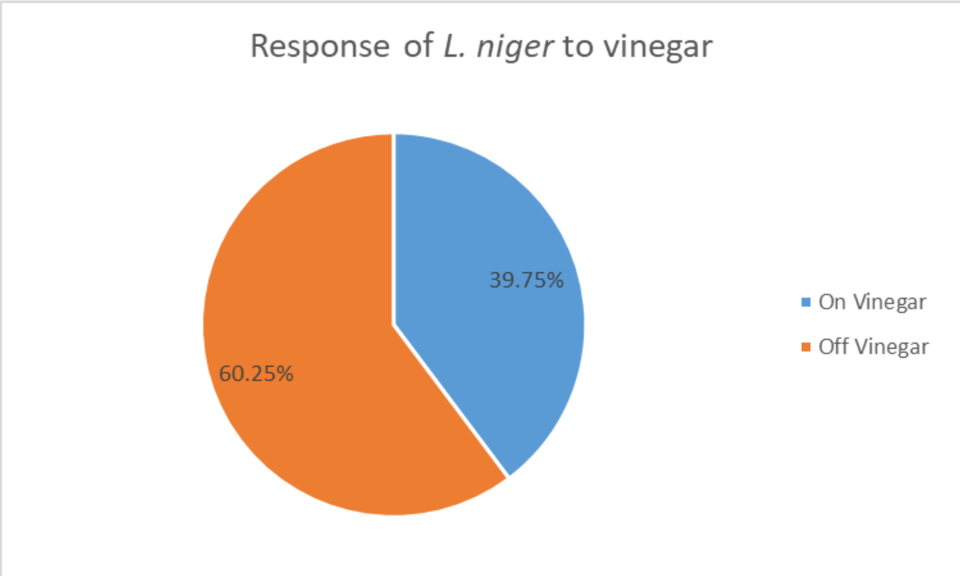


Figure 4.11: *L. niger* workers in an arena in which vinegar had been placed showed a clear preference for the dry side of the arena, but this was not significantly different to the wet/dry control test (Treatment 3) (n=20, T-test, p=0.270).

All of the results are summarised in Table 4.2 below.

Household substance	Tested species	Percentage of frames on trial substance	Percentage of frames not on trial substance	p value
None (Treatment 1)	*L. niger*	(left hand side)[1]	(right hand side)[1]	n/a
None (Treatment 2)	*L. flavus*	(left hand side)[1]	(right hand side)[1]	n/a
Water (Treatment 3)	*L. niger*	43.96	56.04	0.016[2]
Water (Treatment 4)	*L. flavus*	51.49	48.51	0.459[2]
Lemon (Treatment 5)	*L. niger*	42.90	57.10	0.777[3]
Lemon (Treatment 6)	*L. flavus*	48.65	51.35	0.555[4]
Vinegar (Treatment 7)	*L. niger*	39.75	60.25	0.270[3]

1 – as these are controls, they cannot be "on the trial substance", therefore the ants were monitored to find the percentage of frames they were moving on the left hand and right hand side of the arena.

2 – comparing the wet side to the dry side across all trials.

3 – compared against Treatment 3

4 – compared against Treatment 4

Table 4.2: Summary of the results from the ant behavioural trials.

Discussion

As far as I am aware, this is the first time that a Raspberry Pi computer has been used to observe the behaviour of an animal in response to chemicals in its environment. Overall, the hardware worked as expected, providing really useful data, and cut down dramatically on time needed to input manually collected timings.

However, there were some issues with the experimental design which could be masking more conclusive results, due to specific behaviours that relate to ants. These issues could be overcome with some adjustments to the design, and also provide opportunities to think about experimental design and limitations.

Firstly, there is the issue of edge (or wall) following. A study by Dussutour, Deneubourg and Fourcassie (2005) has shown that individual ants actively seek out edges to stay in contact with, and that this individual choice is amplified over time in a colony context. Although this test only looked at individual ants, edge following was clearly seen during the trials. In the circular arenas used in this study, there is a continual wall for the ants to remain in contact with and they can just circle the test environment. The physical sensation of being near the wall may be more important to the ant than avoiding the marked area. At the very least, the role of wall following might affect how likely an ant is to avoid the trial condition. Each individual would have to make a choice between avoidance of a substance in an environment and spending time in contact with the edge of the arena.

Secondly there is the issue of escapes. Actual escapes from the arena were rare due to the fluon lining the walls, but ants were able to climb onto the side. They generally fell off after a short amount of time, but if an ant escaped, it was discounted from the experiment. The fluon was replaced regularly throughout the tests to ensure a good covering up the arena walls. Occasionally, ants escaped under the paper disc, and again, these were discounted from the experiment. Escaping under the paper was more commonly seen in *L. flavus*, due to their smaller size and underground nature, while *L. niger* were more likely to climb and escape the arena itself.

In this experiment, avoidance was defined as time spent in the half of the arena that had not been treated with the chemical. However, this definition doesn't allow for the fact that an ant could be avoiding the treated paper by attempting to climb up the arena walls. If avoidance is defined as time not spent in direct contact with the treated surface, then the results could be even more substantial, and the results displayed in this chapter could be under representative of the actual effect.

In its current state, the programme only records an X position for the ant, however, at any given X position, the ant could have one of many Y and Z positions (e.g., the ant could be climbing on the wall of the arena). In hindsight, also tracking the Y position to give an absolute 2D position of each ant would provide more helpful data, as the effect of wall following could be quantifiably established.

Both wall following and wall climbing could be hiding a more significant result, and could explain why T-Mazes are more commonly used to study ant behaviour. Indeed, this same set up could be used to monitor and detect ant choices made on a T-Maze with a little modification.

A T-maze (Figure 4.12) uses plastic/other materials to create a (generally) elevated path for ants to walk across and explore, and at the far end (i.e. the one furthest from the starting point) the ants make a choice to either go left or right. These arms can be treated before the experiment so the ant will make a choice between two situations. Ants can be trained, for example, to a food source on one arm, and other time, the workers will repeatedly make that choice at the junction, through trail laying behaviours. You can then use this to collect trail marked papers to use in other tests, or move the marked paper to influence decision making. These T-mazes can become more complicated, to find more complex decision making patterns.

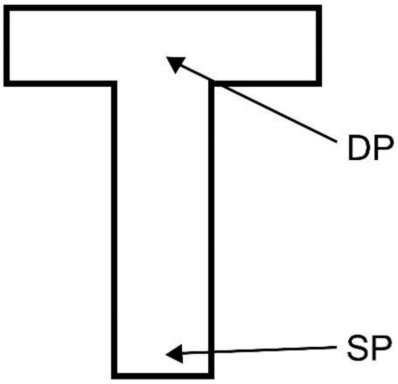


Figure 4.12: A T-Maze, with the starting point (SP) and decision point (DP) marked with an arrow. When a forager arrives at the decision point, it has to choose between the left or right arm.

T-mazes are quite hard to put together, quite flimsy and wouldn't be ideal in a school setting. They take time for the ants to arrive on them, and ants are prone to escaping by climbing or falling. However, a DIY approach could be used to make a simple binary choice for the ants, which may produce a more reliable outcome in a school setting. Toy bricks could be used to produce a thin corridor with treated paper on the arms after the decision point. The plastic bricks could be coated with fluon to prevent escapes, and the blocks could be reused and would be cheap compared to producing acrylic bridges.

Even with the design of the arena put aside, there are some important things to take from the study.

Firstly, the control tests came very close to 50:50 in every trial. This matches other results where ants are given free choices (Devigne and de Biseau, 2012) and shows that there was no inherent bias in my apparatus.

When looking at the results of the treatments, there are some interesting insights that can be taken away. *L. flavus* doesn't really show much of a change between the response to water and to lemon, although there is more of an avoidance of lemon juice. However, *L. niger* shows a much stronger avoidance of any wet surface, and this avoidance increases as the treatment increases in similarity to components of ant defence substances. Vinegar is made up of acetic acid, a simple carboxylic acid (CH_3COOH), a compound closely related to formic acid (HCO_2H). Formic acid is the defensive compound of *L. niger*. The presence of vinegar caused strong avoidance by *L. niger*. Again, while this result was not significantly different from the control tests carried out, there was an increase in avoidance and this test provided the most substantial result throughout the experiment.

L. flavus did not show a strong avoidance or attraction to either wet surfaces in general, or to the lemon juice. *L. flavus* uses a trail component that has a similar scent to lemon. I had expected to see some sort of attraction to the lemon juice, but this was not apparent. A study by Steinmeyer, Pennings and Foitzik (2012) suggests that *L. flavus* do not show particularly aggressive responses to non-nestmates, so the presence of foreign substances may not necessarily elicit much reaction at all. Due to time constraints, the effect of vinegar was not tested on *L. favus*, but it would be interesting to see if the same trend continues.

Ideally, I was hoping to see very clear differences in the behaviour of ants in response to different household items, rather than the non-significant (but trend revealing) results that were obtained. However, this just means that the online methods may not be as effective as claimed, or maybe not enough was being used. There's also the noise caused by edge following, stationary ants and wall climbing that all introduce difficulties in coming to a clear cut answer to the question "do these every day household items act as effective natural ant repellents?"

Clear cut results could have been more beneficial for teachers and students, although the limitations of the current set up could prompt discussion in classes, introducing the ideas of repeats, reliability and improving designs as you go along and to teach the important scientific fact that experiments often don't "work".

The Pi itself worked as it was expected to, and was very reliable, capable of tracking individual worker ants. Once a school has invested in a Raspberry Pi, it isn't limited to just doing one job. A whole range of other potential practicals become possibilities, including monitoring biodiversity in a school playground, DIY light gates or pretty much anything else that a school might want to do, using the wide range of peripherals that can be purchased and connected to the Pi.

In conclusion, the experiment described here is a good starting point for a practical set up for schools. The lack of a clear, decisive attraction to or avoidance of some easily obtainable household products – though there may still be some household substance that would give clearer results – is not in itself a negative thing.

The hardware for this project could be used to teach other aspects of animal decision making; a solid potential opportunity could be to use the set up to discuss decision making in woodlice, with respect to moisture in the environment. The effects of light and dark could also be studied, with some slight adaptations to the setup of the equipment. Another option would be to use the T-maze described earlier which may provide cleaner data with less noise introduced.

Chapter 5 – Conclusions and Further Work

There has been a lot covered throughout the work that has gone into this Masters Project. While fairly wide ranging, the main premise has been clear; can we produce, using easily available resources, pieces of equipment that can be used to help boost science education in the classroom in a cost effective and cross-curricula way?

While the individual studies contained in this thesis have not produced particularly successful results they have led to some interesting avenues for further study and have shown the potential of the hardware developed in this project.

It is really important to consider that for less than £100, I have shown it is possible to purchase the necessary hardware to make a multi-purpose tool that could be used for all sorts of experiments in the science classroom, from microscopy to studying animal behaviour. With some basic coding (all of which can be learnt online easily), a school science technician could easily set up a classroom visualiser, a temperature probe, a humidity sensor, a soil saturation sensor or a DIY digital microscope. The hardware can be adapted to be used in a range of practicals, across the sciences, for all age groups.

I have also shown that the hardware used in this project can successfully track insect movement patterns and use this to analyse ant choice making behaviour. This would cost a lot more using conventional hardware and the equipment is nowhere near as portable, easy to use or customisable as the set up described here.This could be an ideal method for students studying animal behaviour, right up to University level.

This project has also shown that simple DIY equipment can be used to visualise the microscopic world and to monitor the behaviour of animals less than half a millimetre in size to a high level of accuracy, while simultaneously recording information about the local environment of the animal.

The equipment produced in this project has been used in a range of settings, from the laboratory, in schools in teaching sessions, as part of undergraduate teaching projects and as part of the British Science Festival 2017 in a hands-on microscopy session where it has received positive feedback.

I had hoped that the ant choice paradigm would lead to repetitive choices being made, specifically with the home-made substance tests, as this would be easily reproducible (and safe), making it ideal to carry out in a school science lesson. While the hardware worked exactly as it was programmed to perform, the ants do not seem to make a clear choice to avoid certain substances when exposed to them. There may be other substances (like essential oils) which may lead to ants showing avoidance behaviour and lead to a statistically significant result. However, even without a clear choice, the project described in Chapter 4 can be used to teach students about the importance of controls and repeating a test to increase the validity of results, and teach some basic statistical analysis.

It has been very interesting to observe ant behaviour using this equipment. There are several studies that could use this equipment to look at choice making behaviour, especially around biological compounds produced by ants that could have interesting pharmacological properties.

Some species of ants, including *Lasius flavus* produce a chemical called micromolide (Butterfield, 2017). Micromolide has been shown to have anti-microbial properties, including the ability to combat tuberculosis bacteria. It appears that this compound is laid by *L. flavus* when it walks around and is passively laid through pores on the legs.

If this is the case, paper that has been trampled by *L. flavus* should have micromolide present. It is believed that the ants use this to help protect the colony from disease. If this is the case, do worker ants specifically treat larvae with the chemical? An interesting choice test would be to see if workers move larvae into an area that contains trampled paper and is therefore perhaps more sterile. Another choice test would be to investigate how ants respond to paper treated with artificially produced micromolide as opposed to an area that has been untreated (or actively treated with some negative compound).

This could lead to further studies into how ants use chemicals to treat diseases. It has already been shown that ants can learn to self-medicate and (individually) show self-control (Wendt and Czaczkes, 2017) so it would be interesting to know how ants use chemicals that they produce in terms of keeping future generations safe.

In hindsight, some other practical tests could have been carried out to look at decision making behaviour in ants. While looking at ants, some trail pheromone marked paper could have been used to show the ants ability to detect and follow trail pheromone compared to virgin paper. This hopefully would have led to a more clear-cut result and does not require any chemicals to carry out (but does require a large nest, bridges to be constructed and would take significantly longer to do).

Other animals could also have been studied using the same equipment. For example, monitoring the choices of woodlice presented with a damp and a dry environment (making sure that humidity in the air is also recorded simultaneously to interpret the results). This practical would also need to think about how to record the times that movement is detected to account for any time that the woodlouse remains stationary when it has found its preferred environment.

In terms of educational usage, there are almost endless uses for the Raspberry Pi in a school setting. These have been alluded to throughout this thesis, in terms of additional hardware that could be purchased or adapted for use. Along with 3D printing and robotics, there are hundreds of different projects that could be carried out in a school and be used across many subjects in the curriculum.

Utilising the external hardware available for the Raspberry Pi, there are experiments to enhance the knowledge of tardigrade behaviour that could be carried out. Does the temperature in the environment affect tardigrade behaviour? Do individuals change speed in response to a drop or increase in temperature? Do individuals aggregate in response to such conditions? Does the humidity of the environment affect tardigrade aggregation? Does the size of the arena alter this? Taking this even further, looking at aggregation behaviours over a long-time period using this DIY microscope could give us insights into how groups of tardigrades cope with desiccation and how they interact with each other in close proximity.

Overall, this thesis has shown that the Raspberry Pi (and other similar small computers) can be utilised to teach traditional science and computer science in a novel way, at all levels from Primary through to University level education and beyond to a range of people with no computer science background. My hope is that this thesis will encourage science educators to experiment with open source hardware to explore new ways to teach students and to increase their digital literacy, a key skill for future generations of scientists.

References

Adafruit. (2018a) *Raspberry Pi Sense Hat - for the Pi 3 / 2 / B+ / a+.* Available at: https://www.adafruit.com/product/2738 (Accessed 3 September 2018).

Adafruit. (2018b) *Thermocouple Amplifier Max31855 Breakout Board (Max6675 Upgrade)*. Available at: https://www.adafruit.com/products/269 (Accessed 3 September 2018).

Adafruit. (2018c). *DHT11 Basic Temperature-Humidity Sensor + Extras.* Available: https://www.adafruit.com/products/386 (Accessed 3 September 2018).

Adafruit. (2018d) *DHT22 Temperature-Humidity Sensor + Extras.* Available: https://www.adafruit.com/products/385 (Accessed 3 September 2018).

AQA. (2016). GCSE Computer Science (8520) Specification. Available at: https://filestore.aqa.org.uk/resources/computing/specifications/AQA-8520-SP-2016.PDF (Accessed 3 September 2018).

Ariniello, L. (1999). 'Protecting paradise fungus-farming ants ensure crop survival with surprising strategies and partnerships'. *BioScience,* 49(10)**,** pp. 760-763.

Astro Pi. (2018). *Astro Pi*. Available at: https://astro-pi.org/ (Accessed 3 September 2018).

Atriplex82 (2015). *File:Cosmarium201512081550.JPG.* Available at: https://commons.wikimedia.org/wiki/File:Cosmarium201512081550.JPG (Accessed 9 August 2019).

BadenLab. (2014). *Biropette: customisable, high precision pipette.* Available at: https://www.thingiverse.com/thing:255519 (Accessed 3 September 2018).

Bardski, G. (2000). *The OpenCV Library* Available at: www.drdobbs.com/open-source/the-opencv-library/184404319?queryText=Bradski (Accessed 3 September 2018).

Barker, D., Ferrier, D. E., Holland, P. W., Mitchell, J. B., Plaisier, H., Ritchie, M. G. & Smart, S. D. (2013). '4273π: Bioinformatics education on low cost arm hardware'. *BMC Bioinformatics,* 14**,** 243.

BBC. (2016). *BBC Micro:Bit groundbreaking initiative to inspire digital creativity and develop a new generation of tech pioneers* Available at: https://www.bbc.co.uk/mediacentre/mediapacks/microbit (Accessed 3 September 2018).

BBC. (2018a). *School spending on pupils cut by 8%, says IFS*. Available at: https://www.bbc.co.uk/news/education-44794205 (Accessed 3 September 2018).

BBC. (2018b). *Schools cutting posts amid 'funding catastrophe'*. Available at: https://www.bbc.co.uk/news/education-43569389 (Accessed 3 September 2018).

Beisser, D., Grohme, M. A., Kopka, J., Frohme, M., Schill, R. O., Hengherr, S., Dandekar, T., Klau, G. W., Dittrich, M. & Müller, T. (2012). Integrated pathway modules using time-course metabolic profiles and EST data from *Milnesium tardigradum*'. *BMC Systems Biology,* 6(1)**,** p. 72. Available at: https://bmcsystbiol.biomedcentral.com/articles/10.1186/1752-0509-6-72 (Accessed 3 September 2018).

Beta News. (2017). *Raspberry Pi founder Eben Upton talks sales numbers, proudest moments, community projects, and Raspberry Pi 4 [Q&A]* Available at: https://betanews.com/2017/07/19/raspberry-pi-eben-upton-qa/ (Accessed 3 September 2018).

BlueJ. (2018). *Bluej.* Available at: https://www.bluej.org/ (Accessed 3 September 2018).

Bonnet, K., (1773). *Herrn Karl Bonnets Abhandlungen Aus Der Insektologie*. Translated by: Johann August Ephraim Goeze. Halle: Bey J.J. Gebauers Wittwe und Joh. Jac. Gebauer. Available at: https://doi.org/10.5962/bhl.title.47534 (Accessed 3 September 2018).

Boothby, T. C., Tenlen, J. R., Smith, F. W., Wang, J. R., Patanella, K. A., Osborne Nishimura, E., Tintori, S. C., Li, Q., Jones, C. D., Yandell, M., Messina, D. N., Glasscock, J. & Goldstein, B. (2015). 'Evidence for extensive horizontal gene transfer from the draft genome of a Tardigrade'.

Proceedings of the National Academy of Sciences, 112(52)**,** pp. 15976-15981. Available at: http://www.pnas.org/content/112/52/15976.long (Accessed 3 September 2018).

Bordenstein, S. (2017). *Tardigrades (Water Bears)* Available at: https://serc.carleton.edu/microbelife/topics/tardigrade/index.html (Accessed 3 September 2018).

Bos, N., Sundström, L., Fuchs, S. & Freitak, D. (2015). Ants Medicate to Fight Disease. *Evolution*, 69(11), pp.2979-2984. Available at: https://doi.org/10.1111/evo.12752 (Accessed 3 September 2018).

Butterfield, T. (2017) The Behaviour and Chemistry of Recruitment and Alarm in Social Insects (Thesis). Doctor of Philosophy, University of Sussex.

Caldas-Morgan, M., Alvarez-Rosario, A. & Rodrigues Padovese, L. (2015). 'An autonomous underwater recorder based on a single board computer'. *PLOS ONE,* 10(6)**,** e0130297. Available at: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0130297 (Accessed 3 September 2018).

California Academy of Sciences. (2017) *New species discovered in Malaysian rainforest during unprecedented, top-to-bottom survey.* Available at: https://www.calacademy.org/press/releases/new-species-discovered-in-malaysian-rainforest-during-unprecedented-top-to-bottom (Accessed 3 Septembr 2018).

Cavalcanti Da Rocha, C. M., Cavalcante Leite Dos Santos, É., Lima Gomes Júnior, E., Da Rocha Moura, J., Santana e Silva, L. G. & Ferreira Barbosa, D. (2013). 'New records of marine tardigrades from Brazil' Journal of Limnology, 72(1s), e12. Available at: https://doi.org/10.4081/jlimnol.2013.s1.e12. *2013*. (Accessed 3 September 2018).

Cesari, M., Bertolani, R., Rebecchi, L. & Guidetti, R. (2009) 'DNA barcoding in Tardigrada: the first case study on Macrobiotus macrocalix Bertolani & Rebecchi 1993 (Eutardigrada, Macrobiotidae)' *Mol Ecol Resour*. 9(3), pp. 699-706. Available at: https://onlinelibrary.wiley.com/doi/full/10.1111/j.1755-0998.2009.02538.x (Accessed 3 September 2018).

Chagas, A., Baden, T., (2015) *Open Source Toolkit: Hardware.* Available at: https://blogs.plos.org/collections/open-source-toolkit-hardware/ (Accessed 3 September 2018).

Collett, T. S., Dillmann, E., Giger, A. & Wehner, R. (1992) 'Visual landmarks and route following in desert ants'. *Journal of Comparative Physiology A,* 170(4) pp. 435-442. Available at: https://link.springer.com/article/10.1007/BF00191460 (Accessed 3 September 2018).

CopabX. (2013) *Openfuge* Available at: https://www.instructables.com/id/OpenFuge/ (Accessed 3 Stptember 2018).

Czaczkes, T. (2015). *Pheromone deposition by the ant Lasius niger* Available at: https://www.youtube.com/watch?v=lxgw74SizFY (Accessed 3 September 2018)

Czaczkes, T. J., Gruter, C. & Ratnieks, F. L. (2015). Trail pheromones: an integrative view of their role in social insect colony organization. *Annu Rev Entomol,* 60(1)**,** pp. 581-99. Available at: https://doi.org/10.1146/annurev-ento-010814-020627 (Accessed 3 September 2018).

D'Ettorre, P., Heinze, J. & Ratnieks, F. L. W. (2004). Worker policing by egg eating in the Ponerine ant *Pachycondyla Inversa*. *Proceedings of the Royal Society of London. Series B: Biological Sciences,* 271(1546) pp. 1427-1434. Available at: https://doi.org/10.1146/annurev-ento-010814-020627 (Accessed 3 September 2018).

De Vylder, J., De Vos, W. H., Manders, E. M. & Philips, W. (2011). 2D mapping of strongly deformable cell nuclei-based on contour matching. *Cytometry A,* 79A(7)**,** pp.580-588. Available at: https://doi.org/10.1002/cyto.a.21055 (Accessed 3 September 2018).

Debian. (2018). *Debian Logos*. Available at: https://www.debian.org/logos/ (Accessed 3 September 2018).

Degma, P., Bertolani, R. & Guidetti, R. (2018). *Actual checklist of Tardigrada species (2009-2018, 34th Edition: 30-06-2018)*. Available at:

http://www.tardigrada.modena.unimo.it/miscellanea/Actual%20checklist%20of%20Tardigrada.pdf (Accessed 3 September 2018).
Degma, P., Bertolani, R. & Guidetti, R. (no date) *Actual checklist of Tardigrada species.* Available at: http://www.tardigrada.modena.unimo.it/miscellanea/Actual%20checklist%20of%20Tardigrada.pdf (Accessed 3 September 2018).
Dell. (2016). *Linux Laptops with Linux Ubuntu*. Available at: https://www.dell.com/learn/uk/en/ukbsdt1/campaigns/dell-linux-ubuntu-en?c=uk&l=en&s=bsd#campaignTabs-1 (Accessed 3 September 2018).
Devigne, C. & de Biseau, J. C. (2012). The differential response of workers and queens of the ant *Lasius niger* to an environment marked by workers: ants dislike the unknown. *Behav Processes,* 91(3)**,** pp.275-81. Available at https://doi.org/10.1016/j.beproc.2012.09.008 (Accessed 3 September 2018).
Dietemann, V., Peeters, C., Liebig, J., Thivet, V. & Hölldobler, B. (2003). Cuticular hydrocarbons mediate discrimination of reproductives and nonreproductives in the ant *Myrmecia gulosa*. *Proceedings of the National Academy of Sciences,* 100(18)**,** pp. 10341-10346. Available at: https://doi.org/10.1073/pnas.1834281100 (Accessed 3 September 2018).
Downey, M. J., Jeziorska, D. M., Ott, S., Tamai, T. K., Koentges, G., Vance, K. W. & Bretschneider, T. (2011). Extracting fluorescent reporter time courses of cell lineages from high-throughput microscopy at low temporal resolution. *PLOS ONE,* 6(12)**,** e27886. Available at: https://doi.org/10.1371/journal.pone.0027886 (Accessed 3 September 2018).
Dussutour, A., Deneubourg, J.-L. & Fourcassié, V. (2005). Amplification of individual preferences in a social context: the case of wall-following in ants. *Proceedings of the Royal Society B: Biological Sciences,* 272(1564)**,** pp. 705-714. Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1602045/ (Accessed 3 September 2018).
Ecorys UK. (2016). *Digital Skills for the UK Economy.* Available at: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/492889/DCMSDigitalSkillsReportJan2016.pdf (Accessed 3 September 2018).
Fabrizi, A. (2016). *Dropbox Uploader* Available at: https://andreafabrizi.it/2016/01/01/Dropbox-Uploader/ (Accessed 3 September 2018).
Fedora Project. (2018). *Logo* Available at: https://fedoraproject.org/wiki/Logo (Accessed 3 September 2018).
Fletcher, G., (2013) 'Febuary 2013 Expedition - Battling Heat, Broken Equipment and a Sunburnt Back' *Chagos Conservation Trust,* 3 March. Available at: https://chagos-trust.org/blog/febuary-2013-expedition-battling-heat-broken-equipment-and-a-sunburnt-back (Accessed 3 September 2018).
Franklin, E. L., Richardson, T. O., Sendova-Franks, A. B., Robinson, E. J. H. & Franks, N. R. (2011). Blinkered teaching: tandem running by visually impaired ants. *Behavioral Ecology and Sociobiology,* 65(4)**,** pp. 569-579. Available at: http://dx.doi.org/10.1007/s00265-010-1057-2 (Accessed 3 September 2018).
Franks, N. R. & Richardson, T. (2006). Teaching in tandem-running ants. *Nature,* 439(7073)**,** p. 153. Available at: http://dx.doi.org/10.1038/439153a (Accessed 3 September 2018).
Fraunhofer. (2017). *Algae survive heat, cold and cosmic radiation*. Available at: https://www.fraunhofer.de/en/press/research-news/2017/february/algae-survive-heat--cold-and-cosmic-radiation.html (Accessed 3 September 2018).
Frisch, K. V. (1967). *The dance language and orientation of bees.* Cambridge, Mass., Belknap Press of Harvard University Press.
Gabriel, W. N., McNuff, R., Patel, S. K., Gregory, T. R., Jeck, W. R., Jones, C. D. & Goldstein, B. (2007). The tardigrade *Hypsibius dujardini*, a new model for studying the evolution of development. *Dev Biol,* 312(2)**,** pp. 545-59. Available at: https://doi.org/10.1016/j.ydbio.2007.09.055 (Accessed 3 September 2018).

Garcia-Sierra, C. (2013). *How Jetson Development Platform helps automakers build smarter cars*. Available at: https://blogs.nvidia.com/blog/2013/03/22/how-jetson-development-platform-helps-automakers-build-smarter-cars/ (Accessed 3 September 2018).

Gąsiorek, P., Zawierucha, K., Stec, D. & Michalczyk, Ł. (2017). Integrative redescription of a common arctic water bear *Pilatobius recamieri* (Richters, 1911). *Polar Biology,* 40(11)**,** pp. 2239-2252. Available at: https://doi.org/10.1007/s00300-017-2137-9 (Accessed 3 September 2018).

Giraldeau, L., Vlaone, T. J. & Templeton, J. J. (2002). Potential Disadvantages of Using Socially Acquired Information. *Philosophical Transactions of The Royal Society B Biological Sciences.* 357(1427), pp. 1559-1556. Available at: https://doi.org/10.1098/rstb.2002.1065 (Accessed 3 September 2018).

Giraud, T., Pedersen, J. S. & Keller, L. (2002). Evolution of supercolonies: the Argentine ants of Southern Europe. *Proceedings of the National Academy of Sciences,* 99(9)**,** pp. 6075-6079. Available at: https://doi.org/10.1073/pnas.092694199 (Accessed 3 September 2018).

Gladstone, D. E. (1981). Why there are no ant slave rebellions. *The American Naturalist,* 117(5)**,** pp. 779-781. Available at: https://doi.org/10.1086/283759 (Accessed 3 September 2018).

Goldstein Lab (2007). *Hypsibius exemplaris*. Available at : https://www.flickr.com/photos/11562437@N03/1139784900 (Accessed 9 August 2019).

Graham, P. & Mangan, M. (2015). Insect navigation: do ants live in the now? *The Journal of Experimental Biology,* 218**,** pp. 819-823. Available at: http://jeb.biologists.org/content/218/6/819 (Accssed 3 September 2018).

Halfacree, G. (2015). *File:Raspberry Pi Zero (23021815320).png.* Available at: https://commons.wikimedia.org/wiki/File:Raspberry_Pi_Zero_(23021815320).png (Accessed 9 August 2019).

Hashimoto, T., Horikawa, D. D., Saito, Y., Kuwahara, H., Kozuka-Hata, H., Shin-I, T., Minakuchi, Y., Ohishi, K., Motoyama, A., Aizu, T., Enomoto, A., Kondo, K., Tanaka, S., Hara, Y., Koshikawa, S., Sagara, H., Miura, T., Yokobori, S.-i., Miyagawa, K., Suzuki, Y., Kubo, T., Oyama, M., Kohara, Y., Fujiyama, A., Arakawa, K., Katayama, T., Toyoda, A. & Kunieda, T. (2016). Extremotolerant tardigrade genome and improved radiotolerance of human cultured cells by Tardigrade-unique protein. *Nature Communications,* 7**:**12808. Available at: https://www.nature.com/articles/ncomms12808 (Accessed 3 September 2018).

Hengherr, S., Heyer, A. G., Kohler, H. R. & Schill, R. O. (2008). Trehalose and anhydrobiosis in Tardigrades - evidence for divergence in responses to dehydration. *Febs j,* 275(2)**,** 281-288. Available at: https://doi.org/10.1111/j.1742-4658.2007.06198.x (Accessed 3 September 2018).

Hengherr, S., Worland, M. R., Reuner, A., Brummer, F. & Schill, R. O. (2009). High-temperature tolerance in anhydrobiotic tardigrades is limited by glass transition. *Physiol Biochem Zool,* 82(6)**,** pp. 749-755. Available at: https://doi.org/10.1086/605954 (Accessed 3 September 2018).

Herzner, G., Schlecht, A., Dollhofer, V., Parzefall, C., Harrar, K., Kreuzer, A., Pilsl, L. & Ruther, J. (2013). Larvae of the parasitoid wasp *Ampulex compressa* sanitize their host, the American cockroach, with a blend of antimicrobials. *Proceedings of the National Academy of Sciences,* 110(4), pp. 1369-1374. Avaiable at: https://doi.org/10.1073/pnas.1213384110 (Accessed 3 September 2018).

HESA. (2018). *Higher education student statistics: UK, 2016/17 - subjects studied*. Available at: https://www.hesa.ac.uk/news/11-01-2018/sfr247-higher-education-student-statistics/subjects (Accessed 3 September 2018).

Hickling, R. & Brown, R. L. (2000). Analysis of acoustic communication by ants. *The Journal of the Acoustical Society of America,* 108(4)**,** pp. 1920-1929. Available at: https://doi.org/10.1121/1.1290515 (Accessed 3 September 2018).

Hoffman, C. (2014). *Android is based on Linux, but what does that mean?* Available at: https://www.howtogeek.com/189036/android-is-based-on-linux-but-what-does-that-mean/ (Accessed 3 September 2018).

Hölldobler, B. & Wilson, E. O. (1990). *The Ants,* Cambridge, Belknap Press of Harvard University Press.

Holman, L., Jørgensen, C. G., Nielsen, J. & d'Ettorre, P. (2010). Identification of an ant queen pheromone regulating worker sterility. *Proceedings of the Royal Society B: Biological Sciences*, 277(1701). Available at: https://doi.org/10.1098/rspb.2010.0984 (Accessed 3 September 2018).

Horikawa, D. D. (2010). *Anhydrobiosis in tardigrades (Ramazzottius varieornatus) - desiccation and rehydration.* Available at: https://www.youtube.com/watch?v=3nFee4iu5Us (Accessed 3 September 2018).

Horikawa, D. D., Cumbers, J., Sakakibara, I., Rogoff, D., Leuko, S., Harnoto, R., Arakawa, K., Katayama, T., Kunieda, T., Toyoda, A., Fujiyama, A. & Rothschild, L. J. (2013). Analysis of DNA repair and protection in the tardigrade *Ramazzottius varieornatus* and *Hypsibius dujardini* after exposure to UVC radiation. *PLOS ONE,* 8(6)**:** e64793. Available at: https://doi.org/10.1371/journal.pone.0064793 (Accessed 3 September 2018).

Horikawa, D. D., Sakashita, T., Katagiri, C., Watanabe, M., Kikawada, T., Nakahara, Y., Hamada, N., Wada, S., Funayama, T., Higashi, S., Kobayashi, Y., Okuda, T. & Kuwabara, M. (2006). Radiation tolerance in the tardigrade *Milnesium tardigradum*. *Int J Radiat Biol,* 82(12)**,** pp. 843-848. Available at: https://doi.org/10.1080/09553000600972956 (Accessed 3 September 2018).

Jönsson, K. I., Rabbow, E., Schill, R. O., Harms-Ringdahl, M. & Rettberg, P. (2008). Tardigrades survive exposure to space in low Earth orbit. *Current Biology,* 18(17)**,** pp. R729-R731. Available at: https://doi.org/10.1016/j.cub.2008.06.048 (Accessed 3 September 2018).

Kanagasuntheram, R., Geh, N. K. T., Yen, C. C., Dheen, S. T. & Bay, B. H. (2018). A composite 3D printed model of the midcarpal joint. *Anatomical Science International,* 94(1), pp.158-162.Available at: https://doi.org/10.1007/s12565-018-0469-3 (Accessed 9 August 2019).

Kellner, K., Trindl, A., Heinze, J. & D'Ettorre, P. (2007). Polygyny and polyandry in small ant societies. *Mol Ecol,* 16(11)**,** pp. 2363-2369. Available at: https://doi.org/10.1111/j.1365-294X.2007.03297.x (Accessed 3 September 2018).

Kent, H. R. & Bacon, J. P. (2016). Microsco-Pi: A novel and inexpensive way of merging biology and I.T. *School Science Review,* 98**,** pp. 75-82. Available at: http://sro.sussex.ac.uk/66499/1/Kent%20and%20Bacon%202016.pdf (Accessed 3 September 2018).

Kikuta, N. & Tsuji, K. (1999). Queen and worker policing in the monogynous and monandrous ant, Diacamma sp. *Behavioral Ecology and Sociobiology,* 46(3)**,** pp. 180-189. Available at: https://doi.org/10.1007/s002650050608 (Accessed 3 September 2018).

Kinchin, I. M. (1994). *The Biology of Tardigrades*, London ; Chapel Hill, NC, U.S.A. : Portland Press.

Koutsovoulos, G., Kumar, S., Laetsch, D. R., Stevens, L., Daub, J., Conlon, C., Maroon, H., Thomas, F., Aboobaker, A. A. & Blaxter, M. (2016). No evidence for extensive horizontal gene transfer in the genome of the tardigrade *Hypsibius dujardini*. *Proc Natl Acad Sci USA,* 113(18) pp. 5053-5058. Available at: https://doi.org/10.1073/pnas.1600338113 (Accessed 3 September 2018).

Lahav, S., Soroker, V., Hefetz, A. & Vander Meer, R. K. (1999). Direct behavioral evidence for hydrocarbons as ant recognition discriminators. *Naturwissenschaften,* 86**,** pp. 246-249. Available at: https://naldc.nal.usda.gov/download/6164/PDF (Accessed 3 September 2018).

Lee, W. M., Upadhya, A., Reece, P. J. & Phan, T. G. (2014). Fabricating low cost and high performance elastomer lenses using hanging droplets. *Biomedical Optics Express,* 5(5)**,** pp. 1626-1635. Available at: https://doi.org/10.1364/BOE.5.001626 (Accessed 3 September 2018).

Lenoir, A., Depickere, S., Devers, S., Christides, J. P. & Detrain, C. (2009). Hydrocarbons in the ant *Lasius niger*: from the cuticle to the nest and home range marking. *J Chem Ecol,* 35(8)**,** pp.

913-921. Available at: https://doi.org/10.1007/s10886-009-9669-6 (Accessed 3 September 2018).

Loarie, D. (2014). *Rotifer Phylum Rotifera.* Available at: https://www.flickr.com/photos/57556735@N08/13521461703 (Accessed 9 August 2019).

Lundstrom, J. & Stvensson, L. (2006). *The effect of dehydration rates on anhydrobiotic survival and trehalose levels in Tardigrades.* Kristianstad University.

Mach, M. (2002). *Species Identification (Ii)* Available at: http://www.baertierchen.de/jan2002.html (Accessed 3 September 2018).

Madden, V. & Goldstein, B. (2012). *Goldstein Lab – Tardigrades.* Available at: https://www.flickr.com/photos/waterbears/8220321129/ (Accessed 3 September 2018).

Mathôt, S. (2013). *Using Opensesame to run experiments on the Raspberry Pi*.

Mathôt, S., Schreij, D. & Theeuwes, J. (2012). Opensesame: an open-source, graphical experiment builder for the social sciences. *Behavior Research Methods,* 44(2)**,** pp. 314-324. Available at: https://doi.org/10.3758/s13428-011-0168-7 (Accessed 3 September 2018).

McLeod, L. (2018). *Using the salt test to calibrate a hygrometer in a pet cage* Available at: https://www.thesprucepets.com/how-to-calibrate-a-hygrometer-1239116 (Accessed 3 September 2018).

MedLine Plus. (2018). *Radiation sickness.* Available at: https://medlineplus.gov/ency/article/000026.htm (Accessed 3 September 2018).

Meijering, E. (2018). *Mtrackj: an Imagej plugin for motion tracking and analysis.* Available at: https://imagescience.org/meijering/software/mtrackj/ (Accessed 3 September 2018).

Meijering, E., Dzyubachyk, O. & Smal, I. (2012). Chapter Nine - Methods for cell and particle tracking. *In:* Conn, P. M. (ed.) *Methods in Enzymology.* Academic Press.

Mersch, D. P., Crespi, A. & Keller, L. (2013). Tracking individuals shows spatial fidelity is a key regulator of ant social organization. *Science,* 340(6136)**,** pp. 1090-1093. Available at: https://doi.org/10.1126/science.1234316 (Accessed 3 September 2018).

Microsoft. (2017). *Ubuntu now available from the Windows Store!* Available at: https://blogs.msdn.microsoft.com/commandline/2017/07/10/ubuntu-now-available-from-the-windows-store/ (Accessed 3 September 2018).

Microsoft. (2018). *Windows 10 Internet of Things.* Available at: https://developer.microsoft.com/en-us/windows/iot (Accessed 3 September 2018).

MIT. (2002). *Scratch.* Available at: https://scratch.mit.edu/ (Accessed 3 September 2018).

Monnin, T. & Ratnieks, F. L. (2001). Policing in queenless Ponerine ants. *Behavioral Ecology and Sociobiology,* 50(2)**,** pp. 97-108. Available at: https://doi.org/10.1007/s002650100351 (Accessed 3 September 2018).

Moran-Jones, K., Brown, L. M. & Samimi, G. (2015). Inc280, an orally available small molecule inhibitor of C-Met, reduces migration and adhesion in ovarian cancer cell models. *Sci Rep,* 5**,** 11749. Available at: https://doi.org/10.1038/srep11749 (Accessed 3 September 2018).

Morgan, C. & King, P. (1976). *British Tardigrades, Tardigrada,* London, Published for the Linnean Society of London by Academic Press.

Müller, M. & Wehner, R. (1988). Path integration in desert ants, *Cataglyphis fortis. Proceedings of the National Academy of Sciences,* 85(14)**,** pp. 5287-5290. Available at: https://doi.org/10.1073/pnas.85.14.5287 (Accessed 3 September 2018).

Multicherry. (2015). *File:Raspberry Pi 2 Model B v1.1 top new.jpg*. Available at: https://commons.wikimedia.org/wiki/File:Raspberry_Pi_2_Model_B_v1.1_top_new.jpg (Accessed 9 August 2019).

Narendra, A., Gourmaud, S. & Zeil, J. (2013). Mapping the navigational knowledge of individually foraging ants, *Myrmecia croslandi*. *Proceedings of the Royal Society B: Biological Sciences,* 280(1765). Available at: https://doi.org/10.1098/rspb.2013.0683 (Accessed 3 September 2018).

Narendra, A., Reid, S. F. & Raderschall, C. A. (2013). Navigational efficiency of nocturnal Myrmecia ants suffers at low light levels. *PLOS ONE,* 8(3)**,** e58801. Available at: https://doi.org/10.1371/journal.pone.0058801 (Accessed 3 September 2018).
NEN. (2015). *E-Security: managing and maintaining cyber-security in schools.* Available at: www.nen.gov.uk/advice/e-security-managing-and-maintaining-e-security-cyber-security-in-schools (Accessed 3 September 2018).
Nguyen, V., Lilly, B. & Castro, C. (2014). The exoskeletal structure and tensile loading behavior of an ant neck joint. *Journal of Biomechanics,* 47(2)**,** pp. 497-504. Available at: https://doi.org/10.1016/j.jbiomech.2013.10.053 (Accessed 3 September 2018).
Normand, L. (2018). *How to calibrate your hygrometer.* Available at: https://www.neptunecigar.com/tips/how-to-calibrate-your-hygrometer (Accessed 3 September 2018).
Pageau, C. (2018). *Motion Track - Windows, Unix, Raspberry Pi demo.* Available at: https://github.com/pageauc/motion-track (Accessed 3 September 2018).
Padro, E (2011a). *Specimen: CASENT0179923 Lasius flavus.* Available at: https://www.antweb.org/bigPicture.do?name=casent0179923&shot=p&number=1 (Accessed 9 August 2019).
Padro, E (2011b). *Specimen: CASENT0179929 Lasius niger*. Available at: https://www.antweb.org/bigPicture.do?name=casent0179929&shot=p&number=1 (Accessed 9 August 2019).
Pan weterynarz (2010). *File:Hurtnica pospolita.jpg.* Available at: https://commons.wikimedia.org/wiki/File:Hurtnica_pospolita.jpg. (Accessed 9 August 2019)
Pearce, J. (2013). *Open-source lab : how to build your own hardware and reduce research costs*. Waltham: Elsevier.
Pest Kill. (2018). *Get rid of those ants – how to do it, and what to know.* Available at: http://pestkill.org/ants/ (Accessed 3 September 2018).
Picturepest (2014). *Paramecium bursaria - 400x.* Available at: https://www.flickr.com/photos/picksfromoutthere/13263096305 (Accessed 9 August 2019).
Pigon, A. & Weglarska, B. (1955). Rate of metabolism in tardigrades during active life and anabiosis. *Nature,* 176(4472)**,** pp. 121-122. Available at: https://doi.org/10.1038/176121b0 (Accessed 3 September 2018).
Proverbs 6:6-8, Holy Bible, New International Version
Raspberry Pi Foundation. (2013a). *15,000 Raspberry Pis for UK schools – thanks Google!* Available at: https://www.raspberrypi.org/blog/15000-raspberry-pis-for-uk-schools-thanks-google/ (Accessed 3 September 2018).
Raspberry Pi Foundation. (2013b). *Long term, deep water, satellite connected ocean monitoring system*. Available at: https://www.raspberrypi.org/blog/satellite-ocean-monitoring/ (Accessed 3 September 2018).
Raspberry Pi Foundation (2014). *Penguin lifelines.* Available at: https://www.raspberrypi.org/blog/penguin-lifelines/ (Accessed 3 September 2018).
Raspberry Pi Foundation. (2016). *Raspberry Pi 3 on sale now at $35.* Available at: https://www.raspberrypi.org/blog/raspberry-pi-3-on-sale/ (Accessed 3 September 2018).
Raspberry Pi Foundation (2017). *Journeying with green sea turtles and the Arribada Initiative.* Available at: https://www.raspberrypi.org/blog/sea-turtles/ (Accessed 9 August 2019).
Raspberry Pi Foundation. (2018a). *Installing Operating System Images.* Available at: https://www.raspberrypi.org/documentation/installation/installing-images/ (Accessed 3 September 2018).
Raspberry Pi Foundation. (2018b). *Getting started with Picamera.* Available at: https://projects.raspberrypi.org/en/projects/getting-started-with-picamera/4 (Accessed 3 September 2018).

Raspberry Pi Foundation. (2018c). *Root User/Sudo.* Available at: https://www.raspberrypi.org/documentation/linux/usage/root.md (Accessed 3 September 2018).

Raspberry Pi Foundation (2018d). *Apt.* Available at: https://www.raspberrypi.org/documentation/linux/software/apt.md (Accessed 3 September 2018).

Raspberry Pi Foundation (2019a). *Getting started with the Camera Module.* Available at: https://projects.raspberrypi.org/en/projects/getting-started-with-picamera (Accessed 9 August 2019).

Raspberry Pi Foundation (2019b). *Raspberry Pi Zero Time-Lapse Camera.* Available at: https://projects.raspberrypi.org/en/projects/raspberry-pi-zero-time-lapse-cam/5 (Accessed 9 August 2019).

Raspbian. (2018). *Raspbian.* Available: https://www.raspbian.org/ (Accessed 3 September 2018).

Reader's Digest. (2018). *13 simple solutions to help you get rid of ants for good.* Available at: http://www.rd.com/home/cleaning-organizing/13-common-items-for-getting-rid-of-ants/ (Accessed 3 September 2018).

Rebecchi, L., Altiero, T. & Guidetti, R. (2007). Anhydrobiosis: the extreme limit of desiccation tolerance *Invertebrate Survival Journal,* 4**,** pp.65-81. Available at: http://www.isj.unimo.it/articoli/ISJ137.pdf (Accessed 3 September 2018).

Robinson, E. J. H., Smith, F. D., Sullivan, K. M. E. & Franks, N. R. (2009). Do ants make direct comparisons? *Proceedings of the Royal Society B: Biological Sciences,* 276(1667). Available at: https://doi.org/10.1098/rspb.2009.0350 (Accessed 3 Septembeer 2018).

Robson, J. F. & Barker, D. (2015). Comparison of the protein-coding gene content of *Chlamydia trachomatis* and *Protochlamydia amoebophila* using a Raspberry Pi computer. *BMC Research Notes,* 8**,** 561. Available at: https://doi.org/10.1186/s13104-015-1476-2 (Accessed 3 September 2018).

Rosebrock, A. (2015). *Install Opencv and Python on your Raspberry Pi 2 and B+.* Available at: https://www.pyimagesearch.com/2015/02/23/install-opencv-and-python-on-your-raspberry-pi-2-and-b/ (Accessed 3 September 2018).

Sadun, E. (2012). *Happyshutter brings smile recognition to iOS*. Available at: https://www.engadget.com/2012/01/05/happyshutter-brings-smile-recognition-to-ios/?guccounter=1 (Accessed 3 September 2018).

Savage, J. (2011). *Cross-curricular teaching and learning in the secondary school,* Milton Park, Abingdon, Oxon

Schley, R. (2010). *File:Nematoda Fadenwurm.jpg.* Available at: https://commons.wikimedia.org/wiki/File:Nematoda_Fadenwurm.jpg (Accessed 9 August 2019).

Schneider, C. A., Rasband, W. S. & Eliceiri, K. W. (2012). NIH Image to ImageJ: 25 years of image analysis. *Nat Methods,* 9(7)**,** pp. 671-675. Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5554542/ (Accessed 3 September 2018).

Schneider, S. A., Giliomee, J. H., Dooley, J. W. & Normark, B. B. (2013). Mutualism between armoured scale insects and ants: new species and observations on a unique trophobiosis (Hemiptera: Diaspididae; Hymenoptera: Formicidae: *Melissotarsus* Emery). *Systematic Entomology,* 38(4)**,** pp. 805-817. Available at: https://onlinelibrary.wiley.com/doi/full/10.1111/syen.12033 (Accessed 3 September 2018).

Schultz, T. R. (2000). In search of ant ancestors. *PNAS*, 97(26), pp. 14028-14029. Available at: https://doi.org/10.1073/pnas.011513798 (Accessed 3 September 2018).

SCORE. (2013). *Resourcing practical science at secondary level.* Available at: http://www.score-education.org/media/11805/score%20resourcing%20secondary.pdf (Accessed 3 September 2018).

Seckbach, J. & Rampelotto, P. (2015). Polyextremophiles. *In:* Bakermans, C. (ed.) *Microbial Evolution under Extreme Conditions.* DeGruyter. pp. 153-170. Available at: https://www.degruyter.com/viewbooktoc/product/211345 (Accessed 3 September 2018).

Seki, K. & Toyoshima, M. (1998). Preserving Tardigrades under pressure. *Nature,* 395**,** pp. 853-584. Available at: https://www.nature.com/articles/27576 (Accessed 3 September 2018).

SimonWaldherr. (2018). *File:Raspberry Pi Zero.jpg.* Available at: https://commons.wikimedia.org/wiki/File:Raspberry_Pi_Zero.jpg (Accessed 9 August 2019).

Smith, A. A., Holldobler, B. & Liebig, J. (2008). Hydrocarbon signals explain the pattern of worker and egg policing in the ant *Aphaenogaster cockerelli*. *J Chem Ecol,* 34(10)**,** pp. 1275-1282. Available at: https://link.springer.com/article/10.1007%2Fs10886-008-9529-9 (Accessed 3 September 2018).

Smith, Frank W., Boothby, Thomas C., Giovannini, I., Rebecchi, L., Jockusch, Elizabeth L. & Goldstein, B. (2016). The compact body plan of Tardigrades evolved by the loss of a large body region. *Current Biology,* 26(2)**,** pp. 224-229. Available at: https://doi.org/10.1016/j.cub.2015.11.059 (Accessed 3 September 2018).

Sømme, L. & Meier, T. (1995). Cold tolerance in Tardigrada from Dronning Maud Land, Antarctica. *Polar Biology,* 15(3)**,** pp. 221-224. Available at: https://link.springer.com/article/10.1007/BF00239062 (Accessed 3 September 2018).

SourceForge. (2018). *Win32 Disk Imager*. Available: https://sourceforge.net/projects/win32diskimager/ (Accessed 3 September 2018).

Specious Reasons (2012). *Water bear claws.* Available at: https://www.flickr.com/photos/28594931@N03/8122400268/)(Accessed 9 August 2019).

Srinivasan, M. V. (2015). Where paths meet and cross: navigation by path integration in the desert ant and the honeybee. *J Comp Physiol A Neuroethol Sens Neural Behav Physiol,* 201(6)**,** pp. 533-546. Avaiable at: https://link.springer.com/article/10.1007%2Fs00359-015-1000-0 (Accessed 3 September 2018).

Stec, D., Arakawa, K. & Michalczyk, Ł. (2018). An integrative description of *Macrobiotus shonaicus* Sp. Nov. (Tardigrada: Macrobiotidae) from Japan with notes on its phylogenetic position within the Hufelandi group. *PLOS ONE,* 13(2)**,** e0192210. Available at: https://doi.org/10.1371/journal.pone.0192210 (Accessed 3 September 2018).

Steinmeyer, C., Pennings, P. S. & Foitzik, S. (2012). Multicolonial population structure and nestmate recognition in an extremely dense population of the European ant *Lasius flavus*. *Insectes Sociaux,* 59(4)**,** pp. 499-510. Available at: https://link.springer.com/article/10.1007/s00040-012-0244-8 (Accessed 3 September 2018).

Suzuki, A. C. (2003). Life history of *Milnesium tardigradum* Doyere (Tardigrada) under a rearing environment. *Zoolog Sci,* 20(1)**,** pp. 49-57. Available at: http://www.bioone.org/doi/10.2108/zsj.20.49 (Accessed 3 September 2018).

Telford, M. J., Bourlat, S. J., Economou, A., Papillon, D. & Rota-Stabelli, O. (2008). The evolution of the Ecdysozoa. *Philosophical Transactions of the Royal Society B: Biological Sciences,* 363(1496)**,** pp. 1529-1537. Available at: https://royalsocietypublishing.org/doi/10.1098/rstb.2007.2243 (Accessed 3 September 2018).

The Frugal Life. (2014). *Getting rid of ants.* Available at: http://www.thefrugallife.com/ants.html (Accessed 3 September 2018).

The Pi Hut. (2018a). *Raspberry Pi Sense Hat (Astro-Pi).* Available at: https://thepihut.com/products/raspberry-pi-sense-hat-astro-pi?variant=4552311044 (Accessed 3 September 2018).

The Pi Hut. (2018b). *Official Raspberry Pi 7" touchscreen display.* Available at: https://thepihut.com/products/official-raspberry-pi-7-touchscreen-display?variant=4916536388 (Accessed 3 September 2018).

The Pi Hut (2019). *Raspberry Pi Camera Board - Adjustable-Focus Lens (5MP).* Avaiable at: https://thepihut.com/collections/raspberry-pi-camera/products/raspberry-pi-camera-adjustable-focus-5mp?variant=20063161843774 (Accessed 9 August 2019).

Torvalds, L. & Diamond, D. (2001). *Just for Fun : The Story of an Accidental Revolutionary,* New York ; London, Texere.

Trettin, J., Haubner, M., Buschinger, A. & Heinze, J. (2011). Queen dominance and worker policing control reproduction in a threatened ant. *BMC Ecology,* 11**,** 21. Available at: https://bmcecol.biomedcentral.com/articles/10.1186/1472-6785-11-21 (Accessed 3 September 2018).

Tsujimoto, M., Imura, S. & Kanda, H. (2016). Recovery and reproduction of an Antarctic Tardigrade retrieved from a moss sample frozen for over 30 years. *Cryobiology,* 72(1), pp. 78-81. Available at: https://www.sciencedirect.com/science/article/pii/S0011224015300134 (Accessed 3 September 2018).

Twain, M. (1880). A tramp abroad. Toronto: J. R. Robertson.

Ubuntu. (2017). *Ubuntu logo* [Online]. Available at: https://design.ubuntu.com/brand/ubuntu-logo/ (Accessed 3 September 2018).

University of Southampton. (2018). *Raspberry Pi.* Available at: https://cmg.soton.ac.uk/research/categories/computational-platforms/raspberry-pi/ (Accessed 3 September 2018).

University Of Sussex. (2015). *It services questions and answers - 2641: how can I connect a Raspberry Pi to the wi-fi at Sussex?.* Available at: www.sussex.ac.uk/its/help/faq?faqid=2641 (Accessed 3 September 2018).

University of Sussex. (2016a). *Sussex launches robotics Masters course for non-engineers.* Available at: http://www.sussex.ac.uk/broadcast/read/34812 (Accessed 3 September 2018).

University of Sussex. (2016b). *3D printing changes shape of anatomy learning.* Available at: https://www.sussex.ac.uk/news/all?page=5&id=37465 (Accessed 3 September 2018).

University of Sussex. (2018). *Sussex choice.* Available at: http://www.sussex.ac.uk/adqe/curriculum/sussexchoice (Accessed 3 September 2018).

Vásquez, G. M., Schal, C. & Silverman, J. (2008). Cuticular hydrocarbons as queen adoption cues in the invasive Argentine ant. *Journal of Experimental Biology,* 211**,** pp. 1249-1256. Available at: http://doi.org/10.1242/jeb.017301 (Accessed 3 September 2018).

Wales Online. (2017). *How the iconic Raspberry Pi Computer has been given a new lease of life in Wales.* Available at: http://www.walesonline.co.uk/business/business-news/staggering-10-million-raspberry-pi-13384529 (Accessed 3 September 2018).

Weale, S. (2018). *New funding system leaves schools worse off, say Headteachers.* Available at: https://www.theguardian.com/education/2018/may/08/schools-national-funding-formula-worse-off-headteachers-survey (Accessed 3 September 2018).

Wehner, R. (2003). Desert ant navigation: how miniature brains solve complex tasks. *J Comp Physiol A Neuroethol Sens Neural Behav Physiol,* 189(8)**,** pp. 579-588. Available at: https://link.springer.com/article/10.1007%2Fs00359-003-0431-1 (Accessed 3 September 2018).

Wehner, R., Michel, B. & Antonsen, P. (1996). Visual navigation in insects: coupling of egocentric and geocentric information. *J Exp Biol,* 199**,** pp. 129-40. Available at: http://jeb.biologists.org/content/jexbio/199/1/129.full.pdf (Accessed 3 September 2018).

Wehner, R. & Srinivasan, M. (2003). Path integration in Insects. *In:* Jeffery, K. (ed.) *The Neurobiology of Spatial Behaviour.* New York: Oxford University Press.

Welnicz, W., Grohme, M. A., Kaczmarek, L., Schill, R. O. & Frohme, M. (2011). Anhydrobiosis in Tardigrades--the last decade. *J Insect Physiol,* 57(5)**,** pp. 577-83. Available at: https://doi.org/10.1016/j.jinsphys.2011.03.019 (Accessed 3 September 2018).

Wendt, S. and Czaczkes, T. (2017). Individual ant workers show self-control. *Biology Letters*, 13(10). Available at: https://doi.org/10.1098/rsbl.2017.0450 (Accessed 26 October 2018).

Westrich, A. (2015) *Specimen: FMNHINS0000050367 Dinoponera quadriceps.* Available at: https://www.antweb.org/bigPicture.do?name=fmnhins0000050367&shot=d&number=1 (Access 9 August 2019).
wikiHow. (2018). *How to test a hygrometer.* Available at: https://www.wikihow.com/Test-a-Hygrometer (Accessed 3 September 2018).
Williams, K., De Robertis, A., Berkowitz, Z., Rooper, C. & Towler, R. (2014). An underwater stereo-camera trap. *Methods in Oceanography,* 11**,** pp. 1-12. Available at: https://doi.org/10.1016/j.mio.2015.01.003 (Accessed 3 September 2018).
Yong, E. (2013). *Tracking whole colonies shows ants make career moves*. Available at: https://www.nature.com/news/tracking-whole-colonies-shows-ants-make-career-moves-1.12833 (Accessed 3 September 2018).
Yoshinok. (2013). *$10 smartphone to digital microscope conversion!* Available at: http://www.instructables.com/id/10-Smartphone-to-digital-microscope-conversion (Accessed 3 September 2018).

Image licenses

Many of the images in this thesis are my own original work. Where this is not the case, the images have either been provided with permission from the original rights holder or are licensed for reuse under one of the Creative Commons licenses (https://creativecommons.org/licenses/). These are explained below.

CC BY – Attribution: The original author is given attribution.

CC BY-SA - Attribution-ShareAlike: The original author is given attribution, and any derivative work is released under the same license.

CC BY-NC – Attribution-NonCommercial : The original author is given attribution, and the work cannot be used for commercial reasons.

CC BY-NC-SA – Atrribution-NonCommercial-ShareAlike : The original author is given attribution, and any derivative work is released under the same license. The work cannot be used for commercial purposes.

Fig. 1.1 – CC-BY-SA 4.0

Fig. 1.2 – CC-BY-SA 2.0

Fig. 1.7 – CC BY-NC-SA 4.0

Fig. 1.8(a) – CC BY-NC-SA 4.0

Fig. 1.9(b) – CC BY-SA 3.0

Fig. 1.11 – CC BY-NC 2.0

Fig. 1.15 – CC BY-SA 4.0

Fig. 2.1 – CC BY-SA 4.0

Fig. 2.9 – CC BY-SA 4.0

Fig. 2.10 – CC BY-SA 4.0

Table 3.2

Tardigrade - CC BY-NC-SA 2.0
Rotifer - BY-NC-SA 2.0
Protozoa – CC BY 2.0

Algae – CC BY-SA 4.0
Worm – CC BY-SA 3.0

Appendix 1 – School Science Review Paper

Microsco-pi: a novel and inexpensive way of merging biology and IT

Harry R. Kent and Jonathan P. Bacon

ABSTRACT It is well known that schools and colleges often have budget limitations that can hamper the effectiveness of practical education. This article looks at how cheap, off-the-shelf components can be used to produce a simple DIY digital microscope, and how this provides novel opportunities to integrate biology, physics, design technology and computer science in a fun and hands-on way.

Opening comments from Jonathan Bacon

I am convener of a first-year module at the University of Sussex on cell biology that is compulsory for most of our life sciences students. Every year I am surprised by the number of students studying biology, ecology and other life sciences courses who have never seen anything alive (and moving) under the microscope, yet have chosen these subjects at degree level.

Because of this, for the last few years I have offered a final-year project in which third-year students who are thinking about becoming secondary school science teachers spend one term studying microscopic animals in depth, with the aim of developing new practical experiments and demonstrations to accompany GCSE and A-level biology teaching. These students also get the chance to go into local secondary schools and lead practical sessions about hunting for living microscopic animals in moss and pond water.

Opening comments from Harry Kent

I was one of these final-year project students in 2014 and I quickly became fascinated by tardigrades, microscopic extremophiles that live in a variety of environments. I found it very hard to get a clear and stable image of the animals under the microscope using my mobile phone camera, so I tried to find some method of obtaining good images. While doing this, I came across an 'Instructable' guide from a member called Yoshinok (see *Websites*) that detailed how to produce a cheap DIY microscope stand using a smartphone camera to grab images. I built one to use as part of the teaching section of the project. However, it soon became clear that this tool could be improved upon to produce a digital microscope that might possibly become an integral part of my investigation into tardigrade locomotion.

After the final-year project had ended, we received some funding from the Technology Enhanced Learning Innovation Scheme at the University of Sussex to develop the digital microscope further. This article describes how we modified the initial design, lists the equipment required (Box 1) to reproduce one to use in schools and colleges for a range of different practical activities, and suggests further steps that could enhance the use of the microscope as a cross-curricular teaching tool.

We produced an online survey and found that only about a third of the 101 respondents recalled ever looking at living organisms under a

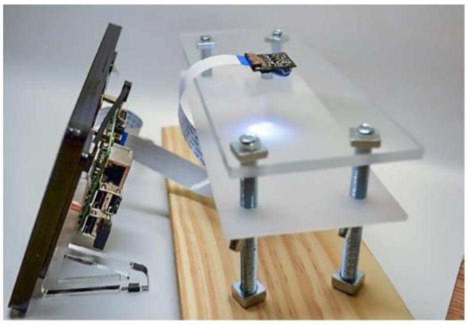


Figure 1 The entire 'Picroscope' assembly, with the stand on the right and the 7 inch touchscreen monitor on the left

Box 1 Equipment list

- wood;
- clear Perspex (200 × 80 × 5 mm);
- opaque Perspex (200 × 80 × 3 mm);
- four M8 roofing bolts 100 mm long;
- eight M8 nuts;
- four M8 wing nuts;
- short torch (approximately 4.5 cm in length);
- power drill;
- drill bit (8 mm);
- Forstner bit;
- Raspberry Pi 2 Model B;
- Raspberry Pi NOIR camera module;
- 7 inch Raspberry Touch Screen (or a monitor and the cable to connect it up);
- power cable;
- microSD card (8 GB minimum);
- Raspberry Pi wifi USB dongle;
- tealight candles;
- glass slides and coverslips;
- slide tool;
- USB keyboard and mouse;
- two laser pointers (or laser lenses);
- nail file;
- tweezers;
- can of compressed air;
- Blu-tack or hot-glue gun.

microscope. We wondered whether costs and staff time were key reasons for the lack of microscopy practical work in schools. SCORE (2013) released a report based on their survey of secondary schools which found that about 70% of schools felt they were not adequately equipped to teach science lessons effectively. One point highlighted was the lack of sufficient numbers of light and digital microscopes in classrooms; we address this issue in the following pages.

Many suitable components might be available from old phones or old computers. Some components will need to be purchased, but they are easily available from chain DIY stores, high street shops or well-known online outlets. The Raspberry Pi Foundation itself is a charity, and also operates its own store.

This article comprises three parts: setting up the Raspberry Pi; building the microscope; and uses and extension. The entire set-up could easily be built in an afternoon, or spread across a few weeks in lessons or as an after-school activity.

Part 1: Setting up the Raspberry Pi

The Raspberry Pi is a small credit-card-sized computer, used largely for education purposes. To date, seven models have been released (the A, B, A+, B+, 2, 3 and Zero) and over 7 million units have been sold worldwide (Code Club Blog, 2015). One of the key aspects of the Pi is the price, with the cheapest (the Zero) costing only £4. The most powerful, the Pi 3, costs around £30. They are very useful tools for teaching students computer science, and many companies have got involved. For example, Google donated 15 000 units to UK schools (Raspberry Pi Foundation, 2013). Raspberry Pis have been used in all sorts of research; the most recent and high profile example is the 'Astro Pi' project (astro-pi.org), which has seen British astronaut Major Tim Peake take two Raspberry Pi computers onto the International Space Station to run programs that school students have developed as part of a recent nationwide competition.

It is important to make sure that everything is set up and plugged in *before* turning the power on.

We used the Raspbian operating system, the officially supported variant of Linux. Raspbian is freely available from the Raspberry Pi Foundation's Downloads page, or you can purchase a card with it pre-installed. Follow the online instructions to get the software onto the microSD card and insert this into the underside of the Pi 2. We used the most recent version (Raspbian Jessie, November 2015) for this article; older versions should work too, but the steps involved will be slightly different.

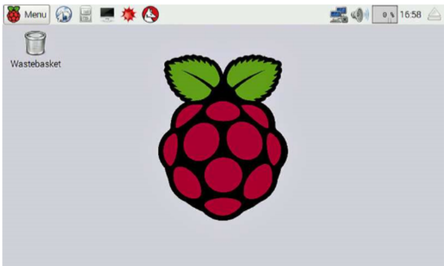


Figure 2 The Desktop of the Raspberry Pi; the newest version of Raspbian boots staight to desktop, as opposed to a login screen in the older versions

Figure 3 The menu bar found at the top of the desktop

Then, plug in your USB peripherals (mouse, keyboard, wifi dongle, etc.) and your HDMI lead. The camera is inserted into the CSI port (labelled as 'Camera' on the board). Finally, plug in your microUSB power cable and turn the power on. You will be guided through the installation process, which should boot straight into the Desktop, and will look something like Figure 2.

If you are using the 7 inch touchscreen or HDMIPi monitors, you will need to follow the instructions available from your manufacturer to set up and connect your monitor. We recommend you get the Raspberry Pi set up using a standard monitor and then, once you have a feel for it, hook up your purpose-built display.

Now, you should have a computer that looks somewhat similar to ones you are used to. The menu bar can be found along the top (Figure 3), along with some important program shortcuts. From left to right, these are the dropdown menu, the internet browser, the file manager, the terminal window, *Mathematica* and *Wolfram*. On the right-hand side of the screen, you will find the internet connection menu, the volume control, an indication of how intensely the Pi is operating (expressed as a percentage) and the time. To the far right is an eject symbol for the safe removal of USB sticks and external media.

To check whether your camera is working, click on the terminal icon and a small black box should appear. This is the terminal, where you will enter code to control the camera. Type

```
raspistill -t 0
```

This should bring up a live preview of whatever the camera is imaging. A list of commands for the camera can be found in Box 2.

One great advantage of using a Raspberry Pi as the basis for the digital microscope is that it can run programs locally, without the need for an additional computer. There are all sorts of programs that can be run on the Raspberry Pi, but Box 3 shows some of those that we recommend for a school setting, and lots of information can be found by looking at the *Websites* section below.

Dropbox Uploader is a program by Andrea Fabrizi that allows the user to send pictures captured on the Pi to a *Dropbox* account, which can then be accessed on any other computer or tablet with a web browser and access to the internet. This is great for sharing work, or for getting images for reports and presentations. The uploader can save new files in the *Dropbox*

Box 2 Camera commands

- `raspistill –o insertyourimagename.jpg`
  - This is the command you use to save an image with the filename you want to choose. –o is the output argument, and is used to name files. It will save by default to your home folder.
- `raspistill –o /home/SSR/Figure1.jpg`
  - This command saves Figure1 into a folder called SSR which is found in the home directory. You can change where files are saved this way, but be careful to check that the folder you want to save an image to has already been created.
- `raspistill –t 600000 –tl 10000 –o timelapse_%04d.jpg`
  - This command produces a time lapse. The –t arguments denote the entire time you want to produce a timelapse over (in milliseconds), -tl defines the time period between photos (in milliseconds) and –o will give the image a filename. By using _%04d in the filename, the Pi will give each image captured a four-digit identifier, which increments by 1 over the time lapse.
  - In this example, a time lapse will be created with pictures captured every 10 seconds for a ten-minute period. The saved files are called 'timelapse_0001.jpg', 'timelapse_0002.jpg', etc., up to 'timelapse_0060.jpg'.
  - If you are planning to analyse these sequences, we recommend that you copy and paste each time lapse into its own folder to make later analysis easier.
- `raspivid –t 10000 –o moviename.h264`
  - raspivid is the command used to instruct the Pi to capture a movie. The –t defines the time in milliseconds for which to capture footage, and –o is used to choose a filename. The movies are in .h264 format, which can be played back on a Linux, Mac or Windows PC using the free *VLC* media. player.

Box 3 Programs recommended for the Pi

- *Dropbox Uploader* (github.com/andreafabrizi/Dropbox-Uploader)
- *ImageJ* (Schneider, Rasband and Eliceiri, 2012; imagej.nih.gov/ij)
- *MTrackJ* (Meijering, Dzyubachyk and Smal, 2012; imagescience.org/meijering/software/mtrackj)

account and also send entire folders across with a simple command. Equally, you can send images from another source, such as a mobile phone, to the Pi itself for analysis.

ImageJ is a free image analysis software package that is often used for scientific research. Installing *ImageJ* is simply a matter of typing the words '`sudo apt-get install imagej`' into the terminal window and then typing '`Y`' when prompted. This will install *ImageJ* onto your device. You will find it under 'Menu'|'Graphics'. The increased processing power of the Pi 2 allows *ImageJ* to run smoothly and to load up a large number of image files. However, this is quite an intensive task that even on high-specification computers can take several minutes, so patience during image loading is required.

To import an image sequence, simply go to 'File'|'Import Image Sequence' and then select the folder in which the images are kept.

MTrackJ is an *ImageJ* plug-in that works alongside *ImageJ* to provide the ability to track changes in images over time. It has been used, for example, to count cells and to look at leg movement coordination in a variety of animals, and we used this plug-in to monitor tardigrade locomotion in various conditions. *MTrackJ* allows the user to highlight one point on a frame. Then, you can load the next frame, and highlight the new position of the thing of interest (e.g. a tardigrade). You then iterate through the entire sequence of images, building up a motion path of the target. You can then save the path as a movie or you can output information about the distance moved.

Part 2: Building the microscope

Now that you have a working computer that is capable of capturing, saving and analysing images, you can start to assemble the microscope. A step-by-step guide for a small-size model is given in Box 4.

Because of the flexibility of the design, you can adapt it to your needs. As this project focused on examining organisms in a small, enclosed area on a slide, we made a smaller base. If you want to examine a larger area, for example a Petri dish, you can make the specimen stage larger so you can move your sample around more freely. The basic mechanism is the same: wing nuts are used to move the specimen stage closer to, or further away from, the camera, essentially acting like a focus knob on a traditional microscope.

Part 3: Uses and extension

As previously mentioned, the Raspberry Pi has been used in all sorts of research projects around the world. Their low cost and power requirements, great flexibility and ease of use make them ideal tools. For example, they have been used to study penguins in the Arctic, to control camera traps to study biodiversity under water, for educational high-altitude ballooning and for monitoring air quality.

This equipment was used at the University of Sussex to study the behaviour of tardigrades when exposed to various environmental conditions. Tardigrades are microscopic members of the Ecdysozoa, and are some of the toughest creatures known. Discovered in 1773 by Goeze, they were initially known as small water bears (Figure 5). They are able to enter an ametabolic state, known as the 'tun' form, in which they can survive pretty much any environmental extreme, including pressure, radiation, temperature, dehydration and even the vacuum of space. For a review of tardigrades' ability to resist environmental change, see Møbjerg *et al*. (2011). However, one of the most incredible things about tardigrades is that they can be found practically anywhere, from the Arctic to tropical regions, and even in the moss on your school's grounds.

Owing to their inherent cuteness and near indestructability, tardigrades are hugely popular with students. Their alien-like qualities make them a really engaging subject, and they lend themselves readily to cross-curricular lesson planning.

As well as finding different, unidentified species in moss, we focused on one species, *Dactylobiotus (Macrobiotus) dispar*, purchased from Sciento, Manchester. A very handy characteristic of *D. dispar* is their transparent nature, which means that the contents of their gut

Box 4 Microscope construction

Firstly, you need to get some lenses. These can be bought (see AixiZ in the *Websites* list) or you can get some by taking laser pens apart. Carefully open up the laser pen using pliers (first making sure that no battery is in the pen and taking care not to get cut on the sharp metal) and remove the lens. It may be in a plastic holder; if possible, remove the lens, or use a hacksaw to cut the plastic down, and then file for a smooth finish. Use the can of compressed air to clean the lenses. We found that you can stack two lenses next to each other to get a good level of magnification (Figure 4A–D), but your lenses may vary.

Take the piece of wood and mark four drill points (which will depend on the size of your stages). This will act as your base. Drill holes through these points and then, once you know the roofing bolts will fit through, turn the base over and countersink the holes using the Forstner bit, allowing the base to sit flush on the work surface. An alternative is to use a spade drill to make suitable holes to inset each screw head before drilling the holes through the wood.

Then, take the Perspex and mark drill points 15 mm from the long edge and 20 mm away from the short edge. The opaque Perspex will act as the specimen stage and is designed to distribute the incident light across the surface (Figure 4E) as opposed to bleaching out the image by beaming directly into the camera. The clear Perspex will act as the camera stage. Once you are happy that these stages will fit onto the roofing bolts and the specimen stage is able to slide up and down freely on the bolts, you are ready for the next step. You may want to use a small file to increase the diameter of the holes, depending on how much freedom of movement your set-up has.

Take the camera stage and drill a hole in the centre. This hole has to be almost the exact same diameter as your lenses, so it will depend on your own set-up. Once this hole has been drilled, simply place the lenses in it. If they do not sit correctly by themselves, either use a file to open the hole a bit more, or use Blu-tack or a hot-glue gun to fix the lenses in place.

Finally, once you are happy that it all fits together, take the stand apart and mark a point directly below the lens with a pencil. This is where your light source will be located. Using the Forstner bit that is a similar diameter to the base of the torch, drill a shallow hole in the upper surface of the wood layer to hold the light source. If you do not have a small torch that will fit, a smartphone's flashlight app also produces a good quality light.

Using the bolts, nuts and wing nuts, assemble the stand as shown in Figure 4E. As it is, this set-up can be used in a classroom or outdoor setting with camera phones to look at microscopic detail. The wing nuts allow you to move the specimen stage relative to the camera stage to bring the image into focus.

To fit the Pi in place, simply line up the camera module with the microscope's lens, and fix with Blu-tack. You could drill some holes to line up with the mounting holes present on the camera board and screw the camera board into place. To position a phone or tablet, open the camera app and line the phone/tablet camera up with the lens.

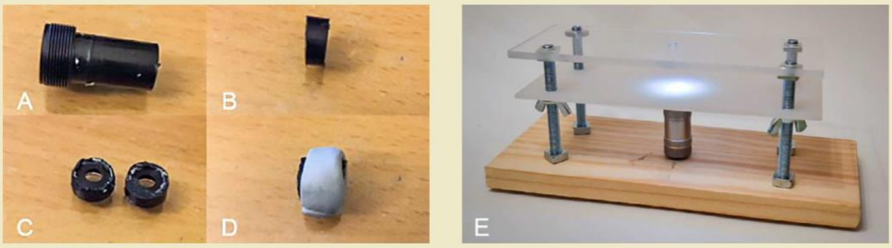


Figure 4 (A) How the lens is mounted in a laser pen; (B) using a hacksaw, you can remove excess plastic and file it down; (C) repeat the process to get a second lens; (D) you can use Blu-tack to hold two lenses together; (E) the complete, assembled DIY microscope stand; without the Pi in place, this can be used with a camera phone or tablet to produce high levels of magnification

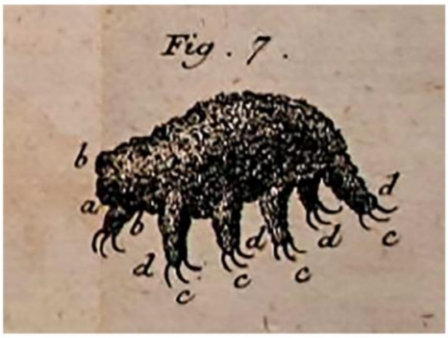


Figure 5 This is the first illustration of a tardigrade by Goeze: a 'Kleiner Wasserbär' (Bonnet and Goeze, 1773)

are very visible if they have been grazing on moss and stand out against the background; this acts as an ideal reference point for image tracking.

The investigation into tardigrades was partly inspired by an article by Shcherbakov *et al.* (2010), who used custom software and expensive equipment to plot tardigrade locomotion. Our DIY version relies on things that you can easily obtain. The set-up is as follows. A layer of wax from tealight candles is poured across a glass slide. To this, a small Perspex sheet with a 5 mm hole drilled through the centre is added. This set-up can be held over a gentle flame to remelt the wax to fix the Perspex and thus to obtain a smooth surface within the observation arena into which the tardigrades are placed.

Then, using the commands from Box 2, a time lapse recording was produced. We took one picture every ten seconds over a ten-minute period, but this can be adjusted to meet your needs. These images can then be loaded into *ImageJ*, and then *MtrackJ* can be used to plot the path of the tardigrade(s) (Figure 6).

You can vary the conditions of the experiment (e.g. light/dark, temperature) to produce an investigation for the students. For example, Figure 6 shows the same tardigrade in the same arena at different temperatures.

Just by taking tardigrades as the focus of a multidisciplinary lesson plan, you can develop a whole project that crosses between biology, physics and chemistry and incorporates some engineering and computer science. We believe that this approach would help students see that the STEM subjects are not separate modules but that research relies on subjects working together to find the solutions to the big issues.

The Pi comes pre-installed with several different development environments to allow students of all ages to learn about coding. A simple drag-and-drop system called Scratch is often used with younger students. There is also Python, a great coding language for beginners. It also has BlueJ installed, which is the Java coding environment that is used to teach programming modules to first-year computer science students at the University of Sussex. The Raspberry Pi has all the software your students will need to start coding. As more GCSE and A-level teaching becomes focused on the principles of coding,

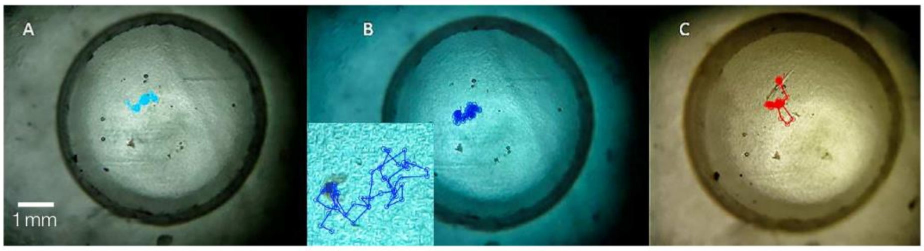


Figure 6 An individual tardigrade was monitored at three different temperatures over a ten-minute period with an image taken every ten seconds. The path was added using *MTrackJ*. The tardigrade was left to acclimatise for ten minutes in each condition before being observed at (A) 8 °C, (B) 20 °C and (C) 32 °C. This tardigrade showed a maximum speed of approximately 3 mm min^{-1} (compared with a body length of roughly 0.25 mm) at 32 °C. Image (B) shows an expanded view of the motion path and the tardigrade is clearly visible as the grey outline on the left.

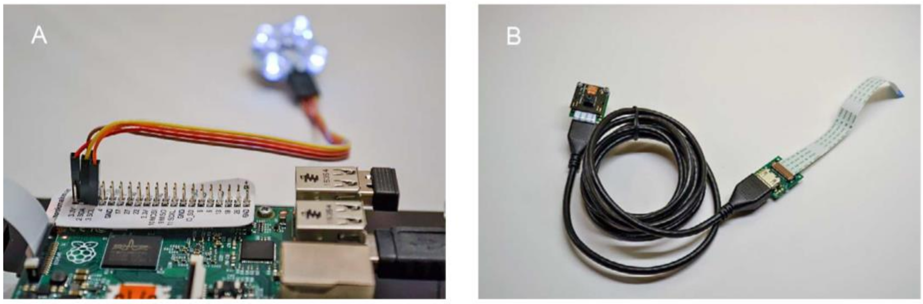


Figure 7 (A) BrightPi set-up, with white LEDs turned on; (B) HDMI camera extension kit

the Pi will become an even greater tool, and its use in different subjects is something that more conventional school equipment cannot easily replicate.

Extension

As has already been stated, one of the best things about this set-up is its flexibility. There are many ways that you could expand upon the basic tool built in this project.

You could use a USB power bank to provide power to the microscope to allow it to be used away from a mains supply. A good-quality power bank that outputs at least 2A should allow the Pi and some screens (such as the 7 inch touchscreen) to be used on the go.

We have subsequently replaced the stubby torch with a BrightPi, a kit produced by Pi Supply, which consists of four bright white LEDs and eight infrared LEDs (Figure 7A). This allows the light levels to be controlled by the user using some commands typed into the terminal window. The BrightPi requires soldering, so students can learn a new skill in the process.

We have also attached an HDMI camera extender (Figure 7B). This allows a standard HDMI cable to be used as the link between the camera and the Pi. The standard cables between the Pi and the camera are about 20 cm long, but 5 m HDMI cables can be readily purchased. Using the infrared camera on the microscope and an infrared light source, the set-up can be placed in a dark environment to monitor the effects of light and dark on animal behaviour, and the long camera lead allows for the user to observe and interact with the touchscreen display in normal working illumination.

We are also currently working on motorising the specimen stage. We are aiming to use an Arduino (another microcomputer) and stepper motors (salvaged from scrap CD and DVD drives) to control the position of the stage in the *x* and *y* planes. This will enable the DIY microscope to be used with greater accuracy and also gives students even more opportunities to learn about basic electronics and coding.

Conclusion

The set-up provides a great starting point for a really easy-to-use mixture of biology and computer science, as well as design technology and also some basic physics. The fact that students can get a chance to see how different subjects can support each other in a classroom setting is really exciting, as many students often get taught subjects in isolation, and the ability to link ideas across disciplines is a skill that is very useful to acquire when young.

The DIY microscope is ideal for use as a long-term project, such as an after-school club, or for use in short in-class practical demonstrations. Once you are comfortable with the Pi, there are many other practical tools that can be produced very cheaply (light gates, thermometers, soil moisture monitors and much more). We hope that this guide will encourage school staff to see that they can produce their own equipment on site at a fraction of the cost of purchasing and maintaining traditional equipment, and get students involved in the production, evaluation and improvement stages of the build.

References

Bonnet, C. and Goeze, J. A. E. (1773) *Herrn Karl Bonnets Abhandlungen aus der Insektologie aus dem Französischen übersetzt und mit einigen Zusätzen herausgegeben von Joh. August Ephraim Goeze*. Halle: Bey J. J. Gebauers Wittwe und Joh. Jac. Gebauer. Available at: www.biodiversitylibrary.org/bibliography/47534.

The Code Club Blog (2015) *Code Club and the Raspberry Pi Foundation join forces!* 3 November. Available at: blog.codeclub.org.uk/2015/11/03/code-club-and-the-raspberry-pi-foundation-join-forces.

Meijering, E., Dzyubachyk, O. and Smal, I. (2012) Methods for cell and particle tracking. *Methods in Enzymology*, **504**, 183–200. Available at: smal.ws/pdf/mie2012.pdf.

Møbjerg, N., Halberg, K. A., Jørgensen, A., Persson, D., Bjørn, M., Ramløv, H. and Kristensen, R. M. (2011) Survival in extreme environments – on the current knowledge of adaptations in tardigrades. *Acta Physiologica*, **202**(3), 409–420.

Raspberry Pi Foundation (2013) *15,000 Raspberry Pis for UK schools – thanks Google!* 29 January. Available at: www.raspberrypi.org/15000-raspberry-pis-for-uk-schools-thanks-google.

Schneider, C. A., Rasband, W. S. and Eliceiri, K. W. (2012) *NIH Image* to *ImageJ*: 25 years of image analysis. *Nature Methods*, **9**, 671–675.

Science Community Representing Education (SCORE) (2013) *Resourcing Practical Science at Secondary Level*. London: SCORE. Available at: www.rsc.org/images/Resourcing-Practical-Science-at-Secondary%20Level_tcm18-230620.pdf.

Shcherbakov, D., Schill, R. O., Brümmer, F. and Blum, M. (2010) Movement behaviour and video tracking of *Milnesium tardigradum* Doyère, 1840 (Eutardigrada, Apochela). *Contributions to Zoology*, **79**(1), 33–38. Available at: repository.naturalis.nl/record/330854.

Websites

AixiZ eBay store: stores.ebay.com/AixiZ-Service-and-International-LLC.

Astro Pi: astro-pi.org.

Dropbox Uploader (Andrea Fabrizi): github.com/andreafabrizi/Dropbox-Uploader.

ImageJ: imagej.nih.gov/ij

MTrackJ: imagescience.org/meijering/software/mtrackj.

Petit Studio – The Raspberry Pi HDMI Camera Extension Cable: petitstudio.blogspot.jp/2015/04/the-raspberry-pi-camera-extension-cable.html.

Pi Supply – Bright Pi – Bright White and IR Camera Light for Raspberry Pi: www.pi-supply.com/product/bright-pi-bright-white-ir-camera-light-raspberry-pi/?v=79cba1185463.

Raspberry Pi Foundation – *Downloads*: www.raspberrypi.org/downloads.

Yoshinok (2013) *$10 Smartphone to digital microscope conversion!*: www.instructables.com/id/10-Smartphone-to-digital-microscope-conversion.

Harry Kent graduated in 2015 with a first-class biomedical science BSc from the University of Sussex, and is now researching for his MPhil at Sussex in the Laboratory of Apiculture and Social Insects. He has a passion for widening participation and tinkering with gadgets. Email: H.R.Kent@sussex.ac.uk
Jonathan Bacon is Professor of Neuroscience, and is currently Head of the School of Life Sciences at the University of Sussex. Email: j.p.bacon@sussex.ac.uk

Appendix 2 – Software Guides

ImageJ (Schneider et al., 2012)

Download and Documentation available from: https://imagej.nih.gov/ij/

MTrackJ (Meijering, 2018)

Download and Documentation available from:
https://imagescience.org/meijering/software/mtrackj/

Dropbox Uploader (Fabrizi, 2016)

Documentation available from: https://github.com/andreafabrizi/Dropbox-Uploader

OpenCV (Bardski, 2000)

Documentation available from : https://opencv.org/

Raspberry Pi Open CV installation guide available from Rosebrock (2015) :
https://www.pyimagesearch.com/2015/02/23/install-opencv-and-python-on-your-raspberry-pi-2-and-b/

Motion Track (Pageau, 2018)

Documentation available from: https://github.com/pageauc/motion-track

Raspbian (Raspbian, 2018)

Download from : https://www.raspberrypi.org/downloads/raspbian/

Historical versions available from: http://downloads.raspberrypi.org/raspbian/images/

Win32 Disk Imager (SourceForge, 2018)

Download and documentation available from: https://sourceforge.net/projects/win32diskimager

Appendix 3 – Motion Track Code

The initial version of this programme was created by Claude Pageau, and edited slightly to ensure the ants were successfully detected and their position was exported in a text file for further analysis.

The Motion Track programme was released under the MIT License (MIT), provided the following notice is included:

motiontest

```
#!/usr/bin/env python

progname = "motion_track.py"
ver = "version 0.7"

print("%s %s using python2 and OpenCV2" % (progname, ver))
print("Loading Please Wait ....")
import io
import time
from picamera.array import PiRGBArray
from picamera import PiCamera
import cv2
import numpy as np

debug = True        # Set to False for no data display
window_on = True   # Set to True displays opencv windows (GUI desktop reqd)
SHOW_CIRCLE = True  # show a circle otherwise show bounding rectancle on window
CIRCLE_SIZE = 10    # diameter of circle to show motion location in window
LINE_THICKNESS = 1  # thicknes of bounding line in pixels
WINDOW_BIGGER = 1   # resize multiplier for speed photo image and if
gui_window_on=True then makes opencv window bigger
                    # Note if the window is larger than 1 then a reduced frame
rate will occur

# Camera Settings
CAMERA_WIDTH = 320
CAMERA_HEIGHT = 240
CAMERA_HFLIP = True
CAMERA_VFLIP = True
CAMERA_FRAMERATE = 25

# Motion Tracking Settings
THRESHOLD_SENSITIVITY = 3
BLUR_SIZE = 3
MIN_AREA = 3    # excludes all contours less than or equal to this Area

#this line creates the text file into which the co-ords are stored.
f = open("/home/pi/ANTTEST/coord.txt", "w")

def show_FPS(start_time,frame_count):
    if debug:
        if frame_count >= 10:
            duration = float(time.time() - start_time)
            FPS = float(frame_count / duration)
            print("Processing at %.2f fps last %i frames" %( FPS, frame_count))
            frame_count = 0
            start_time = time.time()
        else:
            frame_count += 1
    return start_time, frame_count

def motion_track():
```

```
    print("Initializing Camera ....")
    # Save images to an in-program stream
    camera = PiCamera()
    camera.hflip = CAMERA_HFLIP
    camera.vflip = CAMERA_VFLIP
    camera.resolution = (CAMERA_WIDTH, CAMERA_HEIGHT)
    camera.framerate = CAMERA_FRAMERATE
    rawCapture = PiRGBArray(camera, size=(CAMERA_WIDTH, CAMERA_HEIGHT))
    time.sleep(1)
    first_image = True
    if window_on:
        print("press q to quit opencv display")
    else:
        print("press ctrl-c to quit")
    print("Start Motion Tracking ....")
    frame_count = 0
    start_time = time.time()
    for frame in camera.capture_continuous(rawCapture, format="bgr",
use_video_port=True):
        image2 = frame.array
        start_time, frame_count = show_FPS(start_time, frame_count)
        # initialize variables
        motion_found = False
        biggest_area = MIN_AREA
        cx = 0
        cy = 0
        cw = 0
        ch = 0
        # At this point the image is available as stream.array
        if first_image:
            # initialize image1 using image2 (only done first time)
            image1 = image2
            grayimage1 = cv2.cvtColor(image1, cv2.COLOR_BGR2GRAY)
            first_image = False
        else:
            # Convert to gray scale, which is easier
            grayimage2 = cv2.cvtColor(image2, cv2.COLOR_BGR2GRAY)
            # Get differences between the two greyed, blurred images
            differenceimage = cv2.absdiff(grayimage1, grayimage2)
            differenceimage = cv2.blur(differenceimage,(BLUR_SIZE,BLUR_SIZE))
            # Get threshold of difference image based on THRESHOLD_SENSITIVITY
variable
            retval, thresholdimage =
cv2.threshold(differenceimage,THRESHOLD_SENSITIVITY,255,cv2.THRESH_BINARY)
            # Get all the contours found in the thresholdimage
            contours, hierarchy =
cv2.findContours(thresholdimage,cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_SIMPLE)
            total_contours = len(contours)
            # save grayimage2 to grayimage1 ready for next image2
            grayimage1 = grayimage2
            # find contour with biggest area
            for c in contours:
```

```
                # get area of next contour
                found_area = cv2.contourArea(c)
                # find the middle of largest bounding rectangle
                if found_area > biggest_area:
                    motion_found = True
                    biggest_area = found_area
                    (x, y, w, h) = cv2.boundingRect(c)
                    cx = int(x + w/2)   # put circle in middle of width
                    cy = int(y + h/6)   # put circle closer to top
                    cw = w
                    ch = h
            if motion_found:
                # Do Something here with motion data
                if window_on:
                    # show small circle at motion location
                    if SHOW_CIRCLE:
                        cv2.circle(image2,(cx,cy),CIRCLE_SIZE,(0,255,0),
LINE_THICKNESS)
                    else:
                        cv2.rectangle(image2,(cx,cy),(x+cw,y+ch),(0,255,0),
LINE_THICKNESS)
                if debug:
                    print("total_Contours=%2i  Motion at cx=%3i cy=%3i
biggest_area:%3ix%3i=%5i" % (total_contours, cx ,cy, cw, ch, biggest_area))
                    #added these lines to input data into the text file in the
format needed for the plotting programmes
                    data = "{}\n".format(cx)
                    f.write(data)
            if window_on:
                # cv2.imshow('Difference Image',differenceimage)
                cv2.imshow('Threshold Image', thresholdimage)
                if WINDOW_BIGGER > 1:  # Note setting a bigger window will slow
the FPS
                    big_w = CAMERA_WIDTH * WINDOW_BIGGER
                    big_h = CAMERA_HEIGHT * WINDOW_BIGGER
                    image2 = cv2.resize( image2,( big_w, big_h ))

                cv2.imshow('Movement Status', image2)
                # Close Window if q pressed: trying to change this to mouse
click because q is ineffective if no keyboard
                if cv2.waitKey(1) & 0xFF  == ord('q'):
                    cv2.destroyAllWindows()
                    print("End Motion Tracking")
                    #added this line to clean up the camera on closure
                    camera.close()
                    break
        rawCapture.truncate(0)
#--------------------------------------------------------------------------------
----------------
if __name__ == '__main__':
    try:
        motion_track()

finally:
    print("")
    print("+++++++++++++++++++++++++++++++++++")
    print("%s %s - Exiting" % (progname, ver))
    print("+++++++++++++++++++++++++++++++++++")
    print("")
```

Appendix 4 – Microscope GUI

```
from Tkinter import *
from Tkinter import StringVar, IntVar
import picamera
import time
from tkMessageBox import *
import Adafruit_GPIO.SPI as SPI
import Adafruit_MAX31855.MAX31855 as MAX31855
import random
import sys
import motion

#THIS SECTION SETS UP THE PI SOFTWARE SPI CONFIGURATION AND LOADS THE
THERMOCOUPLE
CLK = 25
CS = 24
DO = 18
sensor = MAX31855.MAX31855(CLK, CS, DO)
temp = sensor.readTempC()

###this section makes the actual GUI

class GUI(Frame):
    def __init__(self, parent):
        Frame.__init__(self, parent, background="white")
        self.parent = parent
        self.parent.title("Microscope GUI Alpha V0.5")
        self.pack(fill=BOTH, expand=1)
        self.centerWindow()
        self.goGUI()
###THIS SECTION CHANGES THE SIZE OF THE APP; in older version this forced it to
open in centre of window, hence the name of the method
    def centerWindow(self):
        #altered this line to make it open in full screen, except 80 pixels
along the top for the menu and menu bar
        w = 800
        h = 400
        sw = self.parent.winfo_screenwidth()
        sh = self.parent.winfo_screenheight()

        x = 0
        y = 0

        self.parent.geometry('%dx%d+%d+%d' % (w,h,x,y))

### THIS SECTION CREATES THE GUI ITSELF AND DEFINES THE BUTTONS PRESENT ###

    def goGUI(self):
```

```
        self.parent.title("Microscope GUI")
        #added these lines: instead of having a whole new window, why not just
set a frame to the side of the menu?
        preview_frame = Frame(self, height=360, width=680, bg="white")
        preview_frame.pack(pady=10, padx=10, side=LEFT)
        opencam_button = Button(self, text="Open Camera", command =
self.OpenCam)
        opencam_button.pack(pady=10, padx=5)
        fullscreen_button = Button(self, text = "Preview", command =
self.Preview)
        fullscreen_button.pack(pady=10, padx=5, fill = X)

        takephoto_button = Button(self, text = "Take Photo", command =
self.TakePhoto)
        takephoto_button.pack(pady=10, padx=5)

        makemovie_button = Button(self, text = "Make Movie",
command=self.MakeMovie)
        makemovie_button.pack(pady=10, padx=5)

        timelapse_button = Button(self, text = "Time Lapse", command =
self.TimeLapse)
        timelapse_button.pack(pady=10, padx=5)

        trackbutton = Button(self, text = "Motion Track", command = self.Motion)
        trackbutton.pack(pady=10, padx=5)

        gettemp_button = Button(self, text="Temperature", command =
self.Temperature)
        gettemp_button.pack(pady=10, padx=5, fill = X)

        quit_button = Button(self, text="Quit", command = self.Quit)
        quit_button.pack(pady=10, padx=5, fill=X)

### THIS COMMAND TAKES A PHOTO ###
    def TakePhoto(self):
        print("Taking a photo")
        with picamera.PiCamera() as camera:
            camera.resolution = (800, 480)
            camera.framerate = 24
            camera.start_preview()
            camera.annotate_text_size = 60
            camera.annotate_text = "3"
            time.sleep(1)
            camera.annotate_text = "2"
            time.sleep(1)
            camera.annotate_text = "1"
            time.sleep(1)
            camera.annotate_text = ""
            time.sleep(1)

            camera.capture("/home/pi/ANTTEST/Photo.jpg")
```

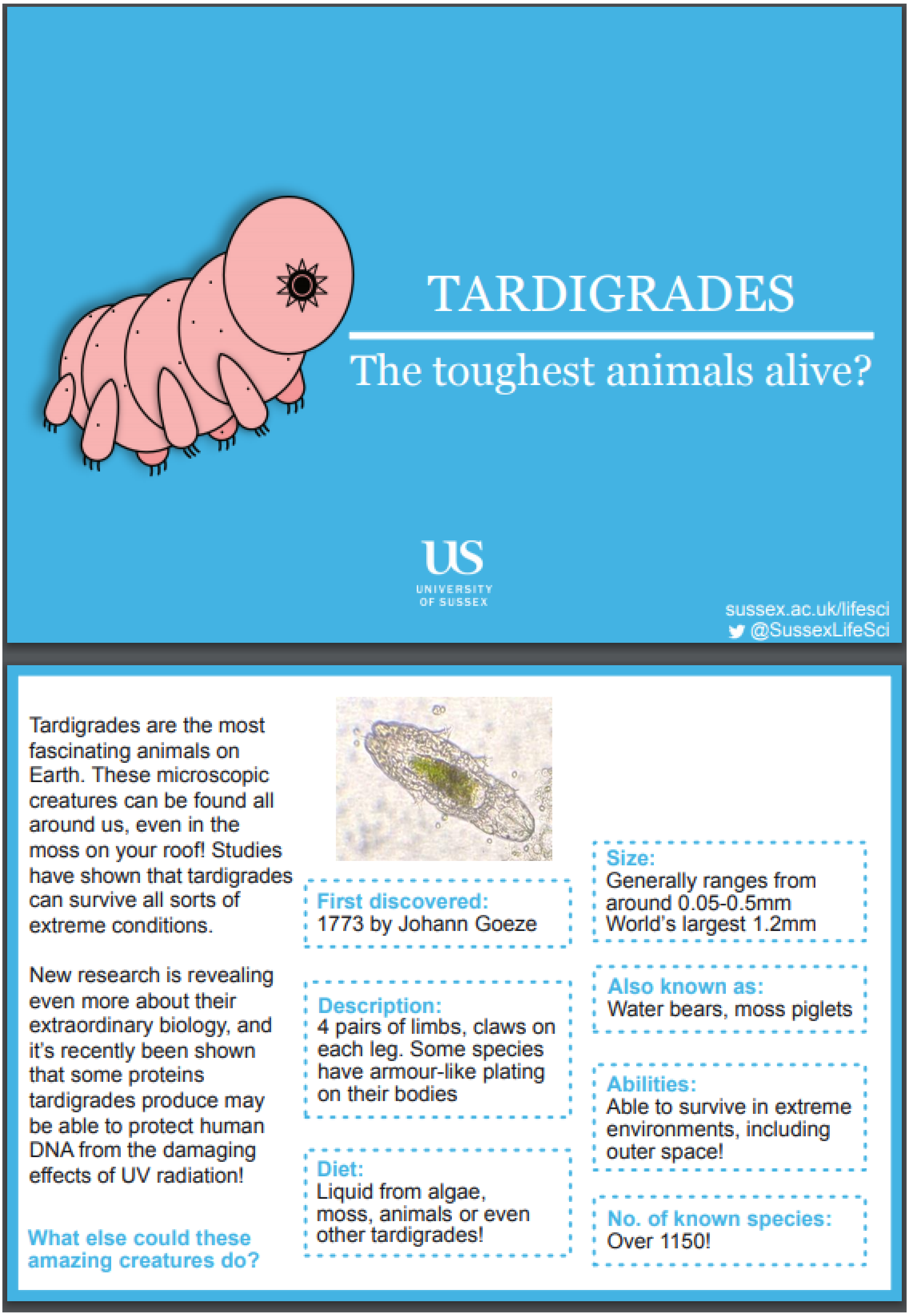
```
            camera.close()
### THIS SECTION DEFINES HOW THE TIME LAPSE WORKS ###
    def TimeLapse(self):
        print("Taking a time lapse; default period of ten seconds, one photo per
second")
        with picamera.PiCamera() as camera:
            camera.resolution = (800, 480)
            camera.start_preview()
            time.sleep(1)
            for i in range(10):
                camera.annotate_text = ("%02d" % i)
                camera.capture("/home/pi/ANTTEST/image%02d.jpg" % i)
                time.sleep(1)
                print "Image",i,"Captured"
### THIS SECTION DEFINES HOW THE APP WILL QUIT ###
    def Quit(self):
        print ("The programme will close")
        time.sleep(1)
        self.parent.destroy()
        #GUI.destroy()

###THIS SECTION DEFINES THE TEMPATURE BUTTON ###
    def Temperature(self):
        temp = sensor.readTempC()
        showinfo(title="Temperature", message=temp)

###THIS SECTIONS DEFINES THE FULLSCREEN CAMERA PREVIEW (DEFAULTS OPEN TO TEN
SECONDS)
    def Preview(self):
        #save a lot of time if I imported this at the top
        with picamera.PiCamera() as camera:
            #this line makes it full screen
            camera.resolution = (800, 480)
            #this line changes how much of the background is visible through the
preview
            camera.preview_alpha = 128
            camera.start_preview()
            time.sleep(10)
            camera.stop_preview()
###THIS SECTION OPENS THE LIVE PREVIEW IN THE GUI
    def OpenCam(self):

        print("this is where cam test was")
        with picamera.PiCamera() as camera:
            print("The button is working")


            camera.preview_fullscreen = False
            camera.preview_window =(20, 50, 640, 480)
            camera.start_preview()
            time.sleep(10)
            camera.stop_preview()
```

```
##this section captures a movie and saves it to the programme folder
    def MakeMovie(self):
        with picamera.PiCamera() as camera:
            print("Filming Movie")
            camera.start_preview()
            time.sleep(1)
            camera.resolution = (640, 480)
            #HERE THE USER COULD EDIT THE FILE NAME
            camera.start_recording('/home/pi/ANTTEST/my_video.h264')
            camera.wait_recording(60)
            camera.stop_recording()
            camera.stop_preview()

   # def CamOn(self):

###this section calls the motion track programme. Added this section!
    def Motion(self):
        showinfo("Motion Track", "Drag the windows into view,\n press Q to
quit\n")
        motion.motion_track()

### THE MAIN SECTION HANDLES HOW THE PROGRAMME LOOPS
def main():

    root = Tk()
    root.title("Picroscope GUI Alpha V0.1")

    ### --- DEFINE THE MENU --- ###
    def aboutbox():
        showinfo(title="About", message=
        """
This program is an early version of a Graphical User Interface (GUI) for the
Raspberry Pi based microscope.
Version 0.3: March 2016
        """
        )


    def helpbox():
        showinfo(title="Help", message=
        """
        This program is the first version of a Graphical User Interface (GUI)
for the Raspberry Pi based microscope.
        "Preview" will bring up a fullscreen preview for 10 seconds
        "Take Photo" will take a photo after a 3 second countdown.
        "Time Lapse" will open a preview window, and take one photo every second
for ten seconds
        Each Photo will be annotated with a 2 digit number to show its
        order in the sequence.
        "Temperature" will bring up the current temperature
        "Quit" will end the programme
```

```
    def help_opencam():
        showinfo(title="Open Camera", message=
        """
Open Camera brings up a small preview of the camera within the GUI
window.Although while it is active, the buttons are inactive, anything you press
will run as soon as the preview ends.
        """)

    def help_preview():
        showinfo(title="Preview", message =
        """
The Preview button opens a full screen preview of the camera, during which time,
the GUI will be invisible and inactive. This preview will end after 10 seconds.
        """)
    def help_takephoto():
        showinfo(title="Take Photo", message=
        """
The Take Photo button will bring up a countdown from 3 to 1, and then will
capture an image called 'Photo.jpg'.
        """)
    def help_makemovie():
        showinfo(title="Make Movie", message=
        """
Pressing Make Movie will produce a .h264 file called 'My_Movie'. This recording
will be 10 seconds long.
        """)

    def help_timelapse():
        showinfo(title="Time Lapse", message=
        """
The Time Lapse function will take one photo every second for 10 seconds, and
name them according to their position in the sequence.
        """)

    def help_temperature():
        showinfo(title="Temperature", message=
        """
The Temperature button will take the current temperature from the thermocouple
and print it into a window on the screen. This does not update in real time in
this programme.
        """)
    def help_motiontrack():
        showinfo(title="Motion Track", message =
        """
This button will open 2 new windows, which will show motion from down the
camera. When you want to end motion tracking, press 'Q' to quit. Data from this
tracking will appear in a new file called "coord.txt"
```

```
        """)
    def help_quit():
        showinfo(title="Quit", message=
        """
This button will cause the programme to close and take you back to the terminal
interface.""")


    menubar = Menu(root)

    filemenu = Menu(menubar, tearoff=0)
    filemenu.add_command(label="Exit", command=root.destroy)
    menubar.add_cascade(label="File", menu=filemenu)

    helpmenu = Menu(menubar, tearoff=0)
    #added this section to improve the help menu 14/3
    helpsubmenu = Menu(helpmenu)
    helpsubmenu.add_command(label="Open Camera", command=help_opencam)
    helpsubmenu.add_command(label="Preview", command=help_preview)
    helpsubmenu.add_command(label="Take Photo", command=help_takephoto)
    helpsubmenu.add_command(label="Make Movie", command=help_makemovie)
    helpsubmenu.add_command(label="Timelapse", command=help_timelapse)
    helpsubmenu.add_command(label="Motion Track", command=help_motiontrack)
    helpsubmenu.add_command(label="Temperature", command=help_temperature)
    helpsubmenu.add_command(label="Quit", command=help_quit)
    helpmenu.add_command(label="About", command=aboutbox)
    #helpmenu.add_command(label="Help", command=helpbox)
    menubar.add_cascade(label="Help", menu=helpmenu)
    helpmenu.add_cascade(label="Help", menu=helpsubmenu)

    root.config(menu=menubar)

    ### --- END OF MENU SECTION --- ###


    #def previewwindow(self):
#       window = tk.Toplevel()
#       window.geometry("700x400")
#       tk.Label(window, text="camera preview").pack(side=RIGHT)

    app = GUI(root)

    root.mainloop()


### CAUSES THE PROGRAMME TO RUN ###

if __name__ == '__main__':
    main()
```

Appendix 5 – British Science Festival Promotional Materials



Post card produced for British Science Festival 2017 Activity Day (Adapted, 2019)