UNIVERSITY OF SOUTHAMPTON

Using software-based acoustic detection and supporting tools to enable large-scale environmental monitoring

by

Peter Prince

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the
Faculty of Physical Sciences and Engineering
School of Electronics and Computer Science

October 2019

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF PHYSICAL SCIENCES AND ENGINEERING
SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

by Peter Prince

Acoustic monitoring tools are often constrained to small-scale, short-term studies due to high energy consumption, limited storage, and high equipment costs. To broaden the scope of monitoring projects, affordability, energy efficiency, and space efficiency must be improved on such tools. This thesis describes efforts to empower researchers charged with monitoring ecosystems, faced with the challenges of limited budgets and cryptic targeted events. To this end AudioMoth was developed: a low-cost, open-source acoustic monitoring device which has been widely adopted by the conservation community, with over 6,600 devices sold as of August 2019.

This thesis covers the development and deployment of three acoustic detection algorithms that reduce the power and storage requirements of acoustic monitoring. The algorithms aim to detect bat echolocation, to search for evidence of a endangered cicada species, and to collect evidence of poaching in a protected nature reserve. Each algorithm addresses a detection task of increasing complexity - analysing samples multiple times to prevent missed events, implementing extra analytical steps to account for environmental conditions such as wind, and incorporating a hidden Markov model for sample classification in both the time and frequency domain. For each algorithm this thesis reports on their detection accuracy as well as real-world deployments carried out with partner organisations. The deployments demonstrate how acoustic detection algorithms extend the use of low-cost, open-source hardware and facilitate a new avenue for conservation researchers to perform large-scale monitoring.

The research also covers an analysis of the accessibility of acoustic monitoring technology, focusing on AudioMoth and its supporting software. This is done using a 75-respondent questionnaire and a thematic analysis done on a series of interviews. The results of both analyses discovered a number of potential methods for improving acoustic monitoring technology in terms of the various forms of accessibility (financial, usability, etc.). The community responses, along with the popularity of AudioMoth and the success of the deployed detection algorithms demonstrate the benefits of providing accessible acoustic monitoring solutions to conservationists.

Contents

List of Figures

Declaration of authorship

I, Peter Christopher Prince, declare that this thesis and the work presented in it is my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Either none of this work has been published before submission, or parts of this work have been published as:
    - P Prince, A Hill, E Piña Covarrubias, P Doncaster, J L Snaddon, and A Rogers. Deploying acoustic detection algorithms on low-cost, open-source acoustic sensors for environmental monitoring. *Sensors*, 19(3):553, 2019. doi: 10.3390/s19030553
    - A P Hill, P Prince, E Covarrubias, C P Doncaster, J L Snaddon, and A Rogers. Audiomoth: Evaluation of a smart open acoustic device for monitoring biodiversity and the environment. *Methods in Ecology and Evolution*, 2017. doi: 10.1111/2041-210X.12955

- E Piña-Covarrubias, A P Hill, P Prince, J L Snaddon, A Rogers, and C P Doncaster. Optimization of sensor deployment for acoustic detection and localization in terrestrial environments. *Remote Sensing in Ecology and Conservation*, 2018. doi: 10.1002/rse2.97
- A P Hill, P Prince, J L Snaddon, C P Doncaster, and A Rogers. Audiomoth: A low-cost acoustic device for monitoring biodiversity and the environment. *HardwareX*, 2019b. doi: 10.1016/j.ohx.2019.e00073
- A P Hill, A Davies, P Prince, J L Snaddon, C P Doncaster, and A Rogers. Leveraging conservation action with open-source hardware. *Conservation Letters*, 2019a. doi: 10.1111/conl.12661

Signature: ....

Date: ....

Acknowledgements

I would like to thank my supervisors: Professor Alex Rogers, Professor Patrick Doncaster, and Doctor Jake Snaddon for their guidance throughout my PhD. I also wish to express how much I have appreciated the teamwork and friendship with Andy Hill as we have worked together on this research for the past few years. Thank you as well to Evelyn Piña Covarrubias for her assistance with the deployments carried out in Belize and Mexico.

For their assistance during the many field tests and deployments carried out over the course of the project I also wish to thank Ruby Lee, Professor Kirk Martinez, Jack Merrifield, Oliver Wearn, BugLife, and the Pook's Hill staff. Thank you to Gabe Miller at San Diego Zoo, for accommodating myself and Andy, and for showing us San Diego.

For helping grow the AudioMoth into the successful conservation tool it is today, I want to thank Alisdair Davies at the Arribada Initiative. His advice, the handling of the group purchasing, and tireless efforts to spread the word about our work have been invaluable.

On a personal note, I want to thank my family for supporting me over the course of my PhD and convincing me to return to the rainforest each time, despite the ever-present risk of snake-bites, heatstroke, and malaria. I also want to thank my friends including (but not limited to) Ibrahim Dirar, Matthew Juggins, Ryan Duffy, and Zack Stephens for their support and for not replacing me when I spent all that time abroad.

This work was supported by an Interdisciplinary Science and Enterprise grant from the University of Southampton Institute for Life Sciences, a Natural Environmental Research Council SPITFIRE DTP award (NE/L002531/1).

Chapter 1

Introduction

> It is said that the opposite of noise is silence. This isn't true. Silence is only the absence of noise.
>
> — Terry Pratchett

According to the Living Planet Index, one of several indices used by the WWF to quantise biodiversity, there was a 58% decline in vertebrate species between 1970 and 2012. When cataloguing species in the LPI database, the WWF includes a record of threats associated with that species. Possible threats include climate change, habitat loss, species over-exploitation, pollution, invasive species and disease. Surprisingly, habitat loss dominates all other threats, affecting 50% of all terrestrial species studied.

As more species become threatened as a result of habitat degradation and ecosystem collapse, it becomes increasingly important for ecologists to possess the data necessary to reduce or at least understand the factors which contribute. One such method of obtaining this data is monitoring that ecosystem. This includes observing species within that ecosystem to study behaviour or population sizes, and monitoring for anthropogenic disturbances such as the presence of poachers or illegal tree-felling. Current ecosystem monitoring techniques work well in certain settings but fail in others, leading to critical gaps in biodiversity data available. One such example is the study of invertebrates, major contributors to most ecosystem processes yet difficult to survey effectively due to the time-consuming, inaccurate nature of most currently-used techniques [Kim, 1993]. Species which are too small or concealed to be picked up reliably by sensing methods such as camera trapping are left relatively unmonitored. Filling species gaps in global ecosystem data is one of the main challenges which monitoring techniques must face.

Monitoring the causes of biodiversity loss can be as integral as surveying its effects on wildlife in that ecosystem. Habitat destruction and degradation in tropical forests is often a result of selective logging and poaching [Laurance, 1999]. This habitat type which is particularly challenging for large-scale manual monitoring in due to the remoteness of these locations and the difficulty of traversing them.

1.1 Problem Statement

Cryptic surveillance is the monitoring of species possessing markings, colouration or size which serve to camouflage it when in its habitat. This form of monitoring can be extremely difficult for visual monitoring techniques. One form of monitoring which fills the niche of cryptic surveillance is the use of acoustics. Acoustic monitoring is a powerful tool for this application which has dramatically grown in popularity in recent years. However its accessibility to researchers with limited funding is minimal and projects able to utilise it are often limited in scope [James et al., 1999]. This is because of the high investment costs involved in either acquiring the necessary commercial equipment or developing bespoke tools.

Two factors which contribute to the high initial investment cost of many acoustic monitoring devices are the storage and energy requirements of long-term monitoring. Even if the cost of the hardware itself is kept low, sufficiently large batteries and data storage can drastically increase the unit cost of acoustic monitoring devices. Alternatives such as using smaller storage mediums but replacing them at regular intervals still require large batteries and come with the additional issue of further habitat disruption with each replacement. This methodology is also unfeasible for deployments in difficult to access locations.

1.2 Relationship to Existing Approaches

It could be assumed that simply broadening the scope of existing monitoring techniques and deploying them more extensively could deal with the previously-listed challenges facing ecosystem monitoring, dedicating more resources to the act of monitoring these previously unobserved species and human disturbances. Unfortunately, in areas where these issues are most prevalent and in research departments tasked with studying these ecosystems, the resources are rarely available [James et al., 1999]. Anthropogenic disturbances in tropical forests are cryptic and will occur in remote, difficult-to-access locations, limiting the effectiveness of monitoring techniques such as camera trapping and the use of satellite imagery.

Remote acoustic monitoring is an alternative technique which can be used to perform large-scale monitoring of cryptic events. Devices designed to listen for small-scale cryptic events such as pest insects are commonplace. Implementing similar techniques for applications involving sound travelling through air requires dealing with additional issues such as extraneous noise sources creating more complex acoustic environments. These complications must be dealt with in order to apply similar remote acoustic monitoring techniques to large-scale monitoring tasks. When applied effectively, acoustic monitoring can be used for a wide variety of applications such as monitoring dispersed elephant populations through annual acoustic surveys [Keen et al., 2017], analysing the vocabulary of humpback whales to monitor behavioural patterns [Stimpert et al., 2011] and tracking visually-identical owls across large ranges by identifying their distinct vocalisations [Grava et al., 2008].

In these applications, noise refers to any sounds produced besides the targeted vocalisation or acoustic event. Manual inspection and automatic detection algorithms can both be used to filter out this noise, discarding irrelevant recordings or samples. Automatic detection can be done with either reactive monitoring devices or analysis after collection.

Passive acoustic monitors (PAMs) used purely for data collection constantly record throughout a period when the target sound is likely to occur, recording large quantities of irrelevant audio. This data can then be filtered out, leaving just recordings relevant to the original study once the deployment is complete [Stowell et al., 2016]. Because all processing is done after a deployment, the monitors themselves can be extremely simple, requiring nothing in terms of functionality besides the ability to keep track of the time and record.

After retrieving the recordings, computationally-intensive analysis processes can be used to identify target sounds, because computing power is unlikely to be a limitation. These processes can incorporate processing techniques such as spectrogram generation, signal energy entropy, and mel-cepstrum frequency coefficients (MFCCs) [Blumstein et al., 2011] as well as various deep-learning techniques [Cakir et al., 2017] to achieve high levels of detection accuracy.

One of the benefits of producing large recording datasets through passive monitoring is the availability of data for future research. While vocalisations of non-target species may be considered noise for one application, a large set of audio recordings could be applied to subsequent studies targeting other acoustic events in the same location. With the recent push towards open science and sharing collections of “big data” in ecology, historical datasets can help contribute to subsequent research projects [Hampton et al., 2013].

Researchers utilising passive acoustic monitoring in this way report using a variety of different acoustic equipment, the most common of which being high-cost commercial sensors such as the SongMeter series from Wildlife Acoustics [2018]. These devices are extremely high quality, reflecting their price, and catering to projects requiring high fidelity recordings. However, by exclusively targeting this market a niche is left for projects prioritising affordability over audio quality. Affordable devices increase the potential scope of research done as well as making monitoring techniques more accessible. This is especially applicable to communities in developing countries where the adoption of acoustic monitoring technology is unlikely due to the minimal coverage possible given the high unit cost of commercial devices relative to their limited budgets.

Despite the high cost of commercial devices, they often lack features or specifications required for specific applications. Companies producing monitoring equipment very rarely produce highly specialised devices, one example of such a device is the SM4-BAT[1], sold to cater to the large bat conservation community. Extremely specialised devices such as the SM4BAT are an uncommon occurrence and exist due to demand from a sufficiently large userbase. It is extremely difficult to design and manufacture products with small target audiences while still making a profit [Kwok, 2017]. Without sufficient demand, companies generally focus instead on features likely to be applicable to a wide variety of users.

In response to the high unit costs of commercial devices, there is a growing trend in the conservation community towards developing sensors specifically for a given monitoring application [Troscianko and Rutz, 2015, Prinz et al., 2016]. This is likely due to the growth of user-friendly micro-computers such as the Raspberry Pi. Pi-based devices possess a full Linux distribution which makes programming a smart sensor a much more accessible task, significantly lowering the barrier to entry in terms of necessary technical expertise [Tashakkori et al., 2017]. This ease-of-use comes at a cost in terms of energy efficiency, with the energy required to run the full operating system forcing long-term deployments to make use of automotive batteries to get sufficient temporal coverage.

Another form that homemade, bespoke devices have taken is custom hardware. Acoustic sensors using custom electronic designs typically use computationally-constrained micro-controllers which run without an operating system to enable low-power operation. These devices trade accessibility for lower energy consumption, requiring a greater level of programming and electronic engineering knowledge in order to design and build such devices from scratch. In research institutions this can be made possible through cross-disciplinary projects, however for many conservationists this technical expertise is not readily available. As a result, the only available option is often the high-cost, high-fidelity commercial devices which limit the scope of monitoring projects due to the large investment required.

For any passive acoustic monitoring device, there is a requirement to store large quantities of data, the majority of which is likely to be useless for a specific application, especially when targeting infrequent events. A single 15-second recording at 48 kHz is approximately 1.4 MB in size, meaning a device set to record continuously requires 5.9 GB of storage capacity for each day of deployment. The act of writing a recording to a monitoring device's given storage medium is often one of the most energy intensive tasks it performs. Because of this, deployed acoustic monitors are extremely limited by both storage and energy capacity if set to record continuously.

[1] https://www.wildlifeacoustics.com/products/song-meter-sm4bat

In contrast to post-collection analysis, reactive recording devices filter audio before recording. Acoustic monitoring devices such as this operate similarly to camera traps, recording in response to a pre-programmed trigger. This limits recordings to those containing what it believes is the target sound and drastically reduces storage and energy requirements for long-term acoustic monitoring deployments.

Reactive recording devices require sufficient computational resources (clock speed, available memory, etc.) to process audio in real time, meaning they require more powerful hardware than constantly recording acoustic monitors. Also required is the ability to alter the underlying software to implement detection algorithms which decide whether or not to record. As commercial devices are closed source, adapting them to active monitoring applications is difficult or impossible. As a result, active monitoring is generally done using bespoke devices, built from scratch and based around modular computers such as the Raspberry Pi or custom hardware.

These acoustic monitoring tools offer distinct solutions to the the challenges associated with acoustic monitoring for conservation. However, of those discussed, none singularly aim to fulfil the requirements of accessibility, flexibility, and high energy efficiency. Based on these requirements, this thesis covers both the development of a low-cost, low-energy acoustic monitoring device and multiple acoustic detection algorithms which use the device as a deployment platform.

1.3 Motivation

This thesis aims to show the utility of low-cost acoustic monitoring tools and techniques for improving the accessibility of ecosystem assessment technology. As this technology becomes more accessible both financially and in terms of user-friendliness, the scope and diversity of research questions which can be asked widens and threatened ecosystems gain access to better protection. Three algorithm case studies are used as evidence of this, displaying utility in the detection of both biophony and anthrophony, presenting research applications which could not be pursued at their given scales without the addition of low-cost acoustic monitoring technology.

By improving the accessibility of monitoring methodologies both in terms of the resources required to implement them and their user-friendliness, the field of bioacoustic monitoring takes steps towards local communities being able to affordably monitor their own natural resources. This is an important milestone in the protection of global biodiversity and the prevention of habitat loss.

1.4 Challenges

Developing both detection algorithms and the low-cost monitoring platform on which they were deployed presents a number of challenges. These range from programming implementation difficulties to logistical challenges involved in the development and adoption of an affordable piece of conservation technology. Testing the effectiveness of the monitoring tools (including the detection hardware, the detection algorithms and all necessary supporting software) also presented additional challenges in terms of planning and carrying out real-world deployments.

Once a sufficiently large user community for a piece of technology exists, their feedback can be used to iterate on the software and hardware to create a tool which satisfies the community's requirements. Having users deploying the technology for a variety of applications in a variety of environments creates diverse feedback. As well as the benefits gained in terms of feedback, with enough users wishing to acquire a device, an economy of scale can be achieved in manufacture. Doing so can reduce the unit cost of each device. This is integral to achieving the level of financial accessibility being targeted. These effects can occur once the device and its capabilities are widely-known, however gaining the initial audience is a major challenge. The conservation community must witness a tool or piece of technology being used successfully by others before they're confident enough to dedicate limited research funding to it.

When data collection technology is adopted in this way it can both enable previously unfeasible projects and improve upon existing methodologies. For example, a large proportion of ecological monitoring is carried out using manual surveying. The introduction of certain technologies can make this data collection process faster and reduce the resources required in terms of effort. When introducing technology to established methodologies, the technological expertise of the average user as well as their reluctance to learn a complex new system must be considered. Designing software and hardware with these factors in mind is an additional challenge for this research. The possible levels of technical expertise become significantly more diverse when this technology is used in studies utilising citizen science [Irwin, 2002]. Users can come from no scientific or technological background and the additional resources required to train them can mitigate the benefits of the technology.

In terms of detection algorithm development, different challenges arise for each attempted based on the features of the target sound and the environment it's deployed into. For example, when detecting bats, only sounds in the ultrasonic band (upwards of 22 kHz) are relevant, so background noise, which is predominantly in the audible spectrum, is easy to filter out. However echolocation calls which are used to identify the presence of bats are extremely short and only audible while the bat is briefly within range of a microphone, presenting implementation challenges. While solutions to problems such as these are often transferable, every application comes with a combination of challenges unique to that application which must be factored into the algorithm design.

Further challenges for the algorithm design come from the limitations imposed by the deployment platform. As the aim of the research is to enable acoustic monitoring on low-cost, low-power hardware, resources such as the processing power and memory availability prevent many commonly-used solutions to digital signal processing (DSP) problems from being utilised. When working with the frequency domain of a digital signal, Fourier transforms are used to decompose that signal into the frequencies which compose it. The presence or absence of particular frequencies is a common detection method [Valenzise et al., 2007, Tzanetakis and Cook, 2002]. However, calculating a complete discrete Fourier transform on a set of samples collected at a high sample rate is time-consuming and this process, combined with further DSP techniques will produce a detection algorithm unable to run on low-cost microprocessors in real time. When designing an active monitoring system which records in response to events, running in real time is vital.

Finally, real-world deployments used to test the effectiveness of both the detection algorithms offer further logistical challenges. Deployments carried out must accurately reflect the standard practices employed in modern conservation research. This includes sufficiently rigorous data collection methods. If the newly-developed technology improves this process in terms of efficiency, potential scope or the overall accessibility of acoustic monitoring in a substantial and meaningful way, then it can be deemed successful.

1.5 Contributions

The main contributions of this work are the development of three acoustic detection algorithms capable of running on low-power, low-cost hardware in order to enable affordable acoustic monitoring for a variety of applications. The hardware itself is also a co-created contribution, with its development and all associated supporting software presented as part of this thesis.

1. The development of AudioMoth, an open-source, low-power acoustic monitoring device of our own design, as a low-cost alternative to commercial and bespoke acoustic monitoring tools.

2. The development of multiple acoustic analysis algorithms capable of detecting and reacting to targeted acoustic events. These include:

   - Generalised bat echolocation detection, with the Cuban greater funnel-eared bat chosen as an example species for field-testing.

- New Forest cicada detection, aiming to monitor for the presence of a species thought to be locally extinct within the UK.
- Anthrophony detection, monitoring human interference within a protected scientific nature reserve in the form of illegal hunting, using gunshots to signify a poaching event.
- Real-world deployments for each algorithm to test effectiveness.
- A convolutional neural network designed to detect the presence of gunshots. Created to benchmark the initial gunshot detector against a commonly-used deep learning technique for acoustic detection.

3. Building a community around a low-cost acoustic monitoring platform, encouraging adoption through software directed by user feedback and support including:

   - Programming libraries and open-source code examples designed to simplify future development.
   - Extensive documentation designed to encourage users to specialise AudioMoth to their intended application.
   - A qualitative analysis of a 75-respondent questionnaire, aiming to identify the demographics of the AudioMoth userbase.
   - A thematic analysis of a series of interviews based around the accessibility and applications of the AudioMoth platform.

4. Using a combination of AudioMoth, detection algorithms and supporting software to improve the accessibility of long-term, large-scale acoustic monitoring by minimising unit costs, reducing energy and storage requirements, and developing in response to user requirements.

These contributions were also in aid of the completion of three real-world deployments aiming to achieve distinct goals. These include monitoring the behaviour and movements of the Cuban greater funnel-eared bat for a preliminary acoustic survey, confirm the presence of the New Forest cicada in the New Forest national park, and assess the level of poaching occurring in Tapir Mountain, a Belizean protected reserve. Each of these are tied to the overall aims of this research and provide evidence of success of the AudioMoth as part of a low-cost alternative framework for acoustic monitoring for conservation.

Variations of the firmware designed for these deployments have been used with AudioMoth in a number of other applications around the world. These deployments, carried out by third parties, include ecosystem monitoring projects in Kenya and Madeira, ultrasonic background noise surveying across Southampton, and a bat monitoring project carried out by the Bat Conservation Trust across the UK. Each of these projects were made possible by the hardware and software developed as part of this research.

The work in this thesis has contributed in part or in full to the following publications:

- P Prince, A Hill, E Piña Covarrubias, P Doncaster, J L Snaddon, and A Rogers. Deploying acoustic detection algorithms on low-cost, open-source acoustic sensors for environmental monitoring. *Sensors*, 19(3):553, 2019. doi: 10.3390/s19030553
- A P Hill, P Prince, E Covarrubias, C P Doncaster, J L Snaddon, and A Rogers. Audiomoth: Evaluation of a smart open acoustic device for monitoring biodiversity and the environment. *Methods in Ecology and Evolution*, 2017. doi: 10.1111/2041-210X.12955
- E Piña-Covarrubias, A P Hill, P Prince, J L Snaddon, A Rogers, and C P Doncaster. Optimization of sensor deployment for acoustic detection and localization in terrestrial environments. *Remote Sensing in Ecology and Conservation*, 2018. doi: 10.1002/rse2.97
- A P Hill, A Davies, P Prince, J L Snaddon, C P Doncaster, and A Rogers. Leveraging conservation action with open-source hardware. *Conservation Letters*, 2019a. doi: 10.1111/conl.12661
- A P Hill, P Prince, J L Snaddon, C P Doncaster, and A Rogers. Audiomoth: A low-cost acoustic device for monitoring biodiversity and the environment. *HardwareX*, 2019b. doi: 10.1016/j.ohx.2019.e00073

1.6 Thesis outline

In Chapter 2, this thesis presents background information on a variety of topics relevant to the research carried out. This background information includes various monitoring techniques within the field of ecology in Section 2.1, acoustic detection tools across all fields in Section 2.2, acoustic detection algorithms employed by these tools in Section 2.4, and overviews of the monitoring projects carried out throughout the following chapters in Section 2.5. Chapter 3 continues with background information, presenting the hardware specifications, functionality, and history of the AudioMoth: a low-cost, open-source acoustic monitoring device developed as part of this thesis.

It then reports in Chapter 4 the development, performance, and deployment results of two acoustic detection algorithms designed for zoological monitoring applications. These two projects include a bat detection and a cicada detection algorithm. Both are constrained by the hardware limitations of AudioMoth and the need to operate while consuming as little energy as possible, in order to maximise the device lifespan and minimise costs to researchers.

Chapter 5 then shows an extension of the algorithms shown in Chapter 4, describing a third detection algorithm designed to detect gunshots within a tropical rainforest environment using a hidden Markov model. This chapter includes the design, an analysis of its performance, an overview of its deployment in Belize, and a comparison with a deep learning alternative. This alternative (described in Section 5.2) is a depthwise-separable convolutional neural network and was compared with the hidden Markov model in order to provide a benchmark against a commonly-used deep learning technique.

Chapter 6 covers an analysis of AudioMoth as an acoustic monitoring tool in terms of user accessibility. This includes a quantitative study of responses to a questionnaire given to 75 AudioMoth users in Section 6.1, a qualitative analysis of a series of follow-up interviews in Section 6.2, and an overview of a number of tools already used to improve the accessibility in Sections 6.3, 6.4, and 6.5.

Finally, Chapter 7 concludes the thesis, describes how the author plans to continue development of the tools developed as part of the research, and discusses a number of possible directions for future work. There are also a number of following appendices including a list of the questions asked as part of the questionnaire in Appendix A; the questions and talking points used to prompt discussion in the follow-up interviews in Appendix B; a list of media engagements relating to the work discussed (articles, interviews, blog posts, etc.) in Appendix C; and documentation listing the functions included in the AudioMoth detection algorithm library in Appendix D.

Chapter 2

Background

People hearing without listening.

Paul Simon and Art Garfunkel

This chapter covers current monitoring methods used for conservation, ranging from manual surveys to satellite imagery. It also describes the growth and benefits of acoustic monitoring, methods currently deployed for the detection of both zoological and anthropogenic sounds and digital signal processing techniques employed to implement acoustic detection algorithms. Finally, the targeted species and deployment environments for each of the three real-world deployments used to field-test the effectiveness of AudioMoth and the detection algorithms are described.

2.1 Ecosystem monitoring for conservation

Ecosystem monitoring is the use of various techniques to study the functions and actors within a living ecosystem. This can be through tracking aspects of the location such as land-use coverage or biomass estimations, or by studying changes in behaviour or population size of animals which both contribute and benefit from the ecosystem. It is done for a number of reasons including assessing vulnerabilities within the environment, assisting in the development of strategies for more sustainable use of natural resources and studying the effects of major ecological events such as climate change.

Studying the effects of climate change on an ecosystem requires long-term studies to understand the various slow processes which can be put off balance by even minor temperature and weather changes. It is for this reason that there has been a push towards greater levels of ecosystem monitoring in arctic regions likely to dramatically change as the effects of climate change become more pronounced [Ims and Yoccoz, 2017]. Many species within strongly-affected environments are referred to as indicator species as their behaviour reflects the overall health of an ecosystem and are thus invaluable monitoring targets. As a result they are regularly monitored to gauge the effects of threats including both climate change and human encroachment [wa Maina et al., 2016].

As well as studying environmental damage, monitoring in this way can also be used to assess the success of actions carried out to protect an ecosystem. The Parks Canada Agency uses various ecosystem monitoring techniques to improve their park management techniques, provide a baseline for researchers studying unprotected ecosystems to compare to, and maintain ecological integrity in the area [Parks Canada, 2019]. Ecological integrity refers to maintaining *"... a condition that is determined to be characteristic of its natural region and likely to persist"*. By monitoring and maintaining a record of various factors deemed relevant to the ecological integrity, including species richness, population dynamics, and habitat fragmentation, changes can be detected and this integrity can be preserved.

There exists the two major challenges facing ecosystem monitoring: filling species gaps in existing data and effectively monitoring anthropogenic disturbances such as poaching and logging in threatened environments. Ecosystem monitoring is generally performed using either field surveys, which involve sending researchers to the location to observe the ecosystem, or remote monitoring, which uses recording devices and sensors to collect data without the presence of a human.

Field surveys are commonly used in species population studies [Thompson, 2002] and work best in smaller, easily accessible sites. Standardised field survey methods which have existed for decades are well-understood and trusted by the conservation community. Since the 1960's, the Breeding Bird Survey (BBS) has used field surveys to monitor bird populations across the United States to both build range maps of various species and monitor the health of their populations [Robbins et al., 1986].

However, field surveys are limited by both personpower and time available, with large-scale monitoring projects requiring whole teams of observers properly equipped to take samples. The previously mentioned BBS pools data from thousands of bird-watchers across the continent to obtain sufficient coverage. While citizen science techniques such as this can reduce the cost of such surveys, resources are required to confirm the accuracy of these results and the reliability of each contributor. Manual surveys also cannot be constant as it is not feasible to always have surveying teams monitoring an area manually, especially in locations which are difficult or dangerous to access. In order to minimise personpower costs, field surveys are often carried out intermittently, taking the results as an average representation of the area. While this can be effective, large amounts of detail are lost in doing so.

Remote monitoring is a powerful alternative to manual monitoring. Unlike direct sample collection techniques, they do not require the physical presence of an observer to conduct a survey, making them much less intrusive after the initial installation. No longer requiring the presence of an observer is important when monitoring anthropogenic disturbances, such as the presence of poachers. A team of rangers surveying an area could encourage the behaviour of poachers to adapt and only hunt in periods when the surveys aren't being performed, making the observations less representative of the ecosystem.

One such example of remote monitoring is the camera trap. Camera traps are fixed, remotely activated cameras which utilise either motion detection or an infra-red sensor to trigger a photo or video recording [Rovero et al., 2013]. Camera traps can remain monitoring constantly, provided adequate power and data storage. For medium to large terrestrial mammals, camera traps work extremely well for capturing and producing thorough inventories, given a sufficient number of devices. Tobler et al. [2008] took an area of rainforest and was able to perform a survey capturing 86% of all species known for the area in just two months, while previous manual inventories took between 3 and 21 years to complete [Voss and Emmons, 1996]. This form of remote monitoring drastically improved researchers' ability to perform rapid inventories of a large area of forest.

This form of remote monitoring can still cause disruptions when deployed. For example, specialised equipment is required for camera traps to remotely monitor nocturnal species. One solution which is commonly employed to do this is a flash. It has been noticed that flashes can train species such as tigers to become "trap-shy" and avoid deployment areas, disrupting the target species, thus negating one of the major benefits of remote monitoring over manual surveys [Wegge et al., 2004].

Camera traps are also limited by their directionality, range, and the surrounding vegetation which can obscure targeted events. This is not an issue in urban environments with very little vegetation or when there exists a location researchers are certain they will return to, such as a fox den or a baited feeding site [Hegglin et al., 2004]. However, a single camera trap in dense forest will cover a relatively limited area, meaning the positioning of such devices is vital to collecting any quantity of data.

Contrasting camera trapping in terms of this limitation, satellite imagery can be collected on a large-scale for remote locations with relative ease compared to a grid of camera traps or a field survey team. This is especially true in inaccessible locations where biodiversity surveying is almost impossible, but no less vital. Even using low-resolution images, differences in tree coverage and flora over time can be an important tool in studying the health of habitats in remote locations such as deep in amazonian rainforests [Tuomisto, 1998]. Large-scale habitat studies such as these can be done over a long time, tracking changes not immediately obvious with ground-level observations.

The images produced from satellite imagery can cover a large area, providing high-resolution data on real-time habitat changes to detect events such as deliberate deforestation and fire. It is often claimed that Earth observation satellites reduce per-area costs dramatically [Marvin et al., 2016], however the investment required to gain access to this technology can often exceed the budgets assigned to conservation projects. This is even more likely in developing countries, where they are often the worst affected by biodiversity loss yet the least likely to possess the funds to purchase this monitoring tool [Danielsen et al., 2003]. While this has been the case in the past, more satellite imaging is becoming freely available in certain areas, mitigating this limitation. As a result, more researchers now claim that satellite imagery is key for monitoring global biodiversity [Turner et al., 2015].

The biggest limitation of satellite-based monitoring is the inability to detect cryptic events across the monitored area, with cloud cover and the tree canopy obscuring certain events and the presence of insects being impossible to detect. Satellite images are also subject to multiple "noisy" environments, experiencing visual noise from the Earth's surface as well as the atmosphere itself [Kerr and Ostrovsky, 2003]. Extra effort must be dedicated to dealing with these extra sources of noise on any collected data.

2.2 Acoustic monitoring for conservation

Due to the visual nature of camera trapping and satellite imagery, both suffer from similar issues, failing to capture certain events occurring within an ecosystem due to physical obstructions and coverage limitations. It is unreasonable to expect purely visible sensors to detect the presence of an insect species in a large monitoring area. Gaps such as these can be filled through the use of acoustics, capitalising on the reduced effect of obstructing foliage in natural environments. Many conservation projects have turned to acoustics for monitoring both humans and animals operating within their chosen ecosystem. Picking out targeted components of the ecosystem's soundscape ("sound landscape") provides a wide variety of new monitoring opportunities including detection, localisation and species recognition [Gillespie et al., 2008, Briggs et al., 2009].

Components of an ecosystem's soundscape can be divided into three categories: geophony, anthrophony and zoophony. Geophony covers sounds produced by the the non-biological world, including weather and natural water sources. Zoophony covers the various vocalisations by any species of animal or insect within an ecosystem. Anthrophony refers to any man-made sound, both directly from humans and from machinery or tools such as motor vehicles. Acoustic monitoring for conservation generally targets either zoophony or anthrophony and my work developing detection methods of each is discussed in chapters 4 and 5 respectively.

2.2.1 Zoological acoustic monitoring

For both humans and animals, sound is an integral component of social interaction. From the simple exchange of information to more complex processes such as a mating, animals use some form of vocalisation as part of almost all important social behaviours.

As a result, variations in these vocalisations work as excellent indicators of behavioural changes within a species [Laiolo, 2010]. In recent years, research into these behavioural changes has been in the context of reactions to human interference [Rabin and Greene, 2002]. This includes hunting difficulties in areas with substantial traffic noise [Siemers and Schaub, 2010] and animals adapting to noisy environments by altering the frequency [Hu and Cardoso, 2010] or amplitude [Brumm, 2004] of their calls. Vocalisations can also be used to study a species' perception of anthropogenic disturbances by their use of alarm calls usually reserved for predators [Hollen and Radford, 2009].

As well as studies into the behaviour of animals, acoustic monitoring is also an effective method for detecting the presence of a species in a variety of environments. Deliberate sounds for the purpose of communication are often referred to as "non-incidental" sounds, whereas sounds produced as a byproduct of movement are referred as "incidental" [Chesmore, 2004]. Both of these sound types are used for presence detection.

Hydrophones have found extensive use within the field of conservation, allowing for the detection of aquatic mammals over vast distances by listening for their calls. Killer whale calls remain detectable up to 16 km away [Miller, 2006], meaning a small number of monitoring devices can provide coverage for a large area. Acoustic monitoring works well in this context thanks to both the amplitude of these calls and their frequency, which rarely has competing sound sources in a similar band [Stafford et al., 1998]. This form of species tracking is often done to monitor the effects of shipping lanes and offshore drilling [Heidemann et al., 2006]. Both human activities which produce a significant amount of disturbance for a large area of ocean.

For terrestrial monitoring, acoustic detection covers a wide variety of species. For both conservation researchers and citizen scientists a common application of acoustics is in bat monitoring [Johnson et al., 2002]. Acoustics work well for this application for a number of reasons. First being the difficulty of visually detecting bats at night when they are most active and second, the small number of noise sources within ultrasonic frequency bands makes it easy to confirm the presence of a bat. Bat vocalisations vary in frequency depending on the species, ranging from 11 kHz (*Euderma maculatum* [Fullard and Dawson, 1997]) and 212 kHz (*Cloeotis percivali* [Fenton and Bell, 1981]). This variation is generally down to the hunting target preferred by a species. If the wavelength of the echolocation call is longer than the length of their prey's body, then echolocation responses are weaker and thus harder to hear [Thomas et al., 2004]. Because of this, acoustic detection methods can target specific species of bat if their primary diet is known. When carrying out bat surveys, devices are either mobile and carried throughout the survey area to achieve coverage or deployed overnight in a fixed location.

Outside of bat detection, fixed passive acoustic monitors (PAMs) are more commonly used, due to the benefits of remote monitoring over manual surveys discussed previously (see Section 2.1). These PAMs are used for a wide variety of species including primates

[Kalan et al., 2015], birds [Bardeli et al., 2010], and elephants [Wrege et al., 2017]. Similar to many aquatic mammals, large terrestrial mammals such as elephants produce calls which are detectable over large distances [Poole et al., 1988] and are thus excellent candidates for acoustic monitoring, allowing for a small number of acoustic sensors to measure herd size and composition across a large area [Payne et al., 2003]. For all passive monitoring projects intending to filter data after collection, PAMs are deployed and set to either record constantly or record during periods when the target species is likely to be active, based on historical data. Once the data has been collected, it is processed either manually or using acoustic detection algorithms designed to automatically identify recordings which contain targeted audio (see Section 2.4).

As well as targeting specific species, acoustic monitoring can also be used to document biodiversity and ecosystem health as a whole [Sueur et al., 2014]. In order to do so, the large quantity of information contained in extended soundscapes is summarised into characteristics which represent various aspects of the acoustic environment. As well as requiring less space to store than the original samples, indices such acoustic heterogeneity can act as indicators of biodiversity. For example, acoustic heterogeneity can correlate with species richness [Gasc et al., 2015].

2.2.2 Anthropogenic acoustic monitoring

Anthropocentric monitoring within conservation frequently focuses on unlawful incursions into protected areas, often for poaching [Astaras et al., 2017]. A large proportion of illegal hunting done in forests is performed using guns such as shotguns and rifles. These produce a loud, distinctive sound which carries for long distances, making it the primary signal of a poaching event. As most illegal hunting occurs at night, specialist equipment is required for visual sensors such as camera traps to even function for this task. Acoustic sensors operate at night without requiring add-ons such as infrared or camera flashes, and possess the ability to detect gunshots over much greater distances, making it much harder for poachers to avoid monitored areas if they are aware of a monitoring presence. Acoustic sensors also do not require direct line-of-sight with their target, allowing for greater discretion. This is especially useful when attempting to monitor humans likely to steal, destroy, or interfere with devices preventing the continuation of their illegal activities.

While studying the causes of habitat destruction, efforts are split between studying them after the fact to effect change through policy-makers [Malhi et al., 2008] and detection in order to apply preventative measures [Shimabukuro et al., 2007]. Proactive intentions require monitoring methods to not just record an anthropogenic disturbance such as a gunshot, but alert relevant authorities on a timescale such that the damage can be prevented. This creates additional challenges for remote sensing, as devices must be capable of reacting and communicating the occurrence of these events in a timely manner. In terms of manual monitoring, while patrolling rangers in problem areas can enable immediate reactions to events, large temporal and spatial scale monitoring isn't feasible. As well as this, the presence of rangers in an area can cause poachers or loggers to relocate their activities to another location or time period in order to avoid detection, remote acoustic sensors benefit from being almost invisible if effectively deployed.

Real-time gunshot detection is a problem also being solved in urban environments, using sensor networks to target gun crime and terrorism [Khalid et al., 2013]. Due to the nature of the deployment location, hardware requirements differ from deployments in tropical forests. Existing urban gunshot detectors include FireFly, a system deployed by the US military and ShotSpotter. In the United States, ShotSpotter is a widely used urban gunshot detection system which has been deployed in various cities as well as areas surrounding the White House and the US Naval Observatory [Calhoun, 2008]. The ShotSpotter system uses a grid of devices with high fidelity microphones, deployed with access to mains power and high-speed wireless communication. The ShotSpotter nodes constantly listen, reacting to acoustic impulses using machine learning techniques to classify impulse recordings. This classifier is trained against a large dataset of gunshots in a variety of urban environments. If a node is certain of a gunshot, it is reported to a central hub. If multiple nodes within range of each other report gunshots within a certain timeframe, the system is confident it was a gunshot and a team of human analysts then study the transmitted recordings before law enforcement is informed. This is an energy-intensive, multi-stage process which relies on multiple detection techniques including human perception to accurately detect gunshots. Lessons can be learnt from this type of gunshot detection, however significant compromises must be made to create an alternative which is accessible to conservationists.

Monitoring for gunshots in non-urban environments has been previously attempted with distributed wireless sensor networks [González-Castano et al., 2009]. However, due to limited energy availability in these locations and the personpower required to replace devices, system cost is a major limiting factor of these networks. Sensor affordability and effective coverage must be traded off against each other.

2.3 Acoustic monitoring technology

There are a wide variety of techniques available to conservation researchers wishing to utilise acoustics to monitor a given species or ecosystem. The suitability of the technology employed by each of these techniques varies depending on the requirements of the deployment. In terms of the deployment location, these requirements include considerations for ease of access, weather conditions, and the risk of equipment theft. General considerations include the unit cost, cost of any necessary additional components (storage, batteries, etc.), and the technical expertise of those performing the deployment.

This section presents three main groups of acoustic monitoring equipment: smartphones, off-the-shelf monitoring devices, and bespoke technology built by the researcher carrying out the deployment.

2.3.1 Smartphones

Acoustic monitoring on small mobile devices has become a major growth area, using the processing capabilities and sensitive microphones of smartphones in novel ways, from identifying users' stress levels from the tone of their voice [Lu et al., 2012], to detecting deteriorating respiratory health [Sun et al., 2015]. In conservation, projects such as Abuzz by Mukundarajan et al. [2017] have used smartphones to enable citizen scientists to assist in the collection of mosquito recordings in order to understand and model the transmission of diseases such as malaria and dengue fever. Many projects have leveraged the ubiquity of these devices to perform crowd-sourcing projects which require the installation of an app rather than any specialist equipment to gather data. Smartphones already possess both a GPS and wireless communication for data upload, both of which enable a wide variety of large-scale data-gathering conservation projects. One such project is CicadaHunt by Zilli et al. [2014], which seeks to leverage the large number of visitors to the New Forest National Park in the UK, to detect a species of insect thought to be locally extinct.

These applications utilise smartphones as mobile monitoring devices, supplementing field survey techniques while still requiring the physical presence of a researcher. Smartphones have also been deployed as PAMs, with projects such as Rainforest Connection recycling outdated smartphones for deployment in tropical forests [Rainforest Connection, 2019]. These phones are equipped with external microphones and custom software, and are then used to monitor for sounds of deforestation, using the standard mobile communication network to contact a central hub when targeted events are detected.

When deployed as PAMs, the temporal coverage of a monitoring project is limited by the battery life of the smartphones and their resilience to outdoor conditions. Smartphone batteries can last just days when running even simple data collection apps whereas an environmental survey will typically require weeks or months of data collection. The smartphones will also require extra protection to survive conditions such as heavy rain and high humidity that exceed those they would experience during normal use.

There are also problems inherent to re-purposing consumer devices such as smartphones. A number of hardware and software features designed to improve the user experience consume significant amounts of power. Without the technical expertise to completely disable these peripherals or replace the underlying operating system, their energy efficiency will always be inferior when compared to bespoke acoustic sensors.

2.3.2 Off-the-shelf devices

Commercial devices designed for the specific purpose of acoustic monitoring are able to achieve greater energy efficiency by only implementing features useful to the majority of their customers. These commercial devices can be general-purpose and use external microphones to enable audible and ultrasonic recordings, or they can be specialised to a single species or group of species. These specialised devices include aquatic sensors built around hydrophones and bat detectors.

2.3.2.1 General-purpose recorders

Currently, ecology researchers wishing to utilise acoustic monitoring predominantly use high-cost, high-fidelity commercial devices which have been designed and marketed specifically for applications in ecology and conservation. Commonly-used devices include the bioacoustic recorder (BAR) from Frontier Labs [2018] and Song Meter (SM) series from Wildlife Acoustics [2018]. To use either of these devices an external microphone is required as well as the recorder itself.

The Frontier Labs BAR costs $800 for the device and a microphone[1]. It has been used extensively in soundscape ecology, using its high-fidelity microphone (capable of recording at sample rates up to 96 kHz) to perform studies of entire ecosystems by recording across the human audible range. These soundscapes have been used to study short-term effects of human interference in tropical forest environments in research such as that done by Burivalova et al. [2018]. They used BARs to record throughout the day for a small number of days, attempting to find periods where the greatest responses to human presences were found. By identifying these time periods, future research could limit monitoring periods and attempt long-term deployments with the limitations of the hardware. Short soundscapes collected over a large timescale have been used with the BAR by Sugai and Llusia [2019] to create “bioacoustic time capsules”, which act as a record of what a particular environment was like in a certain time period. These datasets allow for future studies to identify long-term trends. Both uses trade off levels of temporal coverage due to the limitations of the Frontier Labs device.

Within the field of conservation, the majority of researchers report on using bioacoustics in their application by using the Wildlife Acoustics Song Meter range. The latest Song Meter, the SM4, costs between $850 and $900, with a further $200 required for a compatible microphone[2]. These high-cost, high-fidelity recorders possess large batteries capable of 400 hours of recording time on internal batteries, with long deployments possible with additional external battery packs. Because of the ubiquity of the SM series

[1] http://www.frontierlabs.com.au/shop/bioacoustics.html
[2] https://www.wildlifeacoustics.com/products

within conservation they have been used for a wide variety of projects in diverse environments including localising birds from recordings in terrestrial environments [O'Neal, 2014], studying breeding phenology of a species of frog in mountainous forests [Willacy et al., 2015], and replacing traditional field survey techniques for surveying nightjars across the UK [Zwart et al., 2014]. The SM series provides both high quality recordings, bespoke supporting software, and the capacity for long-term deployments, however the high unit cost of devices in the SM range means that large spatial scale deployments are rarely feasible.

2.3.2.2 Bat detectors

Ultrasonic frequencies are 22 kHz or higher, outside of the hearing range of the average adult human. Due to the requirements of hearing these calls and the large community of conservationists specialising in bats, a market for specialised acoustic monitoring equipment has grown for this application. As a result, there is a variety of different ultrasonic acoustic recorders available for bat monitoring, each using different techniques to "hear" the bats' ultrasonic calls by bringing them into the audible range [Britzke et al., 2013]. For any bat recorder, signals are either directly saved as raw audio or manipulated through one of these methods either to reduce storage requirements or to make them human audible.

The Anabat from Titley Scientific [2019] does so through frequency division, wherein the device takes in samples then creates a new sound at a fraction of the original frequencies (typically around one tenth). This is done by allowing only a fraction of the original waves into the final recording. This method can require extra filtering steps in order to remove artefacts such as additional harmonics which arise as part of the frequency division process.

To compute the frequency-divided signal, zero-crossing analysis is used, a technique where samples are digitised and the points where the signal crosses zero is saved. As well as a step in frequency division, zero-crossing analysis is also used by many devices as a method of storing ultrasonic audio data while minimising storage requirements. Standard audio, recorded at a sample rate high enough to capture bat echolocation, requires large amounts of storage. For example, a one-minute recording at 384 kHz will use 220 MB of storage. By storing the data as zero-crossing information rather than raw audio, devices set to record constantly for long periods require significantly less storage capacity. However, zero-crossing analysis only tracks the harmonic with the greatest amplitude, meaning that both zero-crossing and frequency division lose important harmonic information [Parsons et al., 2000].

An alternative to frequency division is time expansion, a technique used by many ultrasonic detectors from Pettersson [2019]. Time expansion involves stretching the original signal out, this increases its duration and reduces its frequency. Because the same number of samples are being stretched out over a greater duration, time-expanded recordings are still extremely large in size and storage is still a major limitation. However, as no information is lost through the process of time expansion, the output signal can be used for spectral analysis, unlike a frequency-divided signal [Parsons et al., 2000].

2.3.2.3 Equipment availability

As well as developing technology for various applications, actually getting the developed devices into the hands of people where they are needed is a major challenge for conservation as a whole [Pimm et al., 2015]. These concerns have given rise to worries within the conservation community about "neo-liberal conservation". Defined as the commodification of nature and conservation techniques, as companies compete to produce more advanced and sophisticated equipment, the accessibility of methods and the tools used by researchers become more heavily swayed by free-market capitalism. This can drive innovation in aid of profit as features are added and the fidelity of microphones improves, but it can also drive up the price of technology which would be invaluable in developing countries most in need of it. [Dressler and Roth, 2011]

2.3.3 Bespoke devices

There is a growing trend in monitoring for conservation for researchers to design bespoke equipment based around the requirements of their application when commercial devices are deemed unsuitable. There are a number of reasons researchers turn to building sensors themselves, including commercial devices not fulfilling the requirements of their specific application and the unit cost of existing devices making large-scale deployments impossible. Researchers can work with sensor manufacturers to alter existing devices, however niche products rarely turn a profit [Kwok, 2017]. It's for this reason that the majority of acoustic sensors focus on features which are widely applicable such as high fidelity audio recordings.

To develop a bespoke acoustic sensor both money and technical expertise are required. An initial investment is required to fund the design process, produce prototypes, and perform the necessary testing before deployment. However, unless a bespoke device is highly specialised, such as equipped with a hydrophone, it's likely it can be reused in similar acoustic monitoring projects, allowing research groups to spread costs over multiple projects.

In terms of technical expertise, this can come from paid software developers and electronic engineers, or from cross-disciplinary research projects in the case of university research. In recent years, the barrier to entry for developing sensor equipment has dropped dramatically with the growth of modular micro-computers such as the Raspberry Pi and the Arduino. By providing a platform on which monitoring devices can be built, researchers outside of strictly technical fields can optimise their equipment for a given application without the historic difficulties of building such devices entirely from scratch. The Raspberry Pi in particular has provided a base for a number of self-made monitoring devices [wa Maina et al., 2016, Tashakkori et al., 2017, Prinz et al., 2016] thanks to its flexibility and the ease with which it can be programmed and configured.

The Raspberry Pi's flexibility and ease of use is thanks to its Linux operating system, allows users to implement functionality using high level languages such as Python, taking care of low-level functionality itself. Its suitability has snowballed as the community developing such devices has grown, resulting in more modular components, software and support becoming widely available. As more projects use the Pi, more published research and open-source designs exist which new projects are able to emulate. With recent pushes towards open science, developers of bespoke sensors such as SOLO [Whytock and Christie, 2017] have made their designs and code open-source, allowing future projects to build on their work and reduce development costs [Beason et al., 2018].

Unfortunately, the Raspberry Pi's user-friendliness comes at a price as a result of the full Linux operating system. Running a complete operating system makes devices built around it power-hungry relative to their task. During operation, a Raspberry Pi uses between 80 and 260 mA, depending on the model and currently running task. Because of this high level of energy consumption, applications requiring long-term deployments have had to deploy Pi-based devices with large automotive batteries [Wrege et al., 2017]. The cost of these batteries can often exceed that of the device itself, limiting their applicability as a low-cost alternative.

Even with the easier implementation afforded by devices such as the Raspberry Pi and the Arduino, homemade sensors are still limited in terms of reliability. It's extremely unlikely that any homemade sensor will be tested as rigorously as an equivalent device developed commercially. This reliability is vital when deploying sensors for long periods in locations where regular functionality checks aren't possible.

It's for these reasons that many researchers will opt for more expensive commercial devices with proven effectiveness. Published articles specifically using acoustics for conservation predominantly use commercial PAM devices such as the Wildlife Acoustics SongMeter series over developing their own bespoke devices.

2.4 Acoustic recognition algorithms

Passive acoustic monitors which listen and react to acoustic events are becoming increasingly popular. The utility of constantly recording PAMs are often extremely limited due to the energy and storage requirements of long-term deployments. Researchers have begun to use the computing power of the devices themselves to reduce these requirements, recording only when necessary. In order to do this, sensors must be able to tell the difference between both background noise from the environment and the target sound. This can vary in difficulty depending on the environment.

This form of acoustic analysis for triggering has become an established technique for detecting pest insects in grain and wood [Mankin et al., 2011, Gutiérrez et al., 2010]. For these applications it is important to quickly react to the presence of a pest such as the rice weevil (*Sitophilus oryzae*) as their presence can quickly destroy stored crops and causes significant economic damage around the world [Potamitis et al., 2009]. Acoustic detection is so widely used due to the extreme difficulty of visually detecting insects in large sacks of rice or in wood, as well as the relative ease of discerning the vibrations caused by moving insects in solid substrates. However, acoustic detection becomes significantly more complex when done in air rather than a solid substrate such as wood. This is due to the more rapid attenuation of sound signals and the presence of more competing sounds in the form of geophony and other biophony.

2.4.1 Digital signal processing

To combat the complexity of noisy environments and detect target sounds, acoustic detection techniques must increase in complexity also, using a wide variety of digital signal processing (DSP) tools to analyse and then classify a recording with confidence. These tools can be done in the time, frequency or wavelet domain. Each domain requires different amounts of computation to operate in and offers different advantages. Advancements in DSP techniques have allowed researchers to distinguish not just between species, but between individuals of a species, identifying them by factors such as sex and age [Blumstein et al., 2011].

However these elaborate detection algorithms often require substantial computing power to run in real-time, a requirement of sensors determining whether or not to record audio. This means much of the research done into acoustic detection algorithms is from the perspective of sorting through recordings collected by PAMs after the fact [Schrader and Hammerschmidt, 1997], or using computationally powerful static sensors with access to a constant power source. With increased interest in reactive recording on monitoring devices for conservation, alternative techniques are implemented to perform equivalent DSP techniques within the restrictions of low-energy hardware capable of running for long periods in natural environments.

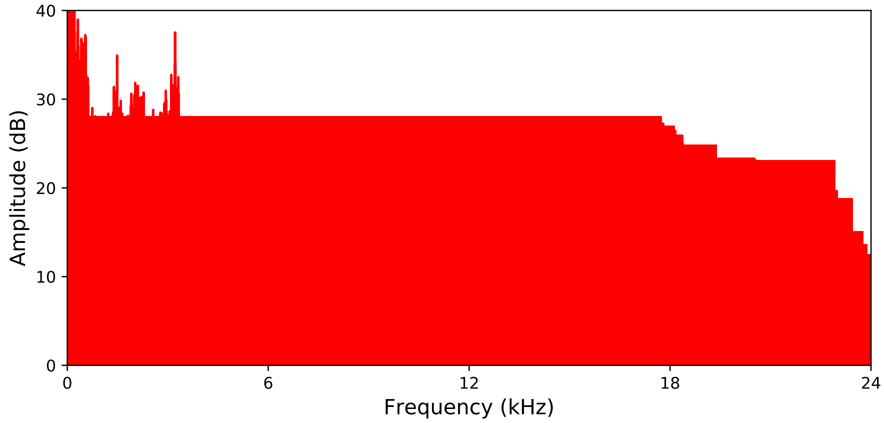


FIGURE 2.1: An FFT plot displaying the frequency components of a 48 kHz recording of a human speaking the words “one, two, three”.

2.4.1.1 Fast Fourier transform

There is a large amount of information which can be gathered by studying an animal’s vocalisations in the frequency domain. While natural environments such as forests can be extremely noisy, animals will often aim to find their own “acoustic niche”, going so far as to move their calls in the frequency domain to prevent them from being lost [Hage et al., 2013]. It’s for this reason that the unique frequency content of a targeted sound is so important, allowing detectors to use frequency presence and amplitude in certain bands as identifiers. This is often referred to as “spectral analysis”.

The Fourier transform (FT) is a commonly used technique which decompose an acoustic signal into a sum or integral of a collection of sine waves. Each of these sine waves represents a different frequency component for spectral analysis (see figure 2.1). A transformed signal consists of a series of complex values, the magnitude of the complex values representing the amplitude of a given frequency bin and the angle (found by using the inverse tangent of the real and imaginary parts) representing its phase offset.

This transformation can be done on a finite signal in the time domain, such as an audio recording, by using a discrete Fourier transform (DFT). A DFT can convert a finite sequence of audio samples into a new signal in the frequency domain. Because the DFT runs on a finite set of samples, it can be implemented with relative ease using what is known as a fast Fourier transform (FFT) implementation. This method of implementing a DFT is commonly used for both acoustic and visual analysis.

The DFT can be implemented as follows for a set of samples $(x_0, \ldots, x_{L-1})$:

$$X_k = \sum_{n=0}^{L-1} x_n e^{-i2\pi kn/L} \qquad k = 0, \ldots, L-1 \tag{2.1}$$

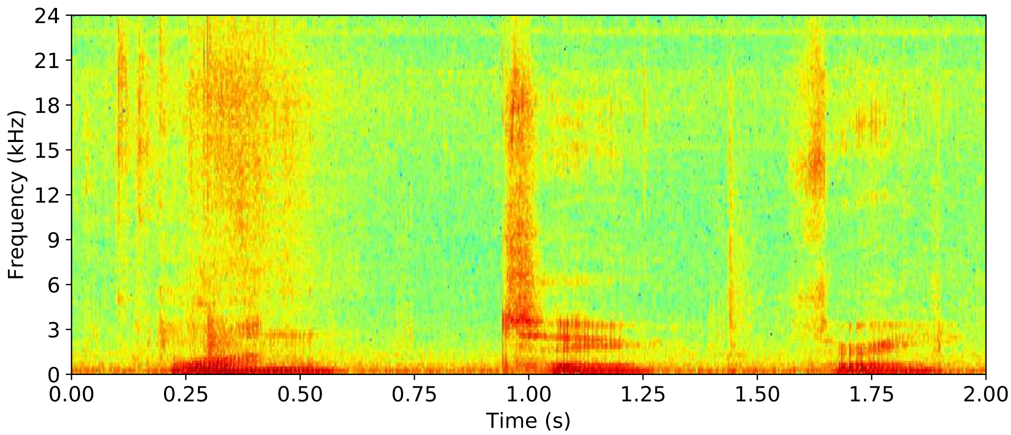


FIGURE 2.2: A spectrogram displaying a 48 kHz recording of a human speaking the words "one, two, three". There are visible differences in the spectral composition of each sound, allowing for acoustic detection techniques to both detect the presence of a human voice and discern between known words.

From this transformation, $(X_0, \ldots, X_{L-1})$ represents the amplitude and phase offset of the frequency components. When studying the presence of various frequency bands within a signal, the amplitude is obtained by applying a modulus function to the complex number. The complex Fourier signal is reversible with an inverse Fourier transform (IFT), however discarding the phase information in this way makes the transform a one-way process.

This algorithm operates with a computational complexity of $\mathcal{O}(L^2)$ on a series of L samples. Various implementations exist which calculate the same result with a complexity of $\mathcal{O}(L \log L)$, any which do so are referred to as a FFT.

One possible use of the frequency components produced by a FT is the production of a spectrogram. These plot the frequency of a signal against time, representing the amplitude of each frequency at each point in time as a pixel colour/intensity. By splitting a signal into frames and performing a series of short-term FTs, the way in which the frequency composition of the signal changes over time can be studied.

Spectrograms are often used in the field of acoustics as humans can be trained to associate spectrograms with their corresponding audio or identify specific structures (see Figure 2.2), this can drastically reduce the time required to manually inspect large quantities of acoustic data as well as allow humans to intuitively analyse audio at frequencies outside of their usual hearing range.

As well as humans visually inspecting audio, this form of spectral analysis also allows computers to do so, applying computer vision techniques to acoustic applications such as music identification [Ke et al., 2005]. By making computer vision techniques applicable to acoustic problems, new analysis methods have been developed including various machine learning techniques based around visually classifying spectrograms.

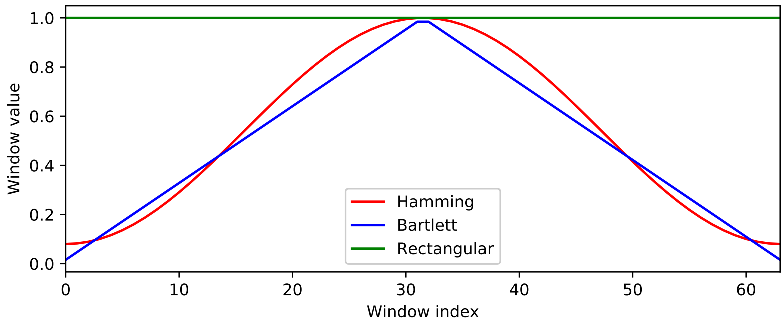


FIGURE 2.3: Three different window types: Hamming, Bartlett and rectangular. Each window affects the frequency distribution of a signal and reduces the effect of spectral leakage.

2.4.1.2 Window functions

When the sample set being processed is a finite signal, windowing is often used to prevent spectral leakage. Frequency extraction methods such as FFTs assume that any data set it is applied to is a single period of a periodic signal. If the samples being analysed happen represent a periodic signal there is no issue as it matches this assumption. However spectral leakage occurs when frequency extraction is applied to a non-periodic signal, and is often more pronounced when there is a significant disparity between the start and end points of the signal. This disparity results in high frequency components not present in the original signal. These high frequency components can also warp the rest of the signal as they can exceed its Nyquist frequency. The frequency extraction method will then alias the errant frequency component down, skewing the rest of the signal's frequency responses.

Window functions can be used to reduce the effects of spectral leakage. These functions reduce the amplitude of the high frequency leakage components. This is done by multiplying the samples by a window the same length as the signal which smoothly varies between zero, a central positive value and then back to zero. Because the start and end of the window are zero, there will never be a sharp incline if there is a disparity between the start and end of a non-periodic signal.

There are a wide variety of window types which affect the frequency distribution of the signal in various ways including Hamming, Bartlett and rectangular (see Figure 2.3). The correct window type for an application is dependent on the predicted frequency characteristics of the windowed signal.

The structure of the window produces a specific frequency distribution for the window itself. Generally this distribution is made up of a large, low-frequency "lobe", followed by several smaller side lobes (see Figure 2.4). The size and shape of these lobes is what dictates the appropriate window for a signal type. For example, if there is likely to be interfering signals near the intended frequency of interest, a window with smaller

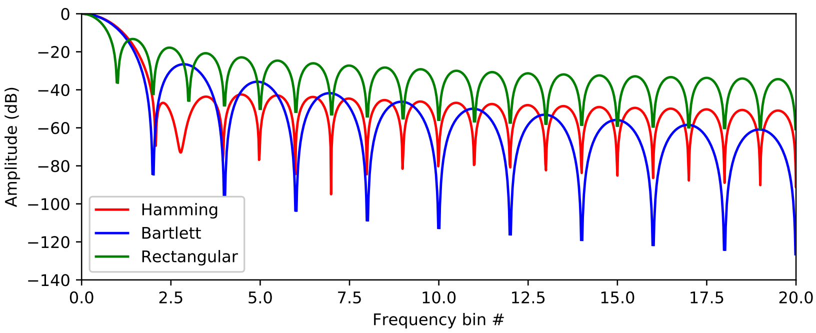


FIGURE 2.4: FFT plots of Hamming, Bartlett and rectangular windows. Each distribution features differently sized primary lobes and secondary lobes which affect the frequency distribution of the windowed signal.

secondary lobes is ideal. If the frequency spectrum of the signal is likely to be flat, a rectangular window is best. In general, both the Hamming and Hann windows work best for applications likely to face a variety of frequency distributions. Both are sinusoidal windows and effectively reduce spectral leakage.

If a window function is used, the window values can be pre-computed to save computations required at run-time, using the sample set length L and two constants $\alpha = 0.54$ and $\beta = 1 - \alpha = 0.46$ (for a Hamming window):

$$h_j = \alpha - \beta \cos\left(\frac{2\pi j}{L}\right) \tag{2.2}$$

2.4.1.3 Goertzel algorithm

As the need for acoustic analysis on low-power hardware increases, more methods for reducing the resources required for common DSP techniques are developed and employed. In the case of spectral analysis, the Goertzel algorithm can be used in place of a DFT. Taking the form of a DSP filter, the Goertzel algorithm was first described by Goertzel [1958]. While a DFT will compute the entire frequency band from 0 Hz to the Nyquist frequency, the Goertzel algorithm can efficiently evaluate a subset of terms from a full DFT. The Nyquist frequency is half the sample rate of the original signal and is the highest frequency which a sampled signal is able to represent.

The filtered signal represents the presence of a chosen band of frequencies. When implemented, the central frequency and bandwidth can be set by changing various precomputed constants. Using one or more Goertzel filters, spectral analysis can be focused on specific frequency bands, reducing the amount of unnecessary calculations done to obtain the amplitude of irrelevant frequencies.

Given a set of samples, the Goertzel filter starts by dividing them into N windows of length L, denoted by $(s_{1,1}, \ldots, s_{1,j}, \ldots, s_{1,L}, \ldots, s_{N,1}, \ldots, s_{N,j}, \ldots, s_{N,L})$ where

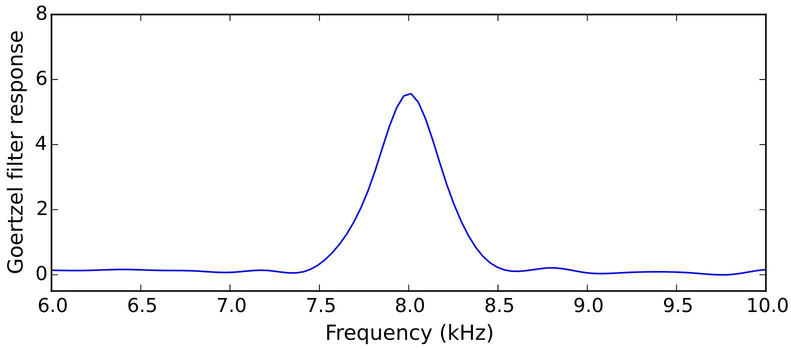


FIGURE 2.5: A single Goertzel filter employed to extract the 8 kHz component of a tone centred around 8 kHz.

$1 \leq i \leq N$ and $1 \leq j \leq L$. The filter length L is set based on the desired bandwidth B and the sample rate of the audio f_s:

$$L = 4\frac{f_s}{B} \tag{2.3}$$

Once the sample set is windowed using a process such as the Hamming window described in Section 2.4.1.2, a Goertzel filter can be applied. First, a temporary sequence y is created using the constant c and Hamming windowed values $(h_0, \ldots, h_j, \ldots, h_L)$ for each of the N windows:

$$y_{i,j} = (h_j \cdot s_{i,j}) + (c \cdot y_{i,j-1}) - y_{i,j-2}. \tag{2.4}$$

The constant c is calculated using the target frequency f and the sample rate f_s. This step can also be done before run-time and hard-coded, saving further computation:

$$c = 2\cos\left(\frac{2\pi f}{f_s}\right) \tag{2.5}$$

Finally, the Goertzel filter uses the temporary sequence y to produce a magnitude m for each window. Each m_i represents the presence of the frequency band of width B around central frequency f within in window i:

$$m_i = y_{i,L}^2 + y_{i,L-1}^2 - c \cdot y_{i,L} \cdot y_{i,L-1} \tag{2.6}$$

The Goertzel algorithm operates with a computational complexity of $\mathcal{O}(L)$, increasing linearly with the total number of samples per window. This is an improvement on both a standard DFT's $\mathcal{O}(L^2)$ and the FFT's $\mathcal{O}(L \log L)$, requiring fewer calculations due to the more focused nature of the filter.

2.4.1.4 Mel-frequency analysis

A technique often used in speech recognition is the mel scale. Based on the non-linear human perception of frequencies, the mel scale uses a series of windows which differ in width depending on the frequency. Lower frequencies are more detailed and receive smaller windows, increasing non-linearly in size as the frequency increases [Tiwari, 2010]. The reason for mimicking human perception and weighting against higher frequencies is the idea that more relevant information is found in the hearing range of humans. In the case of speech recognition, this means human speech is limited to frequencies other humans are likely to hear.

The mel scale is also relevant in any context where the majority of desired acoustic information is found at lower frequencies, making it useful to be more discriminative at these frequencies. Examples of these contexts include large mammals such as whales and elephants, which produce calls able to travel long distances thanks to their low fundamental frequency [Garstang, 2004].

One possible formula of converting a given frequency f from Hz to mels is:

$$m = 2595 \log_{10}(1 + \frac{f}{700}) \tag{2.7}$$

One use of the mel scale is to produce mel-frequency cepstrum coefficients (MFCCs), values which represent the short-term power of a sound. Calculating the MFCCs of a signal is a common pre-processing technique for many forms of acoustic analysis, especially those utilising machine learning. Building on the assumptions of short-term Fourier transforms, MFCCs assume that audio signals vary very little in the short-term and so can be framed together. This simplifies models identifying features in those signals. Each frame produced by the MFCC process accurately represents the short-term energy envelope of the signal, providing a representation of the power spectrum of the signal.

MFCCs are calculated by first dividing the samples into overlapping frames then extracting the frequency components of each frame using a DFT. A series of triangular filters, spaced non-linearly in the mel scale, are then applied to the transformed frames. Finally, a discrete cosine transformation (DCT) is used to break down the frames into the sum of a series of cosine functions. The DCT is a linear function which decorrelates the filter banks (also known as whitening) [Sahidullah and Saha, 2012].

Calculating full MFCCs is not necessary for all applications and the filter bank coefficients can be used as is and for many applications. The DCT was previously done when filter banks were primarily used to pre-process inputs for machine learning tools such as hidden Markov models (HMMs), which are highly susceptible to correlation between frames. However this step has become less necessary as applications have begun to favour various forms of deep neural network over HMMs. These neural networks are less susceptible to the effects of correlated inputs and thus a DCT is often not worth the extra computation required.

2.4.1.5 Time domain signal coding

Techniques such as DFTs and Goertzel filters attempt to use the frequency domain despite the computational intensity of calculating it. However, many analysis tools exist which remain in the time domain in order to minimise complexity. One such technique is time domain signal coding (TDSC) [Chesmore, 2001], based on a digital compression technique for human speech called time-encoded speech (TES) [King and Gosling, 1978]. TDSC builds on TES and was designed for acoustic analysis on low-cost, microcontroller-based systems.

To characterise an acoustic signal, TDSC divides it into a series of epochs taken between zero crossings. These epochs are then defined by their length and shape. The shape of each epoch is described by the number of local minima and maxima present between each zero crossing. These two time domain features are collected into D-S pairs, which are then mapped onto a reduced symbol set and used to form an alphabet for a discrete signal. This alphabet can then be used to define specific acoustic events such as animal vocalisations as a species' calls will contain a similar alphabet distribution.

This technique for defining acoustic signals purely within the time domain has been used by Chesmore [2001] to successfully train artificial neural networks for insect call classification. Thanks to the low computational complexity, TDSC could be deployed on microcontroller-based, handheld devices for species detection in the field.

2.4.1.6 Wavelet transformation

Contrasting DSP techniques which operate in the time or frequency domains, wavelet transformation operates in the wavelet domain. Wavelets are small waves which, instead of recurring endlessly like sine or cosine, are finite and spatially localised (they start and end at zero). Wavelet theory was originally developed to model seismic activity [Grossmann and Morlet, 1984] but has since entered the field of audio engineering. While DFT decomposes a signal into infinitely long frequency signals, a wavelet transformation decomposes a signal into a series of wavelets which are limited in both time and frequency. wavelet transformations can be either discrete or continuous. A continuous wavelet transformation decomposes a signal into various scaled and shifted versions of a "mother wavelet". Discrete wavelet transformations also decompose signals into wavelets, however they use a discrete set of scales to do so.

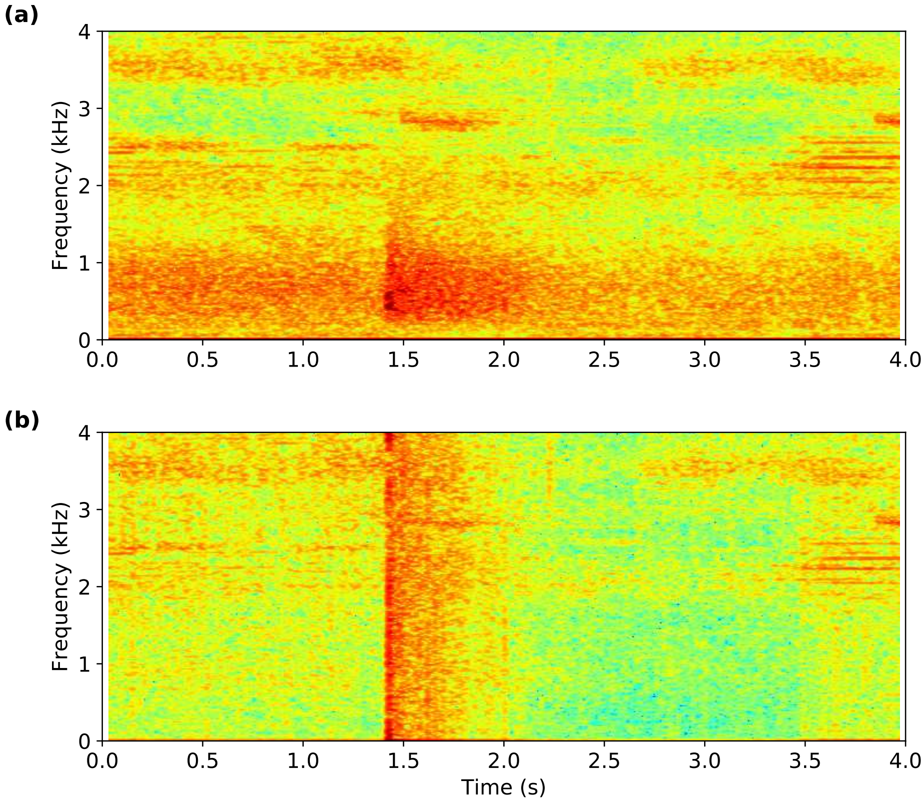


Figure 2.6: Two spectrograms of the same gunshot recording, collected 353 m from the source. These plots show (a) a standard spectrogram and (b) a spectrogram of the samples pre-processed using the Teager energy operator, an operator designed to enhance sudden impulses in signals.

Wavelet transformations have been used extensively in acoustic analysis in solid mediums [Baydar and Ball, 2003] due to the origins of wavelet theory. However acoustic detection in air has also found success with discrete wavelet transformations, using them to detect acoustic events such as passing cars [Averbuch et al., 2009].

2.4.1.7 Other DSP techniques

Picking out acoustic events from a constant stream of recordings from a busy acoustic environment is a computationally intensive task. A wide variety of operators and filters exist which remove unwanted components of an audio signal such as noise or irrelevant frequency bands.

One example of such an operation is the Teager energy operator (TEO). It can be used to enhance the sudden peaking of high frequency components and reduce noise in a recording (see Figure 2.6) and is often used to pre-process signals when attempting to detect sudden impulses [Chacon-Rodriguez et al., 2011]. Other noise-reduction techniques include median filtering, which aims to remove noise from digital signals. It does so by windowing the samples, taking the median value of each set of samples.

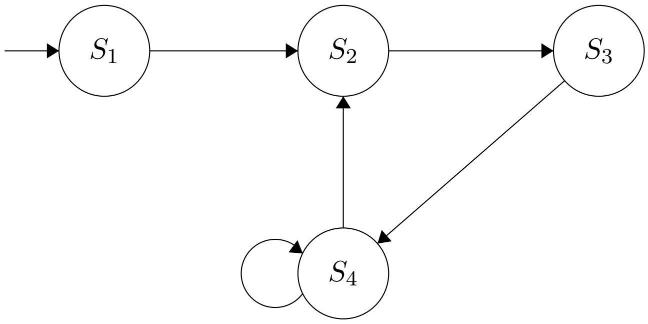


Figure 2.7: An example of a simple finite state machine consisting of four states. A set of observations can be used to predict the combination of states which occurred to produce that collection of outputs.

2.4.2 Hidden Markov models

Once audio signals have been processed, an acoustic detector requires some ability to decide whether the processed inputs contain evidence of the target sound. This can be through a simple classifier or more complex machine learning techniques. These machine learning techniques also vary in complexity and sometimes require specialised pre-processing to function, as mentioned in Section 2.4.1.4.

In order to build an acoustic detector which uses machine learning, a collection of sample recordings is required to train the model. This is where the history of PAM-based monitoring and the increase in open-science support is useful, as datasets for many environments and species already exist.

Hidden Markov models (HMMs) are a machine learning technique which uses inputs as a set of observations, matching the observations to a sequence of states which the observations have the greatest probability of corresponding to. The states can vary from containing a target sound or known elements of a sound's composition (such as the distinctive start and end components of a bird call). State diagrams such as Figure 2.7 are often used to represent the flow of a model between its various states.

Elements required to build a HMM include the probability of transitioning between each state $A_{i,j}$, the probabilities of possible observations in each state $B_{i,j}$ and the probability of a sequence starting in each of the states Π_i. Previously collected recordings can be used to obtain these probabilities, taking these observations, along with the known states associated with them and using them to approximate the distributions.

For a set of N hidden states Q, where $Q = (q_0, \ldots, q_n, \ldots, q_N)$, the initial state probabilities are defined as the probability that the state at time $t = 0$ is i:

$$\Pi_i = P(q_0 = i) \tag{2.8}$$

While the transition probability $A_{i,j}$ is defined as the probability that the current state at time t is j, given that the preceding state was i:

$$A_{i,j} = P(q_t = j \mid q_{t-1} = i) \tag{2.9}$$

Finally, the emission probability $B_{i,k}$ is defined as the probability that, given the current state is q_i, the current observation is o_k:

$$B_{i,k} = P(o_k \mid q_i) \tag{2.10}$$

When classifying audio, emission probabilities are handled either by probability distributions or a series of bins with a probability assigned to each.

Once a HMM has been constructed, the Viterbi algorithm can be used to produce the most likely path of states which a set of observations represents Forney [1973]. The Viterbi algorithm operates on the basis that the optimal path at each step can be deduced from the optimal path at the step before it. This method takes the probability of being in state i previously, of transitioning from that state to the current one $A_{i,j}$ and of obtaining a given observation k at this new state $B_{j,k}$. The algorithm maximises these combined probabilities across all prior states, stepping through the set of observations.

In acoustics, HMMs have been used so extensively in speech recognition that almost all large vocabulary continuous speech recognition (LVCSR) systems use HMMs as an underlying framework [Gales and Young, 2008]. For speech recognition, each state in Q represents a phoneme which can be used to build all other words within the desired language. There are 44 phonemes required to construct the English language, so a HMM for English speech recognition must include 44 states. MFCCs are often used as observations for these models for reasons discussed in Section 2.4.1.4.

Researchers looking to implement acoustic detection in various fields have used the substantial work into speech recognition HMMs as a basis to detect a variety of other acoustic events. These include using a HMM with MFCC features to detect man-made sounds such as coughs in order to diagnose respiratory disease [Matos et al., 2006]. As coughs are also produced in a manner mechanically similar to speech, the mel scale works well for classification. Outside of classifying sounds produced by humans, HMMs have also been applied to biophony, identifying bird calls in large, passively collected datasets [Brandes, 2008].

One method in which detectors have built upon the complexity of basic speech recognition HMMs is by implementing a hierarchical structure of multiple models. In the case of bird detection, an initial HMM can be used to classify recordings as containing birdsong or background noise, followed by a second classification of the birdsong recordings which identifies the species of bird. These hierarchical HMMs (HHMMs) can discern between extremely similar acoustic events such as the sound of a door slamming and the sound of human collapsing [Peng et al., 2009]. As the secondary HMMs are only employed when they're likely to be relevant, these models can run extremely quickly.

While voices and accents can vary, spoken words are generally uniform in order to remain intelligible. For applications such as this and others where events occur in definite, typical orders (such as phonemes in a word), HMMs are ideal. This is not the case for many animal vocalisations and so other decision-making techniques are employed.

2.4.3 Deep neural networks

Deep learning techniques have become used in a wide variety of applications thanks to their flexibility. Previously-collected acoustic datasets has made deep learning especially popular within the field of acoustic detection. With the push towards open science, more of this data is becoming publicly available, allowing researchers to train deep learning models on large datasets which would usually sit unused after analysis for their original research application.

Deep learning is a broad collection of tools which encompasses many model types including various neural network variants, such as standard deep neural networks, recurrent neural networks, and convolutional neural networks. Deep neural networks (DNNs) have multiple layers between their input and output layers which allow them to model complex features from a combination of smaller features. Recurrent neural networks (RNNs) allow data to flow in either direction between these layers and is best used for sequence prediction. Convolutional neural networks (CNNs) work best with image data, mapping two-dimensional input data onto an output. These neural networks have all been applied to the task of acoustic detection, with deep [Hinton et al., 2012], recurrent [Hughes and Mierle, 2013], and convolutional [Abdel-Hamid et al., 2014] neural networks each finding use in speech detection. As the techniques have developed, derivatives of these networks which mix functionality, such as convolutional recurrent neural networks (CRNNs) [Choi et al., 2017], have been used.

The introduction of these deep learning techniques has had a significant effect on wildlife identification and classification. The BirdCLEF bird call identification challenge is an annual contest where entrants compete to develop the best bird identification technique for a given dataset of recordings. Entries are compared using the mean average precision (MAP) of their classifications. In 2014, BirdCLEF's winning entry achieved a MAP score of 0.45 [Goëau et al., 2014]. The following year, the winning entrant was a CNN which achieved a MAP score of 0.69 Joly et al. [2016], a large improvement on previous entries.

CNNs have become increasingly popular for the detection of animal vocalisations. Spectral analysis can produce 2D images which allow acoustic detectors to benefit from developments in computer vision techniques such as CNNs [Goëau et al., 2016]. Variations in amplitude and time are represented as spatial differences in spectral imagery

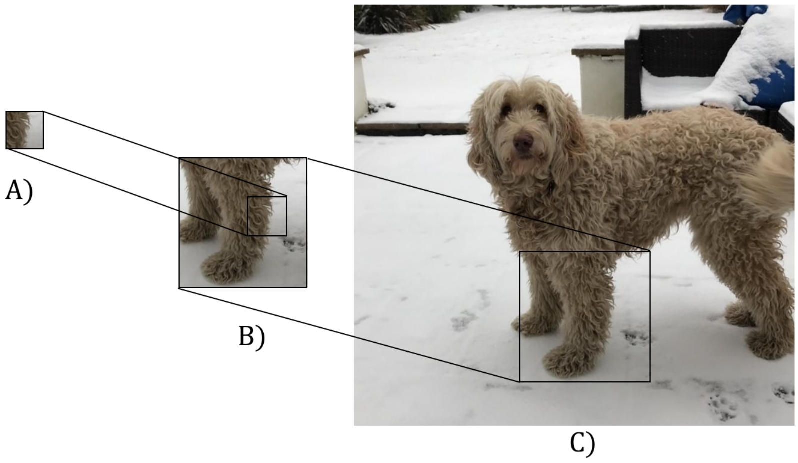


FIGURE 2.8: Each subsequent layer of a CNN uses filters which define a larger area of the target image. Abstract features such as the one shown in (A) come together to define parts of the dog such as its leg (B), which then make up the full definition (C). A CNN is considered a black box and these features are chosen and trained automatically.

such as spectrograms. Humans are easily able to identify these features through visual inspection, so researchers have developed detection techniques which mimic the way the human brain performs this image detection.

CNNs work by breaking a target image into a set of features, these features can appear abstract when viewed individually but together form a full definition of the target. As these features can appear anywhere within the image, a filter is made using the feature which then convolves with the test image as a whole (see Figure 2.8). For audio classification this could be the spectral representation of a target animal vocalisation or background noise. Given enough labelled training data, CNNs are able to take advantage of data-locality to identify patterns in audio data [Piczak, 2015]. In order to obtain a sufficient quantity of labelled data to train a classifier, Mac Aodha et al. [2018] have taken bat recordings from a large number of sources and used citizen scientists to label them, adding to an ever-growing training set.

Examples of successful CNNs for acoustic detection in ecology primarily come as a result of an automatic wildlife detection challenge which pits researchers against each other to develop the best algorithm for classifying a collection of bird call and background noise recordings [Stowell et al., 2016]. Since its inception, the majority of successful entrants have utilised some form of CNN for classification. These include the entry developed by Kong et al. [2017] which runs mel-scaled spectrograms through a standard CNN, a CRNN entry by Cakir et al. [2017] which combines layers of both neural network classes, and an entry by Pellegrini [2017] which makes use of a "DenseNet". A DenseNet is defined as a CNN with short connections between layers close to the inputs and outputs [Huang et al., 2017a]. Instead of representing features using deep or wide networks, DenseNets develop by deliberately reusing features.

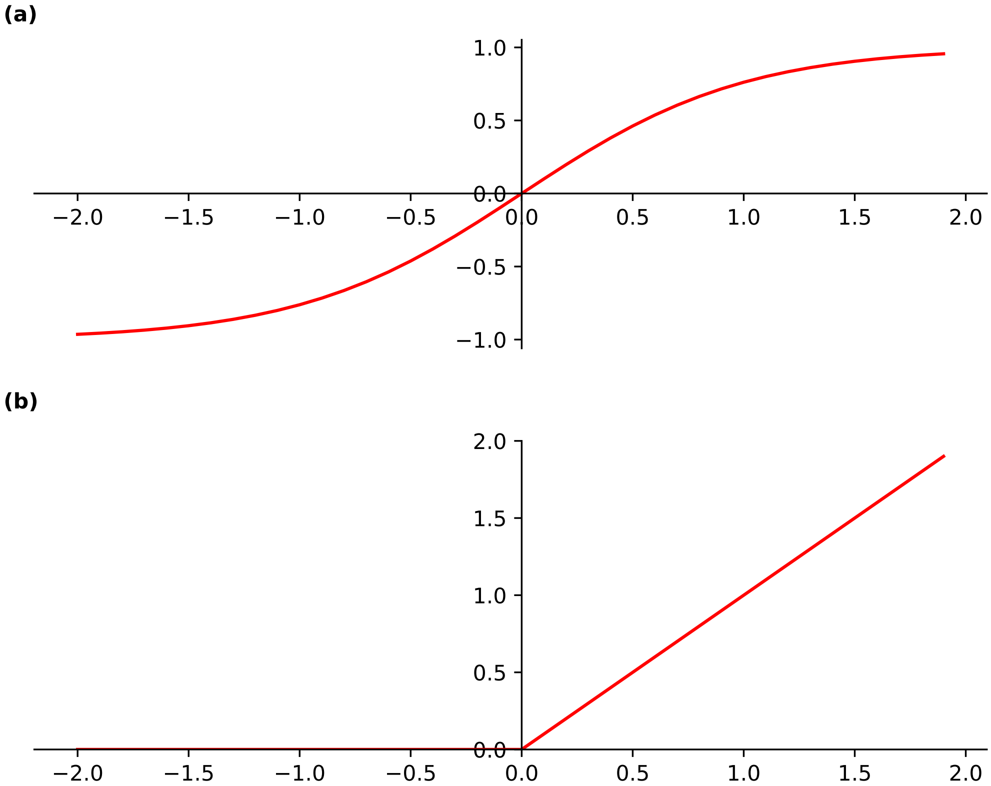


Figure 2.9: Two plots showing different activation functions: (a) the tanh function which possesses a gradient of almost zero before -2 and after 2; (b) the ReLU function which sets values less than zero to zero.

2.4.3.1 The vanishing gradient problem

The vanishing gradient problem is an issue common to training various forms of neural network with gradient-based learning [Ide and Kurita, 2017]. In these learning methods, weights are altered based on the gradient of the error function. If this gradient is extremely small then the weight changes will also be small. On lower layers of a NN, this gradient becomes exponentially small, resulting in models which are slow to train.

This problem is common to NNs which utilise certain activation functions. Activation layers within a network are often used to introduce non-linearity and currently either tanh, sigmoid or rectified linear unit (ReLU) activation function is applied as part of these layers. Both the sigmoid and tanh are S-shaped activation functions, possessing gradients of almost zero at their extremes (see Figure 2.9a). These sections of the function produce the vanishing gradient problem as the activation units within the layer saturate. The ReLU activation function returns any positive value given to it, but sets any negative activations to zero. This results in an L-shaped activation function which can never saturate as the gradient is either one or zero. Most situations a network faces will result in a gradient of one (see Figure 2.9b). However, because the gradient can be zero it is possible for ReLUs to "die", always outputting zero when a large negative bias for its weights is learned. While this is a risk, ReLUs are still used to avoid the vanishing gradient problem.

2.4.3.2 Neural networks on low-powered hardware

The NNs discussed thus far have been run on an existing data set, using the resources of a high-powered desktop machine to classify unseen data. Attempting to perform this classification in real-time provides another set of challenges. A common application of NNs for real-time classification is to trigger more complex speech recognition algorithms. An example of this is on home assistant devices such as Google Home or Amazon Alexa. Devices such as Alexa use elaborate voice recognition algorithms for parsing complex user requests. This is done by transmitting audio to Amazon's servers and performing cloud-based analysis, returning the response.

In order to both mitigate the privacy concerns and bandwidth usage involved in constantly streaming audio to these servers, these devices use on-board keyword spotting (KWS) to identify key phrases such as "Alexa" or "Okay Google" which then trigger more elaborate analysis. Lightweight algorithms, including specialised NN architectures, are often used to pick out key phrases from constant streams of audio. As they must listen constantly, the KWS networks must be extremely energy efficient and so are often run on microcontrollers which limit their complexity.

RNNs have been tested for KWS applications [Fernández et al., 2007], however they often suffer from detection latency, resulting in a poor user experience. As a result, the most common architectures applied to microcontroller KWS are CNNs, implemented such that computation is minimised. These implementations include the CNN variant developed by Fernández-Marqués et al. [2018], which learns to assign weights to binary patterns. These binary patterns are used to represent the network's convolutional filters, limiting the filters' complexity.

Another example is the CRNN developed by Zhang et al. [2017] which combines the functionality of convolutional and recurrent neural networks to achieve high levels of accuracy, while utilising depthwise separable convolutions to reduce both network size and necessary computation. Depthwise separable convolutions are used as an alternative to standard 3D convolutions. For a 2D image (such as an audio spectrogram), each RGB colour channel is convolved separately. Then they are recombined using a pointwise convolution. This method is frequently used to improve the efficiency of image classifiers [Chollet, 2017, Howard et al., 2017] as it requires fewer operations to compute each convolution [Huang et al., 2017b].

2.5 Targeted acoustic events

The acoustic monitoring tasks carried out over the course of this thesis include two instances of biophony detection and one instance of anthrophony detection. Each application was carried out in a distinct environment with unique challenges affecting the detection task. This section provides relevant background information on the target species and/or deployment locations.

2.5.1 Cuban greater funnel-eared bat

The Cuban greater funnel-eared bat (*Natalus primus*) is a species of bat endemic to the Guanahacabibes peninsula in Cuba. Compared to other bat species it is a slow flyer and as such subsists on a diet consisting entirely of insects such as moths. It produces a 60 kHz echolocation call [The Rufford Foundation, 2017], well outside the range of human hearing. Little else is known about the acoustic behaviour of the species due to minimal conservation attention up to this point.

Natalus primus is considered an EDGE species (Evolutionarily Distinct and Globally Endangered). EDGE species are defined by a metric know as their EDGE score, which takes into account the amount of unique evolutionary history (in millions of years) it represents as well as its conservation status. As of 2019, the Cuban greater funnel-eared bat is listed as vulnerable on the IUCN Red List of threatened species [IUCN, 2019], while its evolutionary distinctiveness score is 14.16, resulting in an overall EDGE score of 4.10. For context, the black rhinoceros possesses a score of 6.13 and as such receives significant conservation attention and funding.

Thought to be extinct until 1992, a colony was discovered in a cave known as Cueva La Barca. As bats of its genus are known to congregate in large, conspicuous colonies and all other large caves on the peninsula have been thoroughly surveyed, this colony is believed to be the only one remaining in Cuba. The bats roam the forest surrounding this cave, hunting soft-bodied insects such as moths and crickets. The cave and forest are both enclosed by Guanahacabibes national park, however as both are close to the boundary of the park there are concerns about whether or not its habitat is truly protected.

2.5.2 New Forest cicada

The New Forest cicada (*Cicadetta montana*) is a species of insect thought to be locally extinct in the UK, previously found in various locations throughout the New Forest in Hampshire, England. Previously England's only native species of cicada, its last confirmed sighting was over 20 years ago [Pinchen and Ward, 2002] and as such it is considered a priority species by the Joint Nature Conservation Committee [2007]. First

FIGURE 2.10: The New Forest cicada (*Cicadetta montana*). A species of insect thought to be locally extinct in the UK. Photo taken in Slovenia.

described by Scopoli [1772] in Slovenia (see Figure 2.10), it is still found there as well as other locations throughout Europe.

As with other species of cicada, they spend most of their early lives underground as nymphs, coming to the surface every 7 years to shed their nymph exoskeleton and mate. Due to its life cycle containing many years of inactivity underground, they have undergone long periods without detection in the past. They were believed to be locally extinct in the UK, until their rediscovery in 1962 [UK Forestry Commission, 2004].

As New Forest cicada are also found elsewhere in Europe, it has been possible to study their calls despite their possibly extinct status in the UK. Thanks to studies of them in Slovenia it is known that the males of the species sing using an extended buzz at 14kHz which lasts around 30 seconds [Zilli et al., 2014]. Other behaviours which have been documented outside of the UK include the construction of "turrets". These small cones of earth are built around the tunnels dug by nymphs as they return to the surface at maturity. The call and turret construction have both been targeted by entomologists manually surveying the New Forest for the species in the past.

Efforts by entomologists and conservationists to find the species with the New Forest have historically been carried out using manual surveys. These surveys are focused on locations which either hosted sightings in the past or match documented habitat requirements of the species. These requirements include the presence of certain plants and clearings which experience sufficient direct sunlight.

2.5.3 Poaching in the Tapir Mountain reserve

Tapir Mountain is one of three protected nature reserves in Belize, Central America. It covers approximately 25.4 km^2 and was designated a protected scientific reserve to

FIGURE 2.11: A small poacher's camp, evidence of poaching activity found in the protected Tapir Mountain reserve.

preserve the contained ecosystem in an undisturbed state of wilderness. It is co-managed by the Belize Forest Department and the Belize Audubon Society.

The reserve is home to wide variety of species and access is restricted to scientific and educational purposes only by the management. Despite this, evidence of anthropogenic intrusions is found throughout Tapir Mountain, in the form of discarded shotgun shells and poaching camps which are regularly discovered by researchers in the area (see Figure 2.11). As well as visible evidence, gunshots are regularly heard from the reserve at night, audible in neighbouring reserves.

The majority of illegal hunting is focused on species such as peccary (a pig-like, hoofed mammal) and paca (a large rodent). The poaching of these species is generally done for food and is carried out after sunset. Because of this, it is extremely difficult to protect the reserve from these illegal intrusions. While the reserve is officially protected and managed, resources currently attributed to it are insufficient to manually monitor and police the entire area.

2.6 Summary

This chapter has presented the various methods in which ecosystems are monitored, focusing on the application of acoustics. It has also reviewed the various methods in which conservation researchers implement acoustic monitoring, including commercial acoustic recorders, bespoke recorders based around modular computers such as the Raspberry Pi and custom low-power hardware.

Acoustic monitoring, while extremely useful for filling niches left by other commonly-used forms of ecosystem monitoring such as camera traps and satellite monitoring, is often priced such that large-scale, long-term deployments are unfeasible. As a result, there is a need for low-cost alternatives which can enable conservationists to implement acoustic monitoring within the limited budgets often assigned to conservation projects.

The following chapters present work done to fulfil this need, using AudioMoth as a low-cost, low-power platform for acoustic detection. The AudioMoth itself is described in Chapter 3. By incorporating detection algorithms into an acoustic monitoring tool, the energy and storage requirements which have previously limited bespoke monitoring solutions can be drastically reduced. Chapter 4 describes the development of two such acoustic detection algorithms in the context of detecting first the Cuban greater funnel-eared bat and then the New Forest cicada. Chapter 5 then covers the development of an algorithm which builds upon those algorithms to produce a long-term gunshot monitoring system able to record poaching events within the Tapir Mountain scientific reserve. This algorithm utilises a hidden Markov model which has been benchmarked against a CNN. Finally, Chapter 6 focuses on the accessibility of acoustic monitoring technology, considering the financial limitations as well as the usability of tools developed for conservation.

Chapter 3

AudioMoth: A low-cost, open-source acoustic sensor

> Nature hides her secret because of her essential loftiness, but not by means of ruse.
>
> Albert Einstein

Developed as part of the work covered by this thesis, AudioMoth was developed in order to fill the niche of an open, flexible acoustic monitoring platform, it is a low-cost, low-energy device designed to be applicable to a variety of applications within conservation. It was developed by a team of three: Alex Rogers, Andrew Hill, and Peter Prince (the author). At events and as part of this team's online presence, the name *Open Acoustic Devices* has been used.

All firmware, supporting software and hardware schematics are open-source and available online with non-restrictive licenses[1] which allow them to be altered and extended freely. This allows users to manufacture devices themselves as well as tailor the design to their specific application if required.

Initially named "Soundtrap" during its prototype phase, AudioMoth is the first open-source device of its type to widely adopted within the conservation community. It has been deployed as part of a wide variety of projects by conservation groups, researchers, and NGOs worldwide. This chapter describes the design of its hardware, on-board firmware and supporting software.

[1] https://www.openacousticdevices.info/license

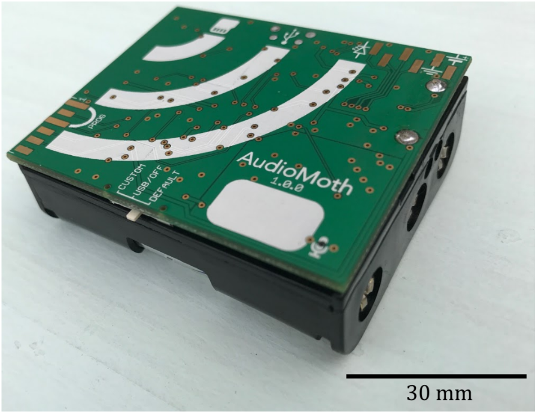


FIGURE 3.1: AudioMoth, a low-cost, low-power acoustic monitoring device designed to provide an accessible platform for acoustic monitoring in conservation.

3.1 Hardware design

The AudioMoth hardware consists of a single printed circuit board (PCB) measuring 58 x 48 x 15 mm and weighing 80 g. The PCB features a slot for a microSD card and supports cards up to 16 GB in capacity. It also features a three-setting, side-mounted switch which allows users to change between the device's three modes: "CUSTOM", "USB/OFF", and "DEFAULT". The behaviour of the device when in each of these modes is defined by the firmware (the behaviour of an AudioMoth with the default firmware in each mode is described in Section 3.2). The switch is indented in the side of the board to prevent it snapping off while devices are in transit.

In terms of power, AudioMoth uses three AA batteries which provide it with 3000 mAh when lithium AA batteries are used. Early prototypes instead used AAA batteries, meaning it possessed a more discreet form factor but reduced battery capacity (1200 mAh). This prototype was used in deployments by some early adopters to test the feasibility of the AudioMoth concept.

Sound is captured through a small drill-hole in the bottom corner of the PCB, denoted by a microphone symbol. It uses a bottom-ported microelectromechanical (MEMS) microphone capable of capturing both audible and ultrasonic audio at sample rates up to 384 kHz. Also in a corner of the PCB is two programming pins and four general purpose I/O pins (GPIO). The programming pins allow users to flash their device with alternate or updated firmware, whereas the GPIO pins enable them to attach and communicate with additional hardware. These pins have been used to successfully integrate LoRa wireless communication to produce a prototype device capable of reacting to acoustic events not by simply recording, but by communicating with a static receiver.

AudioMoth's design is built around an ARM Cortex-M4F microcontroller, which operates with a 48 MHz clock speed and possesses on-board floating-point signal-processing capabilities. In order to support the processing power of the M4F core, a 256 KB SRAM

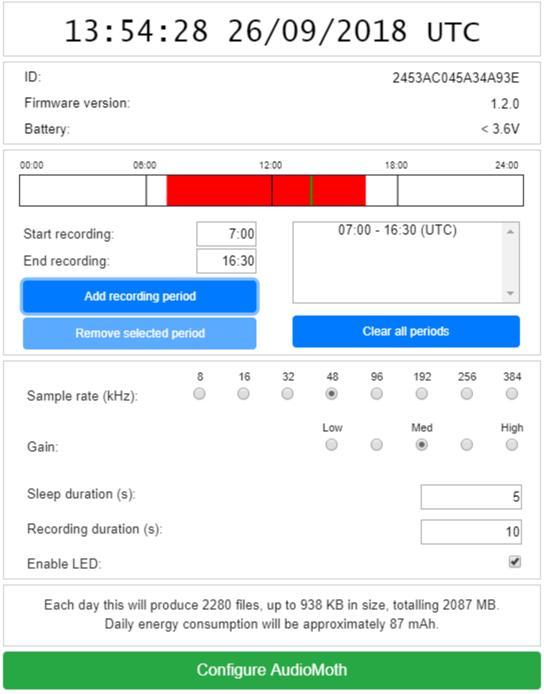


FIGURE 3.2: The AudioMoth Configuration App. An Electron-based application which provides information on an AudioMoth device when plugged in as well as enables the configuration of various settings including the recording schedule, the sample rate and gain control.

chip is used. This extra SRAM is accessible by the M4F processor even in low energy modes using direct memory access (DMA) and is used to temporarily store acoustic data for on-board analysis. By using DMA, the SRAM can be filled without CPU intervention, minimising the energy required to collect audio samples.

3.2 Software design

The firmware provided by default on purchased AudioMoth devices allows an AudioMoth to operate as a scheduled recorder, waking at regular intervals within a schedule to record audio to a provided microSD card as 16-bit uncompressed WAV files before returning to sleep. Each recording possesses a header containing information such as the sixteen-character unique ID of the AudioMoth which created it, the time and date of recording, and the time zone the device was configured to operate in. To keep track of time during operation to guarantee this header information is correct, the AudioMoth uses a high-precision real time clock (RTC) which is set when the device is configured.

Configuring the schedule as well as various settings such as the sample rate and behaviour of the device's LEDs is done using the AudioMoth Configuration App (see Figure 3.2). This application communicates with AudioMoth devices via USB or as an ultrasonic audio signal which can be played and processed by multiple AudioMoth devices simultaneously.

The basic firmware and configuration software are open-source and provide a foundation on which users can develop their own bespoke AudioMoth implementations. This can include interactions with additional hardware, alternative storage formats (zero-crossing, acoustic indices, etc.), and software-based acoustic detection which analyses samples in real-time and reacts in response to these algorithms. All firmware and supporting software has been made available under the open MIT licence[2]. This licence permits the user to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the code, provided the same licence is applied to any derivations of the original code. In order to aid in the development of AudioMoth derivations, all provided code is extensively documented in a publicly available wiki which details the functionality of the firmware as well as instructions for extending it. All firmware code is written in C, providing low-level access to the AudioMoth hardware, along with libraries which implement common AudioMoth functionality and DSP tools. The greater level of control afforded by C allows users to manually control functionality such as the energy modes of the AudioMoth's processor and DMA in aid of much greater energy efficiency.

3.3 Manufacture and purchasing

AudioMoth's affordability is achieved through the simplicity of its construction, consisting of a single printed circuit board (PCB). Without the need for an enclosure, fabrication is simplified and immediately deployable devices can be produced from a single PCB assembler. However, traditional PCB assembly presents a barrier for many users, involving a complex initial setup, and a large one-off-cost for the machinery preparation, which is used to automatically place components. This process is inefficient when ordering a single device, with the one-off cost spread over just one unit.

AudioMoth is accessible through purchasing methods that allow non-technical and financially restricted users to easily acquire devices for their projects. First, group purchasing is used to coordinate the initial device orders and allow users to combine their funds, enabling the one-off setup cost to be spread over a bulk order. This allows them to create an economy of scale, benefiting from orders much larger than would be possible as a lone customer [Wheat et al., 2013]. Using CircuitHub[3] - an online manufacturer specialising in simplifying the process of open-source hardware fabrication - the complex initial setup for PCB manufacture is further simplified. CircuitHub allows the group purchase organisers to order in bulk using an easy-to-use web form. In 2017 and 2018, in collaboration with the Arribada Initiative, these steps were combined into a single

[2] www.openacousticdevices.info/license
[3] www.circuithub.com

FIGURE 3.3: An AudioMoth deployed in a grip-sealed bad with a silica gel sachet to prevent the build-up of moisture, secured to a branch using a cable-tie.

process. The funds were collected by a service called GroupGets[4], which placed the bulk order and distributed the devices to the end-users. Each AudioMoth was sold to the consumer for $49.99 USD, representing a 95% saving on an equivalent single unit order. More than 6,800 AudioMoth devices have been purchased using this combination of group purchasing and online manufacturing since the open-source release in September 2017. This number increases regularly as group purchases continue to run.

Due to the open-source nature of AudioMoth, a number of devices have been purchased and made available by LabMaker[5], a company which specialises in making open science equipment available to researchers. This purchasing method costs more to the user ($79.99) as the benefits of large manufacturing runs are taken as profit by LabMaker. However in exchange the devices are constantly available to users.

3.4 Casing and protection

All of AudioMoth's components sit on one side of the PCB with their placement designed to protect them from knocks. This is achieved by facing the component side of the PCB towards the battery holder and using a bottom-ported microphone. The amount of protection required varies depending on the application. For shorter deployments of a few days or weeks, protection from water damage can be achieved by deploying AudioMoth in water-resistant, grip-sealed bags (see Figure 3.3). These bags are extremely low-cost and easy to deploy in any location with tape and a cable tie. Silica gel sachets can be deployed with each device to prevent condensation inside the bag.

[4] www.groupgets.com
[5] www.labmaker.org

Deploying battery-operated electronics for longer periods in harsh outdoor conditions requires ruggedised, waterproof enclosures to prevent weather damage. With AudioMoth's stature and simple construction it is easy to enclose the device in any waterproof case. To allow sound to pass through an enclosure a drill hole must be made to align with the microphone. Therefore, the only requirement of an enclosure is a flat surface large enough to lay the device against. To prevent water from entering the drill hole a hydrophobic ePTFE acoustic membrane can be placed over it. Finally, for cryptic monitoring the enclosure can be spray painted to camouflage it and protect it against theft. AudioMoth can also accept a wide range of power sources, from 3.5 V to 20 V; therefore, different power options can be used for any case option.

3.5 History

Over the course of this research, AudioMoth has developed from a prototype to a deployed device with several revisions. The timelines in Figures 3.4 and 3.5 show the major design iterations of the AudioMoth as well as the dates of the deployments discussed as part of this thesis.

Early prototypes of the AudioMoth were created under the name "Soundtrap". The first Soundtrap prototype was designed and built by Alex Rogers between January and July 2015. This iteration was built around an mbed LPC1768 microprocessor and was powered by three AAA batteries. The Soundtrap was then updated by Alex Rogers and Andrew Hill in early 2016, switching the mbed microcontroller EFM32 Gecko chip from Silicon Labs.

The mbed and Gecko Soundtrap models were deployed alongside each other in the summer of 2016 as part of the first New Forest cicada detection deployment. All devices were equipped with the cicada call detection algorithm developed by the author and described in Section 4.2. As the mbed is unable to perform floating point arithmetic, the algorithm was implemented on the device using fixed-point arithmetic (see Section 4.2.3). This was not an issue for the Gecko, which possesses a hardware floating point unit (FPU) and so a floating point implementation was used.

The Gecko Soundtrap was used by the Zoological Society of London for the monitoring project described in Section 4.1 in a national park in Cuba. In 2017 they were also used for the second annual New Forest deployment (without the mbed) and the gunshot tests carried out in Belize. These tests produced the dataset which was used to create the detection algorithm also created by the author and discussed in Section 5.1.

In July 2017 it was noticed that the name "Soundtrap" was already used for a widely-used music creation application. In order to prevent future confusion before the public release of the device, it was renamed to AudioMoth.

AudioMoth 1.0.0 had its open-source release in September 2017. The schematics for building this model were publicly released, along with the source code for its firmware. This version was used in 2018 for the third New Forest deployment. In Belize, AudioMoth 1.0.0 was used in a series of final gunshot detection tests (see Section 5.1.5.1) before being used in the yearlong Tapir Mountain monitoring deployment.

Alongside AudioMoth 1.0.0, the AudioMoth Configuration App was released. This Electron application was developed by the author and allowed users to configure the functionality of their devices without directly altering the firmware.

Several months after the open-source release of AudioMoth 1.0.0, a revision was released and made available for purchase. AudioMoth 1.0.1 featured minor design changes including rounded corners and re-positioned switch to prevent it snapping when the devices are in transit.

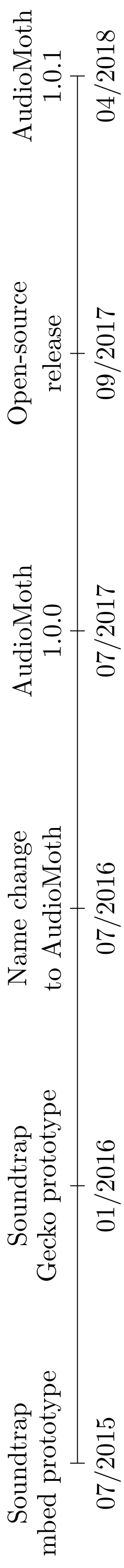


Figure 3.4: Timeline showing the completion dates of each iteration of the AudioMoth. From early prototypes under the "Soundtrap" name, to the updated public release in 2018.

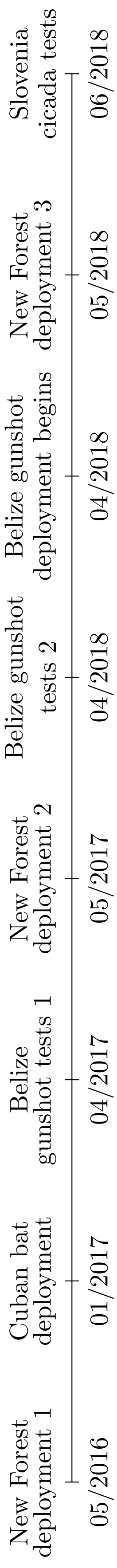


Figure 3.5: Timeline showing the various deployments carried out using the AudioMoth.

Chapter 4

Two algorithms for detecting the presence of bats and a locally-extinct insect species

> Life, uh, finds a way.
>
> Ian Malcolm

The most common application of acoustics within the field of conservation is the detection of animal vocalisations. Most commonly this is done after data collection in order to sort through large collections of recordings faster than is possible for a human to do so with manual inspection. Alternatively, during data collection, acoustic detection algorithms drastically reduce the amount of recordings collected, allowing deployments to run for much longer using the same quantity of storage and energy capacity. It is this second implementation which is discussed in this chapter.

For this purpose, two zoophony detection algorithms are proposed, both aiming to maximise the time for which monitoring devices can be deployed before researchers must return to the deployment sites to replace the storage medium or batteries. In the case of the first algorithm, designed for bat detection, this is motivated by the difficulty in returning to deployment sites and the large storage requirements of recording at high sample rates. The second algorithm, designed to detect the call of the New Forest cicada, was motivated by the need to minimise human intrusions into the species' habitat. As there are already concerns about the species being driven away from areas they were previously found in as a result of guests to the New Forest national park, repeated visits by researchers to isolated areas would only exacerbate the issue.

The goal of the algorithms is not just to accurately and reliably detect the presence of various animal using their vocalisations, but to do so using implementations which run

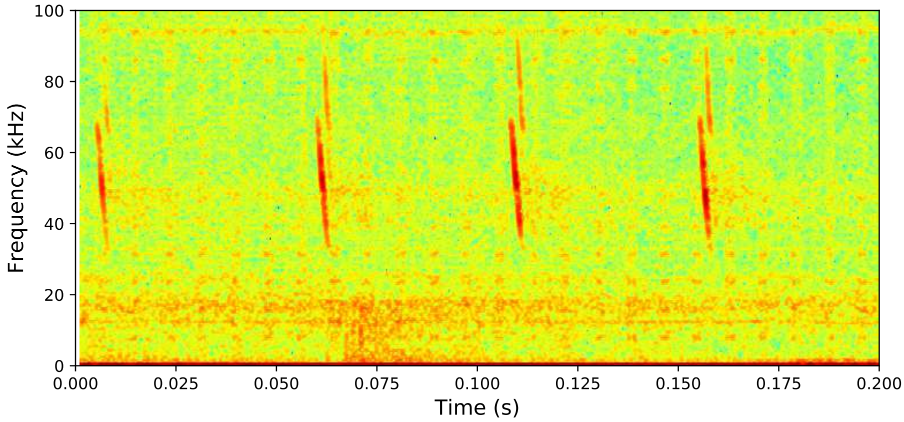


FIGURE 4.1: A spectrogram showing a recording of the echolocation call of a soprano pipistrelle bat. The call consists of a series of two-part pips centred around 50 kHz.

effectively on constrained hardware, such as AudioMoth. This introduces a number of limitations including the clock speed of the device, the amount of available memory (see Section 3.1 for AudioMoth specifications), as well as the availability and efficiency of pre-written libraries for carrying out common mathematical and DSP functions.

The two algorithms are described in terms of their context, the motivation behind the various aspects of their design, their performance both under lab conditions and in the field, as well as details about how they were deployed in the field using the AudioMoth platform. The design decisions of these algorithms then informed the more complex acoustic detection techniques used to monitor anthropogenic disturbances, described in detail in Chapter 5.

The deployment carried out for bat detection was done so by Oliver Wearn and his team at the Zoological Society of London. The New Forest cicada deployment has been repeated in 2016, 2017, and 2018 by the author, Andrew Hill, and Alex Rogers, with the initial deployment locations found with help from BugLife. The hardware and firmware development necessary to prepare AudioMoth for each deployment was done by Andrew Hill and Alex Rogers.

4.1 Bat detection algorithm

The first acoustic algorithm for zoological detection reacts to the presence of bats, recording in response to echolocation calls. These calls generally consist of a series of short pips that act as active sonar for the bat, allowing it to judge distances based on the time difference between a pip and its returning echo.

Echolocation calls vary in structure and frequency depending on various factors, including the acoustic environment and the hunting behaviour of the species. Despite possible

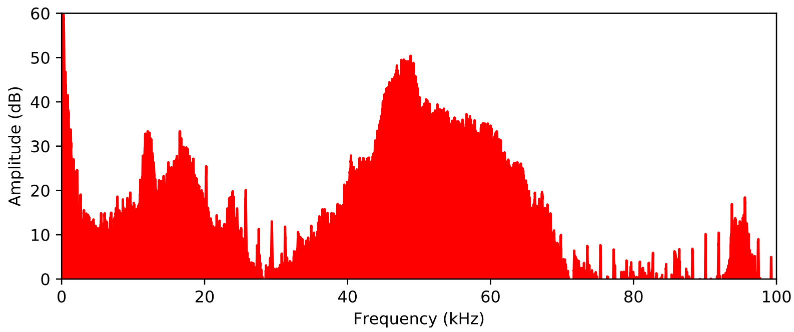


Figure 4.2: FFT plot showing the frequency distribution throughout a one-second recording of a soprano pipistrelle bat. There is a clear peak around the 50 kHz, indicating the echolocation pips are primarily around this frequency.

variation, the majority of echolocation calls are ultrasonic (22 kHz or higher). In natural environments, there are few sources of loud ultrasonic noise, meaning bat detection algorithms can easily remove the majority of noise by using a low-pass or band-pass filter. This is shown in Figure 4.1, where background noise in the recording remains in the audible range below 22 kHz. Because of these acoustic behaviours, a bat detection algorithm that detects short, ultrasonic pips can be extremely versatile and be adapted to a variety of species by centring the detection on different frequencies.

Several sets of bat recordings were collected from various locations in Southampton, UK, in order to develop and test a generalised bat detection algorithm. These recordings included various species including both common pipistrelles (*Pipistrellus pipistrellus*) and soprano pipistrelles (*Pipistrellus pygmaeus*). The soprano pipistrelle's calls were most prevalent within the collected dataset and are centred around 50 kHz (shown in Figure 4.2). The soprano pipistrelle was used as the test case for verifying the detection accuracy of the algorithm during development, before it was adapted for the 60 kHz call of the Cuban greater funnel-eared bat (*Natalus primus*) as part of a field deployment in the Guanahacabibes peninsula, Cuba.

4.1.1 Detection algorithm requirements

AudioMoth is set to record at a sample rate of 250 kHz in order to cover the ultrasonic frequency range used by the majority of bat calls while still producing files of reasonable size. The highest possible frequency detectable in files of this sample rate is 125 kHz (the Nyquist frequency, equal to half the sample rate). The target frequency can be set up to this value to target the majority of bat species.

With a sample rate of 250 kHz, a 10-second recording will require approximately 5 MB of storage. At this sample rate, a 16 GB microSD card will be filled after less than 9 hours of constant recording and will produce 3200 files. In order to improve the storage and energy efficiency of AudioMoth as a bat monitoring device, the detection algorithm had to significantly reduce the quantity of recordings produced by a constantly recording PAM, while maintaining a sufficiently high true positive rate (TPR). As bats are only likely to be in range of a static device for a short period of time, the device must also have a high chance of detecting a call during this brief window.

4.1.2 Sample collection

All algorithms discussed as part of this thesis operate similarly when implemented on AudioMoth hardware. For each iteration of AudioMoth's operating cycle, it collects samples using its MEMS microphone and stores them in the device's external SRAM. This SRAM buffer can be partitioned in various ways, set in the firmware. This enables us to use data structures such as circular buffers which can allow for simultaneous data collection and analysis.

When and how the samples are analysed varies for each algorithm. If an algorithm believes that the target sound is present in samples it has analysed, a recording is made by creating a new file on the microSD card, then a header is generated which contains information such as the time and date of creation, the unique ID of the device which created the recording and its battery level at the time. Samples are then transferred from SRAM into this file, closing it once a preset number of samples have been written. If the device is unable to write this many samples to the file, it is closed prematurely and the header is updated so the file remains readable and no data is lost.

For the bat detection implementation, the SRAM buffer holds a single buffer containing 512 ms of audio at 250 kHz (128,000 samples). The AudioMoth wakes every two seconds to collect these samples, analyse them, and then either record or return to sleep. Listening and recording at high sample rates is energy intensive. This duty cycle is to increase the expected battery life of devices running the algorithm. For deployments where closer to constant listening is required, the sleep period can be reduced in exchange for shorter battery life.

4.1.3 Feature extraction

A single Goertzel filter is used to extract a frequency band centred on the loudest frequency within the targeted species' echolocation call. For a Cuban greater funnel-eared bat, this frequency band was centred on 60 kHz with a bandwidth of approximately 4 kHz. This bandwidth was chosen to cover variations in the higher and lower frequency components within its call.

Using Equation 2.3 with a bandwidth of 4 kHz and sample rate of 250 kHz, the required filter length is 250 samples. This is rounded up to the nearest power to two to give a filter of length 256 ($L = 256$). All 128,000 samples are fed through this filter in windows

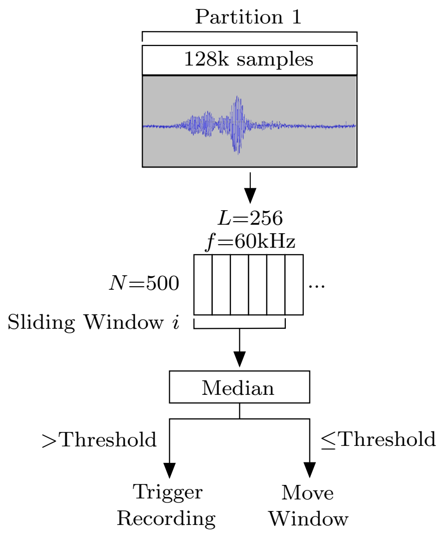


FIGURE 4.3: Data collection and analysis process for the bat detection algorithm. First, 128,000 samples are stored in a single partition in SRAM. Next, the partition is divided into 500 windows containing 256 samples each, which are fed through 60 kHz Goertzel filters. A sliding window then takes 5 filter outputs and compares their median to a threshold. If the median exceeds the threshold a recording is made, else the window slides onward.

of 256, producing 500 Goertzel outputs ($N = 500$). These 500 outputs represent the amplitude within this 60 kHz frequency band over the 512 ms of audio.

Equations 2.3 and 2.5 are used to customise this algorithm for different bat species. By varying the constant c and the length of the filter, the target frequency band can be moved and resized.

4.1.4 Detection stage

Bat echolocation calls vary with both species and current behaviour. When hunting, their calls will shorten to allow for more calling without overlaps with returning echoes. The echolocation calls of the Cuban greater funnel-eared bat while hunting are approximately 5 ms in length. At 250 kHz, one of these calls would be represented by 1250 samples. While multiple calls can occur within a partition, a single call takes up less than a hundredth of the collected samples. In order to make sure a single call isn't lost in the noise which comprises the remaining samples, a sliding window is used to analyse the amplitudes produced by the feature extraction stage (see Figure 4.3).

The sliding window steps through the responses, moving with a step size of half its length. This means that each amplitude is used twice and bat calls will never be split across two windows, becoming too short to detect. Even though each amplitude is being used multiple times, it does not need to be calculated multiple times. All Goertzel responses can be calculated and temporarily stored in memory while the sliding window

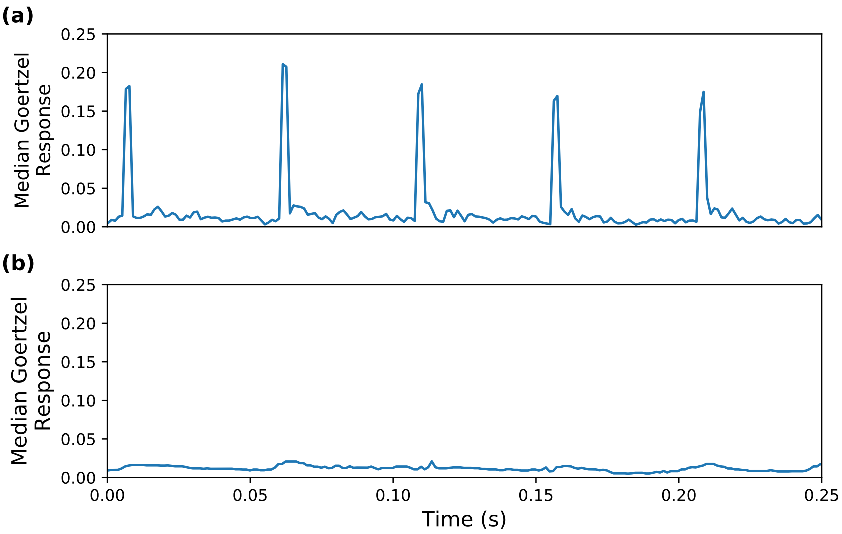


FIGURE 4.4: The median responses of two sliding window configurations for the bat detection algorithm: (a) one with a correctly configured window using 5 values per window and (b) one with too many (15 per window). With too many, bat calls cannot affect the median and are lost.

runs over them. The length of the window is tuned to the likely length of a call, meaning a device deployed in a hunting ground of the species must assume shorter calls.

The window must be no longer than twice the length of a call and no shorter than a single call. If it is too short, then redundant calculations are done and the resulting identifier is noisier. If it is longer, then even if a call is fully in a window, more than half the amplitudes consist of background noise (see Figure 4.4). This becomes an issue in the next stage of the algorithm.

For the Cuban greater funnel-eared bat detector, the sliding window takes five amplitudes ($5 \times 256 = 1280$ samples) and calculates the median response. In order for a bat call to affect the median response, more than half the amplitudes in a window must represent the call. A threshold is applied to the median that determines whether or not a bat is present in the window.

4.1.4.1 Quickselect

In order to efficiently obtain the median, the quickselect algorithm is used to find the central element. This algorithm operates similarly to the quicksort algorithm, choosing pivot elements to partition the data around. Unlike quicksort, quickselect only recurses on the partition containing the target element. This gives it a computational complexity of $\mathcal{O}(S)$, where S is the sliding window length. Quickselect also is an "in-place algorithm", meaning it requires no additional data structures. This is useful when performing a partial sort on a device with limited available memory.

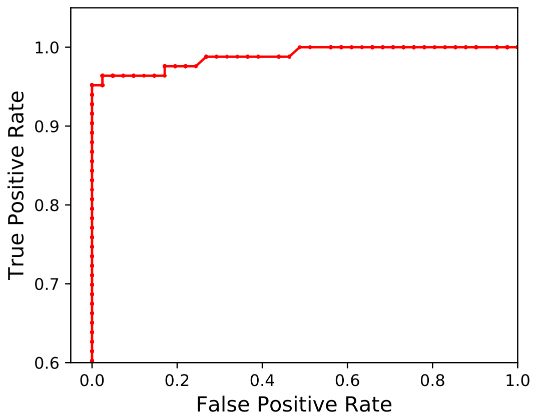


FIGURE 4.5: The bat detection algorithm's receiver operating characteristic curve, plotting the true positive and false positive rates of a range of threshold values used by the algorithm.

4.1.5 Results

The results of this algorithm include both the detection performance and its effectiveness when implemented on AudioMoth hardware, as part of a real-world deployment. The real-world application shows the effectiveness of the bat detection algorithm in enabling the monitoring a specific species of bat. The use of a detection algorithm and low-power hardware meant researchers could cover a large area for multiple nights in a row without temporarily deploying expensive equipment then returning regularly to retrieve it.

4.1.5.1 Performance

The accuracy of the algorithm was assessed using a previously collected set of UK bat species recordings, targeting the soprano pipistrelle. The soprano pipistrelle is a small species of bat which produces echolocation calls around 50 kHz. The dataset used contained 238 recordings, each ten seconds in length, and was comprised of 138 bat calls and 100 background noise clips.

The precision, recall and F_1 score metrics were all used to analyse the algorithm's performance. These three metrics are all standard measures of classification algorithm accuracy. Precision is described as the fraction of correctly classified positive responses among all responses, whereas recall (occasionally referred to as sensitivity) is the fraction of correctly classified positives among just the positive responses. They are calculated as follows:

$$\text{precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}} \tag{4.1}$$

$$\text{recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}} \tag{4.2}$$

The harmonic mean of these two metrics is often used as an overall measure of detection rates and is referred to as the F_1 score:

$$F_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{4.3}$$

Using the soprano pipistrelle dataset, the bat detection algorithm configured to respond at 50 kHz achieved an F_1 score of 0.961. This score was achieved from a precision of 0.994 and a recall of 0.931. This high level of precision is likely due to the low number of false possible positive sources in the data collection location. The data was collected in various green spaces around Southampton, away from pest deterrent devices and electric streetlights, which are common sources of ultrasonic noise in urban environments.

An alternative classification metric for detection algorithms uses a receiver operating characteristic (ROC) plot, taking the area under the curve (AUROC, or occasionally just AUC). In the context of the bat detector, its AUROC is equal to the probability that a randomly chosen recording containing a bat call will be ranked higher than a randomly chosen recording which does not. The detector's ROC curve is shown in Figure 4.5 and presents an AUROC of 0.975. While the AUROC has long been a standard measure of classification accuracy, many researchers question its usefulness for a number of reasons, including the fact it summarises performance over large areas of the ROC space which an algorithm would never operate [Lobo et al., 2008], such as thresholds with unreasonably high false positive rates. It is for this reason that AUROC is used in conjunction with metrics such as the F_1 score.

Loud, ultrasonic noise is relatively uncommon in natural, terrestrial environments, limited to bat echolocation and communication among certain species of frog. While the detection accuracy of the algorithm will vary depending on the rarity of false positive sources in the deployment environment, the tendency of animals to find their own acoustic niches keep the quantity of competing ultrasonic biophony low.

4.1.5.2 Deployment

The effectiveness of the bat detection algorithm was assessed by using it in a real-world deployment, running on AudioMoth in Spring, 2017. The deployment was a preliminary acoustic survey carried out by researchers from the Zoological Society of London (ZSL), aiming to study bat species in Cuba's Guanahacabibes peninsula. The focus of the research was the foraging distribution and behaviour of the Cuban greater funnel-eared bat (for information on the species, see Section 2.5.1), a species endemic to a cave in the peninsula known as Cueva la Barca.

For the deployment implementation, the algorithm was set to monitor a frequency band centred around 60 kHz with a bandwidth of 4 kHz. Deployed devices were set to wake every 2 seconds and listen for 512 ms overnight, between the hours of 17:30 and 7:00 (13.5 hours total). For the remaining 10.5 hours each day, each AudioMoth entered a low-power sleep state consuming just 45 μW. While gathering and analysing samples, AudioMoth used 20.9 mA and 17.7 mA respectively. The deployment was carried out while AudioMoth was still in its prototype phase, meaning AAA batteries were used instead of the standard AAs. These AAAs provided 1200 mAh of power, and given 13.5 hours of active sensing per day, gave a predicted life expectancy of 44 days.

Twenty AudioMoth were deployed in transects along each of the cardinal directions, spaced 20, 40, 60, 180 and 360 metres from a single large cave entrance known to be used by bats. These devices triggered 8,989 times in total, producing 25 hours of triggered audio which required 8.25 GB to store. Had the AudioMoth been set to passively record, waking up at the same intervals, 729 hours of audio would have been collected, requiring 240 GB (a reduction of 96.6%). Of these 8,989 recordings, 7.85% were false positives. The frequency of false positives varied significantly with the position of the device, with a third of all triggered recordings collected by the AudioMoth deployed 180 m north of the cave containing no bat calls. Four devices positioned closer to the cave's entrance produced no false positives.

The experiment required a large number of separate devices listening simultaneously across the bat's habitat. $1049.79 was spent to purchase 21 AudioMoth devices at $49.99 each (20 deployed devices with a single redundancy). A SM4BAT recorder and accompanying microphone similarly costs $1,048[1]. While this single, high-cost device may have a large detection range, it could not compete in coverage with the equivalent cost's worth of low-cost AudioMoth units. The sensors would also have to be left unattended for extended periods of time, meaning damage or theft is a possibility. For low-cost sensors such as AudioMoth, each destroyed device is a loss of $49.99 compared to the SM4BAT's $1,048.

Reliable identification was an issue with a number of triggered recordings as the algorithm was sensitive enough to react to bat calls which were recognisably bats, but too quiet to discern the specific species. This issue was made worse by the fact that, due to their low amplitude, the AudioMoth's MEMS microphone failed to capture all harmonics of the calls, further increasing the difficulty of species recognition.

The survey successfully managed to track bats throughout the monitoring area, mapping the foraging range using call density. The drop-off in density occurred much sooner with

[1] https://www.wildlifeacoustics.com/store/recorders

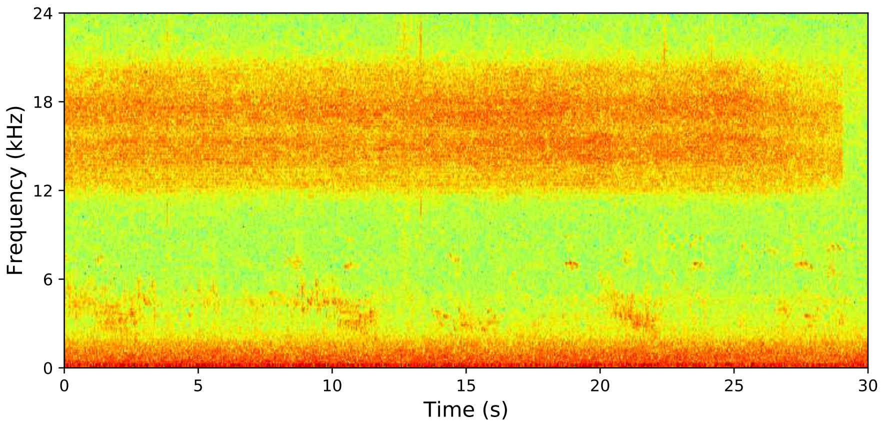


FIGURE 4.6: A spectrogram showing a recording of the extended call of a New Forest cicada. The call covers 12 - 21 kHz and lasts approximately 30 seconds.

devices deployed to the south-west of the cave entrance. As a result, ZSL hopes to deploy more devices in greater density to the north-east of the cave in future.

4.2 New Forest cicada detection algorithm

The second algorithm for animal vocalisation reacts to the call of a specific species of cicada, known as the New Forest cicada. The algorithm was once again deployed on low-cost, low-energy AudioMoth hardware, aiming to automate the annual process of surveying for this possibly-extinct species.

Historically, New Forest cicada surveys have been carried out manually by entomologists each year, travelling to likely sites chosen based on previous sightings and documented behaviour. The 14 kHz call of the cicada (see Figure 4.6) is audible to humans but can be difficult to hear depending on the age of the observer. The commonly stated hearing range of humans is 20 Hz to 20 kHz, with a sharp drop-off in sensitivity with frequencies greater than 15 kHz. As a result, these manual surveys often rely on a combination of call detection and visual surveys for “turrets” (for more information on the species, see Section 2.5.2). Unfortunately the manpower available for these surveys is limited, preventing extensive coverage.

A previous attempt to automate this search using acoustic detection was the CicadaHunt smartphone app, developed by Zilli et al. [2014]. This app used citizen science, leveraging the large number of visitors to the New Forest national park each summer to record possible cicada calls, uploading the results to a central repository. This allows average visitors to collect reliable data without expert knowledge.

A common theory about the lack of recent New Forest cicada sightings is it is due to increased human presence throughout the New Forest, driving the insects away from

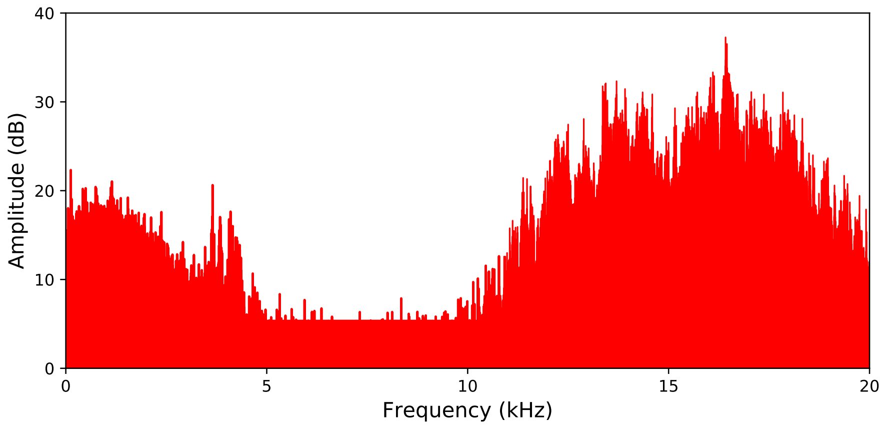


FIGURE 4.7: FFT plot showing the frequency distribution throughout a recording of a New Forest cicada, recorded in Slovenia by the author. The cicada's buzz has a strong response around 14 kHz.

areas frequented by visitors. As these visitors are encouraged to remain on the designated paths, monitoring carried out by them is biased against likely current habitats. Attempting to remedy this by encouraging citizen scientists to go off trails and fill these gaps is irresponsible both in terms of safety and protecting the habitat of the species in question. Another limitation is the reliance on visitors repeatedly returning to sites to obtain constant coverage. Any data collection will have gaps based on when citizen scientists are at a location and using the app.

Performing monitoring remotely solves these issues, requiring surveyors to visit a monitoring site just twice (deployment and collection) while still attaining constant temporal coverage. Thus an AudioMoth deployment with an accompanying detection algorithm was used as an alternative monitoring technique, supplementing CicadaHunt by filling in gaps left at priority locations.

4.2.1 Detection algorithm requirements

The call of a male New Forest cicada is approximately 30 seconds in length and has a frequency range that extends from 12 to 21 kHz (see Figure 4.7). The entire call was required in order to verify the recording's validity, so the AudioMoth was set to record at 48 kHz, giving it a Nyquist frequency of 24 kHz.

The dominant 14 kHz component of the cicada's call is rarely found in the calls of other animals found in the New Forest. One of the few species which produces a call within this frequency band is the dark bush cricket (*Pholidoptera griseoaptera*), which produces short, broad-spectrum chirps covering a similar range. As well as biophony, geophony is also a source of 14 kHz noise as wind produces broad-spectrum white noise in this range. To gain resilience from these possible false positive sources, a second frequency band around 8 kHz was used to classify cicada calls.

To cover the full emergence period of the species each summer, monitoring devices must listen for a three-month period from May until July. During these three months the AudioMoth devices must be fully autonomous and thus require neither battery nor storage replacements. Keeping both energy and storage consumption low enough to make this possible is due to the effectiveness of the detection algorithm and its implementation on AudioMoth.

4.2.2 Sample collection

Because the cicada is documented as calling exclusively during daylight hours when temperatures exceed 20° C, the AudioMoth devices were set to listen for 8 hours a day. The listening period started at 9:00 and finished at 17:00.

As with the bat detection algorithm, the cicada detector operates on a duty cycle even when listening, sleeping for five seconds between listening periods to conserve battery. Each time the AudioMoth wakes up, it collects 8,192 samples and stores them in a single SRAM partition. At 48 kHz, these 8,192 samples represent 171 ms of audio. The short listening period allows the algorithm to repeat multiple times over the course of a 30-second call, giving multiple opportunities for the call to trigger a detection. Running the algorithm multiple times increases the likelihood of successfully detecting the call without additional complexity.

4.2.3 Feature extraction

To detect the cicada call without reacting to broad-spectrum noise, the collected samples are fed through two Goertzel filters at 14 and 8 kHz respectively. These filters possess a bandwidth of 1.5 kHz and thus require a filter length L of 128 samples (see Equation 2.3). These 8192 samples produce 64 windows ($N = 64$) which are then duplicated and fed through the 14 and 8 kHz Goertzel filters to produces two sets of 64 amplitudes.

For the 2016 deployment in the New Forest, a prototype version of the AudioMoth was used. This prototype was powered by three AAA batteries (1200 mWh) and was built around a microcontroller with no capacity for floating-point operations. In order to implement the DSP required to perform the algorithm as well as reduce the computation (and thus energy consumption) required, the feature extraction process was implemented using fixed-point arithmetic.

Fixed-point representation stores the fractional and whole parts of numbers in a set number of bits. Real values can be stored as integers using fixed point representation by losing some precision. This process involved pre-calculating the c, α and Hamming factor constants in this representation and replacing all operations with fixed-point implementations. Almost all values used are stored as 32-bit integers with eight bits dedicated

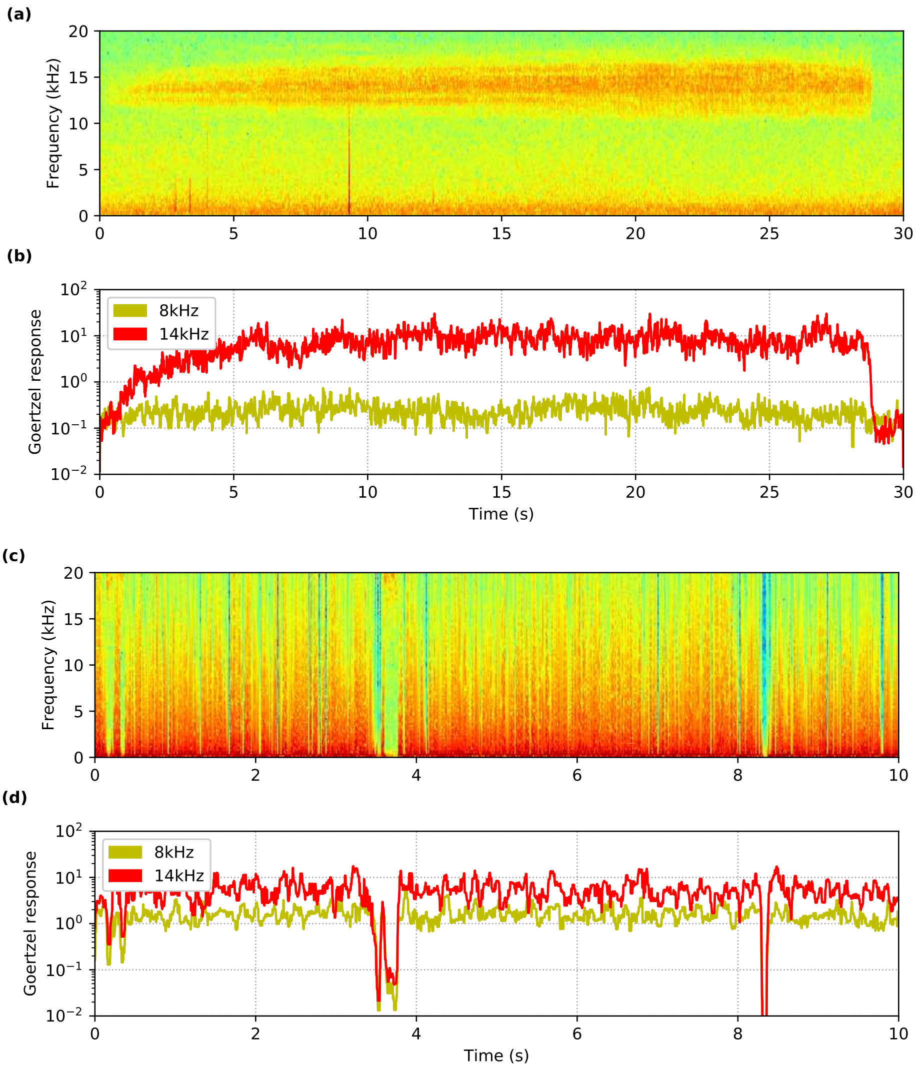


FIGURE 4.8: Plots showing: (a) spectrogram of a New Forest cicada recording; (b) the Goertzel responses of the same recording; (c) a recording of strong wind; (d) Goertzel responses from the wind. The cicada is detected in the recording shown in (a) thanks to the presence of 14 kHz sound without the broad spectrum noise shown in the wind recording (b), which stretches into 8 kHz.

to the fractional part. This leaves 24 bits for the whole part, giving a resolution of 2^{-8} (1/256) and a range of $[-(2^{24}), 2^{24} - 2^{-8}]$. As Hamming factors are all positive values less than 1 they are instead stored as 8 bit fixed point values, with all 8 bits representing the fractional part of their value. In the 2017 and 2018 deployments a floating-point implementation was used as well as devices equipped with AA batteries.

The amplitudes produced by the two filters are zipped together to create 64 pairs of Goertzel responses. To produce a single identifier which could be used to decide whether or not to record, first the ratio of each pair is found and then the median ratio value is taken, comparing it to threshold T to form a classification C:

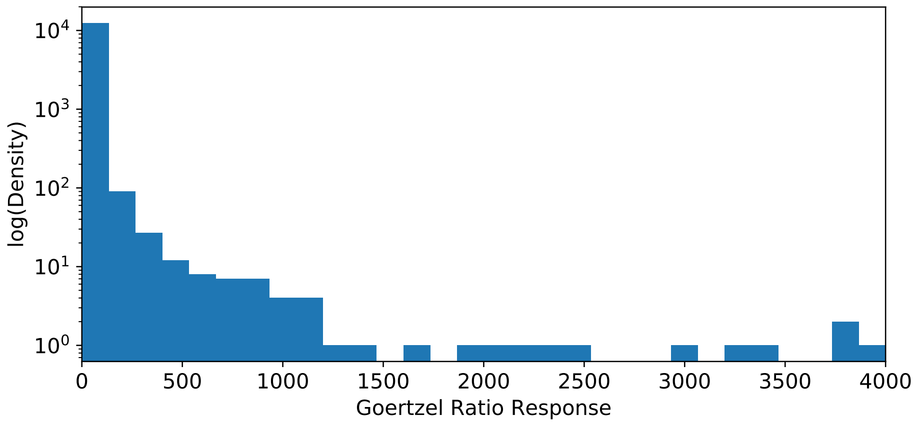


FIGURE 4.9: Histogram showing the distribution of ratio responses in a recording of a dark bush cricket.

$$C = \begin{cases} 1, & \textbf{if } \text{Median}\{m_i^{14}/m_i^8 \quad | \quad 1 \leq i \leq N\} > T \\ 0, & \textbf{otherwise} \end{cases} \tag{4.4}$$

The ratio of the two frequency components produces an identifier which is robust to broad-spectrum noise. Sounds which cover both 14 and 8 kHz produce low ratio values, allowing them to be distinguished from cicada calls which stop at 12 kHz (see Figure 4.8).

4.2.4 Feature extraction alternatives

Alternate feature extraction methods were also assessed when developing the algorithm. Four were tested, using different methods to combine the two frequency responses into a single set of values and then that set into a value which could trigger a threshold. These included taking the difference between the two frequencies rather than their ratio:

$$C = \begin{cases} 1, & \textbf{if } \text{Median}\{m_i^{14} - m_i^8 \quad | \quad 1 \leq i \leq N\} > T \\ 0, & \textbf{otherwise} \end{cases} \tag{4.5}$$

The two remaining variants used the sum of the collected values for both the ratio and difference values, rather than the median. Using the sum with a scaled threshold is functionally identical to the mean, while requiring fewer operations:

$$C = \begin{cases} 1, & \textbf{if } \sum_{i=1}^{N} m_i^{14}/m_i^8 > N \times T \\ 0, & \textbf{otherwise} \end{cases} \tag{4.6}$$

Identifier	Method	TPR	Completion time (ms)
Ratio	Median	0.998	11.58
Difference	Median	0.996	11.50
Ratio	Sum	0.924	11.25
Difference	Sum	0.707	11.11

TABLE 4.1: Table of completion time and TPR for four detection algorithm variants, each using a different feature extraction technique.

$$C = \begin{cases} 1, & \textbf{if } \sum_{i=1}^{N} m_i^{14} - m_i^{8} > N \times T \\ 0, & \textbf{otherwise} \end{cases} \quad (4.7)$$

Taking the median provides robustness to shorter calls such as that of the dark bush cricket. If the mean is used, then a particularly loud cricket call could skew a set of ratios enough to put it over the threshold. The median requires at least 50% of the samples to represent the features of a cicada call in order to react. The distribution of Goertzel ratio values for a dark bush cricket recording is shown in Figure 4.9, where the small number of high ratios could skew a mean and trigger a detection, without having an effect on the median. Calculating the median ratio using an algorithm such as quickselect (see Section 4.1.4.1) is computationally more expensive than calculating the sum.

The four algorithm variants (median ratio, median difference, sum of ratios, and sum of differences) were assessed in terms of accuracy and completion time before the final design was decided upon. To test each algorithm variant, random sets of samples were taken from a collection of test recordings and classified. Sample sets either contained a cicada call, recorded in Slovenia as part of the work by Zilli et al. [2014], while the other contained forest background noise devoid of cicada singing. This was repeated 1,000 times for each algorithm, recording both the accuracy of the classifications and the completion time (see Table 4.1).

In terms of performance, the median of ratios performs the best with a TPR of 0.998, while its difference-based equivalent achieved a TPR of 0.996. Even though the completion time of the median ratio calculation is 0.47 ms longer than the sum of differences, the difference in completion time is negligible at less than half a millisecond.

Overall, the combination of median classification with a ratio-based identifier produced the highest accuracy with minor time differences from the other three identifier/classifier combinations. Because the algorithms vary by so little in terms of time, the design using the median ratio was used as it attained the highest accuracy with negligible completion time differences.

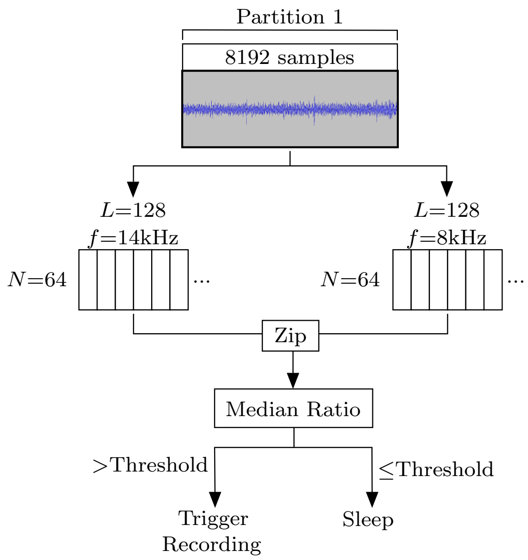


FIGURE 4.10: Data collection and analysis process for the New Forest cicada detection algorithm. First, 8192 samples are collected and stored in SRAM. Next, the samples are fed through two Goertzel filters in windows of 128. The resulting amplitudes are zipped together and the ratio of each pair is calculated. If the median of these 64 ratios exceeds a threshold, a recording is made.

4.2.5 Detection stage

Once the median ratio has been calculated, it's compared to a preset threshold. If it exceeds that threshold a 30-second recording is made, otherwise the AudioMoth is put into sleep mode until the next iteration. This full process can be seen in Figure 4.10.

In order to choose an appropriate threshold, the ROC curve shown in Figure 4.11 was generated. To produce this plot, a set of cicada recordings and background noise recordings was required to test each classification threshold. Recordings collected by Zilli et al. [2014] were used for this[2]. Each point along a ROC curve represents the TPR and FPR associated with a given threshold. Typically the "best" threshold is the one represented by the point closest to the top left of the curve, a point which balances maximising TPR while minimising FPR. However, the algorithm is designed such that it will be repeated multiple times over the course of a single 30-second cicada call. This means that a lower TPR can be accepted, given the cumulative detection probability and the need to minimise energy usage (each false positive consumes energy as a unnecessary recording is made). Listening for eight hours, a FPR of 0.01 produces approximately 60 recordings per day. The threshold was set using a 0.01 FPR, attaining the highest possible TPR given this restriction.

By allowing a reduced TPR, the algorithm is less proactive in terms of data collection, resulting in an increased false negative rate (FNR). The problem of missed cicada calls

[2]Recordings were collected again in 2018 to confirm the results produced by these recordings.

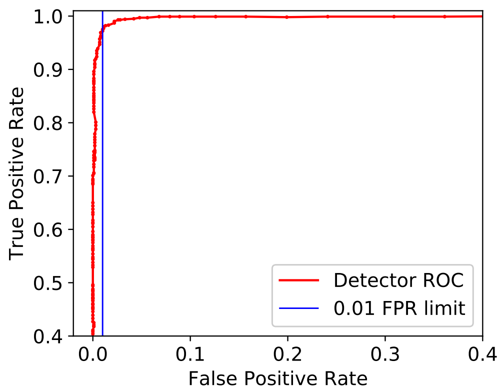


FIGURE 4.11: Receiver operating characteristic curve showing the performance of the New Forest cicada detector at various thresholds. The final threshold was chosen by capping the false positive rate to 0.01 and taking the threshold which corresponded with the highest attainable true positive rate.

can be mitigated somewhat through tactical device deployment. By positioning devices with overlapping ranges, the probability of detection increases in response to the FNR.

4.2.6 Results

As with the previous detection algorithm, the success of the New Forest cicada algorithm was measured by both its performance on a collected dataset and as part of a real-world deployment for conservation research.

4.2.6.1 Performance

The accuracy of the finalised detection algorithm design was tested using a set of recordings collected from a site in Slovenia. The site was previously used by Zilli et al. [2014] to collect recordings for their work. This dataset was collected by deploying a number of constantly recording AudioMoth devices at the site then manually sorting through the recordings later. This produced a set of 196 30-second cicada recordings and 228 background recordings, collected over multiple days. These labelled recordings were then used to calculate the recall, precision and F_1 score.

Using equations 4.1, 4.2, and 4.3, the algorithm was found to possess an F_1 score of 0.982 from a precision of 1.0 and a recall of 0.964. This perfect precision meant that there were no false positive responses to the background noise recordings. The high F_1 accuracy is a result of this insensitivity to false positives, as the algorithm had been designed to prioritise resilience to the small number of likely FP sources (wind and other insect calls) over TPR. The AUROC accuracy metric also reflected this high level of accuracy, with the algorithm achieving a score of 0.998.

Algorithm target	Bat	Cicada
Listening time per cycle (ms)	276	27
Sleeping time per cycle (s)	2	5
Recording time (s)	10	30
Listening current (mA)	17.7	18.0
Sleeping current (mA)	0.5	0.2
Recording current (mA)	24.0	24.0
Listening time per day (hours)	13.5	8.0
Sleeping time per day (hours)	10.5	16.0
Lifespan on 1200 mAh (days)	44	103
Lifespan on 3000 mAh (days)	110	258

Table 4.2: Energy consumption values and expected lifespans for both the bat and New Forest cicada detection algorithms, given a FPR of 0.01.

As well as collecting data for assessment in Slovenia using AudioMoth configured to act as PAMs, a number of devices were deployed with the algorithm itself. Three triggered devices were deployed at various locations around the site and were both manually monitored for when they triggered and had their resulting recordings compared to the labelled continuous data. The AudioMoth consistently responded to calls within approximately 10 m of the source.

4.2.6.2 Deployment

To assess the effectiveness of the algorithm for enabling accessible acoustic detection it was integrated into the ongoing efforts of the New Forest Cicada Project[3]. This project initially covered just the citizen science work done with the CicadaBuzz app (see Section 2.3.1), but has since expanded to encompass the annual deployment of AudioMoth around the New Forest. This deployment has occurred in 2016, 2017, and 2018, and is intended to continue in perpetuity.

Starting in 2016, 87 prototype AudioMoth devices were equipped with AAA batteries which provide 1200 mAh of power. In subsequent deployments, AudioMoth 1.0.0 was deployed with AA batteries which instead provide 3000 mAh. Each device was also deployed with a 8 GB Kingston Class 10 microSD card. The make and class of microSD card have a significant effect on the energy consumption of an AudioMoth due to the varying write speeds. Recording audio to a Kingston Class 10 card uses 24 mA.

The requirement of the deployment was to deploy devices for a minimum of three months in order to achieve sufficient temporal coverage. With a recording current of 24 mA, listening current of 18.0 mA and sleeping current of 0.2 mA, AudioMoth was able to achieve a lifespan of 257.8 days given AA batteries (see table 4.2). This far exceeds

[3] www.newforestcicada.info

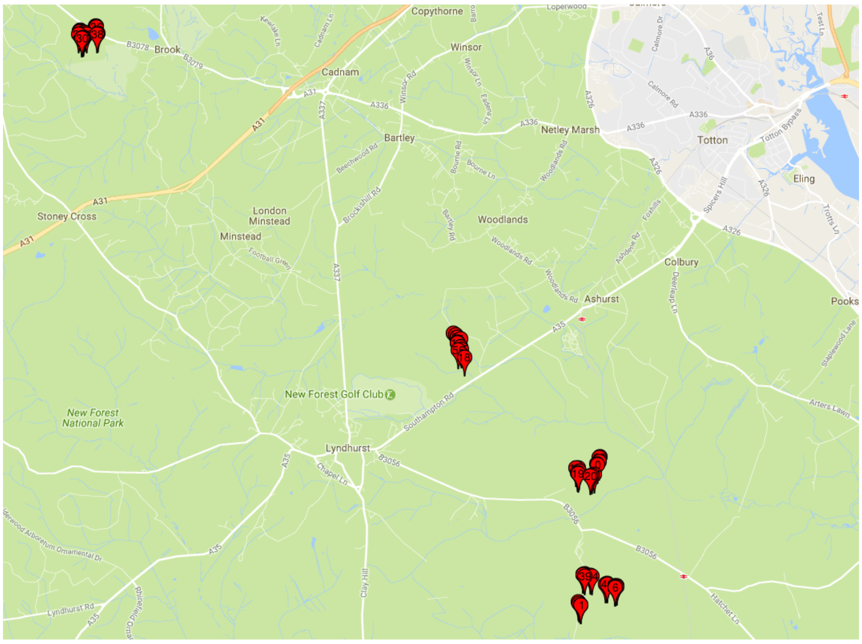


FIGURE 4.12: A map showing the four sites used to deploy 87 AudioMoth devices in 2016 to monitor for the New Forest cicada. These sites were reused in 2017 and 2018.

the required 90 days. However this buffer provides redundancy in the event a device is deployed in a location which the cicada returns to or has a nearby source of unexpected false positives.

In 2016, 2017, and 2018, devices were deployed at four sites: Denny Wood, White Lodge, Botley wood and woodland surrounding Bramshaw cricket ground (see Figure 4.12). These survey sites were chosen based on advice given by entomologists from BugLife in 2016. At each site, devices were positioned around forest clearings, facing east to receive direct sunlight earlier in the day.

Unfortunately, as of summer 2018, the New Forest cicada has not been found over the course of these deployments. However, thanks to the introduction of reactive recording technology, surveys with a much greater degree of spatial and temporal coverage have been carried out. Despite the New Forest cicada's conservation status as a priority species, very few resources have been dedicated to the annual search. Despite the present lack of success in detecting the species, AudioMoth and the accompanying acoustic detection algorithm have made it possible to continually carry out low-cost, remote monitoring of the area and maximise the probability of detecting the New Forest cicada if it still remains in the UK.

4.2.7 Supporting software

In order to support the ongoing process of monitoring the New Forest, additional software was developed which supplements the work done by AudioMoth and its on-board

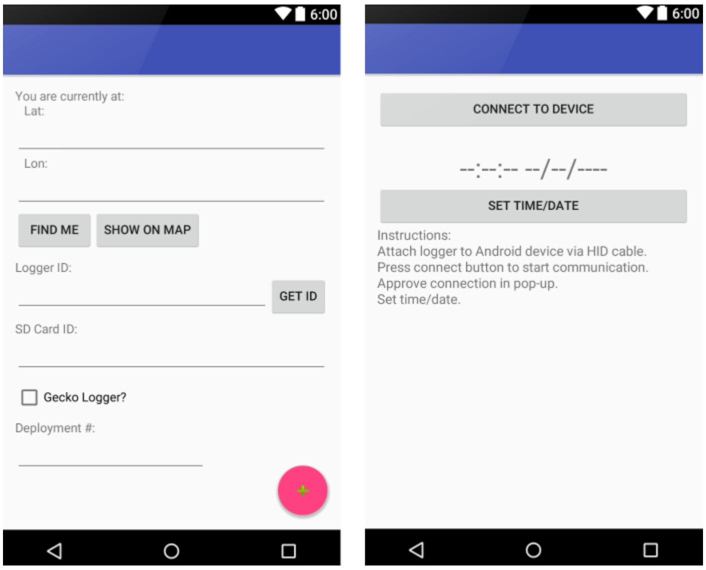
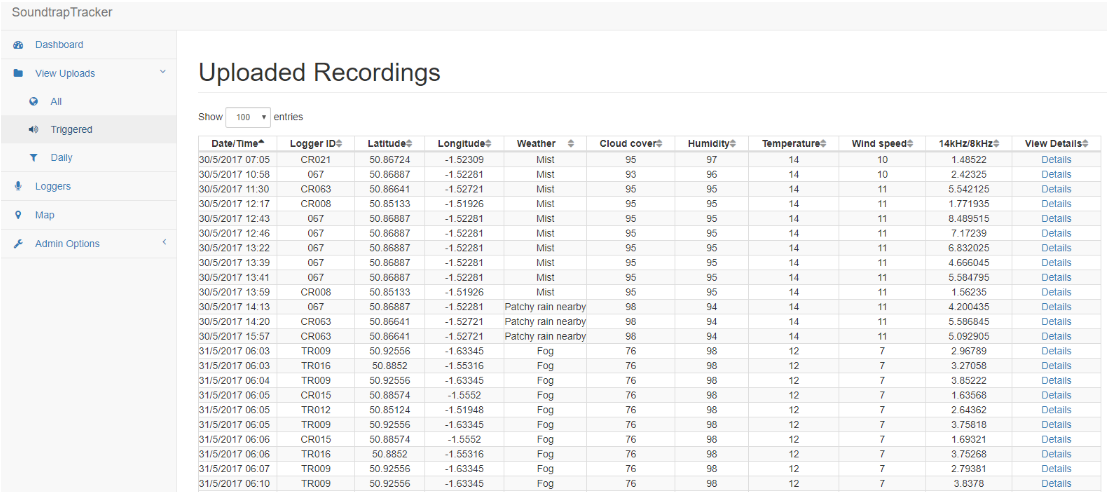


FIGURE 4.13: Screenshots from two pages of a smartphone app designed to automate the process of recording and collating AudioMoth deployment information. As well as this, the app also initialised devices in the field by setting the on-board clock.

detection to improve monitoring accessibility. This included a smartphone application (see Figure 4.13) designed to both initialise devices in the field and automate the process of logging each AudioMoth placement. Using the USB On-The-Go (OTG) specification, an Android app was designed to communicate with AudioMoth devices using the HID protocol. As the firmware for this deployment hard-coded configuration settings such as the sample rate and the sleeping and listening times, communication was limited to setting the on-board clock and obtaining the AudioMoth's 16-character unique ID. This ID, along with the GPS co-ordinates obtained by using the Android device's built-in GPS, was used to register each deployed AudioMoth.

The New Forest is home to a population of approximately 5,000 horses and ponies which are free to graze on open land within the forest. These horses and ponies are owned by properties within the New Forest which possess Common Rights. As a result of these free-roaming equines, AudioMoth devices were required to be deployed at least two metres off the ground, outside of their "reach". Because of this, deployed AudioMoth were installed above eye-level, in the branches of trees. This made them difficult to spot and recover, even if the locations of the sites were recorded. It was for this reason the smartphone app recorded the GPS location of each deployment tree. This streamlined the retrieval process, as maps detailing the exact deployment locations could be generated.

The records for a deployment were stored locally on each smartphone running the app until the end of the initial installation process. Once this process was complete, the records were uploaded to a central server. This meant that multiple groups of researchers were able to deploy simultaneously and easily collate deployment information later.



Date/Time	Logger ID	Latitude	Longitude	Weather	Cloud cover	Humidity	Temperature	Wind speed	14kHz/8kHz	View Details
30/5/2017 07:05	CR021	50.86724	-1.52309	Mist	95	97	14	10	1.48522	Details
30/5/2017 10:58	067	50.86887	-1.52281	Mist	93	96	14	10	2.42325	Details
30/5/2017 11:30	CR063	50.86641	-1.52721	Mist	95	95	14	11	5.542125	Details
30/5/2017 12:17	CR008	50.85133	-1.51926	Mist	95	95	14	11	1.771935	Details
30/5/2017 12:43	067	50.86887	-1.52281	Mist	95	95	14	11	8.489515	Details
30/5/2017 12:46	067	50.86887	-1.52281	Mist	95	95	14	11	7.17239	Details
30/5/2017 13:22	067	50.86887	-1.52281	Mist	95	95	14	11	6.832025	Details
30/5/2017 13:39	067	50.86887	-1.52281	Mist	95	95	14	11	4.666045	Details
30/5/2017 13:41	067	50.86887	-1.52281	Mist	95	95	14	11	5.584795	Details
30/5/2017 13:59	CR008	50.85133	-1.51926	Mist	95	95	14	11	1.56235	Details
30/5/2017 14:13	067	50.86887	-1.52281	Patchy rain nearby	98	94	14	11	4.200435	Details
30/5/2017 14:20	CR063	50.86641	-1.52721	Patchy rain nearby	98	94	14	11	5.586845	Details
30/5/2017 15:57	CR063	50.86641	-1.52721	Patchy rain nearby	98	94	14	11	5.092905	Details
31/5/2017 06:03	TR009	50.92556	-1.63345	Fog	76	98	12	7	2.96789	Details
31/5/2017 06:03	TR016	50.8852	-1.55316	Fog	76	98	12	7	3.27058	Details
31/5/2017 06:04	TR009	50.92556	-1.63345	Fog	76	98	12	7	3.85222	Details
31/5/2017 06:05	CR015	50.88574	-1.5552	Fog	76	98	12	7	1.63568	Details
31/5/2017 06:05	TR012	50.85124	-1.51948	Fog	76	98	12	7	2.64362	Details
31/5/2017 06:05	TR009	50.92556	-1.63345	Fog	76	98	12	7	3.75818	Details
31/5/2017 06:06	CR015	50.88574	-1.5552	Fog	76	98	12	7	1.69321	Details
31/5/2017 06:06	TR016	50.8852	-1.55316	Fog	76	98	12	7	3.75268	Details
31/5/2017 06:07	TR009	50.92556	-1.63345	Fog	76	98	12	7	2.79381	Details
31/5/2017 06:10	TR009	50.92556	-1.63345	Fog	76	98	12	7	3.8378	Details

FIGURE 4.14: A table generated by SoundtrapTracker. Using GPS and weather data, large quantities of recordings can be sorted and prioritised for analysis.

As well as basic deployment information, the central server hosted an online hub named SoundtrapTracker, which provided support for both data collection and recording analysis by storing and presenting the collected audio. In the 2016 deployment 124.5 hours of audio was collected (14,941 recordings). This is a drastic reduction in quantity compared to a system using PAM, given that 87 devices recording constantly for eight hours a day for a three-month period would produce 62,640 hours, given sufficient storage. Despite this, significant manpower would still be required to manually verify the presence of the cicada in any of the 124.5 hours of recorded audio.

The GPS data collected allowed triggered recordings collected at sites with environmental conditions more conducive to a New Forest cicada sighting (direct sunlight, 20 °C) to be prioritised for analysis. Recordings could be sorted according to factors such as cloud cover, temperature, and even the Goertzel ratio response using an implementation of the detection algorithm (see Figure 4.14).

Visual inspection can be a valuable tool for quickly analysing large quantities of audio data. Plots such as the raw waveform, Goertzel ratio plots and spectrograms can all present information about a 30-second recording which can be parsed by a human faster than listening to the recording itself. SoundtrapTracker employed this technique by presenting groups of recordings in the form of spectrograms for researchers to rapidly analyse and verify or refute the presence of a cicada call in each (see Figure 4.15).

The various functionality provided by SoundtrapTracker and its supporting smartphone app further reduced the requirements of the New Forest Cicada Project in terms of both time and manpower. Between 2016 and 2018, the system was used to analyse thousands of recordings collected by AudioMoth across each of the four sites. Software applications such as these can be used as a design basis for a wide variety of supporting software for improving the accessibility of large-scale acoustic monitoring projects.

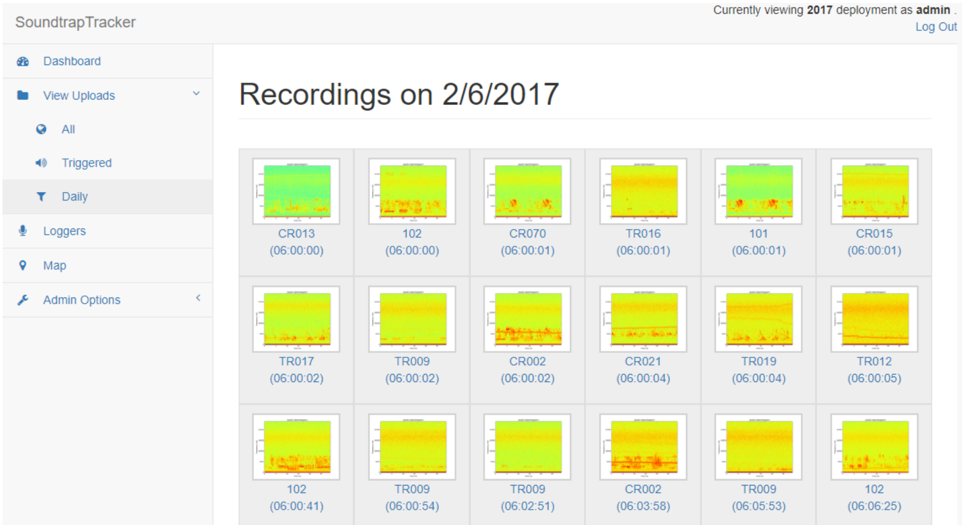


FIGURE 4.15: A grid of spectrograms taken from a subset of recordings collected from the 2017 New Forest deployment. Visual inspection of these plots was used to rapidly analyse each year's dataset.

4.3 Summary

This chapter presented two algorithms designed to detect two distinct animal vocalisations: bat echolocation calls and the prolonged buzz of a male New Forest cicada. Because of the nature of the target sounds and the deployment environment, the two algorithms had differing requirements in terms of the type of data they were collecting and how it was collected. These requirements included sample rates high enough to capture ultrasonic audio and long-term monitoring using just AA batteries.

Existing solutions for bat detection exist, however most widely-used devices are too expensive to be deployed in large enough quantities for the intended application [Adams et al., 2012]. For New Forest cicada detection, previous work has been done by Zilli et al. [2014], using a smartphone application and citizen science. This monitoring project is unfortunately limited to locations and times frequented by the general public, meaning large areas of the forest are left insufficiently monitored. AudioMoth, deployed with the algorithms discussed in this chapter, offer alternative solutions to both problems and enable monitoring which fulfils requirements not previously possible.

For both applications, the requirements informed the design of each algorithm as well as implementation techniques employed in order to run them on AudioMoth hardware. The first, a generalised bat detection algorithm, uses a single frequency band to identify the presence of echolocation calls in a set of audio samples recorded at a sufficiently high sample rate. While this algorithm could be applied to any vocalisation limited to a single frequency band, it is likely to perform best at bat detection due to the lack of ultrasonic false positive sources. This algorithm required the detection of short, ultrasonic pips as well as deployment for multiple days without the need for battery or storage replacement. The New Forest cicada algorithm builds upon the first in terms of complexity, using the interaction between two frequency bands to detect in a manner which is robust to known false positive sources. Both detection methods were evaluated in terms of the detection accuracy of the algorithms themselves and their effectiveness as part of a detection system for a real-world deployment, the first in Cuba and the second in the UK.

Supporting the second deployment was a data logging app and a web-based information hub. These pieces of software, combined with AudioMoth and the detection algorithms deployed on it for each application, fulfil the requirement of enabling large-scale acoustic monitoring by reducing the requirements of both the data collection and the analysis side of the monitoring projects.

The following chapter covers the addition of further complexity to low-power acoustic detection algorithms, using the methods described in this chapter to monitor for anthropogenic disturbances in similar environments.

Chapter 5

An algorithm for detecting gunshots in a tropical forest

> If you look at human behaviour around the world, you have to admit that we can be very aggressive.
>
> — Jane Goodall

Environmental monitoring is often focused on studying the effects of ecosystem degradation, including changes in behaviour or population size of a specific species. This includes the work described in Chapter 4. Problems such as ecosystem degradation and biodiversity loss come as a result of a number of factors including habitat destruction, climate change and direct interference by humans in the form of poaching. In order to identify vulnerable areas, it can be more useful to focus monitoring efforts on the causes of these issues rather than the effects.

Acoustic monitoring benefits from being able to detect acoustic events which other, visual-based detection methods cannot without increased costs for a similar deployment scale. The one visual remote monitoring method which does not suffer from a limited scale is satellite imagery. Satellite imagery is used to monitor anthropogenic disturbances in tropical forests, but generally focuses on large-scale damage such as deforestation and struggles to detect the presence of small groups of poachers under the canopy.

As a result, this chapter focuses instead on small-scale anthropogenic disturbances which can collectively have a large effect on ecosystem degradation. It covers the development of an anthropogenic disturbance detection tool which builds on the complexity of the algorithms discussed in Chapter 4. Focusing on gunshot detection, the algorithm incorporates a hidden Markov model to detect the acoustic behaviour of gunshots as they decay over time rather than the frequency components of the sound as a whole. As with the bat and cicada algorithms, the gunshot detection algorithm is described in terms of the application context, the design motivations and the performance, both in terms of its algorithmic accuracy and as part of a real-world deployment.

The detection accuracy of the algorithm is also benchmarked against a CNN; a commonly-used machine learning technique within the field of acoustic analysis. By comparing the algorithm's performance to a widespread approach to various acoustic detection problems, the effectiveness of the algorithm could be better assessed.

All field tests and device deployments were carried out in Belize by the author, Andrew Hill, Patrick Doncaster, Jake Snaddon, and Evelyn Piña Covarrubias. Hardware and firmware development was done by Andrew Hill and Alex Rogers, with the case for the final deployment designed by Andrew Hill.

5.1 Gunshot detection algorithm

This acoustic detection algorithm aims to detect the sound of gunshots at various distances in a specific tropical forest environment. The chosen location is the Tapir Mountain reserve in Belize, Central America (see Section 2.5.3 for more information). This scientific reserve is subject to unknown levels of anthropogenic disturbances in the form of illegal logging and poaching.

As with the cicada detector, the goal was to implement a detection algorithm which made a long-term monitoring deployment possible using low-cost hardware. The basis of the detection algorithm was formed by studying recordings of two different types of shotgun (12 and 16 gauge), fired at various distances and across various terrains within a similar tropical forest environment to the intended deployment location. Analysis of these recordings enabled the identification of the acoustic characteristics which can define a gunshot and could be used to detect it within a set of audio samples.

5.1.1 Detection algorithm requirements

Different varieties of firearm produce distinctly different sounds when fired. This is an issue when developing a detection algorithm which relies on the standard features of the gunshot to identify poaching events. These features must be broader in order to identify a rifle, a handgun and a shotgun firing as the same sound and thus may also be more susceptible to false positives. However, while the firearms used by poachers vary depending on the hunting location and the target, the guns used for a specific target or location are fairly homogeneous due to factors such as availability or the effectiveness of the weapon. Anonymously polled locals who lived in range of the Tapir Mountain reserve claimed that the majority of unlawful hunting in the area is done using shotguns.

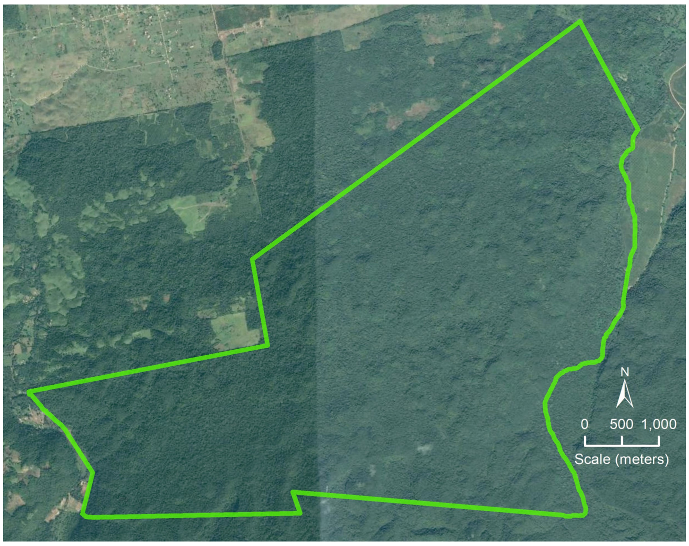


FIGURE 5.1: Map of the Tapir Mountain reserve. The reserve covers 25.4 km^2 of tropical forest and was the deployment location for the gunshot detection algorithm.

Shotguns used for hunting all produce a loud initial muzzle blast which covers a wide frequency range. These frequency components then decay rapidly as the gunshot propagates from its source [Maher, 2007]. The rate at which each component decays varies, with higher frequencies decaying faster as their energy is dissipated and they are absorbed into the air and foliage. The rates at which various frequencies present in a muzzle blast appear then decay can be used as identifying characteristics of a gunshot.

AudioMoth was used to perform large-scale monitoring of the reserve, attempting to quantify the levels of illegal hunting which occurs there. While it is known poaching occurs within the reserve, no quantifiable data exists for the amount or distribution. The results of this deployment could be used to inform future budget assignment and conservation policy for the reserve. To collect sufficient data on the reserve, the research aimed to detect and record poaching events across the entire 25.4 km^2 reserve (see Figure 5.1) over the course of a year.

It was calculated that 79 devices would be required to efficiently cover the reserve (see Section 5.1.5.3). Deploying just 10 of these devices as part of a test deployment took several days due to the difficulty involved in traversing the terrain. This meant battery/storage replacement at regular intervals throughout the year is not feasible and year-long, unattended monitoring is a deployment requirement. While energy harvesting techniques such as solar power and triboelectric nanogeneration exist for sensors such as these, the deployment environment renders these techniques ineffective. For solar power, deploying under the canopy limits the amount of direct sun exposure and there is insufficient constant sound pressure present in the forest to match the AudioMoth's energy consumption with triboelectric nanogeneration.

5.1.2 Sample collection

Unlike cicada or bat detection, poaching events generally involve a single instance of the target sound, meaning constant listening is required to detect them. If the poacher is successful, they are unlikely to fire again in the immediate vicinity, as target animals will likely flee the sound of the initial shot. This means that during these periods devices must listen and analyse constantly for hours at a time and no duty cycle can be implemented to improve energy consumption. In the reserve, illegal hunting of game animals occurs almost exclusively between sunset and and sunrise due to both the nocturnal nature of target species and to avoid detection. This means this constant monitoring must last for 12 hours each day for 365 days (4,380 hours of listening).

A number of steps were taken in order to achieve this without the need to significantly increase battery capacity with an automotive battery. These include reducing the clock speed of each AudioMoth from 48 MHz to 11 MHz. This reduces the energy consumption while performing all actions. However, underclocking the AudioMoth's microcontroller has the effect of increasing the time required to compute the various steps of the algorithm, so no energy is saved despite the lower consumption. The savings come from the baseline energy consumption being lower for all tasks including sleeping and DMA.

It was also necessary to gather samples to both listen to and record at a much lower sample rate. For this implementation, samples are gathered at 8 kHz. This reduced sample rate means that even with the clock set at 11 MHz the AudioMoth is able to perform the complete algorithm on approximately four seconds of audio (32,768 samples) in real-time. The reduced sample rate also improves storage efficiency, as 512,000 recordings at 8 kHz (approximately 62.5 KB each) can be stored on the 32 GB microSD cards each AudioMoth was deployed with.

In order to ensure both constant listening and that no gunshot be split between data collection instances, a three-partition circular buffer is used to collect and analyse samples. Each of these three partitions contains 16,384 samples (approximately two seconds) and are analysed in pairs. While the first and second partitions are analysed, the third is filled with new samples. Once analysis is complete on those two partitions, the samples in the first partition are replaced while the algorithm is run on the second and third partitions. This meant that each set of 16,384 samples is analysed twice and any gunshot split between two is in the centre of the following pair.

5.1.3 Feature extraction

Muzzle blasts are extremely loud compared to the background noise level in tropical forests. When recorded one metre from the source, a gunshot from a 12-gauge shotgun (Remington model 1187) reaches 132.6 dB [Piña-Covarrubias et al., 2018]. However, regardless of the deployment environment, gunshot detection rarely relies solely on simple

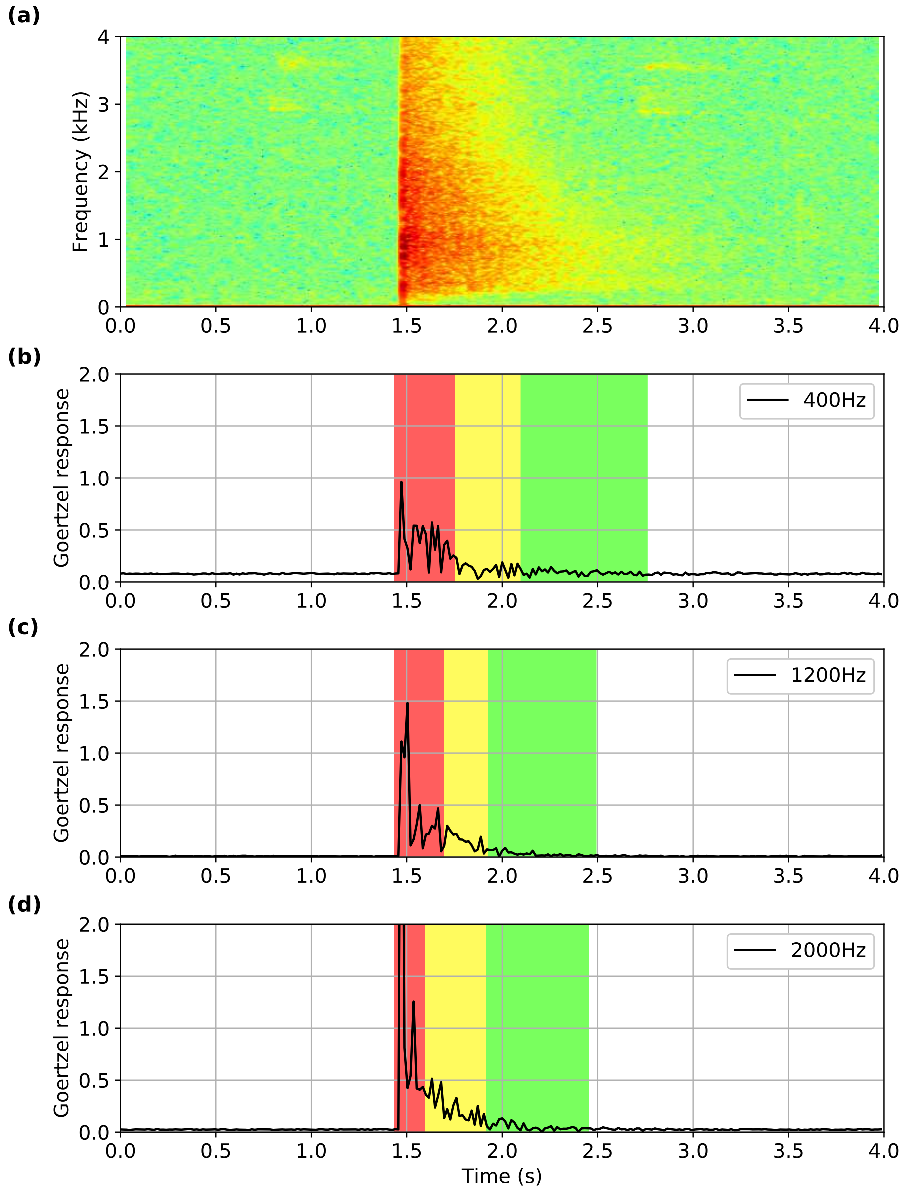


FIGURE 5.2: Plots showing (a) Spectrogram showing a gunshot recorded 255 m from the source at a sample rate of 8 kHz; (b) 400 Hz; (c) 1200 Hz and, (d) 2000 Hz Goertzel responses to the same recording. The impulse state is marked in red, the decay state in yellow and the tail in green.

amplitude peak detection [Chacon-Rodriguez et al., 2011]. This is due to the variation in gunshot amplitude due to factors which have a rapid decaying effect on the signal such as distance or terrain, as well as the wide array of false positive sources such as snapping tree branches.

Instead, other characteristics of the signal are used to identify a gunshot. In this case, the behaviour of various frequency components is used. The muzzle blast of gunshots recorded close to the source possess a wide range of frequency components. As gunshots are audible within recordings up to approximately a kilometre (varying depending on various factors including weather conditions and terrain), frequency band features were

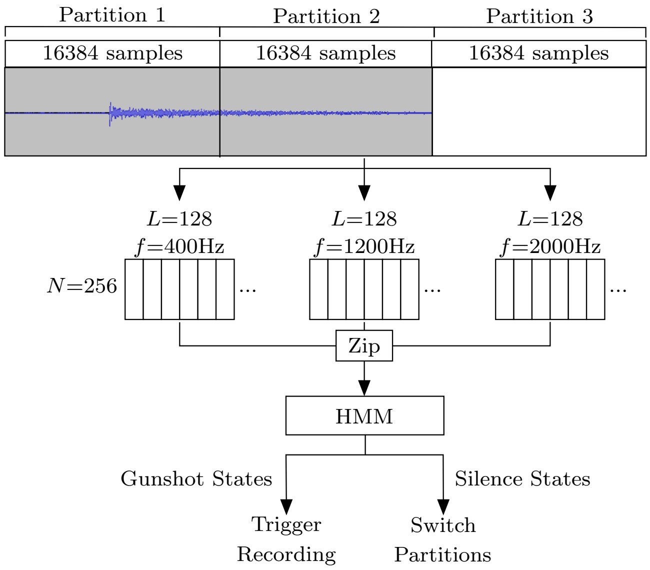


FIGURE 5.3: The sample collection, feature extraction and classification processes which form the gunshot detection algorithm. First, samples are collected and stored in two of the three SRAM partitions. While the third is filled, the extraction process is run on these partitions, zipping the resulting features into 256 observations. These observations are fed into the HMM, classifying a gunshot based on the presence of gunshot states. Once the algorithm is complete, it is run on the samples in the second and third partitions.

chosen which could accurately describe a mid-distance gunshot at around 500 m. For this, Goertzel filters at 400 Hz, 1200 Hz and 2000 Hz are used (see Figure 5.2). While other frequencies are present in a muzzle blast, the majority of these higher frequencies will have decayed to the point where they are inaudible by the time they reach the sensor. Gunshots which do not feature these components are likely to be inaudible and thus undetectable. For each filter, a bandwidth of 250 Hz is used, requiring a filter length of 128 samples ($L = 128$). This meant the 32,768 samples were filtered as 256 windows ($N = 256$).

Note that, as well as the three chosen frequency bands, other features were tested, including a fourth frequency band focusing on the higher frequency components of a gunshot at 3,500 Hz and also the short-term delta energy (STDE). The STDE measures the differences in energy across windowed sections of a signal. This is useful for reducing the effect of noise as consistently noisy signals will have low energy deltas, whereas sudden, impulsive changes will produce large deltas. The precision, recall and F_1 score were used to assess each, resulting in the optimal 400, 1200, and 2000 Hz feature set.

5.1.4 Detection stage

Simplifying the signal down into a single value and performing thresholding on it, as in the case of the bat and cicada algorithms loses valuable temporal information. This temporal information could help tell the difference between a gunshot and the snapping

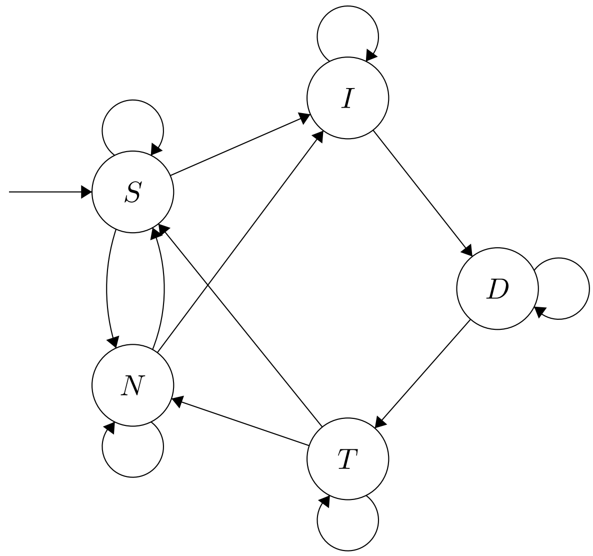


FIGURE 5.4: State diagram showing the five possible states of the gunshot detection HMM: silence *(S)*, noise *(N)*, impulse *(I)*, decay *(D)*, and tail *(T)*.

of a branch close to the sensor. While both feature a sudden, loud impulse across low frequencies, a branch snap will not have the acoustic structure of a gunshot.

Various methods exist for identifying the extent to which the structure of two signals are similar. For detection purposes this means the collected set of samples, processed to a set of feature signals, and a predefined set of values which can represent the target sound. For this detection algorithm, a hidden Markov model (HMM) is used to do this. The structure and standard application of a HMM using the Viterbi algorithm is described in Section 2.4.2.

After feature extraction has been completed and the filter responses have been calculated, the three Goertzel responses are zipped together into 256 triples. These triples are then used to produce a sequence of 256 latent states which represent the most likely set of events which occurred over the course of the recording. Figure 5.3 shows the complete sample collection, feature extraction and classification process.

The HMM classifies observations into a set of five possible latent states: Initial impulse *(I)*, decaying impulse *(D)*, tail *(T)*, noise *(N)* and silence *(S)*. These states and the possible transitions between them are shown in Figure 5.4. While silence and noise could be covered by a single state, splitting them in two allowed each state's distribution to better reflect its possible observations, without encompassing all noisy environments within the extreme end of the silence state.

The observation triples are classified by the HMM using the Viterbi algorithm (described in Section 2.4.2), producing the most likely path through the five possible states. If the 256 states run through the three gunshot states (I, D and T), then a gunshot is detected.

Unlike previously described detection algorithms, when the target sound has been detected in a set of samples, the audio immediately following it is not relevant. For a bat echolocation it is expected that another pip will follow immediately after. For the New Forest cicada, the call lasted 30 seconds, meaning the remainder of the call would follow a detection. For a gunshot, the analysed partitions which produced a positive algorithm response must be saved.

5.1.4.1 Constructing the model

The probability distributions for the three features in each state were calculated based on gunshot recordings collected in Belize between 2016 and 2018. The gunshot files were manually analysed and timestamps for each of the five states were recorded.

The distribution of the three features for the impulse state are shown in Figure 5.5 and are best fit to log-normal distributions. This type of continuous probability distribution possesses a long tail after its initial peak. This tail can reflect the high amplitude responses which occur when a gunshot is fired extremely close to the sensor.

To obtain the transition probabilities, the same gunshot recordings used to produce the emission probability distributions were used. Using the timestamps for each positive recording, the mean time spent in each state was then used to calculate the likelihood of transitioning out of it. In order to test these transition probabilities, they were used to produce sequences of simulated states which were examined to ensure they possessed a similar structure to that of a typical gunshot.

5.1.4.2 Computational complexity

The majority of the algorithm's computation is split between the feature extraction stage (the Goertzel filters) and the detection stage (the HMM). In terms of computational complexity, the feature extraction stage possesses a complexity of $\mathcal{O}(L)$, whereas the underlying HMM's complexity is calculated to be $\mathcal{O}(S^2N)$, with S being the number of states (five in this case) and N being the length of the observation sequence ($N = 256$).

5.1.5 Results

As with the zoological detection algorithms, gunshot detection was assessed first under lab conditions and then as part of a real-world conservation project. Before the deployment which began in 2018, gunshot recordings were collected in and around Tapir mountain. In 2016 these were collected using SM3 recorders, before the completion of the first AudioMoth model. In return trips to the reserve in 2017 and 2018, further audio data was collected with controlled gunshots as part of the performance tests using

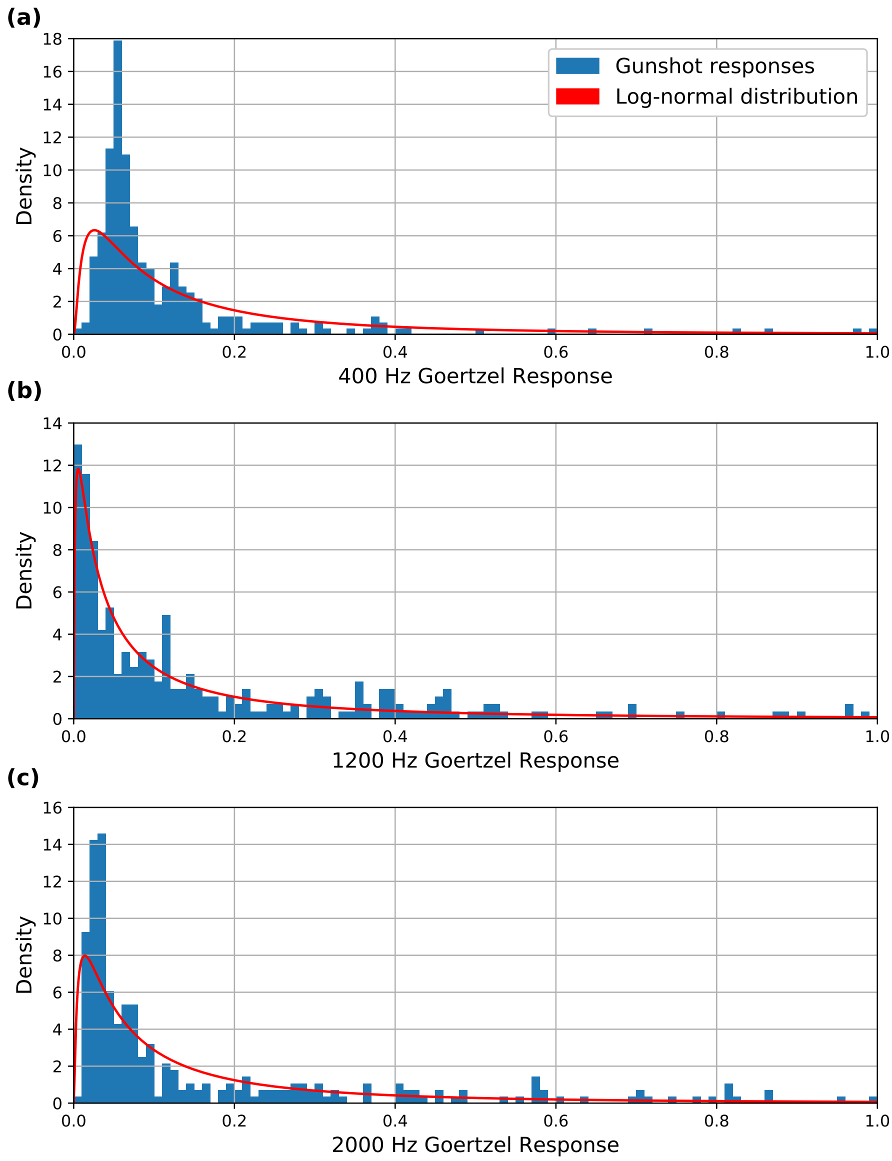


FIGURE 5.5: Histograms showing the densities of different response values for the impulse states of the full dataset of gunshot recordings collected in 2017. The three Goertzel features were centred at (a) 400 Hz, (b) 1,200 Hz and (c) 2,000 Hz. Each feature has been fitted to a log-normal distribution.

AudioMoth. Each year, the data collection built up the available recordings for training the HMM and better representing all possible variations in gunshots a deployed device could hear.

The Tapir mountain reserve features steep, varying terrain which has a significant effect on the rate which the various frequency components present in a gunshot decay. Natural features of the terrain such as hills and valleys can either block the spread of the muzzle blast or even carry it further. By recording at a various distances and in a variety of different terrain types including atop peaks, in valleys, and in hill shadows, the probability distributions are able to reflect a wider array of possible gunshots sounds.

Features (Hz)							
1	2	3	4	Pre-processing	Precision	Recall	F_1
600	1200	1800	-	-	0.75	0.61	0.67
600	1200	2000	-	-	0.77	0.60	0.68
400	1200	2000	-	-	0.79	0.64	0.71
600	1200	1800	3500	-	0.71	0.51	0.59
400	1200	2000	3500	-	0.77	0.63	0.70
600	1200	2000	STDE	-	0.66	0.46	0.54
400	1200	2000	STDE	-	0.70	0.48	0.57
400	1200	1800	STDE	-	0.65	0.49	0.56
400	1200	2000	-	TEO	0.69	0.55	0.61
400	1200	2000	3500	TEO	0.66	0.52	0.58
400	1200	2000	STDE	TEO	0.70	0.38	0.49

TABLE 5.1: Algorithm accuracy metrics for various features given to the HMM. These features include different frequency components, STDE and, frequency bands processed using the Teager operator (TEO). Three bands at 400, 1200 and 2000 Hz achieved the highest F_1 score (0.71) and were used in the deployed model.

5.1.5.1 Performance

In 2017, recordings were collected at various distances between 0 and 800 metres using both a 12-gauge and a 16-gauge shotgun. These recordings were made at both night and day, and formed the dataset used to quantitatively assess the accuracy of the gunshot detection algorithm. This dataset was manually analysed and each four-second recording was classified as either containing a gunshot, silence or significant background noise. In total, 1,170 positive recordings were contained within the dataset.

Using this training set, the detection algorithm achieved a precision of 0.79, a recall of 0.64 and an F_1 score of 0.71 (see Table 5.1). Overall the true positive rate (TPR, equivalent to the recall) was 0.64, whereas the false positive rate (FPR) was 0.015. The lower recall is due to the increased false negative rate (FNR) of long-distance gunshots. The amplitude of these gunshots is significantly reduced by the time they reach the sensor, making it easier for them to be drowned out by background noise. The shots which remain audible are often missing the majority of the higher frequency components which are used to identify the muzzle blast state.

The test set contained 2,970 negative recordings which covered expected background noise as well as a variety of likely false positives. Any sound source which produces a sudden impulse is a possible source of false positive responses, including snapping branches, bird calls and close insect calls. Of these 2,970 recordings, the algorithm achieved a FPR of 0.08, incorrectly classifying insect calls and loud human speech as gunshots 240 times. In the final deployment context, sensitivity to human voices is less of an issue. The presence of human voices within the reserve is extremely unlikely, and will only occur as a result of researchers deploying or retrieving the sensors or illegal

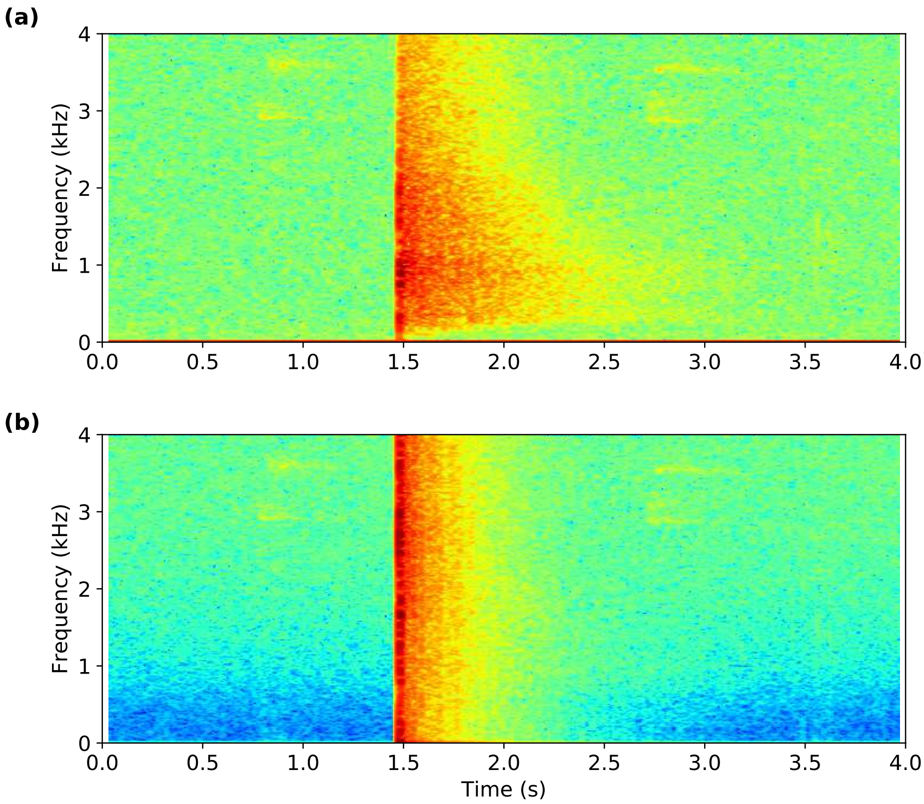


FIGURE 5.6: Two spectrograms of the same gunshot recording: (a) a standard spectrogram and (b) with the TEO applied, emphasising the muzzle blast while erasing the signature decay pattern.

intrusions the sensors are attempting to capture anyway. The triggering insect calls were recorded close to the sensor, and their features were not adequately represented by the noise state. Expanding the distributions further to cover this representation gap could reduce the FPR of the algorithm.

Shown in Table 5.1, the accuracy of alternate features (including short-term delta energy and various frequency bands) was assessed and used to inform the final design. As well as these features, pre-processing the samples using the Teager energy operator (TEO, see Section 2.4.1.7) was also tested. The TEO is design to enhance impulses in signals, filtering out non-impulsive noise in the process. While the TEO makes impulses more pronounced, it frequently erased the decay of the impulse as well as the surrounding background noise (shown in Figure 5.6). While this noise removal cleans the recording and can make it easier to detect the impulse, the missing decay pattern is needed to distinguish between a gunshot and all other impulse-based acoustic events. For this reason samples processed using the TEO produced lower detection accuracies.

Running on an underclocked AudioMoth, the algorithm took 1070 ms to complete, consuming 6.48 mA while it did so. Given this level of energy consumption, an AudioMoth deployed with standard AA-cell batteries, which possess 3,500 mAh of capacity, have a predicted lifespan of 53 days. However, this can be increased to achieve the required 365

Shot distance (m)	100	200	300	400	500	600	700	800	900	1000
Trial shots	12	22	34	34	49	18	31	21	13	18
Detected shots	12	22	33	29	23	5	14	0	9	1
TPR	1.00	1.00	1.00	0.94	0.54	0.25	0.48	0.00	0.67	0.08

Table 5.2: Results from the 2018 test deployment, showing the TPRs for shots fired with increasing distances between the source and the sensor between 100 and 1000 m. Shots fired at distances greater than 500 m often experienced amplitude drop-offs to below 60 dB, resulting in them becoming undetectable by the algorithm.

days by deploying them with higher capacity 6-volt batteries (26,000 mAh), typically used for handheld lanterns, extended this prediction to 398 days.

In terms of storage usage, the algorithm's 0.08 FPR would produce 145 false positive responses per hour. This is relatively high number of false positives, and would be unsuitable for a system transmitting alerts about gunshots in real time. However as each 8 kHz recording is only 63 KB in size, the device would use just 110 MB each night on average, which is a sufficient level of consumption for a deployment aiming to log events. It is also a significant improvement on a continuously recording PAM, which would require 660 MB of storage each night of deployment. With a 32 GB card, a PAM would reach storage capacity in just 48 days, falling far short of the 32,000 mAh battery's capacity and the 365-day requirement.

5.1.5.2 Deployment

To test each iteration of the gunshot detection algorithm in the field, algorithm-equipped AudioMoths were deployed in transects through the forest, approximately 200 m apart. Once deployed, 65 gunshots were fired along the transect using a 12-gauge (Baikal MP-18EM-M) and 16-gauge (Rossi single shot) shotgun. The first iteration of these tests was carried out in Pook's Hill reserve, a private forest reserve immediately adjacent to Tapir Mountain, the intended final deployment location. The first deployed version of the algorithm achieved a TPR of 0.57 on gunshots fired up to 800 m from the sensors across a variety of terrains.

With each set of tests, AudioMoths were deployed in pairs, with one device triggering using the latest iteration of the algorithm while the second acted as a PAM, recording constantly. As all gunshots during these tests were controlled and their exact times recorded, any gunshots missed by the detecting device could be retrieved from the PAM's recordings and used as part of the dataset which formed the next iteration of the algorithm's probability distributions. This iterative approach was used to fill in any gaps in the detection algorithm's acoustic definition of a gunshot. The probability distributions used to define possible observations in each state of the HMM were reconfigured after each test, using this new recording data.

Figure 5.7: AudioMoths equipped with 6 V batteries, ready for deployment in Belize. Each unit consisted of four pieces of laser-cut plastic, a battery and an AudioMoth, held together using two cable-ties. These units were protected by capped plastic pipes and acoustic membrane stickers.

In 2018, the algorithm was tested using more transects, this time in locations which represented a variety of possible terrain scenarios within the Tapir Mountain reserve such as hills between the source and sensor, valleys carrying the sounds further, and open clearings within the forest. In total, 252 gunshots were fired from distances between 100 and 1000 metres from a deployed sensor. The results of this series of tests are shown in Table 5.2. Gunshots within 500 m of a sensor were detected with a TPR of 0.84. For greater distances, factors such as terrain, wind direction, and foliage had pronounced effects on detection accuracy, producing results such as 900 m gunshots achieving a significantly higher TPR than those at 800 m.

The overall effect of the various, unpredictable variables was the increase or decrease in the amplitude of the gunshot at the sensor deployment location. Because of this, the effects of these factors at each location could be recorded by measuring the amplitude of each gunshot. When gunshots fell below 60 dB, the algorithm was unable to detect them as it was unable to pick the sound out of the background noise. This background noise varied between 33 and 55 dB, consisting mostly of wind and insect noise.

The transect trials assisted in the development of the algorithm and understanding the logistics of large-scale deployment in the area. However, the final deployment was in Tapir Mountain itself and began in early 2018. The aim of the deployment was to quantify the illegal hunting known to be occurring in the reserve. The deployment consisted of 79 devices deployed at specific locations across the 25.4 km^2 reserve (the logic behind the number and location of these devices is explained in Section 5.1.5.3).

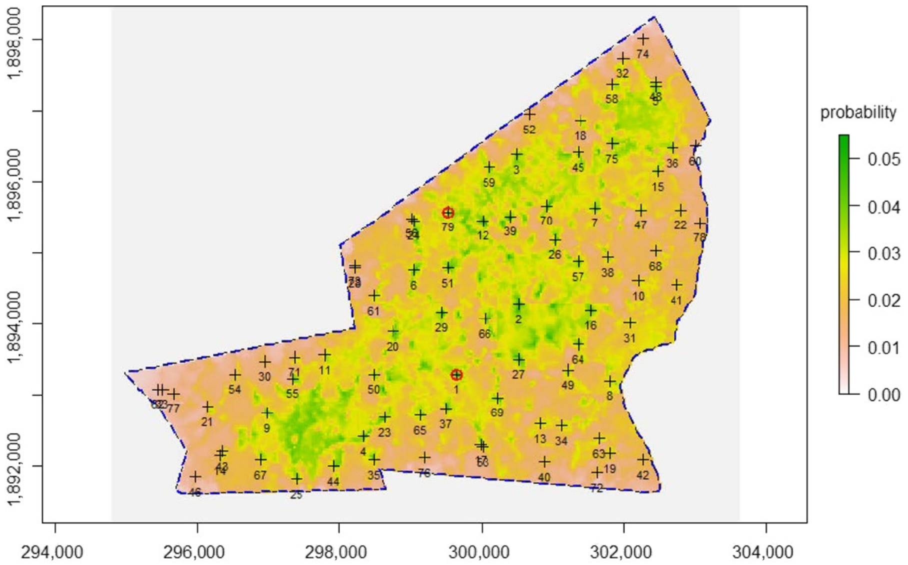


FIGURE 5.8: A map showing the 79 optimal sensor deployment locations within the Tapir Mountain reserve, calculated using a greedy heuristic-based algorithm, developed by Piña-Covarrubias et al. [2018]. Reproduced with permission.

In order to protect the sensors and the large batteries, they were held together using several pieces of laser-cut plastic, housed inside locally-acquired piping (see Figure 5.7). Each pipe was capped at both ends, with a single drill-hole left for the microphone, covered by a waterproof acoustic membrane. Because these short segments of pipe were locally available in the deployment country, the cost of the deployment was reduced as no import fees were paid on bulky bespoke cases. To reduce the sheen of the plastic pipes and provide extra camouflage, each case was coloured using spray paint. In total, each case unit cost approximately £13.50. Allowing conservation researchers to protect a large quantity of monitoring equipment while remaining realistically affordable.

5.1.5.3 Sensor positioning algorithm

The declining costs of ground-level monitoring allow for large-scale deployments not previously possible. Large areas such as the Tapir Mountain reserve require a substantial number of monitoring devices to provide sufficient coverage. As the aim of the deployment was to quantify illegal hunting across the entire reserve, there could be no gaps in the monitoring coverage. However, in order to prove the effectiveness of acoustic detection algorithms on low-cost monitoring devices, this had to be done within a budget realistic within the field of conservation research. Even with low unit costs for monitoring devices, a naïve grid deployment would result in either redundant devices or monitoring gaps due to terrain and foliage altering the effective distance of the sensor and algorithm. This effect is shown in Table 5.2, where the algorithm accuracy varied significantly when the AudioMoth was further than 500 m from the gunshot source.

In response to the decline in algorithm effectiveness, monitoring devices were placed optimally using terrain data, a sound propagation model, and a placement algorithm which has been described in detail by Piña-Covarrubias et al. [2018]. The sound propagation model was used to simulate the spread of gunshots fired from a series of locations evenly spread across the reserve, taking into account the terrain. The placement algorithm assumes that gunshots are equally likely to occur at all locations. This sound map is then used by a greedy heuristic which places sensors in order to achieve complete coverage of the reserve using as few devices as possible. Locations with minimal obscuring terrain require fewer devices to monitor them, whereas more obscured locations require additional devices. This attempts to find an globally optimal placement map which clusters devices in priority locations with low detection probability to achieve coverage without blindspots. The model predicted that 79 algorithm-equipped AudioMoths were required to achieve this level of monitoring coverage. The placement of these devices is shown in Figure 5.8.

5.2 Gunshot detection neural network

In order to benchmark the gunshot detection method described in Section 5.1, a CNN was used to provide a comparison to a technique which has found widespread use in the field of acoustic detection. CNNs are commonly employed to analyse acoustic datasets with a high level of accuracy, operating under restrictions including limited training data [Kiskin et al., 2017] and restrictive hardware constraints [Zhang et al., 2017].

Three CNNs were trained and tested, each using the same four seconds of audio as the HMM as an input, pre-processing these samples differently for each network. The operation of CNNs is described in detail in Section 2.4.3. The first CNN converted each set of samples into a monochrome spectrogram, the second converted them into mel-scale frequency banks, and the final CNN processed the banks further, using a discrete cosine transformation (DCT) to create mel-frequency cepstrum coefficients (MFCCs).

Both the mel scale and MFCCs are explained in detail in Section 2.4.1.4, whereas the steps required to generate standard spectrograms are described in Section 2.4.1.1. The mel scale was chosen for this application as it is designed to be more discriminative at lower frequencies within the range of human hearing [Muda et al., 2010]. The frequency components of a gunshot occur within this range.

These three feature types decrease in size, with each one requiring fewer parameters to detect the defining gunshot features. MFCCs are often used for acoustic detection, however a substantial amount of information is lost during their generation, which can result in valuable information being discarded and gunshots becoming undetectable. Each CNN attempts to aid detection and reduce network complexity by discarding different amounts of input information, without losing necessary identifying features.

5.2.1 Training set collection

The dataset used to generate the gunshot HMM's probability distributions was also used as a training set for each of the three CNNs. While the HMM was trained to identify the gunshot states within a set of samples, the CNNs aimed to classify entire four-second recording as containing a gunshot or not. This distinction meant as well as gunshots recorded across different terrains or distances, the dataset had to represent gunshots occurring at different points within the recording. In order to do this and increase the overall size of the training set, recordings within the set were duplicated and offset in time. This duplication was done equally across all distances collected to avoid biasing the representation towards certain distances.

5.2.2 Feature extraction

Each of the three CNNs used a different pre-processing technique to transform the four-second recordings into images which represent their frequency composition before training/classification. These techniques are standard spectrograms, mel-scale frequency bank images and MFCCs. Converting audio recordings into 2D images representing their frequency composition while weighting lower frequencies is a common technique in speech recognition and keyword detection. This is because, for short duration frames, a speech (or gunshot) signal can be considered stationary. The stationary frames represent the overall signal while enabling the use of the many deep learning techniques developed for image processing.

The three feature types are shown for the same gunshot recording in Figure 5.9. All three were monochrome, with the intensity of each pixel used to represent the amplitude of a given frequency or frequency band. All feature images are 390 pixels wide, each being windowed in the same manner as part of processing. The spectrograms are 257 pixels high whereas each frequency bank image is 40 pixels high. The MFCCs used 12 non-linearly spaced filters, resulting in an image height of 12 pixels.

5.2.3 Network design

Each CNN features three convolution layers, followed by a flattening layer to transform the multi-dimensional tensors used in the convolution stages into a one-dimensional vector, and then finally a fully-connected layer to allow the network to learn non-linear combinations of the convolution features. A small number of convolution layers was chosen to define the features of the small images without over-fitting to the training set. Paired with each convolution layer is a pooling layer and an activation layer. The pooling layers are included to reduce the spatial size of the representation at each step. This has the overall effect of reducing both parameters and computation required. In

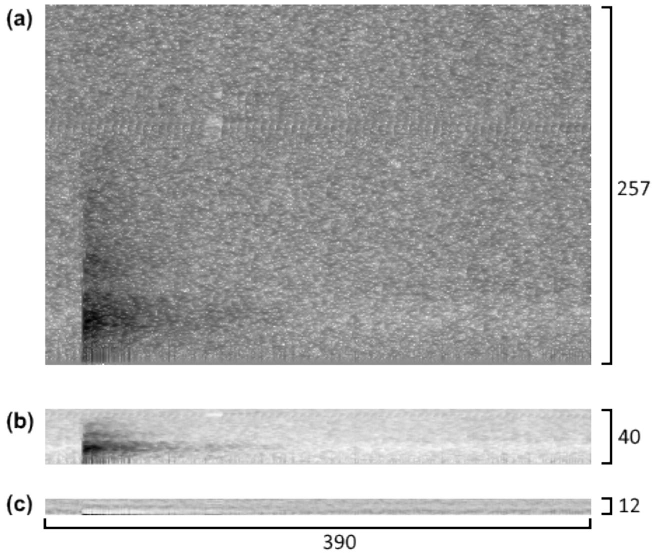


FIGURE 5.9: Three plots showing the same four-second gunshot recording represented by each of the processed images: (a) a standard monochrome spectrogram; (b) mel-scale frequency banks; and (c) mel-frequency cepstrum coefficients.

this implementation the ReLU activation function is used in these layers. As well as introducing non-linearity, the ReLU function helps to reduce the effects of the vanishing gradient problem and improve training time (see Section 2.4.3.1).

The design of the CNNs had to reflect the limitations of constrained hardware and focus on minimising the size and complexity of the network as much as possible. One method employed to do this was replacing standard convolution used by most CNNs with depthwise separable convolution (see Section 2.4.3). DS-CNNs are more efficient than traditional CNNs, reducing both the computations required to convolve the data at each layer and the number of overall parameters. This reduces both the size and the complexity of the network [Huang et al., 2017b].

The primary limitation imposed by running a neural network on constrained hardware is the memory available to store partial results. Zhang et al. [2017] use depthwise separable convolution to aid the development of acoustic neural networks for deployment on microcontrollers. They were able to show that depthwise separable convolutional neural networks (DS-CNN) running on constrained hardware are able to achieve greater levels of accuracy than deep neural networks on identical hardware, while requiring ten times less memory. This is vital on hardware like AudioMoth, where even with additional memory provided by an external SRAM chip, just 256 KB is available for both the detection algorithm and the simultaneous sample collection.

	TPR	FPR	FP per hour	Precision	Recall	F_1 score
Spectrograms	0.86	0.13	228.0	0.85	0.86	0.85
Freq Banks	0.80	0.04	80.7	0.97	0.80	0.88
MFCC	0.78	0.10	185.8	0.86	0.78	0.82
HMM	0.64	0.08	145.0	0.92	0.64	0.75

TABLE 5.3: Detection metrics for three different pre-processing techniques used before training three convolutional neural networks: standard spectrograms, mel-scale frequency banks and MFCCs. Also included for comparison is the same metrics for the HMM. The accuracies were based on the detectors' ability to discern between background noise and gunshots up to 800 m away.

5.2.4 Performance

The three CNNs were tested alongside the gunshot HMM, comparing various accuracy metrics including the expected number of false positive responses per hour, the precision, the recall, and the F_1 score (see Table 5.3). Based on these metrics, the mel-scale frequency bank features produced a network which performed the best of the CNNs and also outperformed the HMM, achieving an F_1 score of 0.88. Its FPR was the lowest and was approximately half that of the HMM. This resistance to false positives resulted in a high level of precision and thus a high F_1 score.

As well as binary classification, the same mel-spaced frequency bank features were used to train a CNN which detected gunshots at different amplitudes. Using classes based around peak amplitude rather than distance was done due to factors such as terrain, foliage, and weather conditions causing overlap between features of different distances.

The classes take the peak amplitude of the file (amp_{peak}), and use the maximum possible amplitude which can be represented by a 16-bit sample (amp_{max}) as a reference to obtain the peak amplitude in decibels:

$$amp_{dB} = 20 * log_{10}(amp_{peak}/amp_{max}) \tag{5.1}$$

The file's class is then found by taking less than -20 dB as *low* amplitude, greater than -20 dB as *medium* amplitude, and greater than -15 dB as *high* amplitude.

The confusion matrix for this model variant is shown in Table 5.4. Despite the success of the original mel-spaced frequency bank CNN, splitting the gunshot class resulted in a less accurate sensor, as low amplitude gunshots were detected with a high FNR (0.17).

Despite the high level of binary classification accuracy attained with the frequency bank CNN, it was achieved at the cost of feasibility in terms of speed. Both the HMM and each CNN were compared in terms of completion time on a high-end desktop machine. The frequency bank CNN required 0.52 seconds to classify four seconds of audio. This

	Low	Medium	High	Background
Low	0.41	0.40	0.01	0.18
Medium	0.11	0.67	0.10	0.12
High	0.02	0.21	0.76	0.01
Background	0.04	0.02	0.02	0.92

TABLE 5.4: Confusion matrix of the mel-scale frequency bank CNN. Gunshots were classified as either *low*, *medium* or *high*, based on the peak amplitude within the file.

included the pre-processing step of converting the samples into mel-scale frequency bands and then the subsequent classification. In the time this method took, the HMM-based classifier was able to analyse and classify 300 recordings, taking approximately 1.75 ms for each classification.

On AudioMoth hardware, the HMM is able to run in 1070 ms. As this is less than the two seconds required to collect the next set of samples the AudioMoth is able to run the algorithm in real time. Assuming the difference in completion time on AudioMoth hardware is similar, this would not be the case for the CNN. This would lead to gaps in the listening period to allow the detection algorithm to catch up, which risks missing gunshot events.

The significant difference in completion time is likely due to the computational complexity difference between each algorithm. As part of the pre-processing stage, an FFT is applied several times to samples to calculate the frequency composition. This step has a computational complexity of $\mathcal{O}(L \log L)$, where L is the length of each overlapping FFT frame ($L = 800$). After pre-processing, the CNN applies convolution to the inputs at each of the three convolution layers. This convolution has a complexity of $\mathcal{O}(WHkk)$, where W and H are the width and height of the inputs (390 x 40 for frequency bank images) and k is the dimensions of the kernel used to apply the convolution (this changes at each layer of the network).

Contrasting this, the HMM-based algorithm first pre-processed the samples using the Goertzel algorithm, which operates with a computational complexity of $\mathcal{O}(L)$. Next, the Viterbi algorithm was used to calculate the most likely sequence of states represented by the processed samples. This algorithm operates with a complexity of $\mathcal{O}(NS^2)$, where N is the number of observations ($N = 256$) and S is the number of states used by the model ($S = 5$) [Backurs and Tzamos, 2017]. Both the pre-processing and the analysis stages of the CNN exceed the original gunshot detection algorithm in computational complexity, explaining the large difference in completion time.

Other research has been able to implement simple CNNs on constrained hardware and achieve high levels of accuracy. The DS-CNN developed by Zhang et al. [2017] for keyword-spotting runs on a Cortex-M7 microcontroller thanks to its low network size and memory requirements. Trained using the extensive Google Speech Commands Dataset, the network is able to classify one-second recordings as one of thirty word classes with 94.9% accuracy, while requiring just 182.2 KB of memory. This implementation uses MFCCs to minimise the number of parameters required and thus produce a smaller, faster model. However, the information lost when performing the DCT on the frequency banks, while unnecessary for discerning between the words in human speech, is required for maximising accuracy of the gunshot detector, given the training dataset. This is shown by the results in Table 5.3. The larger input images used by the mel-scaled frequency banks result in a larger overall model compared to the keyword-spotter.

5.3 Summary

This chapter presented an algorithm which builds on the work of Chapter 4, adding additional complexity to the algorithms discussed to produce a detection algorithm which uses a five-state HMM to react to the sound of gunshots in a tropical forest. This algorithm used the amplitudes of three frequency bands as observations for the HMM and classified four-second audio clips based on whether or not these observations were caused by the three defined gunshot states (impulse, decay and tail).

The requirements of the algorithm included a yearlong, unattended deployment on AudioMoth. This resulted in a greater focus on energy efficiency, leading to underclocking the processor and reducing the sample rate of collection. The second requirement was constant listening, which underclocking had a significant effect on. The algorithm had to complete in under two seconds in order to maintain constant listening throughout the device's 12-hour listening schedule. Despite this limitation, the gunshot detection algorithm applies three Goertzel filters to the audio and then runs these triples through a HMM, achieving a TPR of 0.64 on gunshots recorded up to 800 m away.

To provide context to the accuracy of the detection algorithm, a three-layer CNN was developed to benchmark the algorithm. CNNs are a commonly used deep learning technique for bioacoustic detection, providing a comparison to the state of the art within the field of conservation. While this detection method achieved higher levels of accuracy than the HMM, it was too slow to run on the AudioMoth in real time.

After development and testing, both with a pre-recorded dataset and in the field, the algorithm was deployed on 79 AudioMoth devices. In order to efficiently place the devices such that the minimum number were required while still achieving sufficient coverage, a placement algorithm was used, based on a sound propagation model of the reserve. The AudioMoth deployment using these locations began in Spring 2018 and aims to gather a year's worth of quantifiable poaching evidence within the Tapir Mountain reserve.

AudioMoth, deployed with the detection algorithm, presents a case for reactive recording in acoustic monitoring of gunshots as a tool for measuring levels of poaching within protected areas of forest. Similar devices have been used to monitor poaching in protected reserves, using passive monitoring devices to record in the reserve and then analyse the data after collection [Astaras et al., 2017]. The methods described in this chapter show the benefits of an alternative which can be deployed for longer on lower capacity batteries and produces a dataset which requires less processing after collection.

The following chapter covers the accessibility of acoustic monitoring, analysing the current public perception of AudioMoth through both a questionnaire and a series of interviews. It also reports on the tools used to improve the accessibility of both AudioMoth and the detection techniques described in the preceding chapters.

Chapter 6

Improving the accessibility of acoustic monitoring tools

> As I see it, AudioMoth is about having the freedom to do things yourself, how you need them done.
>
> Anonymous AudioMoth user

No matter how useful or functionality-rich a monitoring device is, it has no effective utility if no-one is able to access this functionality. Enabling more people to make greater use of a piece of conservation technology allows a wider array of research questions to be asked. For a piece of technology, accessibility can be both in terms of cost and usability.

Improving financial accessibility includes reducing unit costs enough that a device becomes viable for deployments with a large quantity of sensors. This can be to cover a large spatial scale or for data collection redundancy. Often large areas of land are threatened by a specific threat (such as poaching or deforestation) and limited funding prevents sufficient quantities of high-cost monitoring equipment from being purchased to monitor and quantify it. This is a common occurrence in developing countries, which are often where conservation monitoring is required most. In these locations, even small-scale acoustic monitoring with most commercial sensors is impossible due to their high unit cost. Improving financial accessibility can allow local communities to monitor their own natural resources.

Accessibility in terms of usability or user-friendliness is important for technology developed for users with a wide range of technical abilities. Overly complex technology can turn off potential users who perceive the benefits of using it to not be worth the initial effort required to learn and incorporate it into established methods. This can result in a reduced the adoption rate.

This chapter presents the assessment of AudioMoth as well as acoustic monitoring technology in general in terms of these forms of accessibility. It includes both quantitative and qualitative analyses, using the results from a questionnaire and a free-form interview for each. As well as these assessments, this chapter also details work done to improve the usability of AudioMoth with interface and development tools.

The questionnaire and interview prompts were designed by the author, with help from Stephen Snow. Questionnaire distribution and analysis was done by the author. The interviews were carried out solely by the author, whereas the thematic analysis following their completion was done in a group by Stephen Snow, Andrew Hill, and the author.

6.1 Questionnaire

To gain an understanding of the demographics of the AudioMoth userbase, their applications, and their justifications for purchasing it over the wide range of alternatives (both commercial and self-built), a short questionnaire was devised. The questionnaire covered a range of topics including their initial awareness of the device, the intended application of the devices they purchased, and the usability of the platform as a whole. The questionnaire is included in full in Appendix A.

The responses could be used to judge the success of AudioMoth and other available sensors in providing for the requirements of researchers wishing to use acoustic monitoring. An analysis of these results could then be used not just to better understand AudioMoth, but to understand how users interact with all acoustic technology.

6.1.1 Respondent demographics

The questionnaire was sent out to people who had previously signed up to the mailing list found on the Open Acoustic Devices (OAD) website. The stated purpose of this mailing list was to inform those on it about updates to the acoustic monitoring software and hardware developed by the OAD team (see Chapter 3 for more information on OAD) as well as news relating to the project as a whole. This meant that those receiving the questionnaire were likely to have a strong interest in the application of acoustic monitoring. Of those contacted, 75 completed the questionnaire.

While 75 respondents is not the complete population of AudioMoth users, the sample size relative to the customer base is considered appropriate. The ideal sample size for an online survey of a population which is constantly changing in size is difficult to nail down. For constantly growing populations like the customer base for a device, between 30 and 500 samples, representing approximately 10% of the overall population is considered

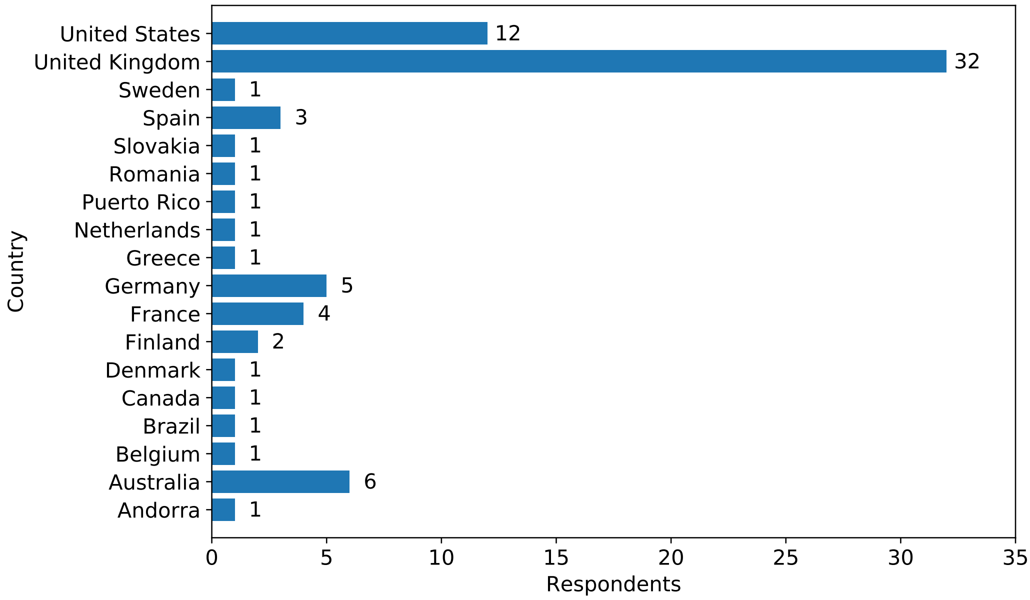


FIGURE 6.1: Questionnaire respondents by location. Due to the method which the respondents were found (through an English-language mailing list), there is a clear bias towards predominantly English-speaking countries.

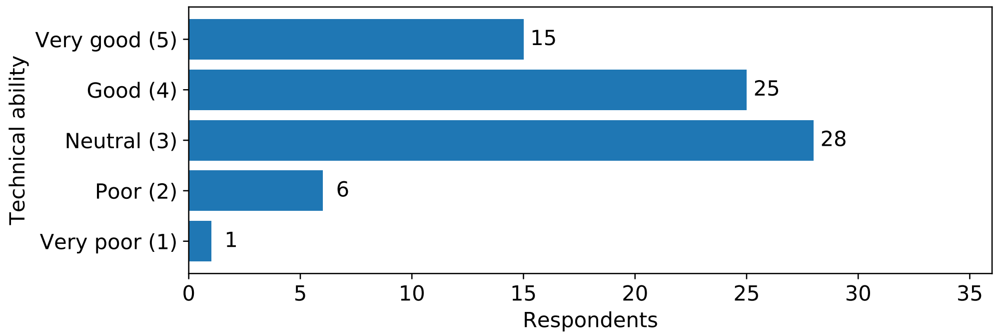


FIGURE 6.2: Respondents were asked to rate their perceived level of technical expertise on a scale of 1 to 5.

appropriate [Hill, 1998, Sue and Ritter, 2012]. As of March 2019, 687 customers had purchased one or more AudioMoth devices, meaning 75 fulfils these criteria.

The 75 questionnaire respondents originated from 18 different countries (see Figure 6.1). As the Open Acoustic Devices website is entirely in English there is an strong bias towards countries where English is the official language (51 out of 75 respondents were from the United States, United Kingdom, Canada or Australia). However, this bias is also inherent to the entire AudioMoth userbase as the majority of publicity events for the project have occurred in the UK, the US, and Australia, and most AudioMoths purchased have been shipped to English-speaking countries.

Of those who responded, 25% worked within academic research, 32% in the private sector, and 47% were unaffiliated citizen scientists or conservation enthusiasts. Academic researchers who responded ranged from undergraduates to professors. As a result, the questionnaire provided a wide spread of opinions across the different forms of ecological research. With enthusiast conservation and observation groups, ecology has a uniquely high number of amateur research groups when compared to other scientific fields, requiring tool designs to consider a wider spectrum of scientific and technical abilities. Getting the opinions of users performing both types of research allowed the results to be used to assess the effectiveness of acoustic monitoring technology in fulfilling the requirements of the entire conservation community.

Respondents were also asked to rate their perceived level of technical expertise (see Figure 6.2). Within conservation there is likely to be a large amount of variation in terms of technical ability amongst users. When introducing technology to an established scientific protocol such as acoustic data collection, it is important to consider both the expected and the minimum level of technical expertise for the average user of that tool or protocol. Introducing technology beyond the technical capabilities of a data collection team could easily negate any potential efficiency gains as resources are expended providing support. Approximately 90% of respondents rated their level of technical expertise either *neutral*, *good* or *very good*, with the remaining 10% considering their abilities *poor* or *very poor*. While this low-ability group makes up a small proportion of the userbase, it is important to acknowledge that these users exist when developing and evaluating technology for the community as a whole. Developing for the bottom 10% provides for the top 90% as well.

6.1.2 Responses

After collecting demographic information, the questionnaire focused on two main areas: why the AudioMoth was chosen for the user's application and the user experience of the device and all supporting software. Why the AudioMoth was chosen can be used to assess how successful the device is for filling the intended niche as a low-cost, large scale monitoring tool.

As part of the first set of questions respondents were asked why they chose to purchase AudioMoth devices over competing acoustic monitoring equipment such as the SongMeter, the Frontier Labs BAR, or a homemade device (see Figure 6.3). The response to this heavily favoured both the cost and the size of an AudioMoth as the primary justification for purchase, with these two factors being chosen by 62% of respondents.

Of the justifications provided, these two have the biggest effect on enabling large-scale monitoring projects. When monitoring equipment is cheaper, a larger number of devices can be purchased and it matters less when they are damaged or lost. When large quantities of equipment are used as part of a remote deployment, the likelihood of some level of hardware failure or loss increases significantly.

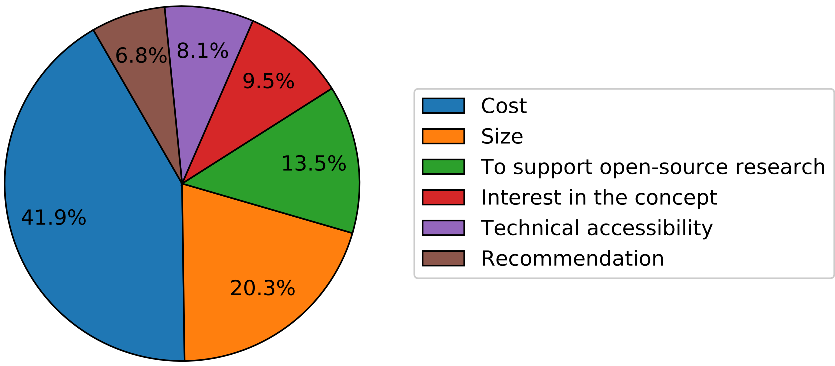


FIGURE 6.3: Pie chart showing the various reasons questionnaire respondents gave for purchasing AudioMoth over other acoustic monitoring equipment. The dominant justifications were the low-cost and small size of the device. Both these factors better enable large-scale deployment.

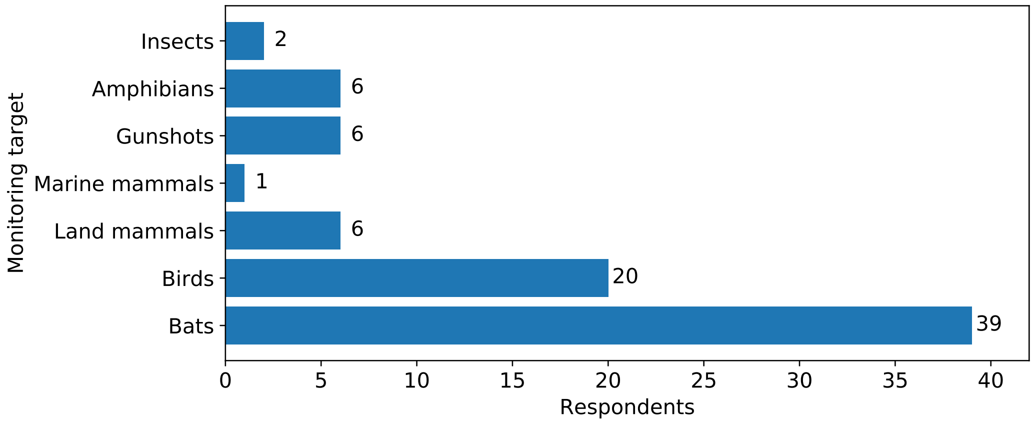


FIGURE 6.4: Targeted species/acoustic events given by respondents. As respondents were allowed to list all intended uses of the device, the total is greater than the number of questionnaire responses.

The detection targets given by the respondents are shown in Figure 6.4. Asking for a user's intended monitoring targets can give the requirements which AudioMoth can fulfil or support users in fulfilling themselves, in order to further satisfy its userbase. For example, a single respondent attempted to submerge their AudioMoth sensors to record marine mammals. Providing support in the form of tutorials for installing additional sensors such as hydrophones, or testing the sensitivity of the device when deployed inside a sealed, waterproof case would assist deployments of this type. Unique deployments have identified under-served applications like this.

Contrasting the results in Figure 6.1, deployment locations shown in Figure 6.5 include a wider range of countries where AudioMoths were deployed. These include four deployments in Africa, despite no respondents purchasing in African countries. While a bias towards English-speaking countries exists with AudioMoth purchasing and awareness,

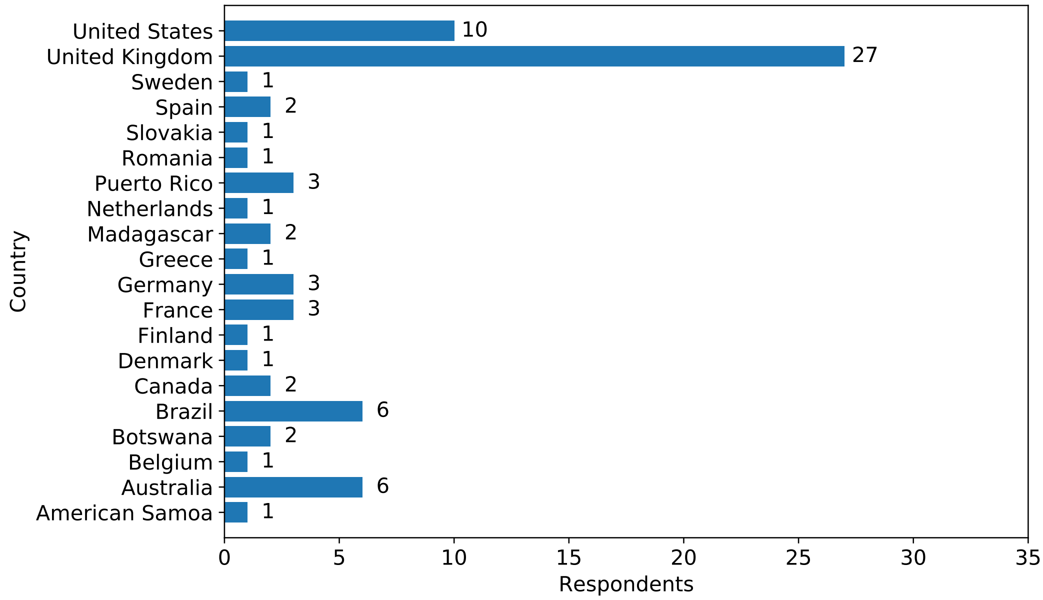


FIGURE 6.5: Respondents by deployment location. Contrasting Figure 6.1, countries outside the UK and US had significantly more deployments than purchases.

deployments do occur outside these countries in locations with greater need of ecological protection. One motivation of AudioMoth's accessibility is to enable users to monitor their own natural resources and increase the ubiquity of acoustic monitoring. This disparity between purchase and deployment location is a signal that further efforts must be put into improving accessibility and awareness of affordable acoustic monitoring tools in these threatened locations. See Section 2.3.2.3 for more information on this issue in the context of conservation technology as a whole.

In terms of user-friendliness, the majority of respondents were content due to the straightforward interface used to interact with the limited features AudioMoth possesses. Aspects of the interface which confused users and were common sources of mistakes have been used to inform future interface designs and support. These minor issues included unintuitive naming conventions and insufficient support documentation.

By competing with acoustic monitoring alternatives on price and physical design, the user interface has been able to remain extremely simple. This simplicity is an important factor in encouraging adoption of new technology, especially when citizen scientists and volunteers are often used to speed up the process of deploying large numbers of sensors. For these projects, a high level of technical knowledge is not guaranteed and rapid education of new users is vital. The minimal confusion with the user interface attests to the success of AudioMoth's simplicity.

6.2 Interview thematic analysis

The final section of the questionnaire asked if the respondent was happy to take part in a short follow-up interview over the phone or Skype. Of the 75 questionnaire respondents, 12 interviews were scheduled and each lasted between 20 and 40 minutes. The standard questions asked as part of each interview are listed in Appendix B. These questions were treated as conversation prompts and further questions were asked depending on where the conversation was taken by the interviewee. The interviews aimed to gain a better insight into the attitudes and opinions of the various types of user which had already invested in AudioMoth.

Due to the nature of free-form interviews, qualitative analysis was required to evaluate the responses. The methodology chosen to do so is thematic analysis; a widely-used technique in many fields for identifying themes in qualitative datasets.

6.2.1 Methodology

A six-step process for carrying out thematic analyses has been developed by Braun and Clarke [2006]. This process was used to break down and analyse the recorded interviews. The first step is for the analysts to familiarise themselves with the data. This includes transcribing the data (in this case the interview recordings), reading through the transcribed dataset, and noting down initial ideas. As part of this first step, analysts must immerse themselves in the dataset, reading it in an active manner (discussing trends, taking notes, etc.) to maximise the chance of identifying patterns.

Step two is to generate the initial codes. A "code" is defined as an element of information which can be assessed in a meaningful way. Common responses and patterns within the series of interviews are a good way of identifying elements likely to make good codes. Code selection was an iterative process, with each member of the analysis team sorting the data separately, to confirm the accuracy of the code classifications.

Once the codes have been generated, they are broadly grouped into a set of potential themes. Themes are found by identifying common relationships between sets of codes. Themes are sometimes referred to as axial codes.

The fourth step is to review the themes, making sure they work with regard to the entire dataset as well as the individual themes fitting well into each one. As part of this step, an analysis map is created, visualising the relationships between the codes and how the relate to each other within the groupings chosen in step three.

Next, the themes are studied and named. The specifics of what each theme tells a reader about the data and the overall story the analysis intends to tell, given these themes.

Finally, a report is produced. To create this report, the overall story is finalised and specific extracts from the data are selected which best reflect the chosen themes and research questions.

The analysis process was carried out by three researchers with an inter-rater reliability (IRR) of 50%. This was achieved by half of all interviews being coded by two or more researchers. The codes were not discussed as a group and all coded interviews were kept hidden from other researchers until that step was complete to ensure open-minded code selection. Once every interview had been annotated at least once, codes were reviewed as a group and a set of 28 was finalised. These codes were grouped into three primary themes, each theme containing between eight and eleven codes.

The themes chosen were "Open-source, open science and community", "Appeal', and "Problems and limitations" (see Table 6.1).

6.2.2 Results

The aim of the thematic analysis was to better understand how people who use acoustic monitoring for conservation research interact with the technology available to them. This is useful for the future of the AudioMoth project as well as for the development of conservation technology as a whole. Identifying under-served niches within the field can result in previously impractical or impossible research questions to become doable. In this section, the themes produced by this analysis (listed in Table 6.1) are described.

6.2.2.1 "Open-source, open science and community"

A recurring topic amongst the interview responses was open science and the open-source nature of AudioMoth. In recent years, open science has become an increasingly prevalent topic within the field of ecology. With the growth of internet communities and a focus on sharing both data and expertise, ecologists and those interested in conservation have come to associate open collaboration with the betterment of the field and their responsibility to feed back into the community. This is reflected in interviewees who went out of their way to seek open-source technologies, rather than using widely-used commercial tools:

> *I was googling open-source wildlife technology and it came up and I thought "Ooh this is interesting". Then I started following the project.*

These quotes highlight people's willingness to engage with open science and shows the existence of a niche within acoustic monitoring technology. AudioMoth fills this niche by being one of few available open-source options for experimentation within acoustic

Code	Individuals
Open-source, open science and community	
Experimentation	8
Housing	8
Hacks	7
Creativity	7
Lay person (attitudes, apprehensions, limitations)	6
Community support, desire to feed back into the community	6
Initial tinkering	5
Citizen science	5
Responsibility (developers versus users)	3
Barriers to entry	2
Ownership and the ethics of open-source	2
Appeal	
Cost	8
Expendable	7
Long-term use	7
Flexibility	6
Usage in the field	6
New opportunities	5
Word-of-mouth	5
Triggering	4
Help/support (developer-driven)	4
Problems and limitations	
Comparison to commercial devices	11
Technical expertise informing use	7
Resilience	7
Noise and audio quality	6
Usability	5
File types and file size issues	5
Security	2
Minimum acceptable performance	2

Table 6.1: Codes applied as part of the interview thematic analysis. The individuals are the number of respondents who mentioned each of the concepts over the course of their interview.

monitoring which don't require building a sensor from scratch. The topic of **experimentation** came up frequently, with eight responses mentioning their intention to do it or listed the ability to experiment as a benefit of AudioMoth.

The Organisation for Economic Cooperation and Development defines open science as "[Making] the primary outputs of publicly funded research results - publications and the research data - publicly accessible in digital format with no or minimal restriction". In terms of conservation, a major motivation for this is to encourage public engagement. Conservation projects which have publicly available publications and results or those which use use citizen science for data collection are more likely to raise awareness of conservation issues. Of those interviewed, five people mentioned intending to use AudioMoth to perform some form of **citizen science**. Examples include:

> *We've been looking for a cheap option because we plan on doing a big citizen science project. Giving them to people so they can record while swimming.*

These responses show AudioMoth's ability to enable public engagement, which can increase the chance of obtaining future funding to study the issue as well as improve the effectiveness of public policy changes made in response.

Due to AudioMoth's development as part of an academic research project rather than from a commercial entity, people have also begun to associate the device with a hacker community. The interviews supported the pervasive power of community and the Six interviewees mentioned **community** in the context of either being a part of it, the AudioMoth representing it or the responsibility of feeding back into it:

> *It's nice people can hack their way through the hardware and make changes. It encourages a hacker community like the Raspberry Pi did.*

Because AudioMoth has been presented to the ecological community as a low-cost tool for acoustic monitoring, this hacker community focuses on low-cost, low-tech solutions for problems such as casing their equipment:

> *I've got some little plastic ice cream containers that good old Aldi were selling which were just the right size for AudioMoth.*

These low-tech solutions mean users perceive a low barrier to entry for both using the AudioMoth and tailoring it to their applications. This has encouraged **creativity** and thus further engagement. This creativity also comes from amateur users who are unlikely to even consider expensive commercial equipment:

> *It's just fun and games. Fiddling with AudioMoth is more of a hobby for me. None of the stuff I do is paid, it's just a bit of fun I do with local groups.*

However, this association with open science and open-source technology can have drawbacks. For many security-conscious people, commercial products which act as a black box are thought of as much safer and more secure than open-source technology. This has shown to be the case with AudioMoth:

> *I work in local government and they're a bit afraid of it all. IT departments see open-source projects as security risks.*

As a result, even if security is not a priority for a bird monitoring device, the open-source descriptor carries negative connotations. Two interviewees specifically mentioned **security** as a primary concern when working with open-source technology, both hardware and software. In response to this, effort should be put into giving options which provide security (protecting files on the storage medium, for example), even when not necessary for the majority of projects. Making sure modules and libraries used by the software are kept up to date and fixing security issues as soon as they are discovered in released applications also helps ensure users' trust. This could help reduce the negative perceptions of open-source technology and provide security-conscious researchers more diverse monitoring tools.

6.2.2.2 "Appeal"

A major focus of the interviews was to discover the drivers of users who purchased AudioMoth. The prevalence of certain motivating factors can help decide what should be prioritised with any future conservation technology.

One motivator which became extremely apparent through the study is **cost**. Researchers with limited budgets and amateurs unable to invest much money into their hobby are extremely price-driven. When alternative devices to the AudioMoth were mentioned, it was almost exclusively done in the context of the AudioMoth's price offsetting the benefits of a commercial device. Users are happy to use the more expensive devices for specific applications, however those applications can be limited for a number of reasons, including users being reluctant to leave high-cost equipment unattended:

> *The commercial ones are great, but that's £900 I'm hanging on a tree. I can't afford not to keep an eye on it.*

Interviewees also expressed how the cost factor is integral to citizen science, as they can loan low-cost equipment to citizen scientists despite the high risk of damage or loss. Alternatively, low-cost equipment can be purchased by the citizen scientist themselves, further engaging them with the conservation process by allowing a conservation enthusiast to own comparable tools to professional conservationists.

> *I've loved being able to give devices to my friends and just see what birds are in their gardens.*

This also ties back to the ideas described in Section 6.2.2.1, as public engagement is one of many major motivators for open science.

Interviewees often mentioned sending each other devices and sharing insights. A lot of the knowledge which exists about monitoring technology spreads through **word-of-mouth**. The ability to easily share the devices themselves (due to both cost and size), has encouraged this word-of-mouth spread:

> *I found out about [the AudioMoth] from a friend at my birding group.*

Knowing how information proliferates through conservation groups through word-of-mouth, directly interacting with these groups either face-to-face or on social media is clearly an effective method for improving engagement with technology-based monitoring. Resources which encourage sharing, such as short videos and images which convey the utility of the device are also important for enabling further awareness.

6.2.2.3 "Problems and limitations"

As part of the interview, interviewees were guided towards discussing issues and limitations of AudioMoth. Responses frequently came in the form of direct comparisons to commercial, off-the-shelf devices, with very few interviewees having attempted building their own. The reason for this was predominantly **technical expertise informing use**, with seven interviewees making statements which imply their technical skill has directly informed how they use monitoring technology and which devices they prefer:

> *I am an end user. I'm not going to program any equipment I buy.*

While the idea of flexible technology which can be tailored to a specific application is an appealing factor for many users, a large number are unable to do this tailoring themselves. As a result, many find waiting on the community to develop an appropriate case or relevant functionality a limitation of AudioMoth.

Other limitations which were commonly mentioned included **resilience** (7 mentions), **noise and audio quality** (6 mentions), and **file issues** (5 mentions). Interviewees claimed they were often willing to overlook limitations of a piece of technology due to the unique niche which it fills. High-fidelity recorders come with a high-cost, but often a high level of recording quality is required and this expense is justifiable. For AudioMoth, the audio quality is worse than many commercial devices, but benefits such as its size, price and flexibility make it ideal for many other applications:

> *I wouldn't say the resolution is perfect. It's worse than a commercial system but it's appropriate for our goals.*

When users make exceptions for these issues, they do so with an idea of **minimum acceptable performance**. This concept refers to the minimum requirements of a tool which they are willing to accept and for low-cost devices these requirements scale directly with price. This balance of acceptable performance against price is especially sensitive for low-cost equipment, because affordability is the quality all other trade-offs are made in aid of.

6.2.3 Discussion

Overall, the interview responses show that AudioMoth exists within its own area of the acoustic sensor market, which historically consisted of just bespoke devices. This niche was previously limited due to the lack of specific technical expertise within the field of ecology. Despite this inability to fully develop open-source technology, the scientific community in general has begun to favour tools and techniques which lend themselves to open science. The interviews reflected this and showed that some users are willing to incorporate additional hardware, develop cases, and alter code within the bounds of their abilities, given the opportunity. When new opportunities are provided, these users are ready and willing to take advantage of them.

This analysis has shown that there is a niche within conservation technology as a whole for tools which lower the barrier to entry just enough for ecologists to take part in open science development. This niche exists alongside high-cost, high-fidelity sensors like Wildlife Acoustics' SongMeter range, without directly competing with them.

The development of both AudioMoth and new technologies should be done with respect to the minimum acceptable performance, given the manufacturing price. Additional functionality and performance can increase the overall cost to the user and result in a net loss in terms of viability as a low-cost tool. Remaining close to the this minimum performance keeps unit costs low, keeps the tool within the targeted niche, and enables large-scale deployments with high risk of equipment loss.

6.3 Acoustic detection algorithm software library

As well as open-source hardware, all firmware and supporting software for the AudioMoth has been made freely available online. These open-source repositories have extensive documentation and aim to encourage users to manipulate the basic firmware to implement additional functionality specific to their application.

One goal of AudioMoth was to be applicable to a wide array of acoustic monitoring projects, maximising its presence in the niche of low-cost monitoring. The value of the device to a specific user could increase if additional features such as wireless communication or lots of specialised detection algorithms were developed and provided as a standard feature. However, the time and effort required for their development, as well as the necessary hardware required to implement them would also increase the cost of each device. Low unit cost is integral to the appeal and utility of AudioMoth. Sacrificing affordability reduces the overall accessibility of the device, despite gaining functionality. Specialised hardware and additional functionality will likely be unnecessary for a large number of applications.

Encouraging bespoke development by users is one way to maintain flexibility and accessibility without this sacrifice. However, as shown in Section 6.2.2, AudioMoth users often do not possess the technical expertise to confidently make changes to the underlying firmware of the device to implement functionality they desire. Bridging the gap in technical ability to allow users to make these changes without direct assistance is key to producing an acoustic sensor which utilises detection techniques (such as those discussed in Chapters 4 and 5) and is widely accessible. To achieve this, a C software library was developed which provided users with implementations of DSP functions and filters likely to be used in the development of various acoustic detection algorithms.

Using these high-level functions, the algorithms discussed in previous chapters could be implemented in a more human-readable form. For users new to this type of programming, human-readable examples of code with visible results is key to understanding. By providing simplified access to complex DSP tools, the accessibility of acoustic detection techniques discussed as part of this research improves. The documentation for this library is provided in Appendix D.

6.4 Configuration application

Because all AudioMoth firmware is open-source, configuring a device to operate using a specific schedule or record at a given sample rate can be done by altering the code and hard-coding these behaviours. For an experienced programmer, setting up the development environment, compiling the new firmware and flashing a device is relatively straightforward. Detailed tutorials provided online[1] greatly simplify this process.

For users unfamiliar with C or programming in general, making substantial changes to even heavily annotated code is not trivial. As this is the case for the majority of ecologists likely to make use of AudioMoth, device configuration must be abstracted to make using the device more accessible. As well as accessibility, a simplified user interface makes applying a configuration to a large number of devices a more efficient process, even for more technically-skilled users.

[1] https://www.openacousticdevices.info/getting-started

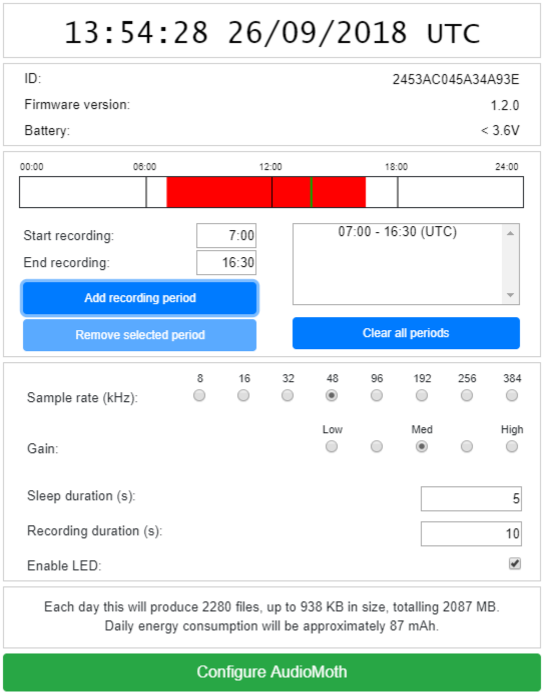


FIGURE 6.6: The AudioMoth Configuration App. All user interaction with the AudioMoth outside of switching between the three operation modes is done through this application, communicating with the device over USB HID.

The AudioMoth Configuration App was created to handle all configuration interactions with the AudioMoth. Its user interface is shown in Figure 6.6. Built using the Electron framework [Github, 2019], the configuration application uses USB HID to communicate with AudioMoth devices. This communication is done using simple, easily-extendable data packets. Users wishing to add additional functionality to the AudioMoth which requires communication with the device can easily append to these packets.

The configuration application and its accompanying firmware also act as an example of a single possible application of the AudioMoth hardware. That of a scheduled acoustic recorder, capable of recording at a range of sample rates. This basic use of the AudioMoth hardware is applicable to a wide variety of monitoring projects and reflects the functionality of many commercial acoustic monitoring tools. Because of this it is sufficient for a large number of users. Using the online documentation and these implementations as an example, users have already tailored the AudioMoth to their deployments, programming features such as adaptive recording schedules which change with the sunset.

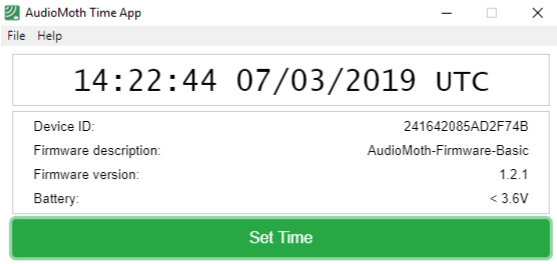


Figure 6.7: The AudioMoth Time App. Any AudioMoth firmware using the foundation firmware provided online is able to interact with this application, display device/-firmware information, and update the clock.

6.5 Time application

As well as the configuration application, which configures the operation of the default AudioMoth firmware, a second application was also developed which supports the AudioMoth's basic functionality. This application sets the on-board clock and displays various useful information, including the device ID, battery level, and a description of the firmware which can be hard-coded by a developer (see Figure 6.7).

While the configuration application functions as part of a single AudioMoth implementation, the time application is designed to interact with any possible implementation. It does so by using functions included in the foundation of all AudioMoth firmware. This firmware foundation is open-source[2] and implements the AudioMoth's lower level functions, such as turning on the microphone, flashing LEDs, and requesting the time over USB. Users wishing to create bespoke firmware for an application use it as a foundation to build all other functionality on top, confident that these functions will always be available.

6.6 Summary

This chapter presented various techniques by which the accessibility of AudioMoth and acoustic monitoring technology has been both assessed and improved upon. This assessment included a questionnaire which received 75 responses, aiming to qualitatively analyse the demographics of the AudioMoth userbase. Respondents came from a range of backgrounds in and out of academic research, from 18 different countries, and with a variety of technical abilities. Future developments to AudioMoth and any new technology within the field must consider the needs of these demographics if accessibility is to be maintained.

[2] https://github.com/OpenAcousticDevices/AudioMoth-Project

Analysing the results of the questionnaire also revealed a disparity between the purchase location and deployment location, with a number of African and South American countries containing multiple monitoring deployments but few purchases originating there. This could be a signal that future work aimed at improving accessibility could do so by improving awareness in developing countries where monitoring tools are needed.

Building on the results of the questionnaire, a series of interviews were also carried out. The interviews were transcribed and qualitatively assessed using thematic analysis. This process produced three major discussion topics: "open-source, open science and community", "appeal" and "problems and limitations". These themes formed the basis of a study which aimed to better understand the opinions of AudioMoth users on the device and accessibility of acoustic monitoring in general.

This chapter also presented previous efforts to improve the accessibility of AudioMoth and its functionality. These efforts included a software development library, designed to simplify the process of iterating on the open-source AudioMoth firmware. The library was created in response to users stating their inability to implement needed features, such as specialised detection algorithms and adaptive recording schedules. By reducing the requirements in terms of technical ability, a greater number of users gain increased utility of device.

Other efforts to improve accessibility discussed include two desktop applications. The configuration application handles user interactions with the device without requiring direct interaction with the underlying firmware or requiring flashing the device to alter its behaviour. It has been downloaded over 2,100 times and continues to receive regular updates as new functionality is developed. The time application is designed to set the clock and display important device information. If bespoke firmware is built using the recommended firmware foundation then it will be compatible with this application.

Chapter 7

Conclusions and future work

> Its hard to see things when you're too close. Take a step back and look.
>
> Bob Ross

The research described over the course of this thesis has covered the topic of enabling large-scale environmental monitoring by using acoustic detection algorithms deployed on low-cost, low-energy monitoring devices. Acoustics provide many benefits over alternative monitoring methods such as camera trapping and satellite imagery, however the extent to which these benefits are taken advantage of is limited by the accessibility of available tools and techniques.

Detection algorithms provide accessibility in a number of ways: by reducing energy and storage requirements, reducing unit costs for monitoring equipment, and by producing smaller datasets. These accessibility improvements result in improved feasibility for large-scale ecological monitoring projects. This can be a large temporal or spatial scale. The improvements enable devices to be deployed for extended periods of time without replacement and can reduce the unit cost such that a greater number of monitoring devices can be purchased.

The work describes a number of monitoring applications which deploy acoustic detection algorithms for this purpose, providing examples of research questions which would typically require significantly more resources to answer. These applications act as evidence for the effectiveness of acoustic detection algorithms on low-cost hardware in providing accessible, large-scale environmental monitoring.

Chapter 2 reviewed the literature surrounding environmental monitoring in general and then tightened focus on acoustic monitoring. For environmental monitoring this included methods commonly employed in conservation to monitor both zoological and anthropogenic events. These methods have been used successfully for a large number of

FIGURE 7.1: Discussing the AudioMoth with Prince William at the 2018 Illegal Wildlife Trade Conference, as part of the "How the wildtech sector can support the fight" stand.

deployments, but are limited by potential coverage and are thus not applicable to every monitoring project. Acoustics offer an alternative range of tools which fill niches left by other monitoring techniques.

Chapter 2 also reported on how acoustic monitoring is implemented, describing projects which use smartphones, commercial acoustic sensors, and bespoke technology. It also reviews a number of processing and analysis techniques used by people wishing to automatically detect acoustic events, rather than just passively record them. A number of these techniques are employed in following chapters to develop detection algorithms.

Chapter 3 described the design, hardware specifications, and supporting software of the AudioMoth, a low-cost, low-energy acoustic monitoring device which was the deployment platform for the detection algorithms discussed in this thesis. As an environmental monitoring tool, AudioMoth has become widely-used within the conservation community, partly due to its financial accessibility when compared to existing commercial sensors, such as WildLife Acoustics' SongMeter series. AudioMoth applications have extended outside of those described in this thesis. Using the device and supporting software discussed in Chapter 6, a wide array of conservation projects have been carried out around the world, with minimal direct assistance from any members of the AudioMoth development team. AudioMoth is an example of how low-cost, open-source technology can greatly expand what is possible for conservation researchers within the constraints of a limited budget. In 2018, AudioMoth represented itself and the growing community of "wildtech" at the London Illegal Wildlife Trade Conference (see Figure 7.1).

Chapter 4 presented two acoustic detection algorithms aiming to detect acoustic events from zoological sources, deployed on AudioMoth hardware. The first, a bat detection algorithm designed to monitor the Cuban greater funnel-eared bat. The algorithm uses the species' echolocation calls to detect its presence. These vocalisations are ultrasonic pips which vary in length and frequency depending on a bat's current action. While hunting, the Cuban greater funnel-eared bat's call is approximately 5 ms long and is centred around 50 kHz. Because few other acoustic sources produce sound in that ultrasonic frequency band, the 50 kHz component alone was used to identify the call. The detection algorithm applied a threshold to this extracted component, using a sliding window to analyse sample sets multiple times and prevent the call from being lost. Using these steps, the algorithm achieved a high level of detection accuracy, with an F_1 score of 0.961 and an AUROC of 0.975. The algorithm was then field tested as part of a real-world deployment in Cuba which relied upon acoustic detection to study the movements and behaviour of the target species.

The second algorithm discussed in Chapter 4 aimed to detect the presence of the New Forest cicada, a species of insect thought to be locally extinct. The search is commonly referred to as The New Forest Cicada Project[1]. The algorithm operated similarly to the bat detector, building on its design to produce an algorithm robust to wind noise. Instead of just the central frequency of the cicada's call (14 kHz), the algorithm took the ratio of this frequency and the 8 kHz component. White noise produced by wind contains both 14 and 8 kHz frequency components and its 14 kHz component can be loud enough to trigger a false positive recording. However, when the wind is loud enough to do so, the 8 kHz component will also be loud and will produce a small ratio value. This extra analysis step requires few additional calculations while removing a major source of false positives. This robust algorithm achieved an F_1 score and AUROC of 0.964 and 0.998, respectively. By relying on a detection algorithm rather than passively recording and manually analysing after collection, the device requirements in terms of energy and storage were dramatically reduced. This algorithm was deployed on AudioMoth sensors for three-month periods each summer in 2016, 2017, and 2018. Despite not being found in any of these years, annual deployments are planned to continue. The New Forest Cicada Project presents a second example of a large-scale monitoring application which was made possible by the introduction of both low-cost monitoring equipment and accompanying acoustic detection algorithms.

In Chapter 5 these algorithms are extended further, describing the implementation of a HMM-based detection algorithm which responded to the sound of gunshots in a protected tropical forest reserve. Unlike the previous animal call detectors, a gunshot does not possess a single, unique frequency component, so in order to reliably detect it the temporal characteristics of the sound were used. The HMM was used to identify the initial impulse and the decay of a gunshot, using the behaviour of three frequency components to recognise each state. The detection algorithm varied in accuracy depending on

[1] http://www.newforestcicada.info/

FIGURE 7.2: Twelve AudioMoth devices deployed in cases constructed from plastic piping, prepared for a series of field tests aiming to calculate the accuracy of the device and algorithm at various distances.

distance, however on a dataset of 1,170 recordings ranging from 0 to 800 m, it achieved an F_1 score of 0.71.

Due to the difficulty of traversing tropical forests and the need to obtain a representative quantity of poaching data in the Tapir Mountain reserve, the devices had to remain deployed for a year. This, along with the need to cover the entire reserve, further stressed the importance of low-cost, low-power monitoring, especially in a location such as Belize, where money assigned to protecting a scientific reserve like Tapir Mountain is limited. Devices had to both listen throughout the entire deployment year without replacement and remain affordable (storage and batteries included) when deployed in large quantities. AudioMoth devices, equipped with the HMM-based algorithm, were field-tested in Belize (see Figure 7.2) in early 2018, preceding their deployment for the yearlong monitoring project.

To benchmark the gunshot HMM, an alternative algorithm was developed using deep learning techniques commonly used in acoustic detection for ecology [Stowell et al., 2019]. This alternative detection method was also described in Chapter 5. The algorithm was built around a DS-CNN which used mel-weighted frequency bands as input features. The DS-CNN outperformed the HMM in terms of accuracy with an F_1 score of 0.88, however the time required to analyse a set of samples meant that constant listening was not possible on current AudioMoth hardware.

Chapter 6 then illustrated the effectiveness of AudioMoth in improving the accessibility of acoustic detection tools. This was done with a quantitative analysis of questionnaire responses and a qualitative analysis of a series of interviews. The quantitative analysis was given to people from a wide array of demographics who had purchased at least one AudioMoth. These responses revealed the awareness bias of the technology towards western, English-speaking countries, expressing a need to improve accessibility in terms of knowledge in developing countries where low-cost monitoring for conservation is likely to be needed. They also revealed a number of other factors to consider when aiming to improve the accessibility of conservation technology in general, including the need to accommodate for users with extremely low levels of technical expertise. Doing so simultaneously provides for more able users.

The interview results were studied using a thematic analysis which grouped the topics and themes of the responses to highlight commonly held opinions. These opinions included the importance of open-source design, showing how ecologists are now actively seeking open-source tools which reflect the concept of open science. Users wish to take advantage of open-source tools, however they are limited by their own technical abilities and the accessibility of doing so. Other themes revealed by thematic analysis include the importance of financial accessibility. This included enabling users to obtain sufficient equipment quantities as well as the reduced value enabling deployments which risk equipment, such as long-term deployments and projects involving citizen science.

Finally, Chapter 6 described two tools employed to make different aspects of AudioMoth's usage more accessible: the configuration app which streamlines user interaction with the AudioMoth and a detection algorithm library, designed to simplify the process of developing custom detection functionality. Tools such as these improve the accessibility of AudioMoth's more technically-demanding functions, closing the knowledge gap between users and the functionality they desire.

In conclusion, AudioMoth, along with the research described in this thesis present an argument for the use of low-cost sensors and acoustic detection algorithms to enable accessible, large-scale monitoring for conservation. This argument is backed up by the widespread use of AudioMoth for acoustic monitoring, the deployments presented in Chapters 4 and 5 which were made possible at scale thanks to bespoke detection algorithms, and a number of publications about various aspects of the research in both ecology and engineering journals. The interview responses confirm the need for a low-cost alternative to the widely-used commercial acoustic monitors. While these devices are applicable to a wide variety of conservation problems, the accessibility of low-cost conservation technology provides for an under-served niche with many unanswered research questions.

7.1 Continuing development

The AudioMoth's success has necessitated continued support for the device, beyond the work discussed in this thesis. The AudioMoth project and Open Acoustic Devices will be continued in the form of a start-up, following the completion of this research. The three-man team aim to sell AudioMoth devices as well as additional products and services including cases and consultation, in order to fund the continued development and support of AudioMoth hardware and software. Open Acoustic Devices intends to iterate on both to better provide for the userbase grown over the past few years.

Hardware development will include the design of the next iteration of the AudioMoth, using both user feedback and experience gained conducting the research described in this thesis to inform the design. This includes hardware better suited to bat monitoring, due to the success of the work described in Chapter 4 and the popularity of AudioMoth for that task, as shown in Chapter 6. Since the original AudioMoth design has been released, better components have become available or entered a price range which makes them usable for a low-cost device. This means that the specifications of the device can improve without sacrificing affordability.

In terms of software developments which will be pursued, the detection algorithms discussed are only applicable to a small subset of detection targets. While the open-source code, documentation, and library functions make tailoring these algorithms to new applications more streamlined, many ecologists still lack the programming ability to do so. This limits the functionality of AudioMoth to code which has been made freely available. A possible future extension could be abstracting the process of building a detection algorithm, using a user interface to simplify the process of setting up triggers. This has been done with an upcoming iteration of the configuration application, extending it to include configuring a single, Goertzel filter threshold (see Figure 7.3). Generalised detection algorithms which can be manipulated using simple user interfaces could bring bespoke acoustic detection to users and bridge the gap in technical skill required.

As hardware and software are developed to ease the process of deployment and analysis, barriers between users and effective monitoring are removed. By continuing to support AudioMoth in this way, other such low-cost, open-source conservation technology are encouraged. This brings benefits discussed throughout this thesis to other tools used by ecologists to monitor, record, and protect the natural world.

7.2 Future work

While the techniques and tools proposed provide evidence for the effectiveness of low-cost acoustic monitoring, there are multiple possible avenues of future work which could be explored.

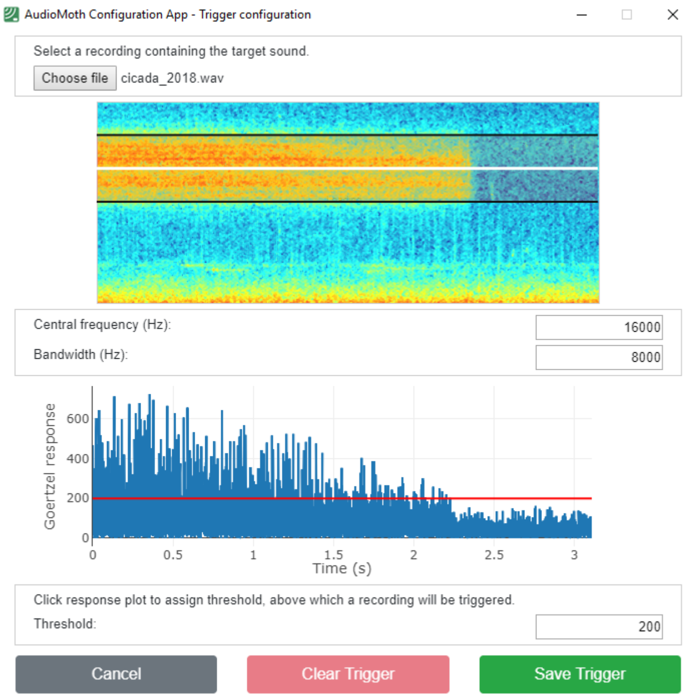


FIGURE 7.3: Trigger selection UI within the AudioMoth Configuration App. This allows users to implement simple detection, based around applying a threshold to a single Goertzel filter response.

One such avenue would involve further experimentation with deep learning techniques on constrained hardware. In order to train a deep learning model for a specific application, a sufficiently large, labelled dataset is required. Thanks to growing support for open science, lots of previously-collected acoustic data is becoming freely available. As well as this, low-cost devices such as the AudioMoth can be deployed for extended periods of time to gather a large quantity of data for training. However, before recordings can be used as part of a training set for a neural network, they must be classified and labelled. To train model for acoustic detection which accurately reflects the range of possible sounds in an environment, examples of as many variations in the target sound and false positive sources are needed. This means that hours of audio must be manually assessed to produce a training set of recordings. Future work could include possible methods of solving the issue of obtaining sufficiently large labelled datasets, using methods such as crowd-sourcing to make the process of analysing easier.

Crowd-sourcing has been previously seen success in classifying large collections of visual data collected by camera traps [Swanson et al., 2015]. Zooniverse provides a generalised tool for this labelling, focusing on image recognition [Cox et al., 2015]. The tool also supports video and audio files for classification also, however its focus is primarily on image recognition. As shown in Section 4.2.7, large quantities of audio data can be quickly labelled through visual inspection. Many Zooniverse projects have used this to great effect, tasking users with labelling spectrograms. As well as standard spectrograms, a specialised audio labelling platform could present audio in a variety of forms, pre-processing the audio depending on the requirements of the application. This has been done to great effect in projects targeting a single application in bioacoustics, such as Bat Detective [Mac Aodha et al., 2018]. Bat Detective uses a web platform to apply labels to recordings as containing bat calls as well as the type of calls within that classification, using both acoustic and visual inspection. Future work could expand to support researchers uploading acoustic datasets and requesting labels for any given application, incorporating the benefits of a specialised bioacoustic platform into the Zooniverse model. The interview responses discussed in Section 6.2 also show a desire for many ecologists to feed back into the monitoring community in the spirit of open science. Labelling other researchers' open acoustic data to enrich available datasets could provide a way for researchers and conservation enthusiasts to fulfil that desire. Crowd-sourcing through citizen science comes with additional problems which could be explored in future work, such as verifying the validity of labels and encouraging enough users to take part.

Once a training set has been obtained, there still remains challenges in building deep learning models which will effectively run on constrained hardware. Investigating all possible methods of reducing the size of various deep learning techniques to run on such hardware lies beyond the scope of this thesis. As with the depthwise-separable convolutional neural network used in Section 5.2, many techniques described in the available literature come from computer vision applications. Exploring these methods and assessing their effectiveness in the context of acoustic detection provides another potential avenue for future work.

As well as the software development side of deep learning for acoustic detection, the hardware which these detection algorithms are deployed on could also be explored. Microcontrollers with sufficient computing power for deep learning are becoming more affordable and more energy efficient. For example, the Silicon Labs Giant Gecko, which has a 72 MHz processor, consumes 77 μA/MHz while active. Other low-power microcontrollers include the Ambiq Micro Apollo3 microcontroller, which has a 48 MHz processor and consumes just 6 μA/MHz while active. The Apollo3 was designed for use in sensor-based edge computing and is supported by Tensorflow Lite, a deep learning framework specifically designed to run on embedded devices. This support includes extensive documentation for setting up Tensorflow Lite on the Apollo3. This microcontroller and those like it could enable many new opportunities for deep learning on constrained hardware. An investigation of these possible platforms for deep learning could be a good starting point for future work.

Outside of deep learning for acoustics, future research could expand upon the work done using automatic detection to bring affordable monitoring to the field of ecology with tools which fill other niches. One example would be visual monitoring, which currently makes frequent use of simple tripwires to react to any movement in front of the sensor, rather than a specific target species. Attempting to bring the ideas of detection algorithms on constrained hardware into computer vision could greatly improve the accessibility of another vital tool for conservation.

Following on from the user requirement studies carried out in Chapter 6, it would be informative to extend the questionnaire to ask users in a wider variety of locations. This could include third world countries where the effects of low-cost conservation equipment are likely to be strongly felt.

Appendix A

Questionnaire

From November 2018 until January 2019, the following questionnaire was carried out amongst users of the AudioMoth. They were found through the Open Acoustic Devices mailing list.

User Information

1. Position/job title?
2. Highest level of education attained?
3. How would you rate your perceived level of technical expertise?
    - Very inexperienced
    - Inexperienced
    - Neutral
    - Experienced
    - Very experienced
4. How would you describe your place of employment?
    - Academia
    - Large business
    - Small business
    - Contractor
    - Other
5. What is your area of research/work?
6. In which country is your place of employment primarily based?

7. Do you currently own any AudioMoth devices?
    - Yes. I am currently using them/have used them previously.
    - No but I would like to in future.
    - No and I do not plan to.
8. How many people in your team/company directly interact with an AudioMoth?

Initial Purchasing

1. How did you initially become aware of Open Acoustic Devices and AudioMoth?
    - Word of mouth
    - Blog post
    - News article
    - Academic publication
    - At an event (conference presentation, etc.)
    - WWF acoustic monitoring recommendation
    - Other
2. Approximately when did you purchase your AudioMoth(s)?
3. How many AudioMoths do you currently own?
4. Did you purchase any devices to trial before a larger order for deployment?
    - Yes
    - No
5. Why did you decide to purchase AudioMoth over other acoustic loggers?

Application description

1. Are you attempting to monitor animal or human sound (biophony or anthropophony)?
    - Human (anthrophony)
    - Animal (biophony)
    - Other
2. Briefly describe the target sound(s).
3. Why are you attempting to monitor this target?

4. Are the monitoring targets vocalisations limited to a certain time of day?
    - Yes
    - No
5. In which country are you deploying?
6. In what type of environment are the AudioMoths deployed?
    - Urban
    - Green spaces near human habitation
    - Light forest
    - Dense forest
    - Grassland
    - Desert
    - Other
7. What is the climate of the deployment location?
    - Temperate
    - Mediterranean
    - Tropical
    - Polar
    - Other

Software Usability

1. Have you installed and used the AudioMoth Configuration Application?
    - Yes
    - No
2. Which operating system do you use to run the application?
    - Mac
    - Linux
    - Windows
3. How would you rate the application's overall ease of use?
    - Very poor
    - Poor

- Neutral
- Good
- Very good

4. How would you rate the clarity of the user interface?

   - Very poor
   - Poor
   - Neutral
   - Good
   - Very good

5. Are there any features which were not initially intuitive? If so, how could they be improved?

6. Are there any common mistakes which are made when configuring an AudioMoth?

7. Are there any features which you feel the AudioMoth Configuration Application is currently lacking?

8. Would you use an Android or iOS implementation of the configuration application?

   - Yes. It would entirely replace the computer-based version
   - Yes. I would use both versions
   - No. I do not have access to a smartphone capable of running such an application
   - No. I have no need for a smartphone version

9. Have you attempted to customise the AudioMoth firmware using the code available online?

   - Yes
   - No. It is not necessary for my application
   - No. I was not aware I could do so
   - No. I do not possess the expertise to do so

Overall Usability

1. How happy are you with the AudioMoth in general?

   - Very unhappy
   - Unhappy

- Neutral
- Happy
- Very happy

2. What are some examples of features you feel the current version of AudioMoth lacks?

Further Contact

1. Would you be willing to take part in a further short interview about your application and experience using AudioMoth?
2. Enter an email address which you are happy with us using to contact you to co-ordinate a brief interview for further information.

Appendix B

Interview prompts

Each interview was free-form, allowing the interviewee to expand on their questionnaire answers and express opinions on AudioMoth and acoustic monitoring technology in general. They were asked a series of open questions intended to prompt these responses.

1. Describe the aims of your AudioMoth deployment.
2. Describe the location where you deployed AudioMoth devices.
    - Internet availability?
    - Difficulty of access?
3. Step through the process you went through to prepare your AudioMoth devices for deployment.
4. What are some examples of issues which hindered the progress of this process?
    - Can you think of any solutions to these issues?
5. Have you previously used any other acoustic monitoring devices?
    - If yes, why did you decide to use AudioMoth for this application?
    - If yes, how does AudioMoth compare to previously used devices in terms of usability?
    - If yes, are there any issues common to using any acoustic sensors?
6. Have you previously used any other forms of environmental monitoring?
    - If yes, why did you decide to use acoustics for this application?
    - If yes, are there any issues common to both types of biological monitoring?
7. With low-cost, open-source devices, do you feel like your expectations in terms of quality of support and accessibility are lower or changed when compared to commercial devices?

8. Do you use the AudioMoth in a way which you think the average user wouldn't? This could be how you case them, deploy them, etc.

9. If you could add a single piece of additional functionality to AudioMoth, what would it be?

Appendix C

Media engagements

The AudioMoth project has been featured in a large number of journalistic publications, reporting projects which were enabled by the work done as part of this thesis.

- New Scientist article describing a project which used AudioMoth-equipped drones to listen to bats, aiming to study in-flight behaviour.
  April 2017.
  New Scientist

- New Scientist article about the bat detection project described in Chapter 5.
  May 2017.
  New Scientist

- Journal of Animal Ecology press article about new technologies for listening to and monitoring bats.
  October 2017.
  Journal of Animal Ecology - Animal Ecology in Focus

- Oxford Sparks podcast featuring an interview with Peter Prince, Andrew Hill and Alex Rogers about the search for the New Forest Cicada (see Chapter 5).
  November 2017.
  Oxford Sparks

- Medium article describing a project where AudioMoth was used to monitor forest birds in Kenya.
  December 2017.
  Medium

- British Ecological Society press release describing a presentation by Evelyn Piña Covarrubia on the work done with AudioMoth in Belize.
  December 2017.
  British Ecological Society

- World Wildlife Fund (WWF) guidelines for acoustic monitoring, featuring AudioMoth as part of a case study.
  2017.
  WWF

- Mongabay article about the 2018 Methods in Ecology and Evolution publication.
  January 2018
  Mongabay

- Sparkfun blog post describing how AudioMoth has scaled to enable global conservation projects.
  May 2018.
  Sparkfun

- Medium article about how AudioMoth is manufactured and supported.
  May 2018.
  Medium

- Alar Ecology blog post announcing the start of a test AudioMoth deployment aiming to monitor nocturnal birds.
  July 2018.
  Alar Ecology

- Remote Sensing in Ecology and Conservation blog post and video describing the AudioMoth placement publication.
  October 2018.
  Remote Sensing in Ecology and Conservation

- Presence at the London 2018 Illegal Wildlife Trade Conference as part of the wildlife technology stand.
  October 2018.

- Chinadialogue article describing various pieces of technology which are being deployed in aid of wildlife conservation, including AudioMoth.
  November 2018.
  Chinadialogue

- Silicon Labs interview with Alex Rogers, describing the AudioMoth's development and how a Silicon Labs processor enables low-power acoustic monitoring.
  February 2019.
  Silicon Labs

- Hackerfarm article in the Tech 4 Good series, reviewing technology designed for humanitarian, development, or environmental applications.
  March 2019.
  HackerFarm

- Wight Link press release by Karen Woods where Adrain Bicker describes using AudioMoth to monitor bat migration.
  April 2019.
  Wight Link

- Article describing a population survey of endangered bats carried out using AudioMoth. Several AudioMoth are visible in the article images.
  August 2019.
  BBC News

Appendix D

Detection algorithm library documentation

The following appendix lists the externally accessible functions available in the basic AudioMoth detection algorithm library. These functions are the bare minimum required for users to implement detection algorithms similar in scope to those described in Chapter 4.

```
/*
 * Function:  generate_hamming_factors
 * --------------------
 *  Create N Hamming window values
 *  Alternatively, with a fixed N, calculate and hard-code Hamming factors to
 *  reduce calculation
 *
 *  N: number of values the Hamming window will be applied to
 *
 *  returns: A length N array of values representing a Hamming window.
 */

float * generate_hamming_factors(int N);

/*
 * Function:  calculate_w
 * --------------------
 *  Calculate the required value of the constant w for a Goertzel filter using
 *  the equation:
 *      w = 2.0 * cos(2.0 * PI * target_frequency / sample_rate)
 *
 *  target_frequency: The central target frequency of the filter
 *  sample_rate: The sample rate of the samples the filter will be applied to
 *
 *  returns: The constant w, used in the process of running a Goertzel filter
 */

float calculate_w(int target_frequency, int sample_rate);
```

```
/*
 * Function:  add_goertzel_filter
 * --------------------
 *  Add a Goertzel filter to an array of Goertzel filters
 *
 *  filters: Pointer to an existing array of Goertzel filters
 *  filter_count: Pointer to number of filters in the array, passed by reference
 *  w: The constant value w, hard-coded or generated by the function 'calculate_w'
 *  length: The length of the Goertzel filter in samples
 */

void add_goertzel_filter(goertzel_filter *filters, int *filter_count, float w,
    int length);

/*
 * Function:  update_single_goertzel_filter
 * --------------------
 *  Given a single Goertzel filter, update using a single sample
 *
 *  filter: The filter being updated
 *  hamming_factors: Pointer to array of values representing a Hamming window
 *  sample: The sample value to update the filter with
 */

void update_single_goertzel_filter(goertzel_filter *filter, float sample);

/*
 * Function:  update_goertzel_filters
 * --------------------
 *  Run update on all current filters using a single sample
 *
 *  filters: Pointer to an array containing all Goertzel filters being used
 *  sample: The sample value to update the filter with
 */

void update_goertzel_filters(goertzel_filter *filters, int filter_count,
    float sample);

/*
 * Function:  update_filters_hamming
 * --------------------
 *  Run update on all current filters using a single sample, applying a Hamming
 *  filter to the sample first
 *
 *  filters: Pointer to an array containing all Goertzel filters being used
 *  hamming_factors: Pointer to array of values representing a Hamming window
 *  sample: The sample value to update the filter with
 */

void update_filters_hamming(goertzel_filter *filters, int filter_count,
    float* hamming_factors, float sample);
```

```c
/*
 * Function:  apply_median_filter
 * --------------------
 *  Replace each value with the median of it and its surrounding values
 *
 *  values: Pointer to array of values to be filtered
 *  kernel_size: Size of median filter
 */

void apply_median_filter(float *values, int kernel_size);

/*
 * Function:  partition
 * --------------------
 *  Partition a list around a pivot point, moving all elements smaller than the
 *  last element to its left and all larger elements to its right
 *
 *  values: Array of unsorted values
 *  l: Pivot point and left index
 *  r: Right index
 */

int partition(float values[], int l, int r);

/*
 * Function:  get_kth
 * --------------------
 *  Order the list up to the kth value using Quicksort and return the kth largest
 *  Useful for finding the median of a set of values if the middle element is
 *  requested
 *
 *  k: The target index in an ordered version of the values array
 *  values: Array of unsorted values
 *  l: Left index (start of list)
 *  r: Right index (end of list)
 *
 *  returns: The kth element of the list if it was sorted in ascending order
 */

float get_kth(int k, float values[], int l, int r);

/*
 * Function:  slide_window
 * --------------------
 *  Take a window of values, move all indexes up one and place a new value in the
 *  newly created space
 *
 *  window: Pointer to a set of values in an array
 *  window_length: The number of values included in the window
 *  new_value: The value to be appended to the window once it has been slid along
 */

void slide_window(float *window, int window_length, float new_value);
```

Bibliography

O Abdel-Hamid, A Mohamed, H Jiang, L Deng, G Penn, and D Yu. Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on audio, speech, and language processing*, 22(10):1533–1545, 2014. doi: 10.1109/TASLP.2014.2339736.

A M Adams, M K Jantzen, R M Hamilton, and M B Fenton. Do you hear what i hear? implications of detector selection for acoustic monitoring of bats. *Methods in Ecology and Evolution*, 3(6):992–998, 2012. doi: 10.1111/j.2041-210X.2012.00244.x.

C Astaras, J M Linder, P H Wrege, R D Orume, and D W Macdonald. Passive acoustic monitoring as a law enforcement tool for afrotropical rainforests. *Frontiers in Ecology and the Environment*, 15(5):233–234, 2017. doi: doi:10.1002/fee.1495.

A Averbuch, V A Zheludev, N Rabin, and A Schclar. Wavelet-based acoustic detection of moving vehicles. *Multidimensional Systems and Signal Processing*, 20(1):55, 2009. doi: 10.1007/s11045-008-0058-z.

A Backurs and C Tzamos. Improving viterbi is hard: Better runtimes imply faster clique algorithms. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 311–321. JMLR.org, 2017.

R Bardeli, D Wolff, F Kurth, M Koch, K-H Tauchert, and K-H Frommolt. Detecting bird sounds in a complex acoustic environment and application to bioacoustic monitoring. *Pattern Recognition Letters*, 31(12):1524–1534, 2010.

N Baydar and A Ball. Detection of gear failures via vibration and acoustic signals using wavelet transform. *Mechanical Systems and Signal Processing*, 17(4):787–804, 2003. doi: 10.1006/mssp.2001.1435.

R D Beason, R Riesch, and J Koricheva. Aurita: an affordable, autonomous recording device for acoustic monitoring of audible and ultrasonic frequencies. *Bioacoustics*, pages 1–16, 2018. doi: 10.1080/09524622.2018.1463293.

D T Blumstein, D J Mennill, P Clemins, L Girod, K Yao, G Patricelli, J L Deppe, A H Krakauer, C Clark, K A Cortopassi, S F Hanser, B McCowan, A M Ali, and A N G Kirschel. Acoustic monitoring in terrestrial environments using microphone arrays: Applications, technological considerations and prospectus. *Journal of Applied Ecology*, 48(3):758–767, 2011. doi: 10.1111/j.1365-2664.2011.01993.x.

T S Brandes. Automated sound recording and analysis techniques for bird surveys and conservation. *Bird Conservation International*, 18(S1):S163–S173, 2008. doi: 10.1017/S0959270908000415.

V Braun and V Clarke. Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2):77–101, 2006.

F Briggs, R Raich, and X Z Fern. Audio classification of bird species: a statistical manifold approach. In *2009 Ninth IEEE International Conference on Data Mining*, pages 51–60. IEEE, 2009. doi: 10.1109/ICDM.2009.65.

E R Britzke, E H Gillam, and K L Murray. Current state of understanding of ultrasonic detectors for the study of bat ecology. *Acta Theriologica*, 58(2):109–117, 2013. doi: 10.1007/s13364-013-0131-3.

H Brumm. The impact of environmental noise on song amplitude in a territorial bird. *Journal of Animal Ecology*, 73(3):434–440, 2004. doi: 10.1111/j.0021-8790.2004.00814.x.

Z Burivalova, M Towsey, T Boucher, A Truskinger, C Apelis, P Roe, and E T Game. Using soundscapes to detect variable degrees of human influence on tropical forests in papua new guinea. *Conservation Biology*, 32(1):205–215, 2018. doi: 10.1111/cobi.12968.

E Cakir, Adavanne S, Parascandolo G, Drossos K, and Virtanen T. Convolutional recurrent neural networks for bird audio detection. In *2017 25th European Signal Processing Conference (EUSIPCO)*, pages 1744–1748, Aug 2017. doi: 10.23919/EUSIPCO.2017.8081508.

R B Calhoun. System and method for archiving data from a sensor array, August 12 2008. US Patent 7,411,865.

A Chacon-Rodriguez, P Julian, L Castro, P Alvarado, and N Hernández. Evaluation of gunshot detection algorithms. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 58(2):363–373, 2011.

D Chesmore. Automated bioacoustic identification of species. *Anais da Academia Brasileira de Ciências*, 76(2):436–440, 2004. doi: 10.1590/S0001-37652004000200037.

E D Chesmore. Application of time domain signal coding and artificial neural networks to passive acoustical identification of animals. *Applied Acoustics*, 62(12):1359–1374, 2001. doi: 10.1016/S0003-682X(01)00009-3.

K Choi, G Fazekas, M Sandler, and K Cho. Convolutional recurrent neural networks for music classification. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2392–2396. IEEE, 2017. doi: 10.1109/ICASSP.2017.7952585.

F Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1800–1807, 2017. doi: 10.1109/CVPR.2017.195.

J Cox, E Y Oh, B Simmons, C Lintott, K Masters, A Greenhill, G Graham, and K Holmes. Defining and measuring success in online citizen science: A case study of zooniverse projects. *Computing in Science & Engineering*, 17(4):28–41, 2015. doi: 10.1109/MCSE.2015.65.

F Danielsen, M M Mendoza, P Alviola, D S Balete, M Enghoff, M K Poulsen, and A E Jensen. Biodiversity monitoring in developing countries: what are we trying to achieve? *Oryx*, 37(04):407–409, 2003. doi: 10.1017/S0030605303000735.

W Dressler and R Roth. The good, the bad, and the contradictory: neoliberal conservation governance in rural southeast asia. *World Development*, 39(5):851–862, 2011. doi: 10.1016/j.worlddev.2010.08.016.

M B Fenton and G P Bell. Recognition of species of insectivorous bats by their echolocation calls. *Journal of Mammalogy*, 62(2):233–243, 1981. doi: 10.2307/1380701.

S Fernández, A Graves, and J Schmidhuber. An application of recurrent neural networks to discriminative keyword spotting. In *International Conference on Artificial Neural Networks*, pages 220–229. Springer, 2007. doi: 10.1007/978-3-540-74695-9_23.

J Fernández-Marqués, V W-S Tseng, S Bhattachara, and N D Lane. On-the-fly deterministic binary filters for memory efficient keyword spotting applications on embedded devices. In *Proceedings of the 2nd International Workshop on Embedded and Mobile Deep Learning*, pages 13–18. ACM, 2018. doi: 10.1145/3212725.3212731.

G D Forney. The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, 1973. doi: 10.1109/PROC.1973.9030.

Frontier Labs. Bioacoustic Audio Recorder. http://www.frontierlabs.com.au/index.php?p=1_8_Bioacoustic-Audio-Recorder, 2018. Accessed: 28/12/2018.

J Fullard and J Dawson. The echolocation calls of the spotted bat euderma maculatum are relatively inaudible to moths. *Journal of Experimental Biology*, 200(1):129–137, 1997.

M Gales and S Young. The application of hidden markov models in speech recognition. *Foundations and Trends® in Signal Processing*, 1(3):195–304, 2008. doi: 10.1561/2000000004.

M Garstang. Long-distance, low-frequency elephant communication. *Journal of Comparative Physiology*, 190(10):791–805, 2004. doi: 10.1007/s00359-004-0598-0.

A Gasc, S Pavoine, L Lellouch, P Grandcolas, and J Sueur. Acoustic indices for biodiversity assessments: Analyses of bias based on simulated bird assemblages and recommendations for field surveys. *Biological Conservation*, 191:306–312, 2015. doi: 10.1016/j.biocon.2015.06.018.

D Gillespie, D K Mellinger, J Gordon, D Mclaren, P Redmond, R McHugh, P W Trinder, X Y Deng, and A Thode. Pamguard: Semiautomated, open source software for real-time acoustic detection and localisation of cetaceans. *Journal of the Acoustical Society of America*, 30(5):54–62, 2008. doi: doi:10.1121/1.4808713.

Github. Electron. https://www.electron.atom.io, 2019. Accessed: 24/01/2019.

H Goëau, H Glotin, W-P Vellinga, R Planqué, A Rauber, and A Joly. Lifeclef bird identification task 2014. In *CLEF: Conference and Labs of the Evaluation forum*, 2014.

H Goëau, H Glotin, W-P Vellinga, R Planqué, and A Joly. Lifeclef bird identification task 2016: The arrival of deep learning. In *Working Notes of CLEF 2016-Conference and Labs of the Evaluation forum*, pages 440–449, 2016.

G Goertzel. An algorithm for the evaluation of finite trigonometric series. *American mathematical monthly*, 65:34–35, 1958.

F J González-Castano, J Vales Alonso, E Costa-Montenegro, P López-Matencio, F Vicente-Carrasco, F J Parrado-Garcia, F Gil-Castineira, and S Costas-Rodriguez. Acoustic sensor planning for gunshot location in national parks: A pareto front approach. *Sensors*, 9(12):9493–9512, 2009. doi: 10.3390/91209493.

T Grava, N Mathevon, E Place, and P Balluet. Individual acoustic monitoring of the european eagle owl bubo bubo. *Ibis*, 150(2):279–287, 2008. doi: 10.1111/j.1474-919X.2007.00776.x.

A Grossmann and J Morlet. Decomposition of hardy functions into square integrable wavelets of constant shape. *SIAM journal on mathematical analysis*, 15(4):723–736, 1984. doi: 10.1137/0515056.

A Gutiérrez, V Ruiz, E Moltó, G Tapia, and M del Mar Téllez. Development of a bioacoustic sensor for the early detection of red palm weevil (Rhynchophorus ferrugineus Olivier). *Crop Protection*, 29(7):671–676, 2010. doi: 10.1016/j.cropro.2010.02.001.

S R Hage, T Jiang, S W Berquist, J Feng, and W Metzner. Ambient noise induces independent shifts in call frequency and amplitude within the lombard effect in echolocating bats. *Proceedings of the National Academy of Sciences*, page 201211533, 2013. doi: 10.1073/pnas.1211533110.

S E Hampton, C A Strasser, J J Tewksbury, W K Gram, A E Budden, A L Batcheller, C S Duke, and J H Porter. Big data and the future of ecology. *Frontiers in Ecology and the Environment*, 11(3):156–162, 2013. doi: 10.1890/120103.

D Hegglin, F Bontadina, S Gloor, J Romer, U Müller, U Breitenmoser, and P Deplazes. Baiting red foxes in an urban area: a camera trap study. *The Journal of Wildlife Management*, 68(4):1010–1017, 2004. doi: 10.2193/0022-541X(2004)068[1010:BRFIAU]2.0.CO;2.

J Heidemann, W Ye, J Wills, A Syed, and Y Li. Research challenges and applications for underwater sensor networking. In *IEEE Wireless Communications and Networking Conference*, volume 1, pages 228–235. IEEE, 2006. doi: 10.1109/WCNC.2006.1683469.

A P Hill, P Prince, E Covarrubias, C P Doncaster, J L Snaddon, and A Rogers. Audiomoth: Evaluation of a smart open acoustic device for monitoring biodiversity and the environment. *Methods in Ecology and Evolution*, 2017. doi: 10.1111/2041-210X.12955.

A P Hill, A Davies, P Prince, J L Snaddon, C P Doncaster, and A Rogers. Leveraging conservation action with open-source hardware. *Conservation Letters*, 2019a. doi: 10.1111/conl.12661.

A P Hill, P Prince, J L Snaddon, C P Doncaster, and A Rogers. Audiomoth: A low-cost acoustic device for monitoring biodiversity and the environment. *HardwareX*, 2019b. doi: 10.1016/j.ohx.2019.e00073.

R Hill. What sample size is enough in internet survey research. *Interpersonal Computing and Technology: An electronic journal for the 21st century*, 6(3-4):1–10, 1998.

G Hinton, L Deng, D Yu, G E Dahl, A Mohamed, N Jaitly, A Senior, V Vanhoucke, P Nguyen, T N Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 29(6):82–97, 2012. doi: 10.1109/MSP.2012.2205597.

L I Hollen and A N Radford. The development of alarm call behaviour in mammals and birds. *Animal Behaviour*, 78(4):791–800, 2009. doi: 10.1016/j.anbehav.2009.07.021.

A G Howard, M Zhu, B Chen, D Kalenichenko, W Wang, T Weyand, M Andreetto, and H Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

Y Hu and G C Cardoso. Which birds adjust the frequency of vocalizations in urban noise? *Animal Behaviour*, 79(4):863–867, 2010. doi: 10.1016/j.anbehav.2009.12.036.

G Huang, Z Liu, L Van Der Maaten, and K Q Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269. IEEE, 2017a. doi: 10.1109/CVPR.2017.243.

J Huang, V Rathod, C Sun, M Zhu, A K Balan, A Fathi, I Fischer, Z Wojna, Y Song, S Guadarrama, and K Murphy. Speed/accuracy trade-offs for modern convolutional object detectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3296–3297, 2017b.

T Hughes and K Mierle. Recurrent neural networks for voice activity detection. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7378–7382. IEEE, 2013. doi: 10.1109/ICASSP.2013.6639096.

H Ide and T Kurita. Improvement of learning for cnn with relu activation by sparse regularization. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2684–2691. IEEE, 2017. doi: 10.1109/IJCNN.2017.7966185.

R A Ims and N G Yoccoz. Ecosystem-based monitoring in the age of rapid climate change and new technologies. *Current Opinion in Environmental Sustainability*, 29: 170–176, 2017. doi: 10.1016/j.cosust.2018.01.003.

A Irwin. *Citizen science: A study of people, expertise and sustainable development.* Routledge, 2002.

IUCN. Natalus primus conservation status. http://www.iucnredlist.org/details/136777/0, 2019. Accessed: 25-01-2019.

A N James, M J B Green, and J R Paine. Global review of protected area budgets and staff: World conservation monitoring center [wcmc] biodiversity series no. 10. *Cambridge, UK*, 1999.

J B Johnson, M A Menzel, J W Edwards, and W M Ford. A comparison of 2 acoustical bat survey techniques. *Wildlife Society Bulletin*, 30(3):931–936, 2002.

Joint Nature Conservation Committee. Uk list of priority species (uk bap species). http://jncc.defra.gov.uk/page-5717, 2007. Accessed: 10/01/2019.

A Joly, H Goëau, H Glotin, C Spampinato, P Bonnet, W-P Vellinga, J Champ, R Planqué, S Palazzo, and H Müller. Lifeclef 2016: multimedia life species identification challenges. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 286–310. Springer, 2016.

A K Kalan, R Mundry, O J Wagner, S Heinicke, C Boesch, and H S Kühl. Towards the automated detection and occupancy estimation of primates using passive acoustic monitoring. *Ecological Indicators*, 54:217–226, 2015. doi: 10.1016/j.ecolind.2015.02.023.

Y Ke, D Hoiem, and R Sukthankar. Computer vision for music identification. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 597–604. IEEE, 2005. doi: 10.1109/CVPR.2005.105.

S C Keen, Y Shiu, P H Wrege, and E D Rowland. Automated detection of low-frequency rumbles of forest elephants: A critical tool for their conservation. *The Journal of the Acoustical Society of America*, 141(4):2715–2726, 2017. doi: 10.1121/1.4979476.

J T Kerr and M Ostrovsky. From space to species: ecological applications for remote sensing. *Trends in ecology & evolution*, 18(6):299–305, 2003. doi: 10.1016/S0169-5347(03)00071-5.

M A Khalid, M I K Babar, M H Zafar, and M F Zuhairi. Gunshot detection and localization using sensor networks. In *2013 IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)*, pages 1–6. IEEE, 2013. doi: 10.1109/ICSIMA.2013.6717917.

K C Kim. Biodiversity, conservation and inventory: why insects matter. *Biodiversity & Conservation*, 2(3):191–214, 1993.

R A King and W Gosling. Time-encoded speech. *Electronics Letters*, 14(15):456–457, 1978. doi: 10.1049/el:19780306.

I Kiskin, B P Orozco, T Windebank, D Zilli, M Sinka, K Willis, and S Roberts. Mosquito detection with neural networks: the buzz of deep learning. *arXiv preprint arXiv:1705.05180*, 2017.

Q Kong, Y Xu, and M D Plumbley. Joint detection and classification convolutional neural network on weakly labelled bird audio detection. In *Signal Processing Conference (EUSIPCO), 2017 25th European*, pages 1749–1753. IEEE, 2017. doi: 10.23919/EUSIPCO.2017.8081509.

R Kwok. Field instruments: Build it yourself. *Nature*, 545(7653):253–255, 2017. doi: 10.1038/nj7653-253a.

P Laiolo. The emerging significance of bioacoustics in animal species conservation. *Biological Conservation*, 143(7):1635–1645, 2010. doi: 10.1016/j.biocon.2010.03.025.

W F Laurance. Reflections on the tropical deforestation crisis. *Biological conservation*, 91(2):109–117, 1999.

J M Lobo, A Jiménez-Valverde, and R Real. Auc: a misleading measure of the performance of predictive distribution models. *Global ecology and Biogeography*, 17(2):145–151, 2008. doi: 10.1111/j.1466-8238.2007.00358.x.

H Lu, D Frauendorfer, M Rabbi, M S Mast, G T Chittaranjan, A T Campbell, D Gatica-Perez, and T Choudhury. Stresssense: Detecting stress in unconstrained acoustic environments using smartphones. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 351–360. ACM, 2012. doi: doi:10.1145/2370216.2370270.

O Mac Aodha, R Gibb, K E Barlow, E Browning, M Firman, R Freeman, B Harder, L Kinsey, G R Mead, S E Newson, et al. Bat detective - deep learning tools for bat acoustic signal detection. *PLoS computational biology*, 14(3):e1005995, 2018.

R C Maher. Acoustical characterization of gunshots. In *Signal Processing Applications for Public Security and Forensics, 2007. SAFE'07. IEEE Workshop on*, pages 1–5. IEEE, 2007.

Y Malhi, J T Roberts, R A Betts, T J Killeen, W Li, and C A Nobre. Climate change, deforestation, and the fate of the amazon. *science*, 319(5860):169–172, 2008. doi: 10.1126/science.1146961.

R W Mankin, D W Hagstrum, M T Smith, A L Roda, and M T K Kairo. Perspective and promise: A century of insect acoustic detection and monitoring. *American Entomologist*, 57(1):30–44, 2011. doi: 10.1093/ae/57.1.30.

D C Marvin, L P Koh, A J Lynam, S Wich, A B Davies, R Krishnamurthy, E Stokes, R Starkey, and G P Asner. Integrating technologies for scalable ecology and conservation. *Global Ecology and Conservation*, 7:262–275, 2016. doi: 10.1016/j.gecco.2016.07.002.

S Matos, S S Birring, I D Pavord, and H Evans. Detection of cough signals in continuous audio recordings using hidden markov models. *IEEE Transactions on Biomedical Engineering*, 53(6):1078–1083, 2006. doi: 10.1109/TBME.2006.873548.

P J Miller. Diversity in sound pressure levels and estimated active space of resident killer whale vocalizations. *Journal of Comparative Physiology A*, 192(5):449, 2006. doi: 10.1007/s00359-005-0085-2.

L Muda, M Begam, and I Elamvazuthi. Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques. *arXiv preprint arXiv:1003.4083*, 2010.

H Mukundarajan, F J H Hol, E A Castillo, C Newby, and M Prakash. Using mobile phones as acoustic sensors for high-throughput mosquito surveillance. *elife*, 6:e27854, 2017. doi: 10.7554/eLife.27854.001.

B M O'Neal. *Testing the Feasibility of Bioacoustic Localization in Urban Environments*. PhD thesis, University of South Florida, 2014.

Parks Canada. Parks Canada Agency ecosystem monitoring overview. https://www.pc.gc.ca/en/nature/science/controle-monitoring/ecosysteme-ecosystem, 2019. Accessed: 03/01/2019.

S Parsons, A M Boonman, and M K Obrist. Advantages and disadvantages of techniques for transforming and analyzing chiropteran echolocation calls. *Journal of Mammalogy*, 81(4):927–938, 2000. doi: 10.1644/1545-1542(2000)081⟨0927:AADOTF⟩2.0.CO;2.

K B Payne, M Thompson, and L Kramer. Elephant calling patterns as indicators of group size and composition: the basis for an acoustic monitoring system. *African Journal of Ecology*, 41(1):99–107, 2003. doi: 10.1046/j.1365-2028.2003.00421.x.

T Pellegrini. Densely connected cnns for bird audio detection. In *Signal Processing Conference (EUSIPCO), 2017 25th European*, pages 1734–1738. IEEE, 2017. doi: 10.23919/EUSIPCO.2017.8081506.

Y-T Peng, C-Y Lin, M-T Sun, and K-C Tsai. Healthcare audio event classification using hidden markov models and hierarchical hidden markov models. In *IEEE International Conference on Multimedia and Expo*, pages 1218–1221. IEEE, 2009. doi: 10.1109/ICME.2009.5202720.

Pettersson. Pettersson range of ultrasonic detectors. http://www.batsound.com, 2019. Accessed: 08-01-2019.

K J Piczak. Environmental sound classification with convolutional neural networks. In *Proceedings of the IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE, 2015. doi: 10.1109/MLSP.2015.7324337.

S L Pimm, S Alibhai, R Bergl, A Dehgan, C Giri, Z Jewell, L Joppa, R Kays, and S Loarie. Emerging technologies to conserve biodiversity. *Trends in ecology & evolution*, 30(11):685–696, 2015. doi: 10.1016/j.tree.2015.08.008.

E Piña-Covarrubias, A P Hill, P Prince, J L Snaddon, A Rogers, and C P Doncaster. Optimization of sensor deployment for acoustic detection and localization in terrestrial environments. *Remote Sensing in Ecology and Conservation*, 2018. doi: 10.1002/rse2.97.

B J Pinchen and L K Ward. The history, ecology and conservation of the New Forest cicada. *British Wildlife*, 13(4):258–266, 2002.

J H Poole, K Payne, W R Langbauer, and C J Moss. The social contexts of some very low frequency calls of african elephants. *Behavioral Ecology and Sociobiology*, 22(6):385–392, 1988. doi: 10.1007/BF00294975.

I Potamitis, T Ganchev, and D Kontodimas. On automatic bioacoustic detection of pests: the cases of Rhynchophorus ferrugineus and Sitophilus oryzae. *Journal of economic entomology*, 102(4):1681–1690, 2009.

P Prince, A Hill, E Piña Covarrubias, P Doncaster, J L Snaddon, and A Rogers. Deploying acoustic detection algorithms on low-cost, open-source acoustic sensors for environmental monitoring. *Sensors*, 19(3):553, 2019. doi: 10.3390/s19030553.

A C B Prinz, V K Taank, V Voegeli, and E L Walters. A novel nest-monitoring camera system using a Raspberry Pi micro-computer. *Journal of Field Ornithology*, 87(4):427–435, 2016. doi: 10.1111/jofo.12182.

L A Rabin and C M Greene. Changes to acoustic communication systems in human-altered environments. *Journal of Comparative Psychology*, 116(2):137, 2002. doi: 10.1037/0735-7036.116.2.137.

Rainforest Connection. Rainforest connection home page. https://www.rfcx.org, 2019. Accessed:09/01/2019.

C S Robbins, D Bystrak, and P H Geissler. The breeding bird survey: its first fifteen years, 1965-1979. Technical report, Patuxent Wildlife Research Center, Laurel, MD, 1986.

F Rovero, F Zimmermann, D Berzi, and P Meek. "Which camera trap type and how many do I need?" A review of camera features and study designs for a range of wildlife research applications. *Hystrix*, 24(2), 2013. doi: 10.4404/hystrix-24.2-6316.

M Sahidullah and G Saha. Design, analysis and experimental evaluation of block based transformation in mfcc computation for speaker recognition. *Speech Communication*, 54(4):543–565, 2012. doi: 10.1016/j.specom.2011.11.004.

L Schrader and K Hammerschmidt. Computer-aided analysis of acoustic parameters in animal vocalisations: a multi-parametric approach. *Bioacoustics*, 7(4):247–265, 1997. doi: 10.1080/09524622.1997.9753338.

J A Scopoli. Observationes zoologicae. *Annus V, Historico-naturalis Lipsiae*, 1772: 70–128, 1772.

Y Shimabukuro, V Duarte, L Anderson, D Valeriano, E Arai, R Freitas, B F Rudorff, and M Moreira. Near real time detection of deforestation in the brazilian amazon using modis imagery. *Ambiente e Agua-An Interdisciplinary Journal of Applied Science*, 1 (1):37–47, 2007. doi: 10.4136/1980-993X.

B M Siemers and A Schaub. Hunting at the highway: traffic noise reduces foraging efficiency in acoustic predators. *Proceedings of the Royal Society B: Biological Sciences*, 278(1712):1646–1652, 2010. doi: 10.1098/rspb.2010.2262.

K M Stafford, C G Fox, and D S Clark. Long-range acoustic detection and localization of blue whale calls in the northeast pacific ocean. *The Journal of the Acoustical Society of America*, 104(6):3616–3625, 1998. doi: 10.1121/1.423944.

A K Stimpert, W Au, S E Parks, T Hurst, and D N Wiley. Common humpback whale (megaptera novaeangliae) sound types for passive acoustic monitoring. *The Journal of the Acoustical Society of America*, 129(1):476–482, 2011. doi: 10.1121/1.3504708.

D Stowell, M Wood, Y Stylianou, and H Glotin. Bird detection in audio: a survey and a challenge. In *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE, 2016.

D Stowell, M D Wood, H Pamuła, Y Stylianou, and H Glotin. Automatic acoustic detection of birds through deep learning: the first bird audio detection challenge. *Methods in Ecology and Evolution*, 10(3):368–380, 2019. doi: 10.1111/2041-210X. 13103.

V M Sue and L A Ritter. *Conducting online surveys*. Sage, 2012.

J Sueur, A Farina, A Gasc, N Pieretti, and S Pavoine. Acoustic indices for biodiversity assessment and landscape investigation. *Acta Acustica united with Acustica*, 100(4): 772–781, 2014. doi: 10.3813/AAA.918757.

L S M Sugai and D Llusia. Bioacoustic time capsules: Using acoustic monitoring to document biodiversity. *Ecological Indicators*, 99:149–152, 2019. doi: 10.1016/j.ecolind. 2018.12.021.

X Sun, Z Lu, W Hu, and G Cao. Symdetector: detecting sound-related respiratory symptoms using smartphones. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 97–108. ACM, 2015. doi: doi:10.1145/2750858.2805826.

A Swanson, M Kosmala, C Lintott, R Simpson, A Smith, and C Packer. Snapshot serengeti, high-frequency annotated camera trap images of 40 mammalian species in an african savanna. *Scientific data*, 2:150026, 2015. doi: 10.1038/sdata.2015.26.

R Tashakkori, N P Hernandez, A Ghadiri, A P Ratzloff, and M B Crawford. A honeybee hive monitoring system: From surveillance cameras to Raspberry Pis. In *SoutheastCon, 2017*, pages 1–7. IEEE, 2017. doi: 10.1109/SECON.2017.7925367.

The Rufford Foundation. Acoustic detection of Natalus primus, project update february 2015. http://www.rufford.org/files/16094-1%20Feb%20update.pdf, 2017. Accessed: 30-01-2017.

J A Thomas, C F Moss, and M Vater. *Echolocation in bats and dolphins*. University of Chicago Press, 2004.

W L Thompson. Towards reliable bird surveys: accounting for individuals present but not detected. *The Auk*, 119(1):18–25, 2002. doi: 10.1642/0004-8038(2002)119[0018: TRBSAF]2.0.CO;2.

Titley Scientific. Anabat Acoustic Monitor. https://www.titley-scientific.com/uk/, 2019. Accessed: 15/01/2019.

V Tiwari. Mfcc and its applications in speaker recognition. *International journal on emerging technologies*, 1(1):19–22, 2010.

M W Tobler, S E Carrillo-Percastegui, R L Pitman, R Mares, and G Powell. An evaluation of camera traps for inventorying large-and medium-sized terrestrial rainforest mammals. *Animal Conservation*, 11(3):169–178, 2008. doi: 10.1111/j.1469-1795.2008. 00169.x.

J Troscianko and C Rutz. Activity profiles and hook-tool use of new caledonian crows recorded by bird-borne video cameras. *Biology letters*, 11(12):20150777, 2015. doi: 10.1098/rsbl.2015.0777.

H Tuomisto. What satellite imagery and large-scale field studies can tell about biodiversity patterns in amazonian forests. *Annals of the Missouri botanical garden*, pages 48–62, 1998. doi: 10.2307/2991994.

W. Turner, C. Rondinini, N. Pettorelli, B. Mora, A. K. Leidner, Z. Szantoi, G. Buchanan, S. Dech, J. Dwyer, M. Herold, et al. Free and open-access satellite data are key to biodiversity conservation. *Biological Conservation*, 182:173–176, 2015. doi: 10.1016/j.biocon.2014.11.048.

G Tzanetakis and P Cook. Musical genre classification of audio signals. *IEEE Transactions on speech and audio processing*, 10(5):293–302, 2002. doi: 10.1109/TSA.2002.800560.

UK Forestry Commission. Wildlife and conservation; wildlife and its management in the new forest. https://www.forestry.gov.uk/pdf/Wildlife.pdf, 2004. Accessed: 04/02/2019.

G Valenzise, L Gerosa, M Tagliasacchi, F Antonacci, and A Sarti. Scream and gunshot detection and localization for audio-surveillance systems. In *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*, pages 21–26. IEEE, 2007. doi: 10.1109/AVSS.2007.4425280.

R S Voss and L Emmons. Mammalian diversity in neotropical lowland rainforests: a preliminary assessment. *Bulletin of the AMNH*, 230, 1996.

C wa Maina, D Muchiri, and P Njoroge. Cost effective acoustic monitoring of biodiversity and bird populations in Kenya. *bioRxiv*, page 072546, 2016.

P Wegge, Chiranjibi P Pokheral, and S R Jnawali. Effects of trapping effort and trap shyness on estimates of tiger abundance from camera trap studies. *Animal Conservation forum*, 7(3):251–256, 2004. doi: 10.1017/S1367943004001441.

R E Wheat, Y Wang, J E Byrnes, and J Ranganathan. Raising money for scientific research through crowdfunding. *Trends in ecology & evolution*, 28(2):71–72, 2013. doi: 10.1016/j.tree.2012.11.001.

R C Whytock and J Christie. Solo: An open source, customizable and inexpensive audio recorder for bioacoustic research. *Methods in Ecology and Evolution*, 8(3):308–312, 2017. doi: 10.1111/2041-210X.12678.

Wildlife Acoustics. Song Meter range of acoustic monitors. https://www.wildlifeacoustics.com/products, 2018. Accessed: 12/12/2018.

R J Willacy, M Mahony, and D A Newell. If a frog calls in the forest: Bioacoustic monitoring reveals the breeding phenology of the endangered Richmond Range mountain frog (Philoria richmondensis). *Austral Ecology*, 40(6):625–633, 2015. doi: 10.1111/aec.12228.

P H Wrege, E D Rowland, S Keen, and Y Shiu. Acoustic monitoring for conservation in tropical forests: Examples from forest elephants. *Methods in Ecology and Evolution*, 00, 2017. doi: 10.1111/2041-210X.12730.

Y Zhang, N Suda, L Lai, and V Chandra. Hello edge: Keyword spotting on microcontrollers. *arXiv preprint arXiv:1711.07128*, 2017.

D Zilli, O Parson, G V Merrett, and A Rogers. A hidden markov model-based acoustic cicada detector for crowdsourced smartphone biodiversity monitoring. *Journal of Artificial Intelligence Research*, pages 805–827, 2014.

M C Zwart, A Baker, P J K McGowan, and M J Whittingham. The use of automated bioacoustic recorders to replace human wildlife surveys: an example using nightjars. *PloS one*, 9(7):e102770, 2014. doi: 10.1371/journal.pone.0102770.