Maynooth University
National University of Ireland Maynooth

Acoustic Machine Learning Tools and Analysis Software for Advancing Biodiversity Monitoring

A thesis submitted in fulfillment of the requirements for the Ph.D. degree in Statistics

By:

Anthony Gibbons

Under the supervision of:

Professor Andrew Parnell

Professor Ian Donohue

Hamilton Institute
Maynooth University
Ollscoil na hÉireann, Má Nuad
Maynooth, Co. Kildare, Ireland

December 20, 2025

For
Aoibheann Gaughran

and
Davy Gibbons

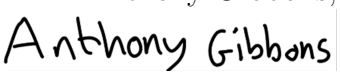
Declaration

I hereby declare that I have produced this manuscript without the prohibited assistance of any third parties and without making use of aids other than those specified.

The thesis work was conducted from September 2021 to September 2025 under the supervision of Professor Andrew Parnell at the Hamilton Institute, Maynooth University (2021 – 2024) and University College Dublin (2024 – present), and Professor Ian Donohue at the Department of Zoology, School of Natural Sciences, Trinity College Dublin. Chapter 6 of this thesis is based on a research trip under the supervision of Samuel RP-J Ross at the Integrative Community Ecology Unit, Okinawa Institute of Science and Technology Graduate University, Onna, Okinawa, Japan.

In particular, chapters 2 to 6 of this thesis are based on research conducted collaboratively with my supervisors and co-authors, and parts of the text reflect this through the use of collective terms such as "we" and "our". Where such language appears, it is intended to acknowledge the collaborative nature of the research, or to include the reader in the discussion and development of the work.

Anthony Gibbons,



Maynooth, Ireland
August, 2025

Sponsor

This work was supported by the Nature+Energy Project, funded by Taighde Eireann - Research Ireland (12/RC/2302_P2), industry partners (Wind Energy Ireland, NTR Foundation, ESB, SSE Renewables, Asper Investment Management, Ecopower, Energia, EnergyPro, Greencoat Renwables, Ørsted) and MaREI, the Research Ireland Research Centre for Energy, Climate and Marine Research and Innovation.

Collaborations

Andrew Parnell: As my supervisor, Professor Parnell (Maynooth University [2021 – 2024], University College Dublin [2024 – present]) supervised and collaborated on the work of all chapters. This includes reviewing and editing all chapters.

Ian Donohue: As my supervisor, Professor Donohue (Trinity College Dublin) supervised and collaborated on the work of all chapters. This includes reviewing and editing all chapters.

Samuel R.P-J. Ross: Dr. Ross (Okinawa Institute of Science and Technology) supervised and collaborated on the work of Chapter 6.

Emma King: Emma (Trinity College Dublin) contributed to chapters 3 – 5, helping critically with data collection and providing interpretation of results.

Francesco Martini: Dr. Martini (Trinity College Dublin [2022 – 2025], Environmental Protection Agency [2025 – present]) conceptualised the work in chapter 5 by providing a skeleton of the software, as well as contributing meaningfully to the written paper.

Cian White: Dr. White (Trinity College Dublin) conceptualised the work in chapter 5, reviewed the software, and contributed meaningfully to the written paper.

Jane C. Stout: Professor Stout (Trinity College Dublin) collaborated on Chapter 5 by assisting with data collection and analysis.

Courtney Gorman: Dr. Gorman (Trinity College Dublin [2021 – 2023], University College Dublin [2023 – present]) contributed in Chapter 3 by providing ecological motivation and insights for the written paper.

Masako Ogasawara: Masako (Okinawa Institute of Science and Technology) contributed to the collection of the data used in Chapter 6.

Publications

Chapters 2 to 6 of this thesis have been published in, or submitted to, peer reviewed journals. or edited books. Chapter 3 has been published in PeerJ, Chapter 5 has been published in Ecological Informatics. Chapters 2, 4 and 6 have all been submitted.

Peer-reviewed journal articles:

- Anthony Gibbons et al. "NEAL: an open-source tool for audio annotation". In: *PeerJ* 11 (Aug. 2023), e15913. ISSN: 2167-8359. DOI: 10.7717/peerj.15913
- Anthony Gibbons et al. "ExActR: A Shiny app for creating ecosystem extent accounts". In: *Ecological Informatics* 87 (July 2025), p. 103072. ISSN: 1574-9541. DOI: 10.1016/j.ecoinf.2025.103072

Currently Submitted:

- Anthony Gibbons et al. *Generative AI-based data augmentation for improved bioacoustic classification in noisy environments*. 2025. arXiv: 2412.01530 [cs.SD]
- Anthony Gibbons et al. "Monitoring with Machines: A review of computational bioacoustics". (2025) Book chapter of *Wildlife Monitoring: Integrating Conservation and Innovation in Human-Altered Landscapes*. To be published in 2026.
- Anthony Gibbons et al. "Acoustic detection of a rarely vocalising invasive mammal from sparse data". (2025) Journal article.

List of Acronyms

ACGAN Auxiliary Classifier GAN

AI Artificial Intelligence

AUC Area Under (ROC) Curve

BTO British Trust for Ornithology

CNN Convolutional Neural Network

CORINE Coordination of Information on the Environment

CSRD Corporate Sustainability Reporting Directive

DDPM Denoising Diffusion Probabilistic Models

DL Deep Learning

DNN Deep Neural Network

EIAR Environmental Impact Assessment Report

FFT Fast Fourier Transform

GAN Generative Adversarial Networks

GIS Geographic Information System

Hz Hertz

ISTFT Inverse Short-Time Fourier Transform

JS JavaScript

kHz kiloHertz

km kilometre

km^2 square kilometre

MFCC Mel-Frequency Cepstral Coefficients

ML Machine Learning

MW Megawatt

MWh Megawatt-hour

OKEON Okinawa Environmental Observation Network

PAM Passive Acoustic Monitoring

PCEN Per-Channel Energy Normalization

ROC Receiver Operating Characteristic

SEEA System of Environmental Economic Accounting

UI User Interface

UN United Nations

VAE Variational AutoEncoder

VQVAE Vector Quantised-Variational AutoEncoder

Acknowledgements

I'd like to start by thanking my supervisors, Professor Andrew Parnell and Professor Ian Donohue, for all your support over the past four years. Thank you for encouraging me by giving or building on ideas with suggestions and comments at every meeting, without which I would still be in the weeds. I'm also grateful for the many conferences, workshops and research trips you enabled me to attend. Andrew and Ian consistently mentored me on all things research, allowing me to grow both as a researcher and a person. As supervisors, they've shown the value of compassion, unwavering support, and superhuman patience. As individuals, they are inspiring role models whose influence will shape all my future endeavours.

I also want to thank Professor Ken Duffy, Professor Andrew Parnell and Professor David Malone, and all involved in creating and managing the Hamilton Institute at Maynooth University. I have learned so much from the seminars, events and overall buzz that takes place there. For organising events and being an invaluable help with bookings, travel plans, and everything else without a second thought, a special thank you goes to Rosemary Hunt and Kate Moriarty.

I'd like to thank Wind Energy Ireland, MaREI and all the project partners involved in the Nature + Energy Project for supporting my scientific research. I'd like to single out those who gave early input to my research as well as helping crucially with maintaining equipment: Sean McGovern at Energia; Julie and Peter Brett at Ecopower; Sheila Layden at Energypro; Carmel Brennan and John O'Sullivan at SSE and Ted Walsh at Ted Walsh & Associates Ltd; Frank Kelly and James Crowe at ESB; Joe Dalton, Terry Tighe and Noel Ruane at NTR.

Thanks to the Nature + Energy Project team: Professor Yvonne Buckley, Dr. Ruth Brennan, Emma King, Siobhan McLoughlin, Dr. Maryam Norouzi,

and Professor Jane Stout for all your guidance and handling the hard stuff. I also received guidance from colleagues in the E3 project Digitising Biodiversity, namely Linta Antony, Meegan Gower, Litty Matthew and Lucy McSweeney, and their supervisors Professor Nicola Marchetti & Professor Adam Narbudowic, Professor Anil Kokaram, Professor Caroline Brophy, and Professor Michael Cronin.

A giant thank you to all of my collaborators across the chapters featured in this thesis. Without your expert knowledge and willingness to help, the work would be worse off and a lot less fun. To Dr. Francesco Martini and Dr. Cian White, thank you for involving me in the ecosystem accounting project and trusting me to get it done (if not on time, to a reasonable standard). To Professor Jane Stout, Dr. Courtney Gorman and Masako Ogasawara, a big thanks for all your help with any questions I had and stopping me from going too far astray.

A special thank you to two of my collaborators in particular.

To Emma King, thank you for all your help and support from the very beginning of the project. You handled so much of the project from an admin point of view, keeping everyone on track, not to mention doing a PhD on top of that in half the time! You were so helpful with testing the software and providing ecological context and motivation for my papers, without which they'd be much worse off. Thank you also for organising and driving us to all the wind farms across the country without hesitation. I promise I'll get my driving license soon.

To Dr. Samuel R.P-J. Ross for the amazing opportunity to visit the Okinawa Institute of Science and Technology. You managed to juggle our project on top of everything else you do, while making sure to introduce me to so many interesting people across the university and never hesitating to bring me to a favourite place for lunch (Okinawa soba and tacos were next level). If this experience was any indication, you'll go on to have a stellar academic career heading projects and supervising many more researchers. Pass on my thanks to Professor David Armitage as well as Amy, Alex, Dev, Maya, Peter, Toyo and everyone in the Integrative Community Ecology Unit for making me feel so welcome. サムさん、沖縄で大変お世話になりました。とてもいい経験をありがとうございました。

Thank you to Prof. Nicola Marchetti and Prof. Rafael de Andrade Moral for kindly agreeing to serve as examiners of this thesis. Your keen interest in the material, lively discussion and questions you raised all contributed to the quality

of this work. Your suggestions have not only strengthened the final version of this thesis, but also gave me new ideas on how to continue my research.

Big thanks to John Anthony, my closest friend for more than 13 years. You're great at taking my mind off work and covering endless topics with our late night chats. Thanks for broadening my interests and inspiring me as you jump into startup world. Another big thank you to my other housemate Shauna for nudging me along or letting me vent. We had a great 3 years at the apartment, thanks for all the fun and memories there. Though I'm not sorry to have missed Watermelongate.

Thank you to the new friends from the Hamilton Institute that I have worked alongside for the past four years, in particular to Akash, Blake, Rachel, Samara, Dara and Jack. I want to single out my office mates Carl, Sylvia and Ojas for making me an honorary chemist. Special thanks to my friends Chris, Daniel, Ellie, Katie, Margherita and Muireann for your continued encouragement over the years. To all my friends and family across the country in Derry, Dublin, Galway, Kildare, Mayo and Meath, and around the world in Auckland, Bielefeld, Los Angeles, Eidsvåg, Everett, Florence, The Hague, Liverpool and Nuremberg, thank you for being there.

I would like to extend my sincere thanks to the JASP team at the University of Amsterdam giving me real-world experience building things in R and being exemplars of a collaborative technical team. My experience with you was possibly the primary reason why I got accepted into the PhD in the first place. In particular, thanks to Eric-Jan Wagenmakers, Tim de Jong, Bruno Boutin, and Don van den Bergh — I appreciate all your help even 6 years later.

A heartfelt thank you to the team at Peroptyx for giving me my hands-on start with Machine Learning and software development, and trusting me to deliver as your startup hit the runway. Special thanks to Paul McBride and and Maeve Bleahene for their continuing support of my education and appreciating any time I can give. Thanks also to Colm and Pat for their infectious collaborative natures and guidance.

This thesis really began long before my PhD did in 2021, and I'd like to thank some of the teachers and professors who have inspired me since the beginning. Thank you to my primary school teachers for giving the building blocks to bring me to where I am now. To my secondary school teachers, in particular Lorraine

Hannon, Pauline Finucane and Cormac McCarthy, thank you going the extra mile. To my university professors at Trinity College Dublin, in particular Donal O'Donovan, Athanasios Georgiadis and Jason Wyse, for challenging me and persuading me to take my interests further.

Without the love and encouragement of my parents Colette and Tom, I would not have had the confidence to start, never mind finish this PhD. Thank you for being my greatest supporters and always providing a place to go home and clear my head. I am so grateful for how hard you've both worked and all you've sacrificed to give me the opportunities that you have. To my brother Joe, thank you for continued encouragement from afar helping to keep me grounded. We'll always remember our family holiday to New Zealand and thank you for making that experience so special. To my sister Sarah, thank you for your support and hilarious stories to take our minds off work, I hope we keep up lunches in Orient. A final big thank you to my grandmother Teresa, who inspires me every day to keep learning. Lastly, we can't forget about Peaches, the best puppy.

Preface

As I finish this thesis, I am drawn to reflect on what brought me here in the first place. My maths undergrad and summer jobs in the civil service, software development, and a tech startup gave me a fascination with automation and data science. Artificial Intelligence was a hot topic in 2021 and I wanted to contribute meaningful, practical solutions to real-world problems using it. While the COVID-19 pandemic was a motivator due to a lack of job openings, my core idea was to work on a green project while continuing the pursuit of learning and honing skills that are becoming increasingly valuable in industry. This led me to seeking out Andrew Parnell in Maynooth, who accepted me to work on his project of machine learning to quantify biodiversity on wind farms via sound.

I began my PhD in 2021, exploring bioacoustics papers that applied machine learning for wildlife monitoring with acoustic sensors. While developing deep learning classifiers for bird species using web-scraped data, we started collecting our own data at wind farm sites. Ian connected me with bird experts in Ireland to assist with labelling audio clips. To streamline this process, I prototyped a Shiny App, inspired by feedback from the Bioacoustics Slack group who highlighted the challenges of manual labelling. Incorporating input from labellers, particularly Mark Shorten, and later Yves Bas and Yvan le Bras, I refined the app through multiple iterations to enhance efficiency and data granularity. Though our initial paper submission was rejected, we improved the manuscript and GitHub repository, leading to successful publication in 2023.

Wind farm construction disrupts landscapes, and as part of a larger project, I worked on quantifying land cover change using the SEEA framework, previously employed by TCD Ecology groups. Ecosystem Extent accounts measure transi-

tions between land cover types over time. Building on Francesco Martini's existing process for the ForES project and Cian White's experience with remote sensing, this side project became an iterative effort we eventually finished and was published in early 2025.

While scaling our bird species classifier to more species, I hit a wall in performance likely due to poor data quality. I explored various techniques such as data augmentation, transfer learning, and adding a little human-labeled data, but none yielded satisfactory improvements. Manual spectrogram data augmentation methods, such as time shifting or mixup, failed to give meaningful generalised improvements. Anil Kokaram at TCD suggested synthetic data approaches like those used in object detection. At the same time, frontier large language models were beginning to leverage AI scaling laws and synthetic data for training [13, 125, 264] and industry leaders were publicly emphasising synthetic data's role in advancing generative AI models [178, 195]. I explored generative adversarial networks (GANs), given their established literature and finding available code to experiment with. Kieran Gibb at the EcoHacK workshop 2023 recommended a paper [67], which got at the same idea, but my implementation could not produce synthetic samples of sufficient quality. This prompted me to investigate newer techniques like Stable Diffusion, whose samples were much more convincing and improved performance.

Towards the end my PhD, I sought out a research visit at the Okinawa Institute of Science and Technology, connecting with Dr. Sam Ross. Sam agreed to host me for a short project focused on detecting the invasive mongoose, which are less vocal and challenging to monitor in Okinawa's diverse rainforest soundscape. We initially explored a few-shot learning approach using a prototypical network, but it proved ineffective for identifying mongoose vocalisations due to our limited dataset. Recognising the need for greater robustness to distinguish these signals amidst the complex acoustic environment, we sought alternative methods to enhance detection accuracy.

Abstract

When investigating the ecological and behavioural patterns of wildlife through sound, bioacoustic studies using machine learning (ML), such as convolutional neural networks (CNNs), are key for analysing large acoustic datasets. For other biodiversity monitoring methods like ecosystem accounting, practitioners often do not have the required technical knowledge. This thesis presents a series of studies focused on the application of ML methods to bioacoustic data, and providing tools to assess ecological impacts.

We begin by conducting a systematic literature review of Passive Acoustic Monitoring (PAM) to investigate how it's used with ML methods and identify trends or gaps present. We find increases in dataset size over time, and spectrograms being the most popular representation of bioacoustic data for inference, but highlight the need for standardized evaluation methods and broader use of open-source to advance the field.

Our second contribution is NEAL, an open-source Shiny R application, which enables granular annotation of audio data for use in training and evaluating species classification models. Its no-code interface and modular design empower use by non-programmers, improving annotation workflows and supporting machine learning model development in bioacoustics.

Later, we use generative AI, specifically Stable Diffusion, to create synthetic spectrograms for use in training bird species classification models. We use a dataset annotated using NEAL to benchmark these models. We demonstrate that supplementing training data with synthetic samples enhances classification performance on the human-labelled test set.

Our next contribution is ExActR, an open-source Shiny R application which en-

ables environmental project managers to quantify land cover changes using geospatial datasets. It supports ecosystem extent accounting without requiring GIS expertise. ExActR facilitates accessible and reproducible ecological assessments, with potential to expand into a comprehensive tool for ecosystem accounting.

Our final contribution investigates the use of a lightweight CNN to classify sparse vocalisations of the invasive small Indian mongoose in Okinawa using audio from camera trap videos. The classifier was then applied to a large acoustic dataset to gain insights into the distribution of the mongoose across Okinawa and aiding conservation efforts.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Onshore Wind Energy and Biodiversity

The wind energy sector in Ireland has experienced rapid and sustained growth over the past two decades, achieving an annualised growth rate of 16.9% since 2000 (see Figure 1.1). This expansion plays a pivotal role in reducing Ireland's reliance on imported fossil fuels and mitigating greenhouse gas emissions [200]. As a cornerstone of Ireland's energy security, the sector is integral to achieving National Strategic Outcome 8 of Project Ireland 2040[1] and the All of Government Climate Action Plan 2021,[2] which aims to increase renewable energy reliance from 30% to 70% by 2030. Furthermore, wind energy is fundamental to Ireland's commitments under the United Nations Sustainable Development Goals, particularly Goals 7 (Affordable and Clean Energy), 13 (Climate Action), 14 (Life Below Water), and

[1] https://www.npf.ie/wp-content/uploads/Project-Ireland-2040-NPF.pdf

[2] `https://www.gov.ie/en/department-of-climate-energy-and-the-environment/publications/climate-action-plan/`

15 (Life on Land).[3]

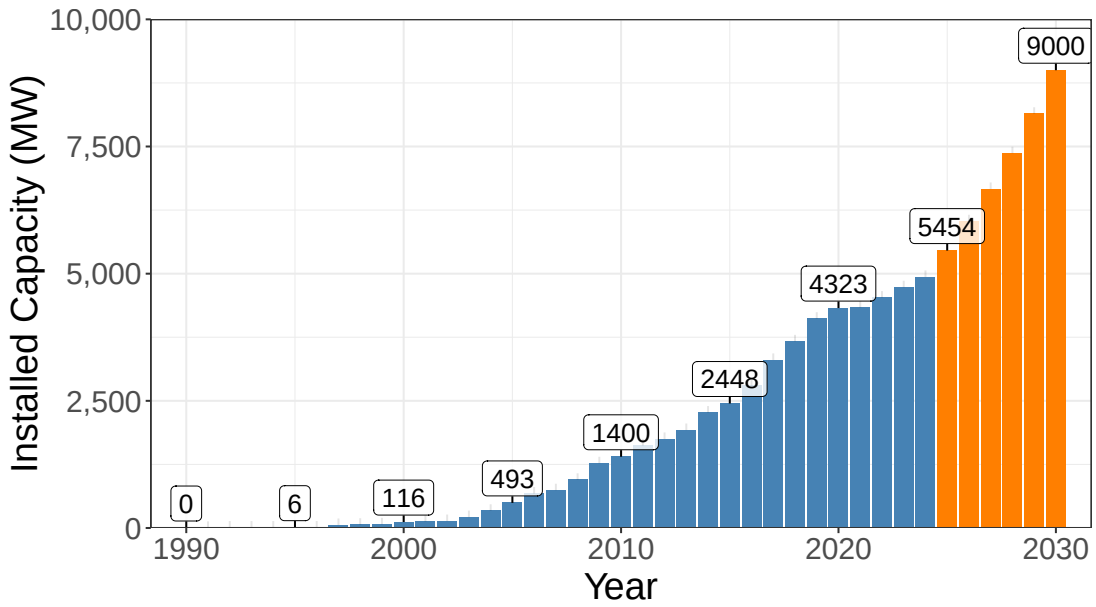


Fig 1.1. Installed wind energy capacity in Ireland since 1990. Illustrating an annualised growth rate of 16.9%. Ireland's first commercial wind farm was commissioned at Bellacorrick, Co Mayo in 1992. The figures in blue are real figures from December of each year, while the figures in orange are estimates based on the above growth rate.

Recent energy policies, such as the EU Renewable Energy Directive [58], have driven the adoption of sustainable solutions to address climate change, with wind energy at the forefront. In 2018, wind energy supplied 30.1% of Ireland's electricity, with the potential to exceed domestic demand and transition to an export-driven model by 2030 [262, 263]. Onshore wind generation in Ireland is among the most cost-effective energy sources in Europe per MWh [14, 263], making its continued development essential for meeting decarbonisation targets cost-effectively. Projections suggest that wind energy could remove up to 450 Megatonnes of CO_2 from the Irish economy by 2050 [263]. However, the sector's sustainability and growth are challenged by a widespread lack of social acceptance [200], necessitating innovative approaches to balance expansion with public support. Strengthening public engagement through initiatives such as community benefit fundsa[4]nd shared own-

[3] https://sdgs.un.org/goals

[4] https://energiagroup.com/news-media/meenadreen-community-benefit-fund-promo/

ership modelsh[5]ave proven to be popular measures to driving Ireland's onshore wind energy industry while maintaining societal support.

Amid the current biodiversity and climate emergencies, onshore wind farms offer more than just energy production. The land they occupy represents significant natural capital, providing habitats that support biodiversity and deliver critical ecosystem services. These services, though currently unaccounted for, are vital for climate change resilience and a sustainable future [169]. With over 300 wind farms [6] currently spanning approximately 290 km^2 (based on installed capacity of 4,934.1 MW and estimate of 16.9 MW/km^2 [294]) and the possibility for expansion to 2,705 km^2, the national wind farm network holds immense potential [128]. If managed appropriately, these sites could enhance climate change mitigation through increased carbon sequestration, improve ecosystem resilience, and support essential services such as pollination across wider landscapes. Realising this potential would yield both direct financial benefits and indirect reputational gains for the sector.

The UN Economics of Ecosystems and Biodiversity (TEEB) Initiative estimates that habitat degradation and biodiversity loss inflict an annual economic cost of US$2 – 4.5 trillion [268]. Preliminary studies [253] indicate that wind farms in Ireland can have both positive and negative impacts on natural capital and biodiversity. Positive impacts arise from habitat creation, with opportunities to enhance these effects through improved land management for conservation. However, birds and bats are particularly vulnerable to the direct negative impacts of onshore wind turbines [200, 269], with effects varying based on wind farm specifications, topography, habitats, and species diversity [253]. Context-specific mitigation strategies and detailed assessments of species and habitat sensitivity are therefore critical areas for further research.

In Ireland, Environmental Impact Assessment Reports (EIARs) are a critical component of the planning and ongoing management processes for wind farm developments, as mandated by national and European Union regulations to ensure sustainable energy projects. These reports comprehensively assess potential environmental impacts on local ecosystems, including biodiversity, water quality, and

[5] https://www.seai.ie/case-studies/co-wind

[6] https://windenergyireland.com/about-wind/the-basics/facts-stats

landscape aesthetics, as well as socioeconomic effects on nearby communities [293]. The process involves detailed studies to identify and mitigate risks to wildlife, such as birds and bats, which are particularly vulnerable to wind turbine operations [111]. Public consultation is also a key element, allowing community input to address concerns like noise or visual impacts [293]. Through rigorous EIAs, Ireland aims to balance the expansion of renewable energy with environmental protection and community well-being [111].

Despite industry efforts to address biodiversity impacts, there remains a pressing need for eco-innovations that simultaneously support the expansion of wind energy, enhance social acceptance, and improve natural capital management. By maximising the positive impacts of wind farms on natural capital and effectively communicating these benefits to stakeholders, including the public, this research aims to address these challenges. This will be achieved through the development of novel natural capital accounting methodologies for widespread adoption across the sector, the construction of a state-of-the-art smart environmental monitoring system to enhance data resolution and reduce operational costs, and the implementation of biodiversity enhancements to maximise the value of natural capital at a landscape scale.

1.2 Nature + Energy Project

The Nature + Energy project outlines specific objectives to guide this effort, as illustrated in Figure 1.2. This thesis focuses on the development of a **WP2: Smart Environmental Monitoring System** and providing tools to help with **WP1: Natural Capital Accounting**.

1.3 Thesis Outline and Contribution

Chapter 2 provides a systematic literature review of computational bioacoustics, with a focus on passive acoustic monitoring (PAM) and a quantitative lens. This chapter evaluates the strengths and weaknesses of PAM compared with traditional manual species monitoring, highlighting its scalability and cost-effectiveness alongside challenges like background noise interference. It explores the signal processing

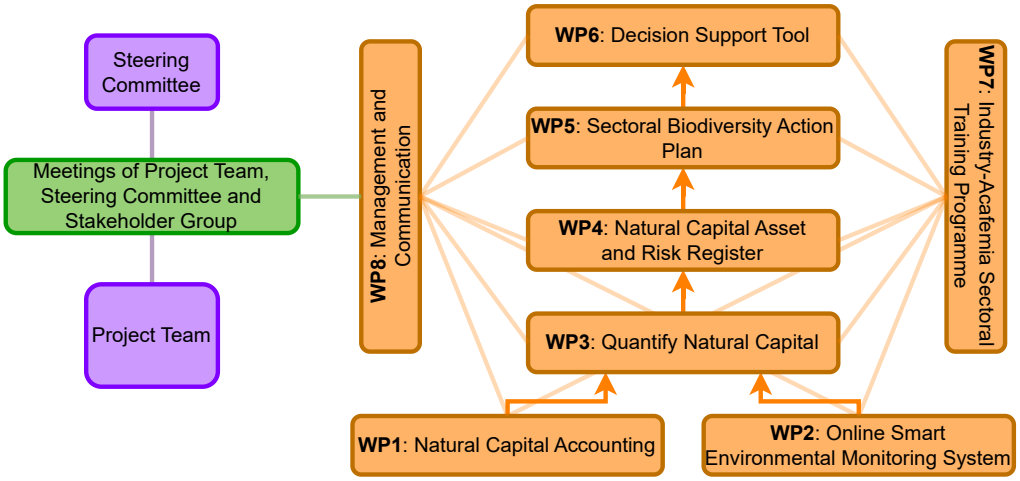


Fig 1.2. Structure and objectives of the Nature + Energy project. The chart outlining key goals for enhancing natural capital and environmental monitoring in Ireland's wind energy sector. The two administrative structure (purple) together provide joint activities (green) leading to project work packages (orange) which each have their own specific deliverables. Reproduction from project proposal

tools commonly employed for acoustic data analysis, and examines how the performance of automated monitoring methods is assessed through quantitative metrics. The chapter investigates the prevalence of open-source code and data practices in bioacoustics research, revealing opportunities for advancing the field.

In Chapter 3, we introduce the Nature+Energy Audio Labeller (NEAL), an open-source Shiny application designed to streamline the audio annotation process for bioacoustic research. NEAL offers flexible deployment, functioning both locally and on servers, and includes advanced features such as frequency filtering on spectrograms to isolate relevant sounds by removing unwanted noise. The application enhances usability by displaying detailed metadata for each audio file, providing critical geographical and ecological context. Users can assign confidence levels to annotations, search and navigate existing labels efficiently, and export annotations in bulk, making NEAL a valuable tool for researchers seeking to improve the accuracy and accessibility of bioacoustic data labelling.

Chapter 4 explores the application of generative AI to enhance bioacoustic classification through synthetic spectrogram generation, leveraging the well-

established Stable Diffusion model from computer vision for bird species monitoring. Our findings demonstrate that incorporating synthetic spectrograms into training datasets improves classifier performance across multiple models, enhancing bird song classification accuracy while reducing reliance on costly expert-labelled data. The hold-out test set from Chapter 3's NEAL annotations validates these improvements.

Chapter 5 presents ExActR, an open-source Shiny application for ecosystem extent accounting, building on techniques developed for NEAL in Chapter 3. ExActR enables users to dynamically select grouping variables from their data and generate extent accounts for two or multiple time points, facilitating detailed tracking of ecological changes. The app provides a suite of visualisations to illustrate shifts in ecosystem extent and supports seamless copying of tables and plots to the clipboard or bulk export of session outputs for reporting purposes. ExActR empowers researchers and conservationists with an intuitive, flexible tool to monitor and communicate changes in ecosystem dynamics.

Chapter 6 investigates the detection of the invasive small Indian mongoose in Okinawa as part of the OKEON project, integrating methodologies from Chapters 2 – 4. We created a novel dataset of mongoose vocalisations extracted from camera trap videos and developed a lightweight Convolutional Neural Network, distilled from a more complex model, to classify mongoose contact and alarm calls. This classifier was applied to approximately 500 hours of audio collected over eight years in Southern Okinawa, enabling the assessment of temporal trends in mongoose vocal activity.

Finally, we conclude our work in Chapter 7 by discussing the contributions, limitations, and promising areas of future work for each of the chapters of this thesis. All proposed methods in this thesis were implemented using R and Python, and their accompanying code are all openly available. In addition, all datasets are either openly available or accessible upon request.

2

Monitoring with Machines: A review of computational bioacoustics

Anthony Gibbons [1], Ian Donohue [2], Andrew Parnell [3]

1 Hamilton Institute, Department of Mathematics and Statistics, Maynooth University, Kildare, Ireland
2 Zoology, School of Natural Sciences, Trinity College Dublin, Dublin, Ireland
3 School of Mathematics and Statistics, University College Dublin, Dublin, Ireland

Abstract

Ecological monitoring is essential for understanding biodiversity amid rapid global change. Within wildlife monitoring, Passive Acoustic Monitoring (PAM) has become a powerful and commonplace tool. Advances in computational meth-

ods—fuelled by affordable data collection and machine learning—have accelerated its growth, yet challenges persist, particularly in data integration, species coverage, and classification model evaluation. In this paper, we systematically review the state of PAM, analysing approximately 80 studies (2005–2024) spanning diverse taxa and ecosystems. We evaluate its strengths, limitations, and applications. Our findings highlight significant increases in dataset sizes over time, with spectrograms dominating preprocessing methods. However, there remains a lack of consensus on evaluation methods for machine learning models, and code sharing has outpaced data availability. We propose a roadmap for future research for closing identified gaps, which includes advocating for open-sourced data and code, improving data preprocessing, and providing accessible model hosting. Broad uptake of these recommendations will enhance impact of computational bioacoustics on ecological science.

Keywords— Bioacoustics, Passive Acoustic Monitoring, Species Recognition, Machine Learning

2.1 Introduction

Wildlife monitoring is critical for understanding ecological dynamics, both on land and in marine environments. Tracking bird species diversity can act as a proxy for habitat health and is a commonly monitored metric for signaling broader environmental stress [242]. Bat surveys safeguard vital roosting habitats, such as caves and forests, protecting vulnerable species from threats [120]. Insect monitoring can function both to support pollinator restoration efforts or with pest control [65]. Similarly, marine surveys can serve either tracking invasive species or detect overfishing or impacts of pollution in the water [97].

Traditionally, species monitoring surveys have relied on human-intensive methods such as camera trapping [26], DNA barcoding [140], capture-recapture [38], and radio tracking [87]. However, the planning and environmental impact assessment of infrastructural developments such as wind farms often necessitate even more detailed monitoring, especially in Special Areas of Conservation where species like birds and bats are of particular concern due to potential mortality risks from wind turbine collisions [296].

In recent years, acoustic monitoring has become a vital tool in wildlife research, driven by advancements in hardware that have enhanced the efficiency and scale of data

collection. *Bioacoustics*, the scientific study of animal sound, involves analysing animal vocalizations and soundscapes and investigating their ecological behaviour. Meanwhile, *Passive Acoustic Monitoring (PAM)* refers to the use of acoustic sensors to non-invasively capture environmental sounds. Although the terms are occasionally used interchangeably in this review, we distinguish Bioacoustics as pertaining to the signal processing and analysis of audio datasets, while PAM specifically denotes the surveillance technique itself.

The adoption of PAM has accelerated the shift toward big data in bioacoustics, enabling researchers to handle increasingly large and complex datasets. Moreover, citizen science initiatives such as Xeno-canto[1] and Cornell Lab's eBird[2] have contributed to this trend by expanding the scale and scope of data collection [63].

The integration of Artificial Intelligence (AI) and Deep Learning (DL) has shifted bioacoustic research away from traditional methods, improving large-scale data processing and species identification accuracy with minimal human effort [5]. This review assesses these advancements, their impact on wildlife monitoring, and future directions, focusing on the migration towards more fully autonomous methods. Four specific research questions are addressed:

1. What are the strengths and weaknesses of PAM and bioacoustics compared with traditional (manual) species monitoring methodologies?
2. What signal processing tools and methods are used to analyse bioacoustic data?
3. How do the diversity (defined here as the variety of taxa monitored), use and performance of automatic monitoring methods compare?
4. What is the level of open-source code and data practices in bioacoustics research?

The structure of this paper is as follows. In the Survey Methodology section, we describe the systematic process used to select and analyse the literature. In the Results section, we present the primary findings of the review based on the above research questions. We conclude with some recommendations of best practice moving forward in using autonomous (in particular machine learning) methods in bioacoustics.

[1] `https://xeno-canto.org/`
[2] `https://ebird.org`

2.2 Survey Methodology

Our review employed a systematic approach to identify and evaluate relevant studies, guided by established protocols for comprehensive literature searches. The methodology encompassed a broad search strategy with a preference towards articles in peer-reviewed academic journals. The process involved defining clear inclusion and exclusion criteria, followed by a structured assessment to investigate variables related to the research questions across the resulting body of literature.

PAM is now a staple of ecological monitoring by utilizing low cost recording and data storage equipment. While audio processing and machine learning methods did not gain much prominence in the bioacoustics field until the late 2000s [118, 211], and deep learning not until 2016 [82], manual surveying by expert ecologists and ornithologists has remained common in fieldwork since the early 2000s [24, 51, 143]. We have therefore constrained our search to papers published since 2000.

We used Google Scholar, Scopus and Web of Science for the initial literature search, which was carried out on 19th November 2024, with the following search string:

```
("passive acoustic monitoring" OR "species monitoring" OR bioacoust*
OR ecoacoust* OR vocali* OR biodiversity OR "soundscape" OR
"biodiversity assessment" OR "species detection") AND (animal OR
bat* OR bird* OR cetacean* OR insect* OR mammal*) AND (acoustic
OR ultrasonic OR harmonic OR "signal processing" OR software OR
"data analysis" OR "classification accuracy" OR "machine learning"
OR "deep learning" OR "statistics" OR "acoustic indices") AND
("automatic monitoring" OR "automated detection" OR "manual methods"
OR "traditional methods" OR "traditional monitoring")
```

This search returned 277, 45 and 50 entries on Google Scholar, Scopus and Web of Science respectively, or 372 entries in total. The literature was then filtered by removing duplicates, entries with missing abstracts, those not in English, and papers unavailable or tied to workshops and conferences. Further screening excluded literature reviews, off-topic papers, and items like theses or non-peer-reviewed preprints. In the final manual filtering stage, some exclusion criteria, such as not being in English or being review papers, were applied again since some results were not identified initially through metadata. Fig. 2.1 gives a visual breakdown of inclusion and exclusion criteria.

This finally yielded 79 entries, ranging from 1 entry in 2005 to 16 entries in 2024.

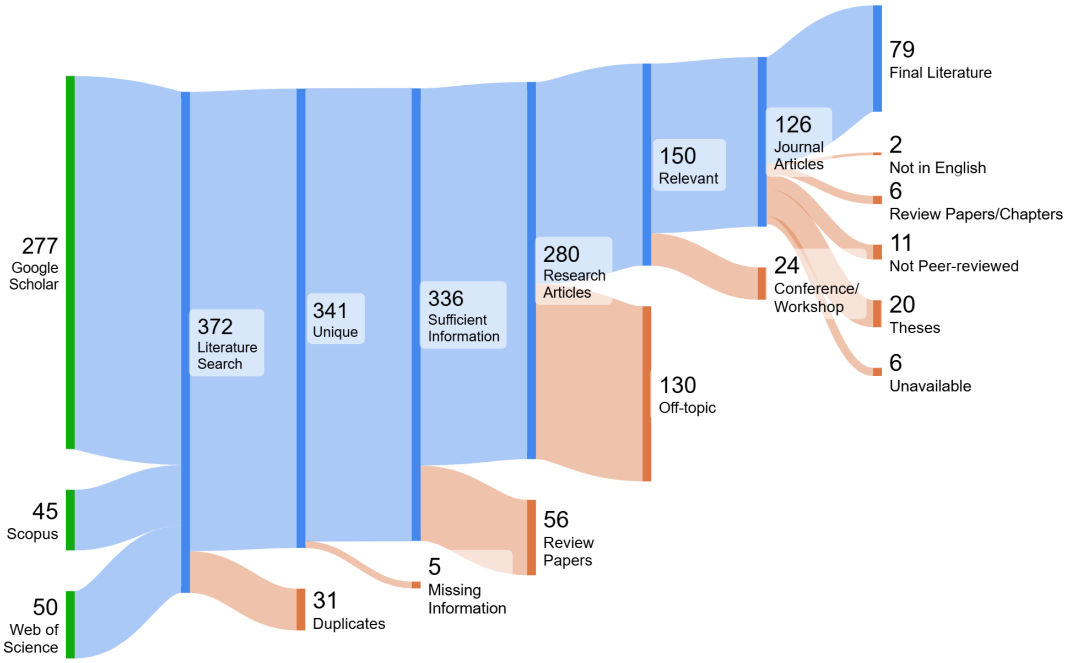


Fig 2.1. Systematic review collection and filtering stages. (created with `https://sankeymatic.com/`)

Five years (2006, 2008, 2010, 2011, and 2012) had no entries. The `.bib` file published along with this paper lists all articles, books, proceedings, and reports resulting from this screening process, along with other articles added for context while writing the review. See `https://github.com/gibbona1/pam_lit_review`. Figure 2.2 provides a detailed depiction of the geographic distribution of study sites referenced in the literature. The points are coloured by their primary ecosystem types studied, with a notable concentration of sites along coastal regions. All the continents, excluding Antarctica, have some level of representation among the included literature.

From each article in the resulting literature list, the following key information was extracted: publication year, monitoring year, location of the study site, monitored taxa, dataset size and duration, preprocessing and analysis methods employed, programming languages used, data and code availability, reported metrics and their corresponding values, evaluation methods used, and research gaps/limits identified. Some of this information was not available in the papers themselves or their supplementary information, so estimates were made when feasible, otherwise fields were left blank. In particular, dataset size often had to be estimated using the length (in hours) of the data collected and the recorder's chosen sampling rate and bit depth. Similarly, the latitude and lon-

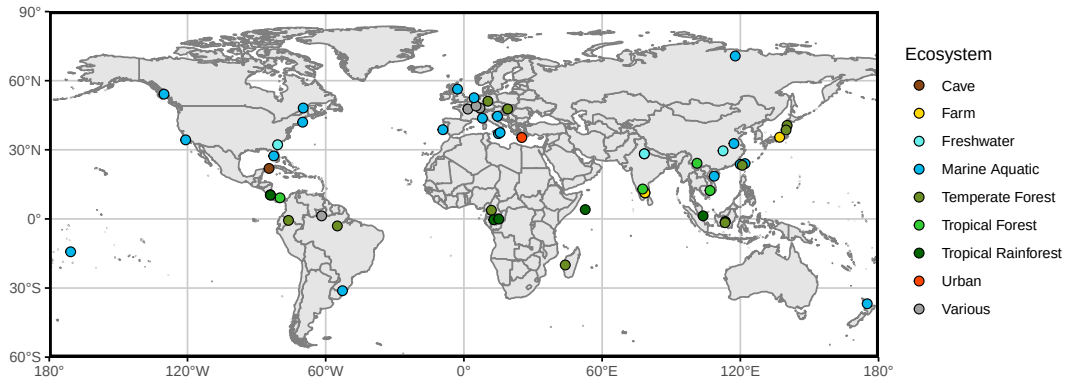


Fig 2.2. Geographic distribution of the study sites in the literature. The map has been cropped to remove everything below 60°S, which included Antarctica, as there were no studies present.

gitude often had to be estimated based on Google Map searches of the reported sites. Furthermore, a number of columns in the extracted data available in the Supplementary Material are not included here as they are free text fields that we used as reference for calculation or categorizing discrete values. Table 2.1 gives a full breakdown of the chosen variables.

2.3 Results

In this section, we begin by comparing traditional manual acoustic monitoring with passive methods and emerging trends seen in data collection. We then analyse preprocessing techniques, analysis methods and automation tools utilized in bioacoustic studies. Next, we explore evaluation methods used to assess acoustic classification model performance. We finish with some comments on the state of open-source in bioacoustics research.

2.3.1 Traditional versus Passive Acoustic Monitoring

Traditional acoustic monitoring surveys were heavily reliant on human expertise and simple tools (e.g., listening or non-programmable recorders), with a history of use akin to traditional trapping. While fully manual data collection and annotation is occasionally still done, it is often in a lab environment ([85] for example).

By manual monitoring, we refer to studies where humans were responsible for collecting and processing all the data. Autonomous methods rely entirely on computational

Table 2.1. Description of variables extracted in this review.

Variable	Description
`Source Type`	Document type, e.g., book chapter, journal article.
`Year`	The year the study was published.
`Monitoring Year`	The year or range of years when data was collected.
`Study Sites`	Description of where the research took place.
`Latlong`	Coordinates (latitude,longitude) of the study sites.
`Dataset Hours`	Total hours of data collection or monitoring time.
`Dataset GB`	Size of the dataset in gigabytes.
`Taxa`	The taxonomic group(s) monitored.
`Analysis`	Automated, Semi-automated or Manual.
`Preprocessing`	Data cleaning or preparation steps before analysis.
`Analysis Methods`	Methodologies and techniques used.
`Automation Tools`	Automation tools or software used in analysis.
`Programming Language`	Programming language(s) used.
`Code Availability`	Whether the code is publicly available (Yes, No).
`Data Availability`	Whether the data can be accessed (Yes, No).
`Evaluation Method`	Method used to assess results or model performance.

techniques, such as machine learning or other software, to process nearly all the data. Semi-autonomous methods combine elements of both approaches. The vast majority of the literature required at least some level of human annotation and inference for at least part of the data collected to act as a ground-truth evaluation method, so we did not categorize papers using this approach as wholly manual. Thus, papers identified as using automated or semi-automated acoustic analysis can be classified as utilizing PAM.

Traditional manual monitoring exhibits several strengths, arising from a longer history of use than PAM. It is effective in targeting individual or a small list of taxa [201]. There is a low technological barrier, usually requiring minimal equipment such as handheld recorders or simply documenting by ear. The data collection process is thus less likely to fail when compared with passive methods. Direct human observation allows for real-time interpretation of sounds, which supports immediate management decisions like crop pest control [139]. Lastly, maintaining the ability to distinguish species by ear or through basic analysis with trained listeners is needed to validate data [50].

Disadvantages of traditional acoustic monitoring methods stem from the limitations of human surveyors. These methods are labour-intensive, requiring active human pres-

ence and listening effort, similar to the demands of sweep netting [201]. They rely heavily on taxonomic expertise, to identify dozens of species by sound, a scarce resource [292]. Even so, human observation introduces subjectivity, leading to observer bias and inconsistent, often unverifiable, results, much like transect walks, which hampers reproducibility [201]. Manual monitoring offers limited scalability due to the increased labour cost, rendering it ineffective for large areas or extended time periods. Finally, temporal constraints mean traditional methods miss ecological activity outside observation windows, such as at night, due to safety and effectiveness risks for surveyors [201]. These strengths and limitations are summarized in Table 2.2.

Table 2.2. Strengths and Weaknesses of Manual Acoustic Monitoring

Aspect	Strengths	Weaknesses
Effectiveness	Works well in targeted settings	Limited scale
Accessibility	Low tech barrier	Labor intensive
Real-time use	Direct observation	Temporal constraints
Precision	Species specificity	Expertise dependent
Consistency	-	Subjectivity

The four papers tagged as performing manual monitoring addressed some drawbacks or limitations to their methods, namely:

- The study by Picciulin et al. [199] analysed the impact of boat noise on fish and dolphin biophony. Future research aims to mitigate anthropogenic noise and expand to species-specific contributions to the soundscape.
- Papin et al. [189] developed an acoustic localization method for monitoring grey wolves at large scales. It mentions integration of automatic detection and localization, as well as expanding to more species.
- Sanchez et al. [236] recorded and analysed the echolocation calls of *Natalus primus* bats to support conservation. The research was constrained by a limited study area and the authors seek to next explore diverse habitats.
- Milanelli et al. [168] examined temporal patterns in the soundscape of a port area within an urban estuary. Plans for further research includes understanding the influence of soundscape on species and assessing long-term anthropogenic noise impacts.

PAM has become the prevailing analysis setup for acoustic-based wildlife monitoring. Fig. 2.3 shows the dominance of automated and semi-automated methods as opposed to purely manual monitoring for the included literature, growing stronger across time. Manual monitoring features sparingly throughout the study period, appearing only four times. These studies focused on bats, carnivores, fish and marine mammals. Semi-automated analysis has been used relatively consistently over time, though at relatively low levels. Both fish and marine mammals had roughly equal amounts of papers using automated and semi-automated analysis, owing to the more difficult data collection process associated with ocean ecosystems. Automated methods are by far the most popular analysis method since 2020. The trend appears to show that the field will continue to grow in line with those using computational methods.

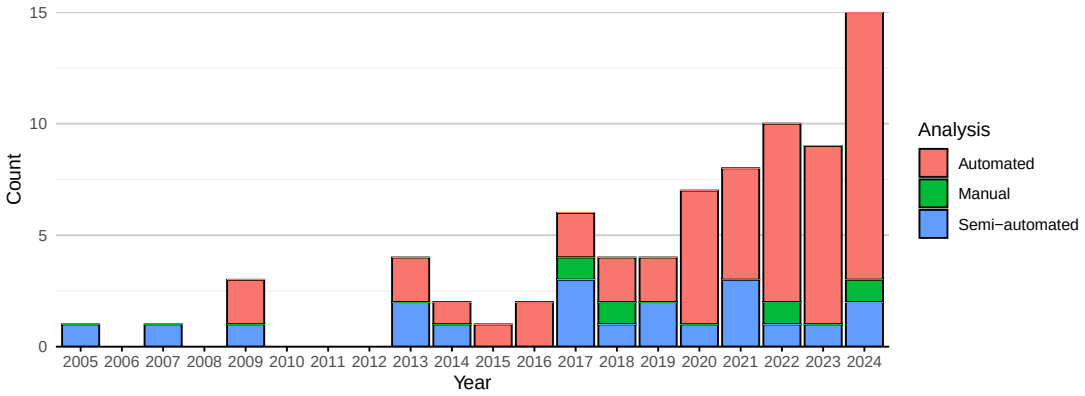


Fig 2.3. Analysis type by year. In the early 2000s, a small number of studies used automated and semi-automated analyses. Over time, automated methods grew rapidly, surpassing other approaches and accounting for over two-thirds of studies by 2024. Studies employing purely manual methods were rare as the volume of acoustic data collected often requires computational tools.

Anthropogenic noise can be reduced with passive acoustic recorders due to the lack of a requirement for a human or machine/vehicle present. Expanding the space- and time-scale of studies quickly outpaces the labour budget and so autonomous data collection is a worthwhile step [187, 241]. The ease of expanding PAM projects compared to the costs of purely manual monitoring efforts allows for greater efficiency in gathering data, calculating species diversity metrics by leveraging economies of scale. The growth in data collection over time exemplified this migration to acoustic monitoring as a 'big data' problem. Exponential dataset growth, facilitated by cheaper audio data collection and storage hardware, quickly surpasses what can feasibly be processed by humans alone,

and so automatic methods are employed to perform inference to match the scope and granularity of data collection. Fig. 2.4 shows the size and length of the datasets used in the studies, when available (note the log scale on the vertical axis).

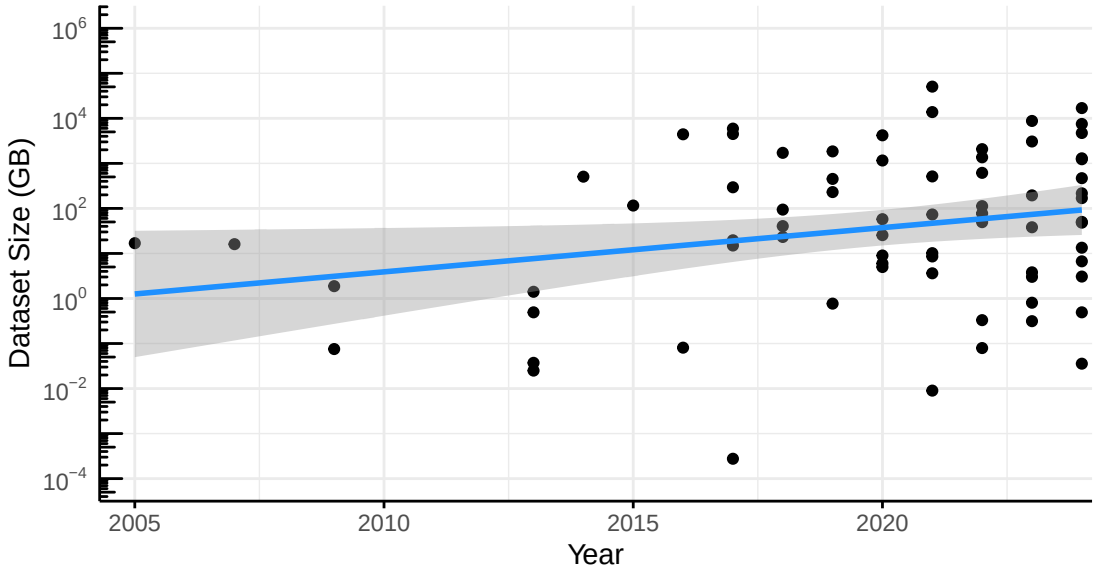


Fig 2.4. Dataset size of studies, in log scale. The dataset sizes in gigabytes, when available, is shown by year. The data size ranges from under 1 MB to 50 TB. The slope of the fitted line is 0.226 (95% CI: [0.016, 0.436]) shows a significant positive trend. Given the log scale, this suggests exponential growth in dataset size over time.

Several of PAM's weaknesses, however, can be regarded as strengths of traditional methods, several of which were mentioned above. The migration towards deep learning (DL) has raised the technological barrier for non-technical ecologists, particularly in terms of data quality, microphone range [81], reliance on a singular mode (sound) compared to humans (sight and sound), and high capital expense of equipment. Studies often employ classifiers with a static taxa list and frequently mention future plans to expand it [213, 261], whereas expert ecologists demonstrate greater flexibility. The question of recorder placement and management often arises [189, 203]. Data management poses additional challenges [6, 272].

2.3.2 Signal processing tools and software

The standard acoustic monitoring pipeline typically involves preprocessing before the bulk of the computation (e.g. DL Models, species classification) is carried out. Spectro-

grams, by far the most common of these techniques used in bioacoustics, are a visualisation of sound that can be treated like an image and can then draw from decades of computer vision research. Spectrograms can be further modified with the mel scale [116], log magnitude [272], per channel energy normalization (PCEN), and extracting Mel Frequency Cepstral Coefficients (MFCC) [277]. Data augmentation is still somewhat rare in bioacoustic studies [116, 188] but is widely used in Computer Vision.

A comparison of preprocessing methods used in each decade in the study period is given in Fig. 2.5. 'Other Methods' are those that only appeared once across the surveyed literature. Frequency Filters, including High-pass [69] and Band-pass [214] filters, feature in each decade. Low-pass filters are much less common in acoustic monitoring studies since the vast majority of biophony occurs at higher frequencies and so that signal should be maintained. Segmentation, in which audio clips are split into (possibly overlapping) segments isolate specific sounds or events. It is a standard preprocessing technique in bioacoustics and may not always be explicitly mentioned in the data or methods sections of the studies; its relative prevalence is, therefore, likely to be underestimated here.

Bioacoustic researchers utilise a variety of programming languages for their data analysis, with Python, MATLAB and R being the most commonly used. They feature in 35, 24 and 16 of the featured literature items, respectively, although MATLAB has become less prominent since the 2020s. MATLAB offers excellent audio functions, signal processing capabilities, and integrated development environment, making it intuitive for beginners. However, it is proprietary and it is unclear if the scripts used could be run in Octave, its open-source equivalent. Python and R are both open-source. Python excels in deep learning applications and provides extensive libraries for audio analysis and signal processing. It has become the default language for deep learning in bioacoustics. The most commonly used Convolutional Neural Network (CNN) architectures such as ResNet, VGG, MobileNet and Inception all have implementations, some even have possibility of loading weights from instances trained on the ImageNET dataset [43]. R is particularly strong in statistical analysis and offers several widely used audio packages. Fig. 2.6 gives the distribution of programming languages used over the survey period.

The counts of analysis methods and automation tools used is shown in Fig. 2.7. Many papers feature more than one analysis method or automation tool and so the sum of counts may not equal the number of papers analysed. 'Analysis methods' describes the procedures or techniques employed in the studies, not the software itself. This includes classification methods, Machine Learning models (CNN, Random Forest, Hidden Markov Model), sub-fields of deep learning (Transfer learning, Data Augmentation, En-

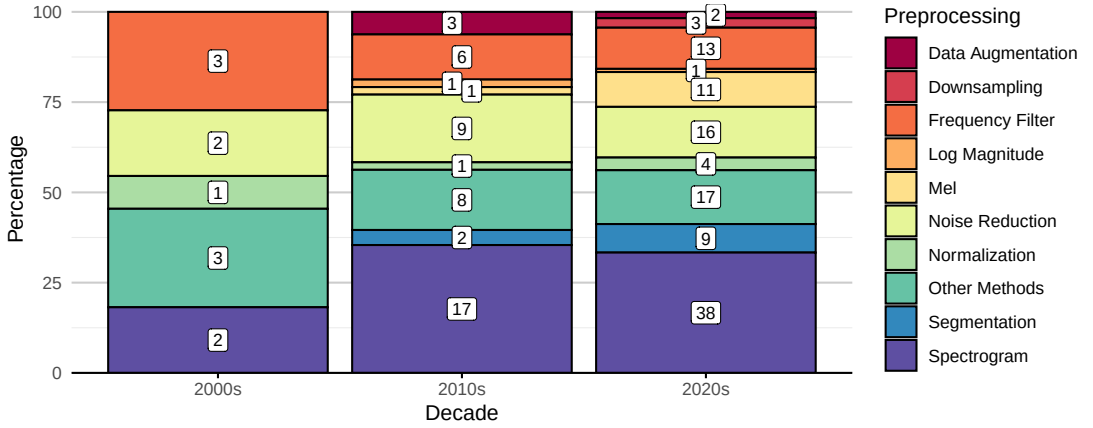


Fig 2.5. Preprocessing methods per decade. The y-axis shows the percentage breakdown of preprocessing methods used per decade. Stacked bars show the number of studies using each method. Since some papers use multiple methods, bar totals do not match the decade's paper count. Spectrograms are the dominant method overall, though Frequency Filters (including manual, band-pass and high-pass filters) were slightly more common in the 2000s. Normalization, common in computer vision, was frequent in each decade. The category 'Other Methods' increases in each decade showing the occasional need for niche techniques. We see the number of unique methods (including 'Other Methods') per decade increasing from 7 in the 2000s to 26 in the 2020s. Papers published in 2024 alone used 14 distinct methods. Mel-based methods (both mel spectrogram and MFCC) showed up once in the 2010s then increased significantly to 11 times in the 2020s.

semble learning), and clustering noise reduction. CNNs dominate here as classification tasks have leapfrogged off computer vision enhancements signalled by the AlexNet [136] paper. Species accumulation curves plot the cumulative number of unique species detected as sampling effort (e.g., survey hours, number of sites, or volume of recordings) increases, and tend to asymptote as most species in an area are documented. They are used to indicate the rate at which a particular site or temporal resolution gains species richness [166]. 'Other Methods', a collection of approaches that each only appeared once in the literature (once or twice for automation tools), shows a large amount of rare applications. Papers tagged with 'Custom Code' refer to studies where the authors developed their own code for specific parts of their work, as opposed to using only readily available software.

Looking at automation tools, these are software used to process the data in some way. The tools were used to facilitate data annotation (Raven [121], BirdNET [123],

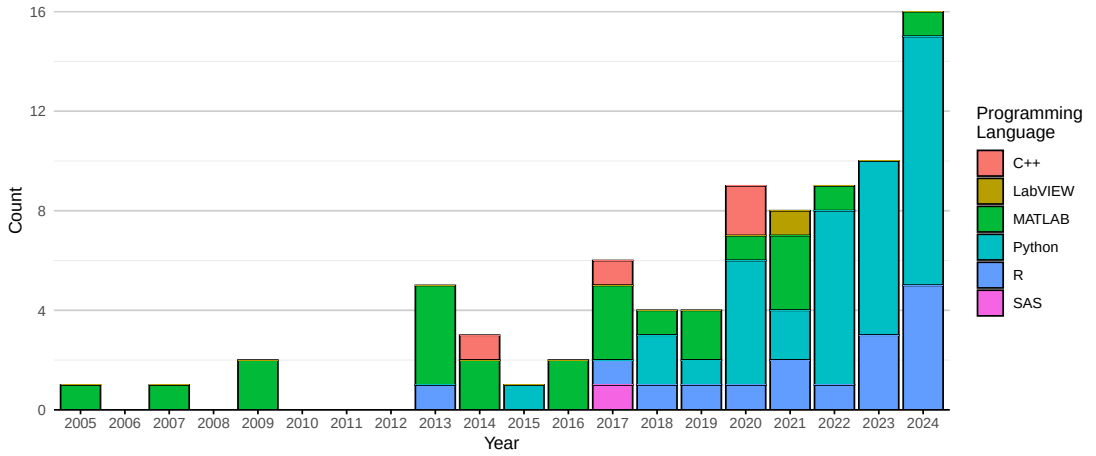


Fig 2.6. Programming language used per year. There is clear movement from MATLAB and C++ to more data science oriented languages, i.e., Python and R. Python's ML frameworks such as PyTorch and TensorFlow dramatically reduce the time taken to train ML models. R also has a range of packages for bioacoustics research such as `seewave` [259]. SAS and LabVIEW each featured once and so it is difficult to say whether they are likely to gain popularity.

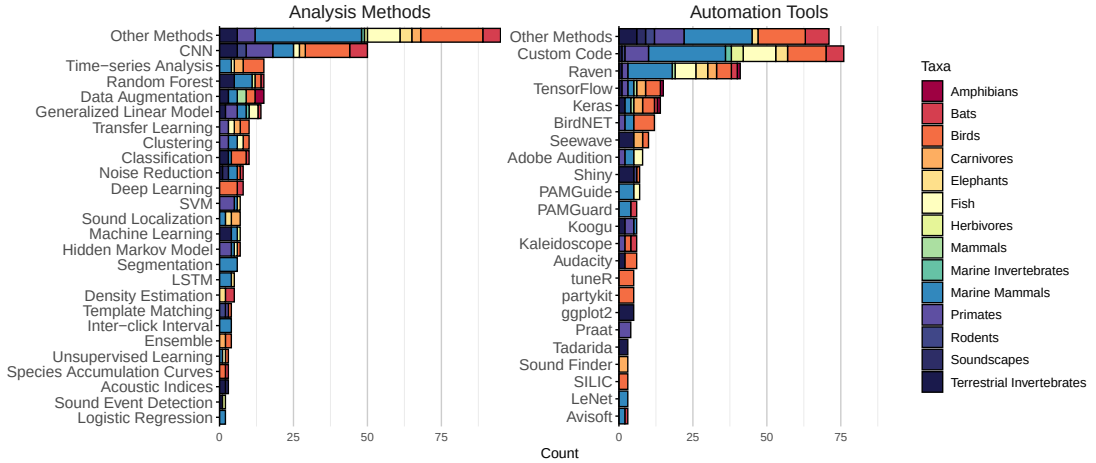


Fig 2.7. Popular Analysis Methods and Automation Tools by Taxa. Most of the obscure Analysis Methods were only used on birds and marine animals. 'Other Methods' was by far the largest category, with the majority going towards fish, marine mammals and birds. For Automation Tools, 'Custom code' features the most, showing the lack of out of the box solutions available for the range of analyses carried out. Marine Mammals require the largest amount of custom code.

Koogu [159], Tadarida [16], Audacity [8], Kaleidoscope [289], BatDetect [157]); as classification model architectures and frameworks (LeNet [145], YOLO [210], TensorFlow [1],

Keras [37], and PyTorch [194]); or helper packages (ggplot2 [286], tuneR [148], partykit [105]). The dominance of custom code shows the field's growing ability to adapt to computer science practice. However, the lack of a standard to make these trained models work off the shelf necessitates such a tailored approach. Accessible user interfaces such as Raven and Audacity as well as the plethora of Shiny apps helps non-technical ecologists and conservationists work on such projects; the literature features two such Shiny apps developed [214, 229]. Raven/RavenPro, TensorFlow, Keras and BirdNET are also popular software used. A large proportion of these tools are packages: PAMGuide and PAMGuard are MATLAB; tuneR, Shiny and seewave are R packages; Keras, TensorFlow, partykit and koogu are python packages. Almost all of the automation tools were used on birds or Marine Mammals, as these are the most common taxa in bioacoustics studies. 'Other Methods' here combine those tools used two times or less across the literature in order to simplify the figure.

2.3.3 Classification models

Species classification models in bioacoustics are evaluated using a suite of methods to ensure robust performance in detecting and classifying biological sounds. Counts of the evaluation methods used in papers in each year of the study period are shown in Fig. 2.8. There is, however, no common evaluation metric for these studies and so comparison of model performance across time or by taxa or method is difficult. Across all taxa, the reported mean reported metrics are: AUC (0.853%, n=2), Accuracy (0.897%, n=34), F1 (0.846%, n=6), Precision (0.921%, n=6), Recall (0.943%, n=4), Sensitivity (0.770%, n=2).

2.3.4 Open Science Practices

The landscape of processing PAM datasets is evolving, with increasing amounts of data and code being openly available, but still falling short of being fully accessible or compatible. Initiatives like DCASE [258] and Kaggle competitions such as BirdCLEF [134] have led the way in promoting open-source practices, requiring participants to share code in their submissions. However, full open-source — requiring model architectures, weights, training code and data being open — remains rare outside of these competitions. Model hosting, on platforms such as Hugging Face[3], has seen only limited uptake in bioacoustics [249] despite its potential to streamline access and use of such state of

[3] `https://huggingface.co/`

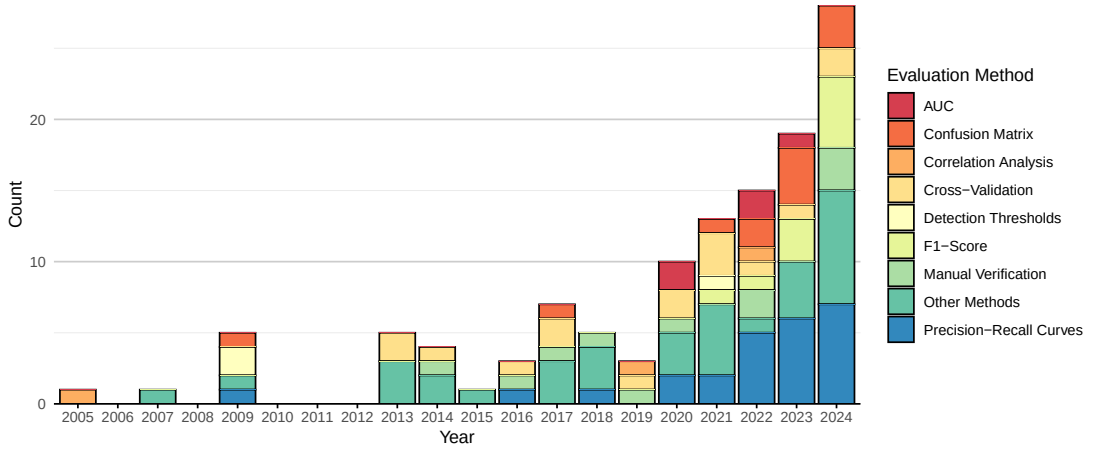


Fig 2.8. Evaluation Methods. There is no clear consensus in evaluation methods over time. AUC and cross-validation are good measures of overall performance, while confusion matrices detail prediction outcomes across categories. Detection thresholds optimise sensitivity, and manual verification confirms results against expert checks. Again the 'Other Methods' category is a collection of methods that only appeared once across the study period. Precision-Recall curves are the most common since 2020.

the art models. Open-source repositories such as GitHub [110], Dryad [52] and Zenodo [60] provide platforms for sharing, preserving, and accessing scientific code, data, and research outputs, fostering collaboration and reproducibility across the bioacoustics field. Fig. 2.9 gives some insight to the adoption of open code and data over our study period.

2.4 Conclusion

Almost all studies included in our survey needed to develop code of some level in order to implement computational bioacoustics methods. While this is an impressive shift towards automated monitoring of ecosystems, there is still a ways to go before even the most common ecological questions (such as species classification) can be answered in a no-code environment. The studies also exhibit a level of short-term bias; the median monitoring length is 237 hours, and the mean monitoring length is 2.6 calendar years.

Spectrograms and normalization are widely adopted preprocessing techniques in PAM, acting as a common starting point for acoustic analysis. Issues with the quantity and quality of labelled data persist, and techniques such as data augmentation have not

(a) Code Availability by Year **(b)** Data Availability by Year

Fig 2.9. Code and Data Availability Trends by Year. (a) Code Availability by Year shows a steady increase in the proportion of studies sharing code over time. (b) Data Availability by Year indicates a slower but positive trend in data sharing practices. With increasing dataset size, it may be less feasible to host large datasets. By 'available code and data' here we mean open-source materials. If papers mention availability upon request, access is limited or the repository was not found during the review, then this was marked as a No.

been widely adopted to address this. The use of synthetic data as an augmentation technique, which is emerging as a promising approach in computer vision and large language models, is even less popular in bioacoustics, but has been shown to be fruitful [67]. Furthermore, the field often relies on custom code for preprocessing and classifier training or inference, despite there being many widely-available standard tools. Given that many preprocessing and analysis steps—such as audio segmentation, generating spectrograms or feature extraction—are common across studies, a wider adoption of intuitive, user-friendly graphical-interface pipelines could greatly reduce duplicated effort and lower technical barriers for new researchers, though several such tools already exist [16, 244].

Classification models in PAM face their own challenges, particularly as the shift toward deep learning accelerates. While these models offer powerful predictive capabilities, there is a lack of standardised evaluation methods, making it difficult to compare performance across studies or ensure reproducibility. This move toward deep learning also sacrifices explainability, a trade-off that complicates ecological interpretation and undermines its ability to be used in tackling broader ecological research questions, e.g., reliably monitoring rare species which we discuss in later chapters. To address these issues, the field would benefit from greater consensus on metrics and benchmarks, which would be helped with the open data and hosted models as mentioned above. As PAM continues to mature, bridging these gaps will be key to unlocking its full potential for wildlife monitoring and beyond.

3

NEAL: An open-source tool for audio annotation

Anthony Gibbons [1*], Ian Donohue [2], Courtney E. Gorman, Emma King [2], [2], Andrew Parnell [3]

1 Hamilton Institute, Department of Mathematics and Statistics, Maynooth University, Kildare, Ireland
2 Zoology, School of Natural Sciences, Trinity College Dublin, Dublin, Ireland

* anthony.gibbons.2022@mumail.ie

Abstract

Passive acoustic monitoring is used widely in ecology, biodiversity, and conservation studies. Data sets collected via acoustic monitoring are often extremely large and built to be processed automatically using Artificial Intelligence and Machine learning models, which aim to replicate the work of domain experts. These models, when operating

as supervised learning algorithms, need to be trained on high quality annotations produced by experts. Since the experts are often resource-limited, a cost-effective process for annotating audio is needed to get maximal use out of the data. We present an open-source interactive audio data annotation tool, *NEAL* (Nature+Energy Audio Labeller). Built using `R` and the associated Shiny framework, the tool provides a reactive environment where users can quickly annotate audio files and adjust settings that automatically change the corresponding elements of the user interface. The app has been designed with the goal of having both expert birders and citizen scientists contribute to acoustic annotation projects. The popularity and flexibility of R programming in bioacoustics means that the Shiny app can be modified for other bird labelling data sets, or even to generic audio labelling tasks. We demonstrate the app by labelling data collected from wind farm sites across Ireland.

Keywords— Bioacoustics, Ecology, Audio annotation, Shiny app, Machine learning, R

3.1 Introduction

Passive acoustic recording is now a staple of ecological monitoring [219, 95, 260, 217]. Remote sensors can collect vast quantities of high quality audio data at a cost affordable to both academic researchers and citizen scientists. However, the volume of data collected can quickly surpass feasible manual labelling ability, and automatic methods are actively being developed [172, 22, 17].

Application of deep learning, in particular convolutional neural networks (CNNs) [145], to image processing for audio classification is growing in both popularity and effectiveness [99]. High performing models have been produced for bat classification [157], insects [305], aquatic mammals [270]; and bird calls, ranging from relatively few [257] to tens [233] to even thousands [123] of classes. While custom CNNs can be trained from scratch, a lack of large *labelled* bioacoustic datasets [12] can impede model performance.

In recent years, audio classification models have benefited from Transfer Learning [181, 315]. This is where a new model is created with the assistance of a neural network pre-trained using a relatively large dataset for a similar task (such as another audio classification [99] or even image classification [247]). The new model adjusts the predictions of the larger model by performing further training with the small amount of labelled data available for the task of interest. In both the custom CNN and transfer learning

settings, some training data are still required to tune to the specific application area, and routine test data should be annotated to monitor performance over time. For novel tasks such as medical imaging classification, manufacturing defect detection, agricultural yield prediction and marine image classification, domain specialists are often needed to label the initial training data and evaluate the model output over time [238, 179, 251, 141].

Here, we focus on bird species detection, which requires experts to manually label files so that they can be input into a supervised learning algorithm. We present an audio annotation tool that aims to reduce the bottlenecks associated with audio annotation, improving the efficiency of the expert's time, which is often at a premium, and providing finer granularity of labelled data (time-frequency limits, species, call type, additional notes) so multiple classification tasks can be carried out on the same data simultaneously.

In much of the existing audio labelling software, supplemental information important to decision making, such as the exact time of the recording, general location, geographic coordinates and shortest distance to the coastline are often not readily available to the user, reducing the effectiveness of the application as a decision tool. Giving annotators this temporal and spatial information can help contextualise hard-to-classify sound clips. In relatively complicated soundscapes, such as wind farms or urban environments, users often lack the flexibility to temporarily filter out noise and focus on the sound of interest, further increasing labelling time. We include the ability to display various metadata and two methods of filtering audio in our app.

We present *NEAL* (Nature+Energy Audio Labeller), an interactive Shiny app designed for audio data annotation. It allows users to visually and audibly interpret audio files and label any sounds observed and offers time and frequency granularity for precise labeling. The app incorporates metadata to inform labellers, as well as labeller confidence for each annotation to provide quality annotations.

Some of the key strengths of NEAL are that:

- it is primarily intended to be used locally but can also be deployed to a server;
- it has automatic frequency filtering of selected areas of the spectrogram to remove unwanted sounds during analysis;
- it displays clear metadata for each audio file, giving context on the geography, habitat and time of year recordings were taken;
- labellers can specify label confidence behind each annotation, as opposed to each

individual classification (e.g. species of a bird vocalisation) being assumed to have 100% confidence associated with it;

- users can search through existing labels and navigate to those of interest;
- user annotations can be downloaded in bulk.

A comparison between NEAL and popular existing tools for bioacoustic annotation projects ([8, 161, 68, 25]) is shown in Table 3.1. While the Shiny app was built to have a user-friendly layout for manual annotation of bird vocalisations with the option of local or server-side use, it doesn't yet have vast functionality in terms of bulk classification or training custom species classification models in-app.

The Shiny App NEAL (Fig 3.1) is presented together with open-source code and sample data to allow for further modification or deployment. Whilst our focus during development was on bird call detection, the app can be easily changed to enable labelling of other audio tasks. This facilitates the adoption of the Shiny app in projects where complicated soundscapes and data quality may differ greatly among sites and equipment.

We provide the overall layout of a labelling project on the Shiny application, as well as a brief overview of the procedures involved and some of the computational workarounds to avoid computation waiting time. We then demonstrate a workflow of classifying bird species on wind farm sites across Ireland, with step-by-step directions of how the app is utilized. We expand on the input and output formats of the data to allow custom projects to be established easily. Source code for the NEAL App, as well as a link to a working demo on an RStudio Server, is available at `https://github.com/gibbona1/neal`. Portions of this text were previously published as part of a preprint (`https://arxiv.org/abs/2212.01457`).

3.2 Methods

3.2.1 User Interface (UI)

The app was built in R [207] using the Shiny [33] framework. Shiny is itself a package in `R`. No knowledge of `HTML`, `CSS`, or `JavaScript` is necessary to build a simple application in Shiny, but small amounts were used here to enhance certain features. One of the many benefits of the Shiny framework is that its end-users do not need any knowledge of R programming to interact with the data and provide annotations. Shiny has already been

Table 3.1. Comparison of NEAL with other free labelling software

Feature	NEAL	Audacity	AviaNZ	Koe	SonicVisualiser
Date released	2022	2000	2019	2019	2005
Platform(s) compatible	✓ ✓ ✓	✓ ✓ ✓	✓ ✓ ✓	✓ ✓ ✓	✓ ✓ ✓
Built with	R, JS	C, C++, Python	Python, C	Python, JS	C++, SML
Label format (time segments, bounding boxes)	boxes	boxes [1]	boxes	segments	boxes
Label confidence slider	✓	✗	✗ [2]	✗	✗
Band-pass filter for selected spectrogram area	✓	✗[3]	✓	✗	✗[3]
Changeable class list	✓	✗	✓	✓	✗
Dynamic class +/-	✓	✗	✓[4]	✗	✗
Site metadata display	✓	✓	✓	✓	✗
In-app label editing	✓	✓	✓	✓	✓
Operates locally	✓	✓	✓	✓	✓
Deployment to server	✓	✗	✗	✓ [5]	✗
Changeable spectrogram colour palettes	✓	✓	✓	✓	✓
Bulk export annotations	✓	✗[6]	✗[7]	✓	✗[6]
Filter labels by multiple fields	✓	✗	✗	✓	✗
Multiple concurrent (collaborating) users	✗[8]	✗	✗	✓	✗
Visualise multiple spectrograms (comparison)	✗	✓	✓[9]	✓	✓
Bulk classification	✗	✗[10]	✓	✓	✗[10]
Analysing sequence structure	✗	✗	✗	✓	✗
Train a species recogniser in-app	✗	✗	✓	✗	✗

[1] not by default
[2] colour codes
[3] Not by default but there may be plugins available
[4] new species added will appear in the list
[5] also operates on its own server
[6] individual *label tracks* for each file can be downloaded but must be named appropriately
[7] per-species annotations can be exported from batch processed files
[8] multiple users can work on a single server but this can be slow and updates are not immediate
[9] Quick review mode after batch processing
[10] Can view multiple files at once but these must be manually labelled

used in developing ecology-related apps and decision-support tools, such as in species-habitat modeling [298], conservation management [193] and forest structure assessment [245].

NEAL makes use of several open-source R packages [7, 11, 197, 30, 31, 90, 198, 2, 151, 246, 148, 259, 286, 72, 32, 240, 288, 287, 64, 301]. The most notable are:

- *shinyjs*, which allows for custom JavaScript (JS) plugins, giving extended functionality using only a small amount of JS code. This includes toggling pause/play

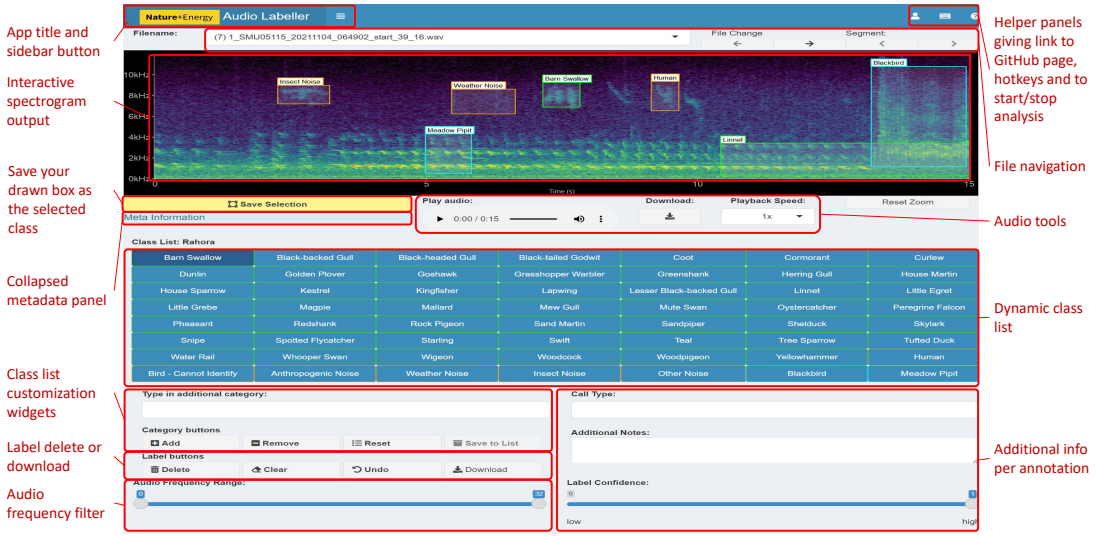


Fig 3.1. Main components of the App User Interface. The main interactive element of the app is the **spectrogram**, providing a visual representation of the sound. This plot can be drawn on with boxes to filter audio or make annotations. The button on the left underneath the spectrogram is for **saving** the current selection on the plot, which will be labelled with the class from the grid of categories below. Various **audio playback** widgets are grouped together to the right of the save button. **File navigation** allows the user to proceed to the next file in the workflow folder, or select from a drop-down menu of available files, the second of which displays the number of annotations present in each file. The collapsed **side panel** on the left hand side has extra configuration settings expected to be changed at most once or twice per session. The **class list** is dynamic: the core (green) categories are selected from a drop-down list in the Configuration tab, while the miscellaneous (orange) categories are static in the current implementation. Custom categories can be added or removed using the category widgets below. Users can provide more information than just the primary category (e.g. bird species) of sound identified; these can include **call type**, free text **additional notes** and **label confidence**.

of the embedded audio element using the spacebar, disabling UI elements and reset buttons.

- *shinyBS* contains extra user-interface (UI) objects such as collapsible panels, giving the app a more compact layout. In particular, the metadata panel, label edit and label summary tables are contained within these collapsible panels and can be opened as needed.

- *shinyFiles* for file and directory navigation. This is especially useful for deployments to server where the folder structure may not be as familiar to users. The folder selected using `shinyDirButton` and `shinyDirChoose` is then searched for audio files.
- *shinydashboard* and *shinydashboardPlus* for the dashboard layout, sidebar and header tabs. Moving the less-used settings and widgets to the sidebar and grouping them into collapsible menus reduces clutter in the Shiny app's main body.
- *tuneR* for reading and writing audio files. It handles the audio files as `Wave` objects during preprocessing - including dB gain, segmentation and normalization - before passing them to seewave functions.
- *seewave* provides audio waveform manipulation functions, such as spectrogram computation. A noise-reduced or frequency-filtered spectrogram can be reconstructed as a `Wave` object using the `istft` function.
- Graphics are implemented using *ggplot2*. The spectrogram is rendered using `geom_raster` and bounding box annotations are drawn with `geom_rect` and `geom_label`. Additional colour palettes are provided via *viridis*.

A full list of packages employed is available in the source code.

Display

Upon opening the app, users are presented with the audio data from the chosen file in the form of a spectrogram, which highlights the sound intensity of many frequency levels over time (Fig 3.1). Spectrograms are one of the most visually perceptible forms for audio data [149]. The audio file can be played with an embedded audio player underneath the visual.

3.2.2 General labelling workflow

The standard use of the app is as follows:

1. Once the user has opened the app and clicked **start labelling**, the first audio file in their workflow is loaded and the corresponding spectrogram is generated.

2. The **spectrogram parameters** such as colour palette and contrast may need to be adjusted until the user is comfortable with the visual distinction of the sounds present.

3. The user plays the audio and, when they come across a sound of interest, they **draw a bounding box** around the vocalisation. This updates the audio player with a **temporary filtered audio file**, keeping only the times and frequencies within the box drawn.

4. Once they have identified the class of sound from the class list, they draw a tight box around the vocalisation and click **save selection**. This will draw a permanent (unless deleted) bounding box labelled with the given class. If they are unhappy with any of the bounding boxes, they can be deleted using the **delete** button in the **label buttons** section.

5. If a new class is being added or deleted, or the base species list is changed, this will affect the **class list** - the grid of classes to choose from. This is only a minor computation and the user should not see any delay unless there are several bounding boxes present in the plot.

6. The user **continues annotating** the file by repeating steps (2) to (5) until no more unlabelled sounds of interest remain.

7. The user clicks on the arrow to **proceed to the next file** and returns to step (2).

The Shiny framework uses reactive programming, meaning that inputs (in the user interface) changed by the user automatically affect those outputs to which they are connected. This gives a smooth experience for users, where the app does not have to be refreshed manually whenever settings are adjusted. In the case of the app, rendering the spectrogram is the most computationally intensive process. Avoiding the plot refreshing every time an unrelated input is changed is key to a seamless user experience. The back-end of the app includes some modular code to split up dependencies (user inputs), which affect outputs of the user interface. We elaborate on how this code works below.

Fig 3.2 shows the full layout of the workflow including some of the back-end components. These components are invisible to the user but are required to reduce redundancy in computation. (1) to (5) correspond to those points in the workflow above.

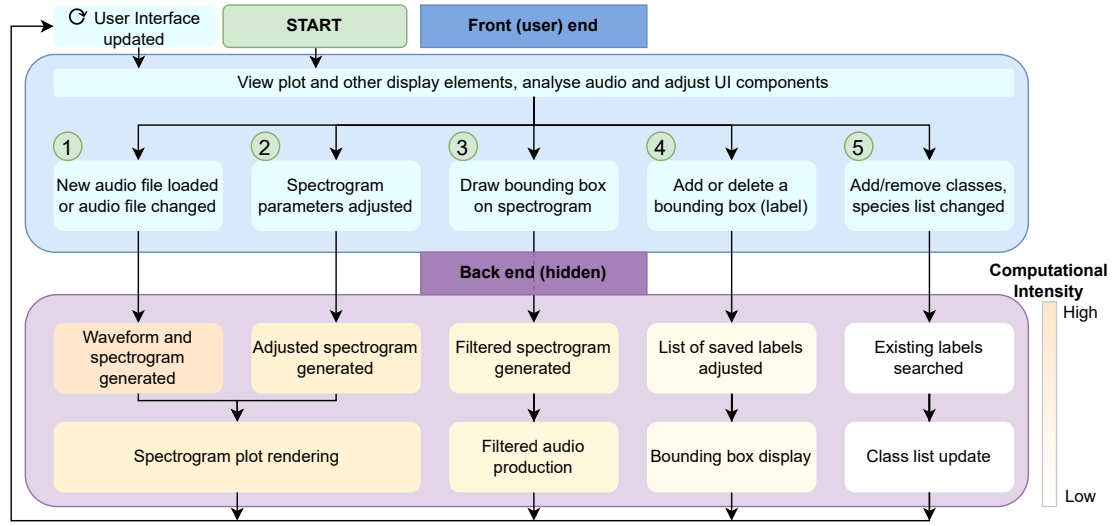


Fig 3.2. Workflow diagram. Demonstration of the front and back-end workflow of the NEAL App. Any analyses or adjustments carried out by the user in the User Interface only affect the relevant processes in the back end. This keeps the user focused only on the elements being updated while reducing downtime compared to the event that the entire page is refreshed every time. Each of the steps (1) to (5) in the front end—the actions and changes performed by the user via the UI—have corresponding processes in the background which are in decreasing order of computational demand. The advantage is that these back-end processes have little to no overlap, meaning they can be run separately when the front-end changes are unrelated. If the change is minor (far right, in white) we do not want this to result in an avoidable refresh involving large computations (shown to the left of the figure in orange).

3.2.3 Back-end workflow computation

Spectrogram plot rendering

While the computation of spectrograms is relatively fast, primarily due to efficiencies gained by implementing the Fast Fourier Transform (FFT) algorithm [21], rendering the result in R using ggplot is slow and creates a noticeable bottleneck for the annotation procedure. As an example, a 15 second audio clip with a sample rate of 24 kHz will produce a spectrogram with dimensions of 128×5622 when applying the `seewave::spectro` function with an FFT window of 256 points and a window overlap of 75% between two contiguous windows. The `geom_raster` function would then have to plot $\approx 720,000$ equally-sized tiles, which is four times that with the linear interpolation needed for ade-

quate resolution, and then colour by the amplitude (in dB) and apply the chosen colour scale.

This computation must be done at least once for each audio file opened, and may be re-run several times when adjusting FFT parameters, colour palette or zooming in on selected areas of the plot. The common ggplot procedure of adding plot components (such as bounding boxes or frequency guides) to the one plot is not ideal here due to the hindrance caused by frequent interactions with, and thus changes to, the spectrogram plot. If even a small parameter was changed (such as adding a class to the class list UI), the entire plot would have to be re-rendered. Avoiding these barriers to labeller efficiency is paramount to the viability of the Shiny app as an audio annotation tool.

To reduce much of this unnecessary computation, we split the computation work of the spectrogram generation into overlapping panels, shown in Fig 3.3. These are often independently generated or refreshed by different widgets in the UI. The panels (c) and (d) have a transparent background and identical axes and padding to the main spectrogram (b). This alignment ensures that all plot layers match exactly.

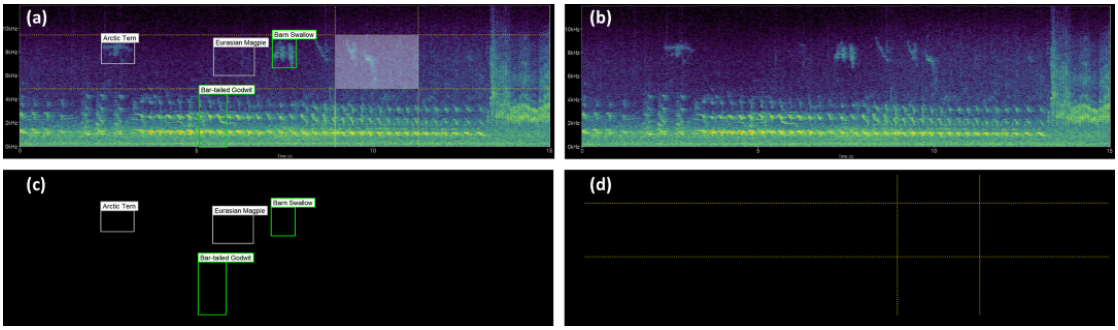


Fig 3.3. Spectrogram plot. The full spectrogram plot (a) is a combination of overlapping panels. From back to front: (b) highlights the main spectrogram, which is the slowest to render; (c) shows the labels panel displaying bounding box annotations; and (d) emphasizes the time/frequency guides which gauge the time and frequency ranges the sound clip occupies, as well as tracing the shape of the bounding box should a label be saved.

If the user wants to get a closer look at a particular sound in the spectrogram, a section can be zoomed in on by selecting the area of interest and double clicking on the selected area. The plot should then re-render to show only the selected ranges in time and frequency. The parameters in FFT settings can be tuned further to investigate more complex sounds at this level of magnification. In particular, increasing the FFT window size can put more emphasis on frequency resolution and less on time resolution.

Decreasing this parameter has the opposite effect. To zoom out, the user double clicks any point on the plot, or clicks the Reset Zoom button.

Filtered audio production

To display the audio to end-users, it is converted to a spectrogram, a visual representation of sound. This is achieved using the Short-Time Fourier Transform (STFT) algorithm [4], which applies the FFT algorithm on successive windows of the audio waveform. The output of this transform is a 2D array of complex numbers, before the modulus is taken and the results (scaled to dB) are printed to the screen.

When selecting the audio area of interest, it is often helpful to remove or significantly reduce obvious noise which can hinder clear identification. By default, the app keeps the selected area of the (complex) spectrogram the same, sets all values outside this area to zero (or near-zero) and reconstructs a filtered audio clip using the Inverse Short-time Fourier Transform (ISTFT). To do this, the complex values of the spectrogram are kept in the back-end, rather than just the magnitudes used for visualisation. These complex values are passed into the `seewave::istft` function, based on a MATLAB implementation [313]. When the user selects some time/frequency range, only those time/frequency segments will be played. Alternatively, if the audio frequency range slider is adjusted, the entire duration of the audio file is included, but only those frequencies within the filtered range. A similar reconstruction is performed through the spectrogram noise reduction techniques carried out by noise reduction of the complex spectrogram data. Fig 3.4 illustrates a comparison of the spectrogram and audio player appearances.

The refined audio can often sound unnatural or distorted when the magnitude of the spectrogram values outside the selection are collapsed to zero, so the app has an alternative option to reduce the magnitude of the audio outside the selected region by a factor of 100. The user deselects *Zero Audio outside Selected* in the **Spectrogram Settings**, and when the plot is clicked outside the selected box, the audio is reset.

Bounding box display

The annotations created by the NEAL app are bounding boxes: rectangular regions defined using *time* and *frequency* coordinates by the lower left and upper right corners of the object. The purpose of providing a bounding box as the main label is to maximize the detail captured in a sound event annotation project. Fig 3.5 describes the varying definitions of sound classification projects, based on the diagram in [254].

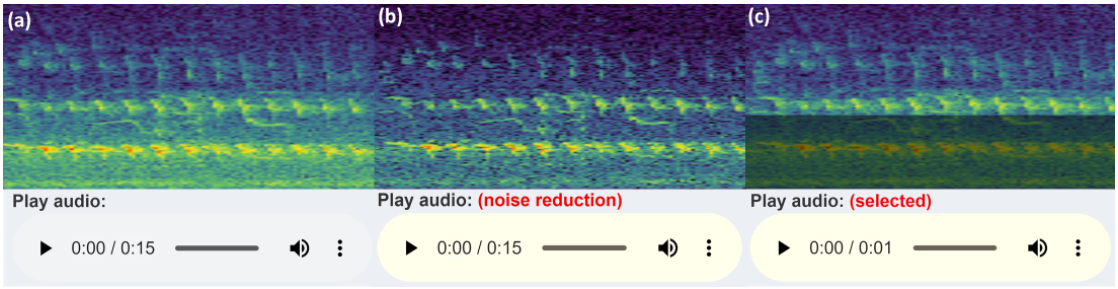


Fig 3.4. Audio Player and spectrogram. The three versions of the audio UI element the user will see (in Chrome), with corresponding spectrogram. The spectrogram has been zoomed in to 0-5kHz (y-axis) and 3-6 seconds (x-axis) to more clearly illustrate the filtering features: (a) shows the default audio player and spectrogram shown to user; (b) conveys row-wise noise reduction applied; (c) displays frequency filtering applied, i.e., the audio within the greyed out box is masked and the remaining audio is reconstructed. (a) shows the raw audio file from the workflow folder, while (b) and (c) show the temporary filtered audio files.

Fig 3.5. Three common types of sound event detection and classification. (a) shows the most common setup where an entire audio clip is labelled either using binary (e.g. bird/not bird) or multi-class classification. It does not give any explicit temporal resolution; (b) displays sound event detection (SED). Labels have fixed time durations, but automatic detection models classify successive time windows (small fractions of the file duration) for each class, merging consecutive windows of the same class into larger sound events. Here, the blackbird classification is longer than the other two detections. This illustration does not include any frequency resolution (the annotations span the entire y-axis) but typically, species occupy specific frequency bins; (c) is object detection setup using bounding boxes. This clearly shows highest resolution detail among the annotation examples shown, with the possibility of overlapping sound events in both time and frequency domains. In this toy example, the two wren annotations have differing frequency ranges which gives richer frequency information. Adapted from [254].

Classifying the sound clip as a whole (a) gives no indication of the duration or regularity (how often it occurs) of a sound event, or whether multiple species are present in a single audio file. Annotation setups where labels transcribe the time dimension

with either no frequency resolution (b) or with a small number of set frequency bins is called sound event detection. This does not have the flexibility to account for overlapping vocalisations in the frequency dimension, or a large number of classes where the frequency bins are difficult to distinguish.

Time- and frequency-specific bounding boxes (c) therefore give millisecond level timestamps (both start and end) for sound events, as well as maximum, minimum and median frequencies of the labelled sound clip. If not used directly for object detection [318], frequency metrics can serve as auxiliary input to train machine learning models for species detection [152].

Class list update

The class list is a group of buttons representing the available classes to label the identified sounds in the audio files. A predefined list of classes is important for consistency within and among labellers, reducing the time taken to repetitively type class names, as well as minimizing the need to correct typos during post hoc analysis.

The list is displayed in CSS `grid` (default) or `flex` containers. The flexbox layout makes the buttons as wide as the text for each class and automatically wraps row-wise. The grid layout has a custom number of columns, which can be adjusted using the *Number of Columns* parameter in **Other Settings**. The grid layout is neater while the flexbox layout is more compact. The differences can be seen in Fig 3.6.

Fig 3.6. Class list layouts. Shows the possible class list layouts for a small example list of classes, as well as the miscellaneous categories and one manually added class. (a) is the default layout of the classes in a grid, with the number of columns decreased to 3 for display purposes. (b) is the same list in flexbox layout. Note the reduced amount of empty space from shorter class names. (c) shows the same flexbox layout with class names changed to their corresponding British Trust for Ornithology (BTO) codes, where applicable. The space is compacted further which becomes more apparent with longer class lists.

The different groups of classes are colour coordinated to distinguish the main classes,

miscellaneous categories and manually added classes. Core classes in the selected site species list have green borders, while classes manually added to the list are in blue. Miscellaneous sounds such as human, insect or weather noise are outlined in orange. If a label is present in the file that is not in the collective class list, its bounding box is coloured grey in the plot.

When a new class is added to the list, the list is searched and if it is not already present, the class is appended. If the class is found, an error message is thrown saying the class is already present in the list.

The British Trust for Ornithology has a list of 2-letter species codes for over 250 bird species. The app has the ability for users to view classes in the displayed list with their corresponding code, if they have a matching species in the BTO list. `bto_codes.csv` acts as a lookup table, which was sourced from `https://www.bto.org/sites/default/files/u16/downloads/forms_instructions/bto_bird_species_codes.pdf`. It has two columns, **bto_code** and **species_name**: the first column has the two letter code associated with the species name in the second column. An example of the conversion is shown on Fig 3.6.

3.2.4 Other workflow features

At the top left of the screen, there is a hidden/collapsed sidebar that contains multiple adjustable drop-down menus of settings. These include **Configuration:** general setup inputs for the annotation project, such as the directory containing the audio files, uploading more files to the user's folder, uploading annotations from previous projects and the column of `species_list.csv` with the relevant classes for labelling, which is expanded upon in Extension to other audio labelling projects. **Sound Settings:** adjustable inputs for processing incoming audio files, including dB gain (or amplification factor) and the length of audio clips to display in the spectrogram (the default is 15 seconds). **Spectrogram Settings:** adjust the visual aspects of the spectrogram plot, e.g. the colour palette, plot height and whether to include time and frequency guidelines for boxes drawn. **FFT Settings:** parameters for the Short-Time Fourier Transform (STFT) calculation via the `spectro` function. **Other Settings:** miscellaneous widgets for adjusting the class list, label summary table and label edit table.

Metadata is a helpful addition to annotation workflows, providing context for users who may not have been involved in the project design or data collection phases. The app attempts to link audio files to a predefined list of recorders that were deployed in order to

give information to end-users on aspects of the study site such as habitat type, location and distance to coastline, then displays them in a collapsible panel to be consulted as desired. For example, the location of the recordings may give important context for detecting species, in particular for migratory birds. Addressing the information needs of labellers by providing all available metadata for context can improve the accuracy of the classifications [173].

With a key objective of the app being labeller efficiency, using keyboard shortcuts instead of repetitive mouse drags and clicks enables increased precision and productivity. This was not included in the main procedure but is a useful addition to efficient labeller workflow. Available hotkeys include saving and deleting labels, navigating to the previous or next file, pausing and playing the current audio file and adding or deleting classes in the class list.

The label edit table is an interactive revision tool that describes all labels for the current file. This can be enabled in the Other Settings tab in the sidebar, where it then appears at the bottom of the app. It will only display when the current file has at least one label. Some of the fields from saved annotations can be edited, such as the time/frequency limits of the bounding boxes, class identified and label confidence.

The summary table is another revision tool providing an abridged description of all files in the workflow folder. It can also be enabled through the Other Settings tab. The user can filter these files by name, number of labels present per file and even split the label counts per file by the classes present. Users can quickly navigate between a large number of files to find other examples of previously labelled species.

When the labelling project is complete and the user wishes to access their labels (especially on a server), they can be exported using the Download button. This is located in the **Label buttons** section towards the bottom of the main body of the app. These labels are exported to the Downloads folder of the user's local machine. This download feature works both on a server or local deployment.

3.3 Case Study

All examples and results in this paper come from data recorded for the Nature+Energy project [160], part of which involves developing an acoustic monitoring system for on-shore wind farm sites in Ireland. Over 1,700 hours of audio was collected between April and July 2022. Wildlife Acoustics Song Meter Mini Bat (`https://www.wildlifeacoustics.com/products/song-meter-mini-bat`) units were mounted at

approximately 1.5 metres above the ground, placed between a wind turbine and some linear feature such as hedgerows or gaps in tree cover. For the first stage of the project, we selected five recording sites (one recorder per location) representing the variety of habitat types present.

Nature+Energy is a collaboration between academic researchers and industry partners that aims to measure and enhance biodiversity at onshore wind farms throughout Ireland. The project focuses on exemplar wind farm sites, located in typical habitats where wind farms are situated in Ireland.

Wind energy is undergoing substantial growth in Ireland, providing over 30 percent of its energy needs in 2020 and expected to reach 80 percent by 2030 [44]. With the increase in wind turbines comes the possibility of biodiversity degradation, with bird and bat populations being particularly vulnerable to collision mortality [212, 36]. The need for monitoring systems is crucial to inform habitat management planning, potentially reducing habitat decline, fatalities and issues specific to each site. Through the Nature + Energy project, a number of acoustic sensors were set up as a non-invasive monitoring method at select study sites across the country. A description of each recorder, and the sites where each was deployed, is given in Table 3.2.

Table 3.2. Summary of Wind Farm Study Sites. The study sites represent a diverse array of land use types, including coastal and inland agricultural lands and commercial forestry. Sites vary significantly in characteristics such as area, number of turbines installed, and hours of acoustic and ultrasonic recordings. The volume of data collected is based on the length of recorder placement and frequency of recorder maintenance.

Site	Land Use Type	Area (Ha)	No. Turbines	Hours recorded	
				Acoustic	Ultrasonic
Carnsore Point	Agricultural (coastal, pasture)	80	14	547	19.6
Cloosh Valley	Commercial forestry on peat substrate	1378	36	438	11.9
Rahora	Agricultural (inland, arable)	24	5	170	14.2
Richfield	Agricultural (coastal, arable)	112	18	850	52.4
Teevurcher	Agricultural (inland, pasture)	60	5	156	1.7
Total		**1654**	**78**	**2161**	**99.8**

Passive acoustic monitoring provides rich insight into species abundance, and temporal and spatial granularity [282]. Due to the nature of where the recorders are located, anthropogenic noise from the wind turbines has an adverse effect on data quality, especially for labelling species with calls occupying lower frequencies. There is a need for frequency granularity to conceal other species or noise sources occurring simultaneously. Metadata, as mentioned in Other workflow features, is a helpful addition to give labellers insight on aspects of the study site such as habitat type, time of day/year, and location.

Project annotation procedure

A version of the app was deployed to an Rstudio Connect server integrating user login via Auth0 (`auth0.com`). Users (bird experts) were assigned a sample of audio files from multiple wind farm sites that were believed to contain sounds of interest. A summary of the actions involved in labelling a selected segment of audio with a bounding box (Steps (3) to (6) in Detailed labeller instructions) is outlined in Fig 3.7 below.

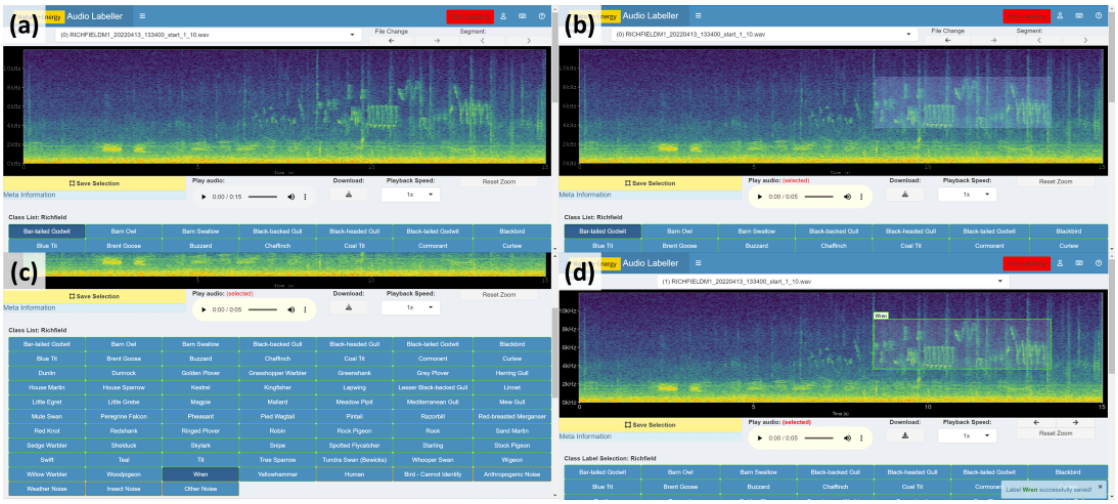


Fig 3.7. Spectrogram labelling example. (a) View spectrogram and play audio. (b) Drag a box around the audio of interest. (c) Select identified class. (d) Click **Save Selection**.

Results of labelling

A small sample of the results obtained from labelling are shown in Fig 3.8. These examples illustrate the diversity of bird vocalisations found on wind farms, even within the same species. The structure, duration and frequency ranges of bird vocalisations are shown to vary, the latter of which could not be extracted via one dimensional labels. Noise from the wind turbines on the study sites is especially noticeable in the examples on the left.

A sample of the count data for non noise-related classes labelled are shown in Table 3.3. There are several zeroes in the table, indicating the diversity of the selected study sites, which cover different geographic regions and habitat types. Further details on the bird species identified by bird experts, using NEAL, during the case study are included in Table S3.1 in Bird conservation status in the appendix. It displays bird con-

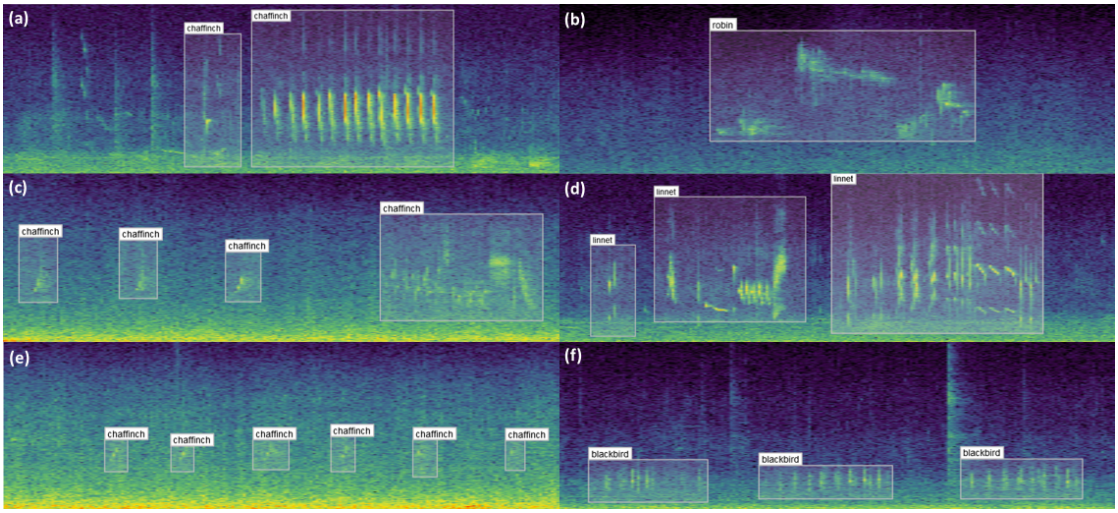


Fig 3.8. Example bounding box annotations. Example annotations of sample audio collected from wind farm sites across Ireland. (a), (c) and (e) in the first column show different vocalisations of the chaffinch. (a) can be readily identified both visually and through listening to the audio, whereas (c) and (e) are increasingly difficult to distinguish from noise. Adjusting the contrast or applying noise reduction in the spectrogram settings may help with identifying examples similar to these. The structure of the vocalisations in (a) and (c) are quite dissimilar despite coming from the same species, which can be optionally captured using the **call_type** input. (b) shows a distinct robin vocalisation which was easily identified given little background noise. (d) includes three separate vocalisations of a linnet, since the sounds are separated by a natural pause. (f) contains three vocalisations of a blackbird. This sits partially in the range of wind turbine noise, and would be difficult to identify if the noise was more prominent or the bird was located further from the recorder.

servation data and information on their distribution across Ireland, and was obtained from [80].

Some common species such as the Blackbird, Meadow Pipit and Robin are present across almost all sites, while the Yellowhammer is only present at one site due to it being a rare species that feeds on cereal crops. The annotations that were discerned to be birds but could not specifically be identified (representing about 8% of non-noise related labels collected, not included in the table) can be returned to the same labeller for deeper analysis or given to another labeller for a second opinion. Both could make use of the output of a trained species recognition machine learning model to inform the decision.

Table 3.3. Example raw species counts from labelled data

Class	Carnsore Point	Cloosh Valley	Rahora	Richfield	Teevurcher	Conservation Status
Blackbird	37	0	107	73	96	Green
Chaffinch	0	141	0	32	0	Green
Dunnock	26	1	0	12	0	Green
Goldfinch	12	0	0	0	0	Green
Hooded Crow	0	0	13	1	16	Green
House Sparrow	0	0	20	0	0	Amber
Linnet	49	0	15	42	2	Amber
Meadow Pipit	17	1	33	15	51	Red
Pied Wagtail	7	0	1	4	2	Green
Robin	0	62	4	49	7	Green
Rook	0	0	10	83	0	Green
Sedge Warbler	11	0	0	1	0	Green
Starling	0	0	37	0	0	Amber
Stonechat	30	0	0	0	0	Green
Wren	26	0	12	143	0	Green
Yellowhammer	0	0	60	0	0	Red
Blue Tit	0	0	0	7	0	Green
Buzzard	0	0	2	0	1	Green
Coal Tit	0	6	0	1	0	Green
Curlew	1	0	0	0	0	Red
Goldcrest	0	3	0	0	0	Green
Grasshopper Warbler	3	0	0	0	0	Amber
Great Tit	0	0	0	5	0	Green
Jackdaw	0	0	0	0	1	Green
Magpie	0	0	1	0	0	Green
Mallard	3	0	0	1	0	Green
Oystercatcher	1	0	0	0	0	Amber
Pheasant	0	0	1	0	0	Green
Skylark	1	0	0	0	0	Amber
Song Thrush	3	0	0	0	0	Green
Swallow	1	0	0	0	9	Amber
Teal	0	1	0	0	0	Amber
Woodpigeon	0	0	4	1	0	Green

The table details the most and least common species identified by NEAL. The top species were those with a total count of more than 10, while less frequent species are shown below. Some bird species, such as the Robin (*Erithacus rubecula*), regularly exploit a wide variety of habitats and food resources, and thus are common and widely distributed. Others however, have more specific habitat requirements. Oystercatchers (*Haematopus ostralegus*), for example, are limited to coastal habitats where they feed on large invertebrates such as mussels. NEAL identified several birds of conservation concern in Ireland (Status **Red**), including the Yellowhammer, Curlew, and Meadow Pipit.

The table details the most and least common species identified by NEAL. The top species were those with a total count of more than 10, while less frequent species are shown below. Some bird species, such as the Robin (*Erithacus rubecula*), regularly exploit a wide variety of habitats and food resources, and thus are common and widely

distributed. Others however, have more specific habitat requirements. Oystercatchers (*Haematopus ostralegus*), for example, are limited to coastal habitats where they feed on large invertebrates such as mussels. NEAL identified several birds of conservation concern in Ireland (Status **Red**), including the Yellowhammer, Curlew, and Meadow Pipit.

Counts of annotations from the first set of data gathered from deployed recorders are shown in Fig 3.9. It displays the activity periods of selected species from the recordings. The recorders were deployed from April-July, which in Ireland is a period when day length is at its greatest; on June 21st, the summer solstice and longest day of the year, Ireland receives approximately 17 hours of daylight. This figure provides an example of the type of data that can be extracted from acoustic recordings, and it could potentially be used to aid management decisions, especially for particularly vulnerable species such as raptors. Some species such as the Yellowhammer tend to be identified throughout the day, whereas others such as the Robin and Chaffinch occur in shorter bursts of activity interspersed with lulls. If multiple recorders were placed on the sites, it could also inform how bird species are using the site throughout the day - for example, to determine whether they are just passing through the study area or are foraging multiple locations throughout the day.

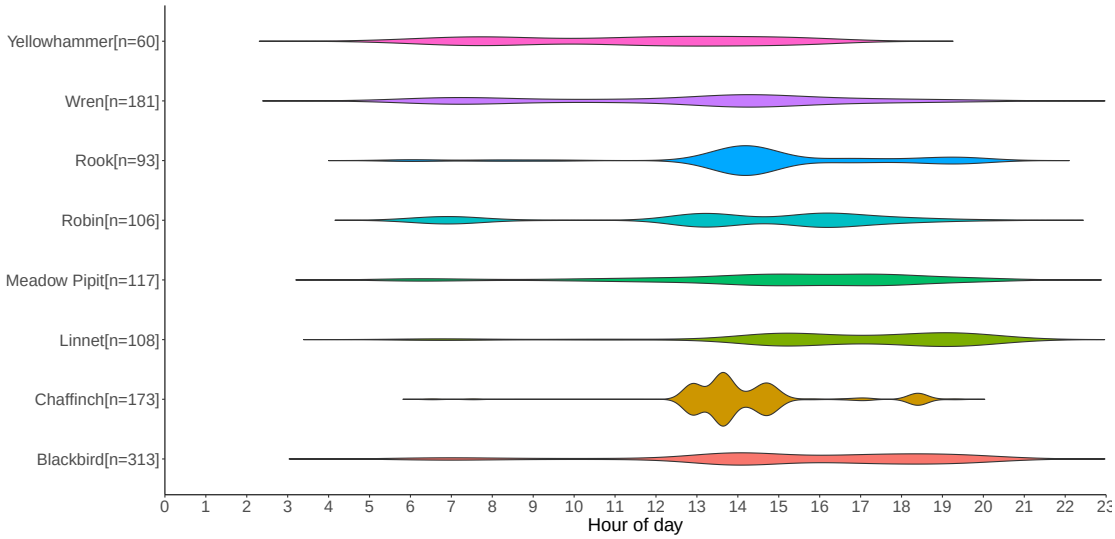


Fig 3.9. Distribution of selected labelled species by timestamp in recording.

3.4 Extension to other audio labelling projects

Our app is built so that it can be configured for other audio labelling tasks beyond the bird species classification demonstrated in our case study. Here we provide some guidance on setting up the input files if they are being expanded to similar areas or changed to cover other domains.

To proceed, a new user merely needs to download (or clone) the GitHub repository. Upon running, the app automatically will create a new user if one is not found in the `www` directory. The user can then add audio files into the XXX directory. The expected audio filename format is `RECORDERNAME_YYYYMMDD_HHMMSS.wav`. If the files are split up into smaller segments, the additional convention `_start_MM_SS` should be appended before the filename extension to indicate the number of minutes/seconds into the recording that they start. This allows the date and time to be parsed from the above filename format and shown in the metadata panel.

Users can copy annotations (or upload if using on a server) from previous labelling projects using the upload button in the Configuration tab in the sidebar panel. These labels could be from manual work or generated by an ML model or some other automatic method. The CSV label data should at least contain the following:

- **date_time:** Date and time label was made in ISO 8601 format (`yyyy-mm-ddTHH:MM:SS+HHMM`)
- **file_name:** Name of the corresponding audio file in the Data Folder
- **start_time:** Start time (in seconds) of the bounding box
- **end_time:** End time (in seconds) of the bounding box
- **start_freq:** Start frequency (in kHz) of the bounding box
- **end_freq:** End frequency (in kHz) of the bounding box
- **class_label:** Class of sound (e.g. bird species) assigned to this bounding box

Labels will be stored in `labels/labels_<username>.csv` if such a folder by this name exists in the above directory. Otherwise the labels are saved in `labels/labels_tmp.csv`. The information contained in each label includes: the time the annotation was made; for which audio file; the start and end time/frequency bounds of each bounding box; the class name; label confidence; labeller and optional call type and additional notes.

The app uses multiple read-only input data tables to accompany the folder of audio files. These include the recorder (study site) location data, species lists and BTO codes. If any of these are edited or replaced for a new annotation project the files should be located in the parent folder of the GitHub repository, and saved in `.csv` format using `UTF-8-BOM` encoding. This ensures that column names and text strings are read correctly.

Species List

NEAL allows for multiple species lists to be included, all of which are stored in `species_list.csv`. The file has one column for each site being studied, with the first entry of each column being the site name. Columns can have different numbers of rows (species), i.e. all those one expects to find at the given site. The species names can be stored in the column in any order as they are sorted alphabetically at run time. Columns can be appended to it by uploading via a widget in the Configuration tab. Each of the columns in the uploaded data not already present in the species data are appended.

The `location_list.csv` file has a list of recorders that were deployed along with accompanying metadata about the sites studied. If the current audio file in the above format matches a recorder in the list, the following columns of the file are passed to the metadata tab:

- **recorder_name:** prefix for audio file names recorded by this device
- **lat:** latitude of the recorder for the study period, in decimal degrees
- **long:** longitude of the recorder for the study period, in decimal degrees
- **location_name:** name of the study site
- **location_county:** county of study site
- **habitat_type:** primary habitat type of the study site
- **dist_to_coastline:** approximate distance to the nearest coastline in kilometres

If any of this information is unavailable, the column name in the metadata panel will still appear but the body of text will be blank. Extra columns can be added to the file, where they will be printed verbatim to the metadata panel.

3.5 Conclusion

NEAL is an open-source application for visually examining and annotating audio data. The tool was designed with the primary goal of improving labeller efficiency and consistency. Vocalisations are tagged with comprehensive annotations providing time and frequency detail, accurate to several decimal places. Call type and an open text field for notes attempt to capture multiple modes of information, with the possibility of performing multiple labelling tasks simultaneously. A labeller providing classifications of different bird species, as well as the call type and identifying sources of noise in each sound clip, has generated three potentially separate sets of labels without expending more effort than generating only one. These can each serve as targets for a machine learning algorithm to predict.

The Shiny package for R allows for an interactive front-end, while its reactive ability reduces unnecessary computational expense. Its no-code interface allows domain experts to interact with the data without any knowledge of R programming. While the app was designed mainly for classifying bird audio, it can be expanded to projects with data focusing on bats, frogs, small mammals and insects - all are popular in bioacoustics research.

There is room for extending the functionality of the app. Expanded contextual information such as weather data, proximity to Special Areas of Conservation and habitat type could prove particularly useful. New interactive tables and visualisations (even the static summary visualisations presented in this paper) could be used to investigate outliers and labeller inconsistency. For example, it may be informative to highlight individual audio clips or entire species where labellers frequently disagree with the majority consensus, or to compare the average label confidence of each species found. The app being open-source means others can contribute or request features, driving innovation for future releases. R is one of the most popular programming languages in bioacoustics and new features to the app can be easily added on through the Shiny app's modular design.

While we did spend a significant amount of time during the development of the app to reduce the rendering time for the spectrogram plots, we could not find a feasible way to get to the speed of the compared-with apps in `R` which would also produce high resolution interactive plots. Future versions of the app hope to find methods to speed up computation to more acceptable levels. Since the app uses `ggplot`, which we have found to be slow in rendering a large number of tiles for the spectrograms, NEAL's

performance should improve in line with its development. If the app continues to lag behind the compared-with software, some `C++` plugins may be able to help with render speeds.

Acknowledgements

We thank our labellers Harry Hussey, David Kelly, Seán Ronayne and Mark Shorten, who annotated the majority of the audio files, as well as providing beta-testing of the app in its early stages. We also thank the on-site personnel and surveyors for carrying out recorder maintenance, i.e. data and battery transfers, as the wind farms are widely spread across Ireland. Finally, we would like to acknowledge the significant work done by Dr. Aoibheann Gaughran on the Nature + Energy project, and for her encouragement and comments on the Shiny app from its inception. This paper is dedicated to her memory.

Funding Information

This publication was produced by the Nature+Energy Project, funded by Science Foundation Ireland (12/RC/2302_P2) and MaREI, the SFI Research Centre for Energy, Climate and Marine Research and Innovation, with additional funding from Microsoft and the SFI CONNECT Centre for Future Networks and Communications (13/RC/2077_P2). In addition, Andrew Parnell's work was supported by: a Science Foundation Ireland Career Development Award (17/CDA/4695); SFI Centre for Research Training in Foundations of Data Science 18/CRT/6049, and SFI Research Centre awards I-Form 16/RC/3872 and Insight 12/RC/2289_P2. For the purpose of Open Access, the author has applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission.

3.A Supplementary Information

Detailed labeller instructions

A more detailed version of the general workflow defined above in General labelling workflow carried out by labellers is below. Items without numbers are optional steps.

1. Go to the deployed app at the given URL. Login via the Auth0 page.

2. Click the user icon in the top right, then start labelling. The first audio file and corresponding spectrogram should load

Nagivate to the next file with little or no annotations. This can be inspected with the dropdown menu, displaying the number of annotations in brackets.

Tune parameters in the sidebar to the desired configuration. In particular, the spectrogram parameters such as colour palette and contrast may be adjusted until the user is comfortable with the visual distinction of the sounds present.

If the user can already visually detect sound events (e.g. bird vocalisations or common forms of noise), these can be annotated as described in steps 3-6.

3. Play the audio until the user comes across a sound of interest. Once identified, place the mouse at the top left corner of the vocalisation you identified and drag to the bottom right of the object, drawing a moderately tight box around it.

4. The audio player now updates to have a **filtered audio file**, reconstructed using only the times and frequencies within the box drawn. This should assist the user in identifying the sound by removing much of the noise present from other frequencies, and cropping the audio to the small clip of interest. To return to the raw audio file, click anywhere on the plot.

5. Once they have identified a sound: if it is in the class list, click the corresponding class's button; otherwise, add it using the text box below and then add it to the list.

If any additional information can be extracted from the audio, such as the call type of the species or miscellaneous notes, or the labeller's confidence of a particular annotation is less than certain, they can be included using the text fields on the bottom right of the app's main page.

6. When the desired class is selected from the class list, redraw a tight box around the vocalisation and click **Save Selection** directly below the plot. This will save the annotation and draw a bounding box with the chosen label onto the plot.

If the user wishes to change any of the bounding boxes, they can be deleted using the **delete** button in the **label buttons** section and and redrawn using the above steps. Alternatively they can be edited using the label edit table which can be

enabled in the **Other Settings** tab. This is described more in the Extension to other audio labelling projects section.

7. **Continue annotating** the file by repeating steps (3) to (6) until no more unlabelled sounds of interest remain.

8. **Proceed to the next file** using the file navigation menu or adjacent navigation buttons and go back to step (3).

9. Once the user finishes their session, click the user icon again, followed by **End Labelling** to finish.

Bird conservation status

Table S3.1. Bird species identified with conservation status and distribution across Ireland

Common Name	Scientific Name	Bird Family	Conservation Status	Ireland Distribution
Blackbird	Turdus merula	Thrushes	Green	Widespread and common resident
Blue Tit	Cyanistes caeruleus	Tits	Green	Widespread and common resident
Buzzard	Buteo buteo	Raptors	Green	Widespread and common resident
Chaffinch	Fringilla coelebs	Finches	Green	Widespread and common resident
Coal Tit	Periparus ater	Tits	Green	Widespread and common resident
Curlew	Numenius arquata	Waders	Red	Declining resident population & winter visitor to wetlands
Dunnock	Prunella modularis	Dunnocks	Green	Widespread and common resident
Goldcrest	Regulus regulus	Kinglets	Green	Common resident (coniferous forests)
Goldfinch	Carduelis carduelis	Finches	Green	Widespread and common resident
Grasshopper Warbler	Locustella naevia	Warblers	Amber	Widespread summer visitor
Great Tit	Parus major	Tits	Green	Widespread and common resident
Hooded Crow	Corvus cornix	Crows	Green	Widespread and common resident
House Sparrow	Passer domesticus	Sparrows	Amber	Widespread and common resident
Jackdaw	Corvus monedula	Crows	Green	Widespread and common resident
Linnet	Carduelis cannabina	Finches	Amber	Widespread and common resident
Magpie	Pica pica	Crows	Green	Widespread and common resident
Mallard	Anas platyrhynchos	Ducks	Green	Common resident (wetlands)
Meadow Pipit	Anthus pratensis	Pipits	Red	Common resident (rough pastures and uplands)
Oystercatcher	Haematopus ostralegus	Waders	Amber	Resident & winter visitor (coast and inland lakes)
Pheasant	Phasianus colchicus	Game Birds	Green	Introduced - Widespread and common resident
Pied Wagtail	Motacilla alba yarrellii	Wagtails	Green	Widespread and common resident
Robin	Erithacus rubecula	Chats	Green	Widespread and common resident
Rook	Corvus frugilegus	Crows	Green	Widespread (absent from expansive uplands)
Sedge Warbler	Acrocephalus schoenobaenus	Warblers	Green	Widespread and common summer visitor
Skylark	Alauda arvensis	Skylarks	Amber	Common resident (uplands and areas of farmland)
Song Thrush	Turdus philomelos	Thrushes	Green	Widespread and common resident
Starling	Sturnus vulgaris	Starling	Amber	Widespread and common resident
Stonechat	Saxicola rubicola	Chats	Green	Widespread resident (scrubland, mainly near the coast)
Swallow	Hirundo rustica	Swallows & Martins	Amber	Common summer visitor
Teal	Anas crecca	Ducks	Amber	Small numbers throughout Ireland
Woodpigion	Columba palumbus	Pigeons & Doves	Green	Widespread and common resident
Wren	Troglodytes troglodytes	Wrens	Green	Widespread and common resident
Yellowhammer	Emberiza citrinella	Buntings	Red	Declining resident mainly in the east and south. Strongly tied to cereal cultivation

Poster

Maynooth University
National University of Ireland Maynooth

Nature+Energy

NEAL: A Shiny App for audio annotation

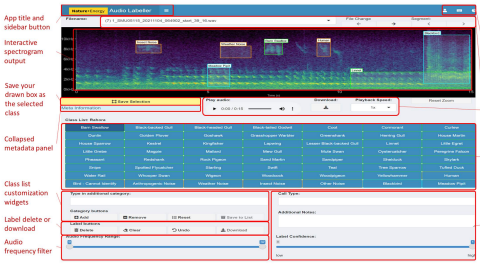
Anthony Gibbons
Supervisors: Andrew Parnell, Ian Donohue
Hamilton Institute, Department of Mathematics and Statistics, Maynooth University

Try it now!

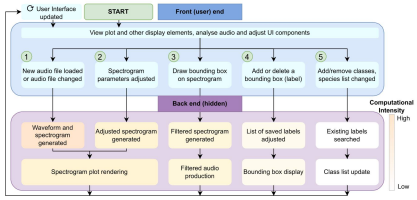
Abstract

Passive acoustic monitoring is used widely in ecology, biodiversity, and conservation studies. Data sets collected via acoustic monitoring are often extremely large and built to be processed automatically using Artificial Intelligence and Machine learning models, which aim to replicate the work of domain experts. These models, being supervised learning algorithms, need to be trained on high quality annotations produced by experts. Since the experts are often resource-limited, a cost-effective process for annotating audio is needed to get maximal use out of the data. We present an open-source interactive audio data annotation tool, *NEAL* (Nature+Energy Audio Labeller). Built using `R` and the associated Shiny framework, the tool provides a reactive environment where users can quickly annotate audio files and adjust settings that automatically change the corresponding elements of the user interface. The app has been designed with the goal of having both expert birders and citizen scientists contribute to acoustic annotation projects. The paper is in preprint on arXiv [1].

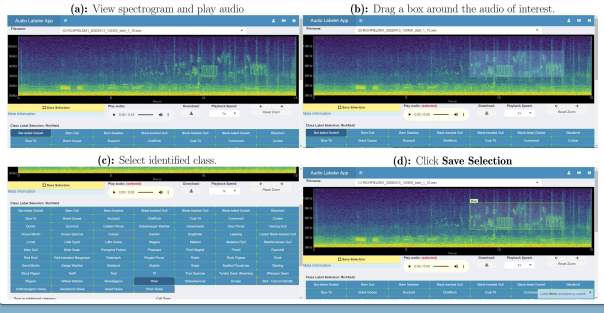
1. App Layout



2. Workflow Diagram



3. Labelling Example

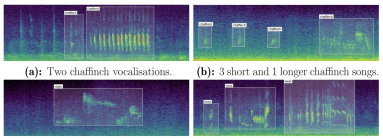


4. Case Study

Wind energy is undergoing substantial growth in Ireland. With the increase in wind turbines comes the possibility of biodiversity degradation. The need for monitoring systems is crucial to inform habitat management planning, potentially reducing habitat decline, fatalities and issues specific to each site. Through the Nature + Energy project, a number of acoustic sensors were set up as a non-invasive monitoring method at select study sites across the country.
A version of the app was deployed to an Rstudio Connect server where users (bird experts) were assigned a sample of audio files from multiple wind farm sites across Ireland.

5. Example annotations

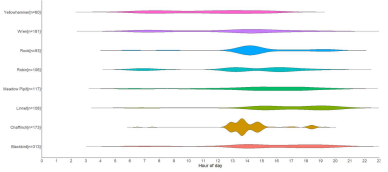
A small sample of the results obtained from labelling. These illustrate the diversity (even within species) of bird sounds found on wind farms.



(a): Two chaffinch vocalisations. **(b):** 3 short and 1 longer chaffinch songs.
(c): Single robin sound. **(d):** Three distinct linnet vocalisations.

6. Label distribution

Counts of annotations from the first set of data gathered from deployed recorders are shown in the figure below. It displays the activity periods of selected species from the recordings. The recorders were deployed from April-July; in Ireland this is when day length is at its greatest.



7. Acknowledgements

This poster was produced by the Nature+Energy Project (`https://www.marei.ie/project/natureenergy/`), funded by Science Foundation Ireland (12/RC/2302_P2) and MaREI, the SFI Research Centre for Energy, Climate and Marine Research and Innovation.

References

[1] Anthony Gibbons et al. *NEAL: An open-source tool for audio annotation*. 2022. arXiv: 2212.01457 [cs.SD]. URL: https://arxiv.org/abs/2212.01457.
[2] Winston Chang et al. *shiny: Web Application Framework for R*. R package version 1.7.1. 2021. URL: https://CRAN.R-project.org/package=shiny.
[3] J. Sueur, T. Aubin, and C. Simonis. "Seewave: a free modular tool for sound analysis and synthesis". In: *Bioacoustics* 18 (2008), pp. 213–226. DOI: https://doi.org/10.1080/09524622.2008.9753600.
[4] Dan Stowell. "Computational bioacoustics with deep learning: a review and roadmap". In: *PeerJ* 10:e13152 (2022). DOI: https://doi.org/10.7717/peerj.13152. URL: https://doi.org/10.7717/peerj.13152.

LaTeX TikZposter

Fig S3.1. Poster for Conference on Applied Statistics in Ireland 2023.

4

Generative AI-based data augmentation for improved bioacoustic classification in noisy environments

Anthony Gibbons [1*], Emma King [2], Ian Donohue [2], Andrew Parnell [3]

1 Hamilton Institute, Department of Mathematics and Statistics, Maynooth University, Kildare, Ireland

2 Zoology, School of Natural Sciences, Trinity College Dublin, Dublin, Ireland

3 School of Mathematics and Statistics, University College Dublin, Dublin, Ireland

* anthony.gibbons.2022@mumail.ie

Abstract

Obtaining data to train robust artificial intelligence (AI)-based models for species classification can be challenging, particularly for rare species. Data augmentation can boost classification accuracy by increasing the diversity of training data and is cheaper to obtain than expert-labelled data. However, many classic image-based augmentation techniques are not suitable for audio spectrograms. We investigate two generative AI models as data augmentation tools to synthesise spectrograms and supplement audio data: Auxiliary Classifier Generative Adversarial Networks (ACGAN) and Denoising Diffusion Probabilistic Models (DDPMs). The latter performed particularly well in terms of both realism of generated spectrograms and accuracy in a resulting classification task. Alongside these new approaches, we present a new audio data set of 640 hours of bird calls from wind farm sites in Ireland, approximately 800 samples of which have been labelled by experts. Wind farm data are particularly challenging for classification models given the background wind and turbine noise. Training an ensemble of classification models on real and synthetic data combined compared well with highly confident BirdNET predictions. Each classifier we used was improved by including synthetic data, and classification metrics generally improved in line with the amount of synthetic data added. Our approach can be used to augment acoustic signals for more species and other land-use types, and has the potential to bring about advances in our capacity to develop reliable AI-based detection of rare species. Our code is available at `https://github.com/gibbona1/SpectrogramGenAI`.

Keywords: Bioacoustics, Data Augmentation, Deep Learning, Generative Models, Generative Adversarial Networks, Passive Acoustic Monitoring, Species Recognition, Stable Diffusion

4.1 Introduction

Species monitoring is a major component of assessing biodiversity and ecosystem health, serving as a barometer of environmental change and conservation needs. The dynamics and diversity of bird communities are key indicators of ecosystem health and environmental change [142]. Birds provide valuable ecosystem services, which include limiting pest populations [71], seed dispersion [162] and pollination [271]. The fields of bioacoustics and ecoacoustics have long studied the behaviour and diversity of bird species activity

via sound. Acoustic monitoring has become increasingly popular due to increased efficiency, and reduced cost, compared to traditional methods [47, 221]. However, many concerns remain surrounding the accuracy and ability of software to identify bird species from acoustic recordings [77, 225].

Computer vision developments have carried over to the field of bioacoustics to detect species via sound. Audio is represented as spectrograms in order to use image classification methods [254]. Labelled data is required to train such classifiers, necessitating audio clips annotated to the degree of detail desired with, for example, content such as species [256], call type [20], individual animals [150], or other audio events present. However, manual labelling of these datasets is both costly and time-consuming to obtain, requiring expert knowledge of bird vocalisations and skills in annotation software. Machine learning, in particular supervised learning, can automate large parts of such classification workflows, and requires high quality training data. Bioacoustics is still a nascent field and lacks standardised labelled datasets [254] to facilitate training these models. To expand on the limited data available, data augmentation is often applied. However, not all data augmentation methods designed for images apply readily to spectrograms due to their inherent time-frequency structure [316].

We investigate both Auxiliary Classifier Generative Adversarial Networks (ACGAN) [184] and Stable Diffusion models [218] as data augmentation methods for enhancing bird call spectrogram classification models, with the aim of reducing the amount of domain-specific labelled data required to train species classifiers. Our key contributions include:

- Application of Stable Diffusion to bird species monitoring. The model generates high-quality and diverse spectrograms of bird sound, as supported by both image and audio quality evaluation metrics. As far as we are aware, this is the first application of stable diffusion to bioacoustic classification of bird species.

- Improved classification. Each classifier we used found some improvement by including synthetic data. Classification metrics generally improved in line with the amount of synthetic data added.

- Ecoacoustic pipelines. We suggest generative models to enhance bioacoustic classification pipelines. Our Stable Diffusion model produces high-quality diverse spectrograms, improving bird song classification accuracy and reducing the need for expensive expert-labelled data.

Our paper is structured as follows. First, we give background on acoustic species classification and data augmentation and place our techniques in the context of existing data augmentation methods. After describing the details of our acoustic dataset, we then detail the audio pre-processing and the generative models we employed. We use the terms stable diffusion, diffusion models, Denoising Diffusion Probabilistic Models (DDPMs), and latent diffusion interchangeably once the stable diffusion model is introduced. We then provide a detailed description of our experimental design and results, and follow this up with in-depth discussion of our results, limitations and paths for future work.

4.2 Existing Methods

Passive Acoustic Monitoring (PAM) is a non-invasive technique where sensors are deployed in the field to record audio, usually on a pre-defined schedule, for weeks, months or years at a time without intervention. The ever-decreasing cost of recording equipment and data storage has allowed much larger PAM datasets to be collected [134, 165, 220]. The advancement of deep learning methods has continued to grow, allowing automatic species detection methods to achieve higher classification performance for the growing volume of data emerging via PAM. The scale and sophistication of these classifiers has advanced rapidly from presence/absence [158] to classifying single digit [215] to tens and hundreds [3, 315] of species. BirdNET [123] and Perch [75], both considered among the current state of the art in terms of bird classification capabilities, were trained on over 6,000 and 10,000 bird species globally, respectively.

The typical bird species classifier's input format will be as spectrograms, a visual representation of sound (see Figure 4.4). The spectrograms are coloured by sound energy levels across time and frequency. These may be standard (linear-frequency), mel (to better align with human sound perception), or log-frequency spectrograms. Other audio representations used as input for classification include Scalogram [239], MFCC [278], Gammatone Filterbank [190], Constant-Q Transform [281] and the raw waveform itself [146].

While generalised solutions for bird species classification such as BirdNET and Perch are widely applicable, fine-tuned approaches are often desired for specific tasks [144]. The benefits include optimising hyperparameters, improving performance and robustness or reducing the number of parameters, which, in turn, reduces the size and computational cost of deploying such models [117, 147]. Training custom classifiers is commonly done through techniques such as transfer learning [312] and knowledge distillation [88, 101].

These training processes necessarily require some amount of labelled training data from the specified task of interest. This can, in the case of acoustic data containing bird species, be time- and labour-intensive to obtain. When the amount of reliably labelled data is limited, additional strategies become appropriate to increase the utility of the available recordings.

Semi-supervised learning is a strategy to remedy class imbalance or issues with low volumes of high quality data, where labelled datasets are supplemented with unlabelled data [89]. It is common, for example, to train a classifier on a small amount of labelled (human-verified) data, run through a large amount of unlabelled data, and take the highly confident predictions to be *pseudo-labels*. Unsupervised learning models are also commonly used to generate these pseudo labels. The model is then retrained with both the labelled and pseudo-labelled data and, now that it has more examples to learn from, this often results in improved performance. Semi-supervised learning has been applied to bird call classification with few shot learning [174], beluga whale detection [314], and individual bird recognition via object detection [62]. In our setting, the entirety of our training and validation datasets are *unlabelled*, i.e. pseudo-labelled by highly confident (over 50% confidence level) BirdNET predictions but not human verified. The test set contains highly confident human annotated data from the same sites, and our aim is for a robust model that performs well on this unseen test dataset.

Data augmentation is widely used to improve the performance of image classification models by artificially increasing the diversity of training data. Common augmentation techniques such as flipping, rotation, cropping, translation, and scaling/zooming help improve generalisation by providing varied perspectives of the same image content [243]. However, these methods are less applicable to spectrograms due to the inherent time (x) and frequency (y) structure of the latter [70]. For example, excessive translation in the form of time or pitch shifting can distort the temporal correlation or pitch accuracy, leading to misleading representations of the audio content [316]. While techniques such as mixup [310] and adding background noise [176] are somewhat common for augmenting spectrograms, they also present challenges. Mixup, which interpolates between two samples, has shown promise [175], but can introduce temporal correlation artifacts or fail to maintain the original signal's integrity. Additionally, adding background noise can degrade the quality of the signal, especially if a large volume of clean samples with birds present is not available, potentially hindering the model's ability to distinguish between the target sound and the noise [131].

Generative models have emerged as a popular technology capable of creating diverse

content, be it text, images, videos, or music/audio. There has been a recent surge in interest in generative AI with the release of Dall-E [208], MidJourney [167] and Stable Diffusion [218], which are being used in personal and commercial projects of all scales. Synthetic image generation started with autoencoders [279, 133] and autoregressive models [275], before Generative Adversarial Networks (GANs) became the state of the art [83]. Many more specific architectures based around GANs were designed and trained for their own specific use cases in image generation, such as GauGAN for generating images based on an initial semantic layout [191]. WaveGAN and SpecGAN generate audio in the form of raw waveforms and spectrograms respectively [49]. A similar approach using mel spectrograms is melGAN [138]. Diffusion models are a more recent advancement that led to Stable Diffusion, involving both an autoencoder and a diffusion model on the latent representation of images. Latent diffusion models have begun to be used for spectrogram generation [106].

Our implementation of generative AI focuses on supplementing training data to improve classification performance that has been demonstrated in bird species classification [93]. In contrast, many generative AI uses are purely focused on image/data synthesis for direct use. ACGAN has specifically been applied for acoustic classification of bird species [34, 67] but we include here since it is a more widely used architecture and is faster at generating samples than diffusion models. Diffusion models have been used to improve classification performance of marine mammals [192] but, as far as we are aware, this is the first implementation of them as augmentation methods for bird species classification, and the first use of either method for species classification at wind farms.

4.3 Materials

In this section, we discuss the audio data collection setup and the steps required to generate pseudo-labelled training and validation sets using a state-of-the-art classification model. We use a set of human-labelled data using NEAL [79], an audio annotation Shiny app, as an unseen test set to assess our classifiers.

For our study, audio data were collected from five wind farm sites across Ireland using the Song Meter Mini Bat recorder from Wildlife Acoustics [290]. The data were collected as part of the Nature+Energy Project, a research collaboration between academic scientists, industry partners, and governmental agencies that focuses on facilitating the enhancement of biodiversity on onshore windfarms in Ireland [84]. Sites were selected as representative of onshore windfarms across Ireland, and the habitats and land-uses in

which they are typically co-located [132].

The acoustic settings were configured as follows: a sample rate of 24 kHz was used, with a recording schedule of 5 minutes followed by a 15-minute non-recording interval to extend deployment time. The right channel gain (the left channel is the ultrasonic microphone) was set to 24 dB. The files were saved in `.wav` format. For this study, recordings from one recording cycle of each of the five wind farm sites were used, totalling approximately 640 hours of acoustic data. Deployments varied in length, as did the frequency of collecting batches of data stored on the SD cards. Some summary statistics about our data set are shown in Table 4.1.

Table 4.1. A description of our wind farm audio dataset.

Site	Location	Date range	# files	# hours
Carnsore	Wexford	06 Jul 2022 - 23 Aug 2022	725	60
Cloosh Valley	Galway	22 Dec 2022 - 19 Jan 2023	920	77
Rahora	Kilkenny	07 Jan 2023 - 16 Apr 2023	2612	218
Richfield	Wexford	05 Jul 2023 - 27 Jul 2023	1504	125
Teevurcher	Meath	30 May 2022 - 26 Jun 2022	1921	160
Total			**7682**	**640**

We ran the classification model BirdNET [123] on each file in the dataset, which partitions the data and classifies each 3 second clip. The model produced 67,163 detections in total, or approximately 56 hours, representing 8.7% of the dataset. The data were refined further by selecting only those bird classes with at least 100 examples. We chose this threshold to ensure reliable estimation of per-class accuracy metrics and to allow meaningful training and validation of both the generative and classification models. For each of these species, we selected up to 500 of the most confidently identified examples (confidence over 0.5) by BirdNET. Our final filtered dataset has 8,248 audio clips involving 27 distinct bird species. A further 17,810 clips with confidence between 0.25 and 0.5 were included in the training and validation sets in later runs for a more generalised model, for a total of 26,058 audio clips.

The BirdNET pseudo-labelled dataset was used as a training and validation set,

with a 90-10 split stratified by class to deal with the imbalanced distribution of the data. Table 4.2 shows a list of the classes together with their split in terms of training, validation and test sets by class.

The test set consists solely of human-labelled audio samples, where experts annotated the data with NEAL [79] software with each annotation containing a confidence score. The annotated data are from the same sites as listed in Table 4.1 but from files that are not included in the training or validation datasets. Keeping the test set distinct ensures that it remains independent, preventing overfitting and data leakage. After some remapping of class names to account for slight discrepancies with custom labels, and including only those with one of the 27 classes mentioned above, we arrived at a test set with 1,334 examples. These labels were in the form of bounding boxes and could be up to 15 seconds long. We thus only took the first 3 seconds of each label to match with BirdNET samples. Cropping the annotations thus has a chance of excluding parts of the vocalisations that occur later in the clips but the annotations always begin when the species is already clearly vocalising. After removing labels with labeller confidence less than 0.9, the final test set had 825 examples.

The test set distribution is different to both training and validation sets, with some of the 27 classes not being in the test set. We chose to keep those extra classes not in the test set in order to 1) give the generative models more data to learn from and 2) train more generalised, robust classifiers, i.e., models capable of handling changes in the data distribution and maintaining performance across varied conditions as would be required in noisy environments such as wind farms. Furthermore, due to the limited size of the test set, we decided not to include any real examples in the training or validation data and preserve the full amount as a human benchmark. During evaluation of the test set we ignored these missing classes when calculating the accuracy of predictions for the classifiers.

4.4 Methods

Our study employs a multi-step approach to generate and evaluate the quality of spectrograms of bird sounds. We begin this section by describing the preprocessing steps involved converting the raw audio into mel spectrograms. We then give background on generative models used, how controllable generation allows for generating (labelled) samples of a specific class, and the differences between the reference models and ours. We assess the quality of synthetic images to decide on the better generative model to

Table 4.2. Distribution of species in the dataset. The data were split via class-stratified random sample: 90% for training, 10% for validation, with human labels as test. The horizontal bars visualise the distributions. Note that some classes are missing in the test set because of the limited amount of human-verified samples. For example, Carrion Crow is quite rare in Ireland, and does not appear in the test set. Examination suggests that these pseudo-labels are likely to be Hooded Crow.

	Common name	# Train	# Val	# Test	% train	% val	% test
1	Barn Swallow	299	33	0	4	4	0
2	Carrion Crow	250	28	0	3.4	3.4	0
3	Common Chaffinch	195	22	112	2.6	2.7	13.6
4	Common Wood-Pigeon	250	28	3	3.4	3.4	0.4
5	Dunnock	450	50	11	6.1	6.1	1.3
6	Eurasian Blackbird	414	46	251	5.6	5.6	30.4
7	Eurasian Blue Tit	383	43	3	5.2	5.2	0.4
8	Eurasian Linnet	450	50	41	6.1	6.1	5
9	Eurasian Magpie	121	13	0	1.6	1.6	0
10	Eurasian Skylark	429	48	0	5.8	5.8	0
11	Eurasian Wren	450	50	96	6.1	6.1	11.6
12	European Goldfinch	379	42	6	5.1	5.1	0.7
13	European Greenfinch	99	11	0	1.3	1.3	0
14	European Robin	450	50	63	6.1	6.1	7.6
15	European Starling	95	10	31	1.3	1.2	3.8
16	European Stonechat	450	50	9	6.1	6.1	1.1
17	Goldcrest	121	13	3	1.6	1.6	0.4
18	Great Tit	123	14	4	1.7	1.7	0.5
19	Hooded Crow	190	21	18	2.6	2.5	2.2
20	Meadow Pipit	450	50	65	6.1	6.1	7.9
21	Redwing	90	10	0	1.2	1.2	0
22	Rook	450	50	52	6.1	6.1	6.3
23	Sedge Warbler	108	12	1	1.5	1.5	0.1
24	Spotted Flycatcher	125	14	0	1.7	1.7	0
25	White Wagtail	232	26	5	3.1	3.2	0.6
26	Willow Warbler	119	13	0	1.6	1.6	0
27	Yellowhammer	251	28	51	3.4	3.4	6.2

use as a data augmentation method. Finally, we give details on the equipment used in the generative and classification model training processes.

4.4.1 Preprocessing

In order to exploit a variety of image classification and generation techniques, the audio samples were converted into a visual format as per standard practise in the field [254]. The audio recordings were transformed into a visual representation known as the *mel spectrogram*. The methods we use require square images to work in the existing framework so we aim to transform the `.wav` files into spectrograms that are approximately square. Mel spectrograms offer several advantages over traditional linear spectrograms; being an effective representation of time-frequency dynamics present in animal vocalisations, offering reduced dimensionality compared to raw waveforms, and aligning more closely with human auditory perception [252].

The spectrogram is computed from the raw waveform using `librosa.feature.melspectrogram` [163] and parameters: 256 mel filters (`n_mels`), a Fast Fourier Transform (FFT) size of 512 (`nfft`), and a `hop_length` equal to 75% of the `nfft` value. The transformation is carried out on 3-second inputs with a sampling rate of 48kHz, consistent with BirdNET. The extra width in the mel spectrogram is then cropped to 256 wide and the resulting square matrix is converted from its amplitude squared to dB, which involves a `log` transform. The resulting spectrogram is then saved using `matplotlib.pyplot.imsave` [109] to give a 256×256 single-channel (greyscale) image with pixel values from 0-255. Alternatively, the spectrograms could have been stored directly as floating-point tensors (e.g., `.pt` or `.npy` files), which would preserve full numerical precision. However, storing them as `.png` images offered several practical advantages: it facilitated rapid visual inspection and quality assessment of thousands of spectrograms during preprocessing, enabled seamless use of standard `torchvision` image-loading and augmentation pipelines, and aligns with widespread practice in the audio representation-learning literature. When the images are read in for training models, a torch transform normalises each spectrogram's pixel values to the range -1 to 1.

Although numerous data-augmentation techniques exist for audio classification tasks (applicable either directly to waveforms or to spectrogram images), we decided not to apply them as additional preprocessing layers for this paper. Preliminary experiments with waveform-level techniques—including UNet-based denoising, mixup with other samples from the same dataset, and random time shifting—yielded no consistent improvement in validation performance. We therefore chose to use entirely generative methods as data augmentation techniques.

4.4.2 Generating Artificial Spectrograms

Generative Adversarial Networks (GAN)

Generative Adversarial Networks (GANs) are a class of machine learning model consisting of two neural networks, that is, a generator and a discriminator, that compete in a two-player minimax game [83]. The generator takes in a latent (or noise) vector z and outputs synthetic data resembling real samples, while the discriminator attempts to differentiate between the real and synthetic data. Both models start out with randomly initialised weights and improve in tandem. As the generator improves, it becomes better at fooling the discriminator, leading to increasingly realistic data outputs, i.e. minimising the loss function conveying the difference in distribution between the real and synthetic data. Conversely, as the discriminator improves, it becomes better at distinguishing real from synthetic data, i.e. maximising this same loss function. At the end of the training process, the generator is used for generating synthetic data, while the discriminator can be discarded or used to determine the quality of the generated images. GANs have gained widespread attention for their ability to generate high-quality images [280, 154], audio [10, 308], and video [122, 232] content as well as biological sequencing [177], camouflage design [265], and classifying bats using radar [285].

Auxiliary Classifier GANs (ACGAN) build on the original GAN architecture by incorporating class information to the framework [184]. The generator's latent vector z is now a concatenation of a one-hot encoded vector of class information y and a noise vector. The discriminator is given the additional task of predicting class labels for both real and generated data, as with a standard image classifier. In an ACGAN, the discriminator outputs both the probability that the input is real or fake, as in the standard GAN, as well as the probability of the image coming from each class. The addition of an auxiliary task has two key benefits: it allows ACGANs to create more realistic and diverse images as the classifier component of the discriminator's loss is backpropagated through both models [231], and the generator is now capable of synthesising fake examples with explicit class information guiding the model. The latter is known as controllable generation.

By incorporating species/class information into the training process, ACGANs allow for more fine-tuned generation of synthetic examples. These samples can later be filtered (if necessary) to keep just the highest quality samples, as determined by the discriminator, which can then be iteratively added to the training data. These larger datasets, depending on their design, can have increased diversity or help address class imbalance, enhancing the robustness of the training process. The ACGAN extension has been ap-

plied to acoustic classification [34, 67] to generate realistic spectrograms to supplement and enhance the existing datasets, which then improves classification accuracy of the models trained on these datasets.

We trained an ACGAN to generate mel spectrograms for supplementing our dataset. We wrote the training script and model definitions in PyTorch. The Generator and Discriminator block architectures were based on [67]. We then modify the setup as follows:

- single channel 256×256 images were used instead of three-channel images, as spectrograms, being time-frequency representations of sound, are inherently greyscale and do not require the RGB colour structure.
- an open source PyTorch implementation that can be adapted for future use. The training loop was based on code from [299].
- using negative log likelihood and log-softmax outputs rather than categorical cross-entropy and raw logits as the discriminator auxiliary loss for more training stability. We found that training on data with significant background noise required this step.

The architecture of the ACGAN we use is shown in Figure 4.1.

Denoising Diffusion Probabalistic Models (DDPM)

Diffusion models are another class of generative AI model that generate data by gradually transforming noise into structured samples. The models utilise a two-phase process, a forward diffusion process (adding noise to the data) and a reverse diffusion process, which undoes the forward process [250]. In the forward process, Gaussian noise is progressively added, t times, to the data x_0 to produce a somewhat degraded image x_t, up to a maximum degradation x_T for some T which would be pure Gaussian noise. The reverse diffusion process uses a UNet model, a convolutional neural network architecture, to predict the noise that was added and produce a reconstruction $\tilde{x}_0$ of the original image from x_t, i.e. $\tilde{x}_0 = x_t - \text{UNet}(x_t)$. During the image generation (inference) phase, some pure Gaussian noise, x_T, is generated and the predicted noise is successively subtracted one step at a time, to generate $\tilde{x}_0$, the final synthetic image. Figure 4.2 shows a schematic of the DDPM process.

DDPMs are a specific type of diffusion model that formalises this noise-reduction process using a probabilistic framework [102]. The training loop involves maximising

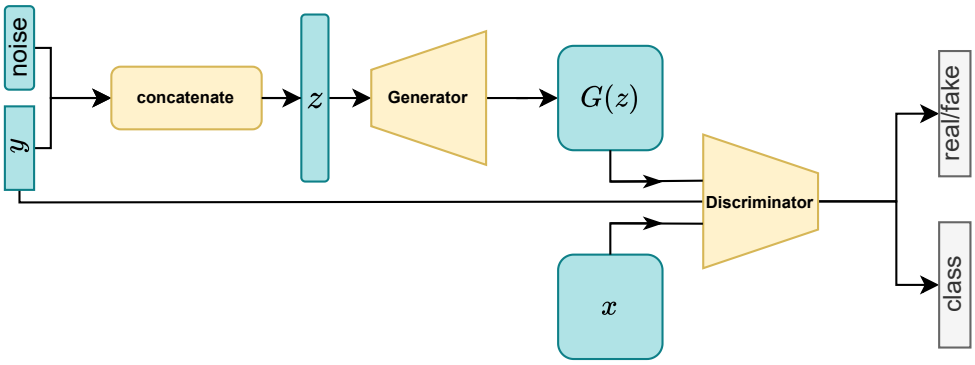


Fig 4.1. Auxiliary Classifier GAN (ACGAN) architecture. For training, real images x and their (one-hot encoded) labels y are given, and sample noise vectors and y are concatenated into latent vectors z. The Generator G generates images $G(z)$ which come from the same distribution as x. Both x and $G(z)$ are passed to a Discriminator D which predicts 1) whether images are real or fake, and 2) their classes. A *two-player minimax game* arises, where G learns to create images to fool D and D learns to better tell x and $G(z)$ apart even as G improves. Images are then generated using the generator as above for subsequent sampling.

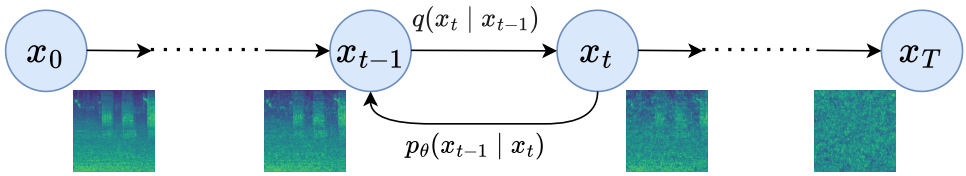


Fig 4.2. Denoising UNet from diffusion model. We have a defined forward process $q(x_t \mid x_{t-1})$ to add noise to an image x_{t-1} to create an image x_t. The reverse process $p_\theta(x_{t-1} \mid x_t)$ is performed by the UNet with parameters θ.

the likelihood of reconstructing the data from its noisy counterpart. The timestep t is also passed to the model allowing the model to learn the conditional probabilities at each step of the diffusion process (from 0 to T), which enables it to denoise the data step by step in reverse. The UNet takes in the noisy image and outputs the predicted noise component, as in the above equation for $\tilde{x}_0$. The loss then compares actual and predicted noise. The model's ability to successively remove noise from input images results in high-fidelity images comparable or superior in quality to other generative models such as GANs. Diffusion models for data generation have seen wide application in medical imaging [29, 108], music synthesis [249], molecule design [283], and filling in

missing or degraded parts of an image [155].

Stable Diffusion models, originally known as Latent Diffusion Models, are similar in design to DDPMs, with the extra step of first encoding the images into a latent space, for example using an autoencoder [218]. The images (in what is called pixel space) are transformed into the latent space, where the forward and reverse (probabalistic) diffusion processes are carried out, before being decoded back to the original image size in pixel space. Since the UNet is the more computationally intensive model, and it is run many (up to T) times during generation, it is much more efficient to carry out this step in the latent space than pixel space, allowing for faster generation and high-resolution outputs.

We employ a stable diffusion model and embed species information for training and generation, as we did with ACGAN. We based our stable diffusion model on the conditional DDPM created by [27] and modify it as follows:

- single channel input images were used instead of three-channel images, as spectrograms are inherently greyscale and do not require the RGB colour structure.
- work with a lower dimensional latent space with tensors of shape $64 \times 64 \times 4$ to train the UNet rather than the 256×256 images themselves, making it a stable diffusion model,
- using a Vector Quantised-Variational AutoEncoder (VQVAE) to encode to and decode from latent space [186], which has the benefits of producing a discrete latent representation, more learnt behaviour in terms of the prior and avoiding posterior collapse (where the decoder ignores the latent structure and outputs generic images), thereby producing better quality samples.

Figure 4.3 shows our model architecture diagram (based on [218]).

4.4.3 Evaluating the Quality of the Generated Spectrograms

We evaluated the quality of the synthetic images generated using two metrics: the Inception Score (IS) [235] and the Fréchet Inception Distance (FID) [100]. IS measures the diversity and clarity of the generated images, ranging from zero (worst quality) to N (best quality), where N is the number of classes. FID assesses the similarity between the generated and real images and ranges from 0 to infinity, with a lower FID indicating closer resemblance to the actual dataset. These metrics were used to compare generated samples from both the GAN and Diffusion models, to evaluate how effectively each set of synthetic examples replicate the characteristics of actual spectrograms.

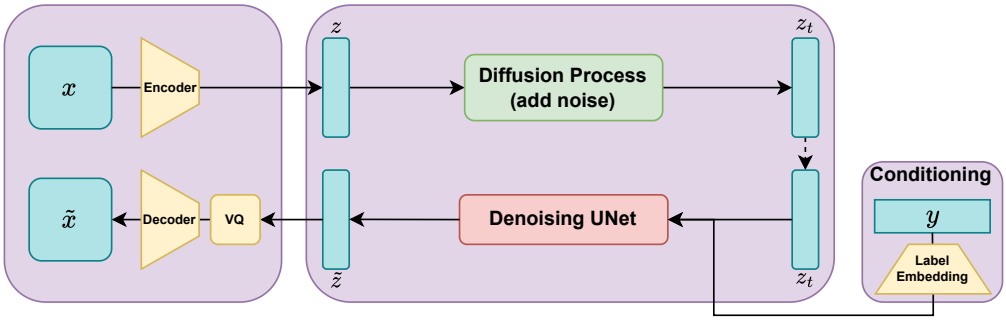


Fig 4.3. Conditional Latent Diffusion architecture. For training, an image x is encoded to a latent z using the VQAE encoder. A random timepoint t from $1, \ldots, T$ is taken. Noise is added to z for t steps giving z_t. The UNet, with inputs z_t, t and y, outputs predicted noise and the loss compares the predicted noise with the actual noise added to z. For sampling, a latent z and class y are generated/given, either pure noise or a partially noised image. For T steps, the UNet runs with z, $t = 1$ and y as inputs and the output subtracted from z itself, updating z. Once denoised T times, this $\tilde{z}$ is vector quantised, decoded to $\tilde{x}$ and saved.

While both image quality metrics have previously been used to measure the quality of spectrograms [67], there have been issues raised with measures like Inception Score [15]. These stem mainly from the fact that its predictions come from the Inception V3 model, which was trained in a different modality and set of classes (in our case ImageNet rather than spectrograms of bird calls). However, it is primarily used as a heuristic to sanity check the quality of training samples after what can be a lengthy training and generation process, rather than as an all-encompassing benchmark. The diffusion samples were more convincing than the GAN samples and so human judgment would have prevailed in either case.

To obtain a more appropriate assessment of audio quality, we converted the generated spectrograms back to audio waveforms using the Griffin-Lim algorithm [91]. We then evaluated the resulting signals with the Fréchet Audio Distance (FAD) [92, 130]. FAD functions like FID, but with embeddings coming from an audio model. We chose embeddings from the CLAP audio encoder [55], which is trained contrastively on diverse audio-text pairs and thus well-suited for general-purpose audio evaluation. Lower values of FAD suggest the generated audio samples more closely align with the real audio dataset in the embedding space.

4.4.4 Experiment Environment

The hardware environment for running our experiments includes a Unix Server, accessed via a web browser, with 566 GB of memory, an Intel(R) Xeon(R) Silver 4214 CPU @ 2.20GHz with 12 cores and 24 threads, and an NVIDIA Tesla T4 GPU with 15 GiB memory. The operating system is Ubuntu 22.04.4 LTS. The programming environment consists of Visual Studio Code version 1.93.1 [39] for prototyping, training and post-processing. Within this, we used CUDA Version: 12.2 [182], Python 3.10.12 [66], and PyTorch 2.0.1+cu117 [267].

4.5 Experimental Design and Results

Both generative models were trained for 200 epochs with square, greyscale images of size $256 \times 256 \times 1$ (see training results in Figure S4.1). After training, both models were set to generate a random collection of images with equal weighting across classes, a total of $250 \times 27 = 6,750$ synthetic images from each model. Figure 4.4 shows some example images synthesised by each generative model. While the ACGAN-generated samples show some clear learning of distinct features for each of the species, they often overly focus on the background noise, with some artefacting or undesirable feature exaggeration observed. Conversely, the DDPM-generated spectrograms are not only convincing but diverse, and appear higher quality than the GAN samples. The structure of the background noise also looks reasonable, with little artefacting. Whilst visually appearing better than the ACGAN samples, next we provide more numerical comparisons of the performance of the generated spectrograms.

Image quality scores support this observation, with DDPM samples showing a 7% increase in IS and 51% decrease in FID score relative to ACGAN samples (Table 4.3). Furthermore, DDPM samples also perform better on the audio quality metric FAD, showing an 8.4% lower score than ACGAN. The scores reinforce our informal visual inspection. We therefore decided to thoroughly evaluate the utility of supplementing our training data with DDPM samples for training acoustic classifiers.

Next, we use our generated samples to determine whether they can assist in improving the classification performance of the model. We set up our experimental datasets as follows: The training data was supplemented with `[0,50,100,150,200,250]` DDPM-generated samples per class, providing up to 6,750 more images (an ∼81% increase in data set size) to learn structures in the spectrograms for distinguishing bird species.

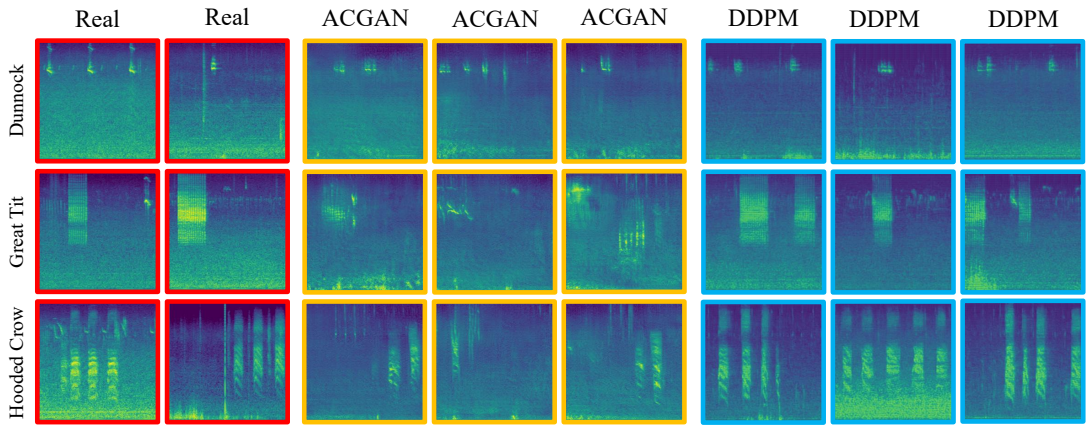


Fig 4.4. Real and Synthetic Spectrograms. The first two (leftmost) red-outlined images in each row are real spectrograms. The next three orange-outlined images are generated by the ACGAN, while the last three blue-outlined images are synthetic samples generated using the DDPM. The x-axis of each spectrogram represents time, ranging from 0-3 seconds. The y-axis represents frequency, ranging from 0 to 12kHz on the mel scale. Brighter colors (using the viridis colour palette) indicate higher energy or loudness. The ACGAN-generated Dunnock and Hooded Crow samples do show some species-specific features, but the Great Tit samples are poor imitations. The DDPM-generated samples show close similarity to the real examples in red but also have variety. The Great Tit examples generated by DDPM are clearly more convincing than the ACGAN samples.

Table 4.3. Synthetic data quality metrics. The baseline IS, FID and FAD scores were calculated on the real data and thus serve as a target value for those created by ACGAN and DDPM. Higher IS scores and lower FID and FAD scores indicate better performance. Note that the FID and FAD scores comparing the dataset with itself are both zero.

Model	IS	FID	FAD
Baseline/target	2.00	0.00	0.00
ACGAN	1.62	64.97	1.07
DDPM	**1.73**	**32.05**	**0.98**

The validation and test datasets remained unchanged to prevent data leakage from using synthetic examples derived from the training data.

For each dataset combining real and (possibly no) synthetic examples, a selection of off-the-shelf classification models were chosen: MobileNetV2 [237], ResNet18 [98], and VGG16 [248]. These models were trained via transfer learning, where early layers were

kept frozen but with the last few layers being trainable. The weights were loaded from a version pre-trained on ImageNet, a large scale computer vision benchmark dataset [42]. The final layer of each was overwritten to match the number of classes for this dataset (27) and whose weights were randomly initialised. To provide an alternative to the pre-trained models, we also included a small custom Convolutional Neural Network (CNN) trained from scratch (see Table S4.1). Finally, once all four of these models were trained (that is, MobileNetV2, ResNet18, VGG16, and our custom model), an ensemble model [317] was constructed by taking the concatenated 108-dimensional probability vectors (27 classes $\times$ 4 models) produced by the individual classifiers and feeding them into a single trainable linear layer that outputs the final 27-class distribution. The linear layer was trained on the same training set while keeping the base models fixed. A comparison of the model sizes is given in Table S4.2.

Each of the five above models used the same training hyperparameters: Adam optimiser, learning rate of 0.001, a batch size of 32, and 50 epochs. For the evaluation of these classifiers, a set of metrics was employed to assess their performance comprehensively: accuracy, precision, recall, F1 score, and top-5 accuracy. Figure 4.5 shows the best validation accuracy for each training process across different models and levels of synthetic data added. While there is an upward trend seen in classification performance by adding synthetic labelled samples, the validation set used also came from BirdNET pseudo-labels.

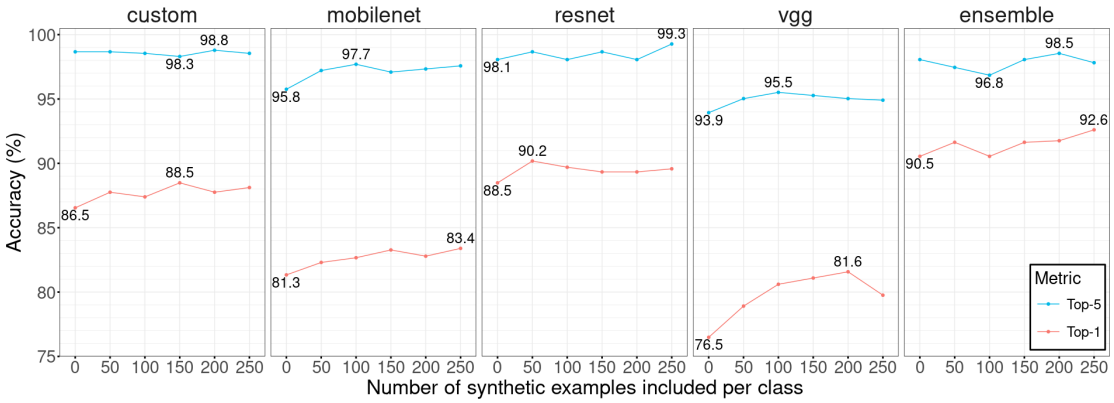


Fig 4.5. Classification Accuracy Results. Top-1 (red line) and Top-5 (blue line) validation accuracy (y-axis) with varying synthetic data per class (x-axis). All models show a positive trend in accuracy improvement when including synthetic examples, particularly VGG. While the ResNet model's accuracy decreases beyond 50 additional synthetic samples per class, these values are still better than with only real data.

A better test of each model's robustness uses human-verified samples. Although the ensemble model achieved the highest validation accuracy, it lagged behind BirdNET on the test set — likely because training relied on only highly confident BirdNET predictions. To build a more robust model, we retrained the ensemble (with and without synthetic examples) using knowledge distillation, where the model learns BirdNET's output distributions rather than one-hot labels. This approach should better transfer BirdNET's insights as a state-of-the-art bird call classifier.

As mentioned in Section 4.3, a larger dataset of 26,058 samples was used, again with a 90-10 split for training and validation. This was done in order for the model to learn the more difficult examples and better approximate BirdNET's distribution than only its most confident outputs. However, including less confident pseudo-labels to enhance generalizability risks incorporating a non-negligible number of incorrect labels, a problem potentially exacerbated by background noise from the wind farm data.

To train using knowledge distillation, the logits from the embeddings were converted into probabilities using the softmax function, with a temperature $T = 3$, Kullback-Leibler (KL) divergence was used as the distillation loss and cross-entropy as the loss for the hard labels. The losses were combined with a weight of $\alpha = 0.7$ for the distillation loss and $1-\alpha = 0.3$ for the cross-entropy loss. Figure 4.6 shows that, while the models trained on BirdNET's predictions from the real data alone do give reasonable performance on the test data somewhat, the models trained with both real and synthetic data outperform the ones trained on real data alone.

4.6 Discussion

The synthetic images generated by Stable Diffusion appeared more realistic than those produced by ACGAN. Training off-the-shelf models on a dataset combining the original data with DDPM-generated samples yielded validation accuracy consistently higher than using real data alone. This suggests that increasing data quantity and quality can be as effective as, or more so than, model-centric improvements or domain-specific features engineering [264].

The classifiers employed included MobileNetV2, ResNet18, VGG16 with ImageNet-pretrained weights and the last few layers unfrozen, and a small custom CNN trained from scratch. An ensemble model was trained, combining outputs from these four models and adding a final linear layer. The ensemble model, augmented by training with 250 extra synthetic examples added per species, achieved the highest accuracy overall on the

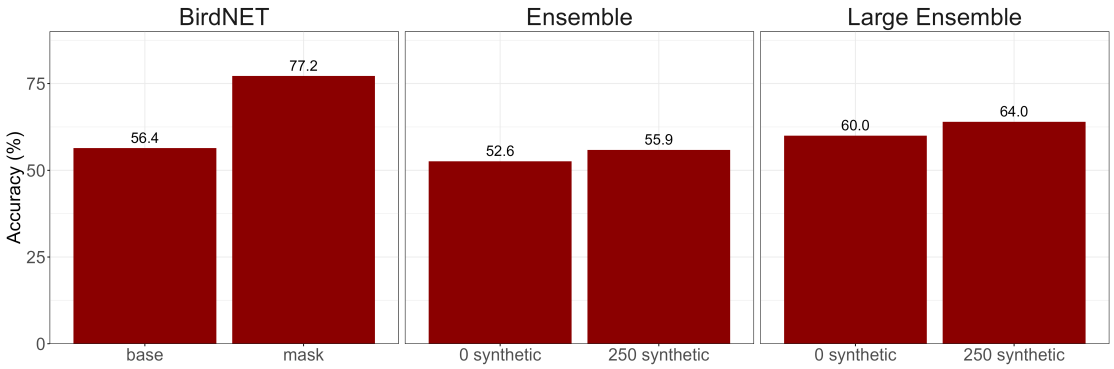


Fig 4.6. Test Set Evaluation. The bars represent the classification accuracy of models on human-labeled test data. Models include: BirdNET-base (comparing against the model's single class prediction), BirdNET-mask (using the maximum probability from the 27 indices in the BirdNET embeddings corresponding to our bird species of interest, rather than all classes), Ensemble (the ensemble of four models above retrained with knowledge distillation, using 0 or 250 synthetic samples per class), and Large Ensemble (ensemble retrained on a larger dataset including less confident BirdNET predictions, again with knowledge distillation and 0 or 250 synthetic samples per class). The Ensemble model with both real and synthetic samples improves accuracy by 3.3% compared to real samples alone but underperforms BirdNET-base and BirdNET-mask. The Large Ensemble model achieves 60.0% and 64.0% accuracy with real and real plus 250 synthetic samples, respectively, surpassing the BirdNET-base but not BirdNET-mask. Further improvements would require more diverse data and higher-quality samples (e.g., including human-labelled data in training and validation).

validation set of 91.3%.

Performance was evaluated on a hold-out test set with human-verified labels from the same wind farm sites as the pseudo-labelled data, comparing six setups: BirdNET, BirdNET with non-target (not in the 27 chosen) classes masked, an ensemble of four models trained on real and real plus synthetic data, and the same with a larger, less confident pseudo-labelled dataset with or without synthetic data. These setups achieved accuracies of 56.4%, 77.2%, 52.6%, 55.9%, 60.0% and 64.0%, respectively. Our results suggest that incorporating high-quality synthetic spectrograms into training data improves bird song classification performance on expert-labelled data.

The largest performance gain came from expanding the dataset to include more real samples (7.4%), but required a 3.2× increase in the dataset size. Augmenting both the initial and larger real datasets with synthetic examples further improved performance

to 3.3% and 4.0%, respectively. However, this only required adding 82% and 26% more samples to the datasets, respectively. Although real data scaling yielded the largest individual gains in classification performance, adding synthetic samples provided meaningful additional improvements more efficiently in terms of data volume, collection effort, and training time.

Our final model achieved accuracy of 64% on a 27-species bird sound classification task with expert-verified labels. While the state of the art is 13% higher, the attained accuracy is already sufficient to provide reliable broad-level species presence and relative activity patterns, even if individual predictions are not always highly confident. We suspect the bird classifier will be ecologically and practically useful overall, supporting scalable, automated biodiversity monitoring where traditional surveys are infeasible.

There still exists a notable gap between BirdNET's performance (77.2% for BirdNET-mask) and the best ensemble classifier (64.0%) on the hold-out test set. Several factors may contribute to this discrepancy. First, our dataset of 26,000 spectrograms corresponds to less than 24 hours in total, which may not be enough to generalise quite well enough even to a handful of habitats on Irish wind farms. BirdNET's was trained on a vast global dataset and likely captures a broader range of acoustic features from a wide variety of habitats and variations in bird vocalizations, enabling it to generalize better to the noisy wind farm environment. Second, the reliance on entirely pseudo-labelled data, processed by BirdNET itself, especially in the larger, less confident dataset, may introduce noise in the training process, as a non-negligible portion of these labels could be incorrect, as noted earlier. Additionally, while synthetic spectrograms improved performance, their quality and diversity may not fully replicate real-world variations, limiting the classifiers' ability to match BirdNET.

We have identified some further limitations with our approach, which we list below to support future work:

- **Automatic data pruning.** Several generated samples were noticeably distorted or contained artifacts, reducing their realism. A pruning method of removing such poor examples at scale would improve the quality metrics further and likely lead to more relevant learning.

- **Pseudo-label bias.** Both the generative and classification models were trained and tuned by BirdNET pseudo-labels, with the embeddings used for the knowledge distillation of the final embedding models. This poses a risk of inheriting BirdNET's biases and systematic errors, especially given it may itself have been

trained on very little data from a domain close to our noisy environment (Irish wind farms).

- **Computational overhead.** The DDPM generation phase is slow, requiring approximately 1,000 denoising steps per image batch. Recent work [93], using a similar VQ-VAE architecture, has improved efficiency in synthetic bird sound generation.
- **More controllable generation.** Generative AI can be guided by adjusting specific input parameters. While species labels are effective here, other metatata such as weather, call type or time information could improve the variety of samples and align better with users' needs. Some Stable Diffusion papers [192] have used an inpainting mask to direct the image generation process to specific areas in the spectrogram image.

The audio classification and generation approach developed for bird species can be adapted for other taxa with distinct vocalizations, such as bats, anurans, or marine mammals. For species with high-frequency calls, such as bats or certain insects, the preprocessing steps should be adjusted to target the relevant frequency range common to all species being classified, enhancing clarity of the spectrogram for deep learning models. For both generative and classification models, the question of whether to train a single model for multiple taxa or a separate models per taxon is worth exploring. By reducing reliance on human-labelled data, this method could support broader conservation efforts through enhanced species classification.

4.7 Conclusion

Incorporating generative AI into bioacoustic pipelines can significantly enhance classification accuracy on domain-specific data, offering ecologists a valuable tool to supplement datasets constrained by data collection logistics and costs. This study demonstrates that augmenting bird species audio classification tasks with high-quality synthetic data generated using a Stable Diffusion model, improves performance in noisy environments compared to relying solely on the original data. Adding up to 250 DDPM-generated samples per class improved classification accuracy by 3.3% for the ensemble model and up to 64.0% for the large ensemble model on human-labeled test data, though it did not surpass BirdNET's accuracy. The DDPM-generated spectrograms were highly convincing,

achieving a 7% higher Inception Score (1.73 vs. 1.62), a 51% lower FID score (32.05 vs. 64.97) and 8% lower FAD score (0.98 vs. 1.07), and underscore the potential of synthetic data augmentation over traditional approaches like model expansion or fine-tuning and so warrants further research. We foresee broader applications of synthetic generation of bird sounds - such as non-invasive bird deterrence and enriching educational tools and public engagement with nature - emphasizing the transformative impact of generative AI on ecological research.

Author Contributions

AG, ID and AP conceived the ideas and designed methodology; AG and EK collected the data; All authors analysed the data; AG wrote the code and made the data available; AG, ID and AP led the writing of the manuscript. All authors contributed critically to the drafts and gave final approval for publication.

Acknowledgments

This paper comprises an output of the Nature+Energy Project, funded by Research Ireland (12/RC/2302_P2), industry partners (Wind Energy Ireland, NTR Foundation, ESB, SSE Renewables, Asper Investment Management, Ecopower, Energia, EnergyPro, Greencoat Renwables, Ørsted) and MaREI, the Research Ireland Research Centre for Energy, Climate and Marine Research and Innovation.

We thank our labellers Harry Hussey, David Kelly, Seán Ronayne and Mark Shorten, who annotated the majority of the audio files in the test set using NEAL. We also thank the on-site personnel and surveyors for carrying out recorder maintenance, i.e., data and battery transfers, as the wind farms are widely spread across Ireland.

Andrew Parnell's work was supported by: the UCD-Met Éireann Research Professorship Programme (28-UCDNWPAI); a Research Ireland, Northern Ireland's Department of Agriculture, Environment and Rural Affairs (DAERA), UK Research and Innovation (UKRI) via the International Science Partnerships Fund (ISPF) under Grant number [22/CC/11103] at the Co-Centre for Climate + Biodiversity + Water.

Data and Code Availability

Code and documentation for training both generative models, all classification scripts are available at `https://github.com/gibbona1/SpectrogramGenAI`. The post-processing scripts, to generate the plots used in this paper, are also available here. The large datasets of spectrograms, embeddings and labels are available at `https://zenodo.org/records/15729847`.

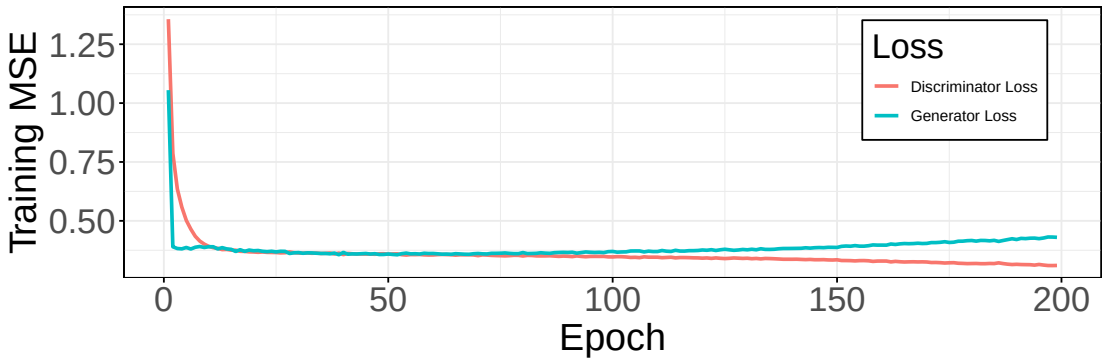
4.A Supplemental Information

Layer Type	Output Shape	Param #	Description
Input	[-1, 1, 256, 256]	-	Spectrogram input
Conv2d	[-1, 16, 256, 256]	160	3x3 convolution, 16 filters
MaxPool2d	[-1, 16, 128, 128]	–	2x2 max pooling
Conv2d	[-1, 32, 128, 128]	4,640	3x3 convolution, 32 filters
MaxPool2d	[-1, 32, 64, 64]	–	2x2 max pooling
Conv2d	[-1, 64, 64, 64]	18,496	3x3 convolution, 64 filters
MaxPool2d	[-1, 64, 32, 32]	–	2x2 max pooling
Conv2d	[-1, 128, 32, 32]	73,856	3x3 convolution, 128 filters
MaxPool2d	[-1, 128, 16, 16]	–	2x2 max pooling
Flatten	[-1, 32768]	–	Flatten to 1D vector
Dropout	[-1, 32768]	–	Dropout layer
Linear	[-1, 256]	8,388,864	Fully connected, 256 units
Dropout	[-1, 256]	–	Dropout layer
Linear	[-1, 27]	6,939	Fully connected, 27 units
Total		**8,492,955**	

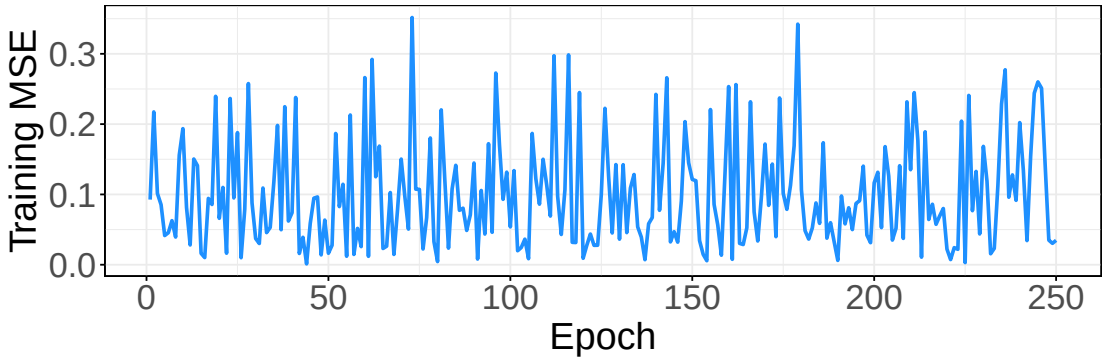
Table S4.1. Architecture of the Custom Model

Model	Total Params	Multiply-Adds (billions)	Est. Size (MB)
ResNet18	11.19M	2.39	92.94
VGG16	134.37M	20.30	648.40
MobileNetV2	2.26M	0.20	30.02
Custom	8.49M	0.24	47.65
Ensemble	156.32M	20.88	762.61

Table S4.2. Key Metrics for Each Model



(a) ACGAN Training Losses



(b) DDPM Training Loss

Fig S4.1. Generative model training losses. ACGAN has generator and discriminator MSE losses. DDPM has a single MSE comparing the actual and predicted noise.

WIND ENERGY IRELAND Annual CONFERENCE 2025

Using Generative AI to improve bird call classification at wind farms

Anthony Gibbons[1], Emma King[2], Ian Donohue[2], Andrew Parnell[3]
1 Hamilton Institute, Department of Mathematics and Statistics, Maynooth University
2 Zoology, School of Natural Sciences, Trinity College Dublin
3 School of Mathematics and Statistics, University College Dublin

Challenge / Research Question

- Birds are key ecosystem health indicators and are increasingly monitored through cost-efficient acoustic methods.
- Machine learning can scale bioacoustic species classification but depends on costly, expert-labeled training data. Wind farms are particularly challenging given the background wind and turbine noise.
- Data augmentation can boost classification accuracy by increasing the diversity of training data and is cheaper to obtain than expert-labelled data, but many of these techniques are not suitable for audio spectrograms.

Methods

- We investigate two generative AI models as data augmentation tools to synthesise spectrograms and supplement audio data: Auxiliary Classifier Generative Adversarial Networks (ACGAN) [1,2] and Denoising Diffusion Probabilistic Models (DDPMs) [3,4].
- Alongside these new approaches, we present a new audio data set of 640 hours of bird calls from wind farm sites in Ireland, approximately 800 samples of which have been labelled by experts.

Results

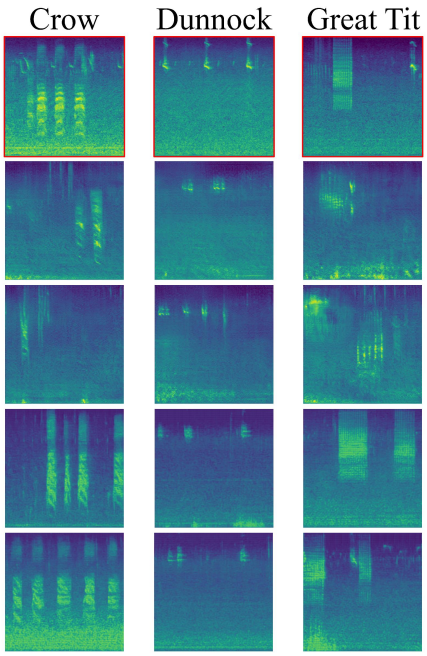


Figure 1: Real and Synthetic Spectrograms. Can you tell which are real and which are fake?

- DDPM performed particularly well in terms of both realism of generated spectrograms and accuracy in a resulting classification task. See Figure 1.
- The x-axis of each spectrogram shows time (0–4 seconds), while the y-axis shows frequency (0–12 kHz) on the mel scale. Brighter colors indicate higher energy or loudness.
- Synthetic images were then generated by the diffusion model, 250 per class, for a total of 6,750 synthetic spectrograms. These will serve as further training data by supplementing the original training data used for the bird species classification models.
- Training an ensemble of classification models on real and synthetic data combined gave 92.6% accuracy (and 90.2% with just the real data) when compared with highly confident BirdNET [5] predictions. See Figure 2.

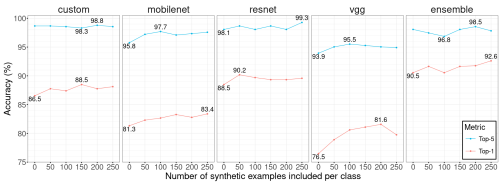


Figure 2: Classification Accuracy Results. Top-1 (red line) and Top-5 (blue line) validation accuracy (y-axis) with varying synthetic data per class (x-axis). All models show a positive trend in accuracy improvement, particularly MobileNet and VGG. While the ResNet model's accuracy decreases beyond 50 additional synthetic samples per class, these values are still better than with no synthetic data added.

Impact

Our approach can be used to augment acoustic signals for more species and other land-use types, and has the potential to bring about a step-change in our capacity to develop reliable AI-based detection of rare species. The paper is in preprint on arXiv [6].

References

[1] Fu, Y., Yu, C., Zhang, Y., Lv, D., Yin, Y., Lu, J. and Lv, D. (2023), 'Classification of birdsong spectrograms based on DR-ACGAN and dynamic convolution', Ecological Informatics 77, 102250.

[2] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. (2014), 'Generative Adversarial Networks'. URL: https://arxiv.org/abs/1406.2661

[3] Ho, J., Jain, A. and Abbeel, P. (2020), 'Denoising diffusion probabilistic models'. URL: https://arxiv.org/abs/2006.11239

[4] Rombach, R., Blattmann, A., Lorenz, D., Esser, P. and Ommer, B. (2022), 'High-resolution image synthesis with latent diffusion models'. URL: https://arxiv.org/abs/2112.10752

[5] Kahl, S., Wood, C. M., Eibl, M. and Klinck, H. (2021), 'BirdNET: A deep learning solution for avian diversity monitoring', Ecological Informatics 61, 101236.

[6] Gibbons, A., King, E., Donohue, I., Parnell, A. (2024) 'Generative AI-based data augmentation for improved bioacoustic classification in noisy environments', URL: http://arxiv.org/abs/2412.01530.

Bord na Móna | OW OCEAN WINDS

WIND ENERGY IRELAND
Annual Conference 2025

Fig S4.2. Poster for Wind Energy Ireland Conference 2025.

5

ExActR: A Shiny app for creating ecosystem extent accounts

Anthony Gibbons [1*], Francesco Martini [3], Cian White [3], Emma King [2], Jane C. Stout [3], Ian Donohue [2], Andrew Parnell [1]

1 Hamilton Institute, Department of Mathematics and Statistics, Maynooth University, Kildare, Ireland
2 Zoology, School of Natural Sciences, Trinity College Dublin, Dublin, Ireland
3 Botany, School of Natural Sciences, Trinity College Dublin, Dublin, Ireland

* anthony.gibbons.2022@mumail.ie

Abstract

Ecosystem accounting is a structured way to integrate nature into sustainable decision-making. The System of Environmental Economic Accounting-Ecosystem Accounting

(SEEA-EA) was adopted by the United Nations as a set of international standards for the collection of habitat data and compiling ecosystem accounts. The ecosystem extent account is one of the four pillars of the SEEA-EA framework, where the spatial composition of an ecosystem accounting area is grouped by habitat types, and the land cover change over time is quantified. Although a variety of tools exist for preparing extent accounts, most of them require moderate to high levels of technical expertise. Here, we present *ExActR* (Extent Accounts in R), an open-source application for generating extent accounts using shapefiles, a geospatial vector data format. The app is built in `R` and the associated Shiny framework, which automatically updates as the user interacts with it. The application supports multiple timepoints, where extent accounts (tables) are generated for consecutive pairs of timepoints, accommodating users' needs for dynamic ecosystem assessments across several periods. Data visualisations are generated in the form of both interactive (leaflet) and static maps of each timepoint, and barplots to illustrate land type composition and change. A version of the app has been deployed (available at `https://gibbona1.shinyapps.io/extent_app/`), offering a space for interactive exploration of ecosystems. Shiny's reactivity, combined with JavaScript plugins for copying tables into multiple formats, including LaTeX and plots, make the application results suitable to insert directly into reports. The app is suitable for using with any spatial grouping variable, such as ecosystem type, elevation, rainfall or population. We test its functionality on small and large study sites on CORINE land cover data [59], as well as land cover maps generated using very-high resolution satellite imagery of a wind farm site in Ireland, during and post construction, demonstrating its ability to adapt to various land cover classification systems. The tool can be used to understand, visualise and track changes in ecosystem assets, aiding interpretation by both scientists and stakeholders.

Keywords— SEEA, Ecosystem Accounts, Ecosystem Extent, Land cover Change, Shiny App, R, Shapefiles, Geospatial Data, Data Visualization

5.1 Introduction

Land cover maps document the area, configuration and category of what is physically present on the Earth's surface in an area of interest. They are useful for environmental monitoring, particularly as comparing two land cover maps of the same location over a period of time allows detection, quantification and monitoring of land cover change, a key driver of biodiversity loss [94, 103, 119]. Loss of natural habitats due to anthropogenic

land use change creates cascading effects that ultimately impact humans through loss of biodiversity and reduced supply of key ecosystem services [28, 54, 156, 273]. In addition to human modifications to natural landscapes, climatic changes can also contribute to habitat degradation and loss [114]. In many cases, the interaction of land conversion and climate change exacerbates biodiversity loss due to synergistic effects between the two pressures [180], although such effects are likely dependent on the taxa under study and local conditions [171, 311]. Although most changes in land cover and land use negatively impact nature, it is also possible to reverse the trends and lead to the recovery of biodiversity through appropriate land management. For example, agricultural landscapes are known to host greater biodiversity when landscape heterogeneity is higher [202] and there are examples of biodiversity recovering following restoration actions [45, 107].

Studies documenting changes in land cover and land use are not new, but until recently there was no agreed standard framework to report them. The EU has demonstrated the importance of systematically documenting and monitoring ecosystems, thereby sparking broader interest in consistent reporting frameworks [113]. In 2021, the United Nations (UN) approved the System of Environmental Economic Accounting – Ecosystem Accounting (SEEA-EA), the first step in providing an agreed and standardized methodology to report the extent, condition, and delivery of ecosystem services (e.g. fresh water and climate regulation) at the national, regional, and local scales [274]. A number of studies have since been published that use the system and many governments are starting to produce the accounts required [9, 61, 230]. The EU's Corporate Sustainability Reporting Directive (CSRD) is another case study where ecosystem accounts can provide a useful metric to inform reporting requirements where reports may need to be produced at scale, and often by non-experts, to measure organisations impact on ecosystems.[1] The CSRD will require organisations, as soon as Jan 1st 2025, to disclose the risks climate change will have on them and the impacts their business can cause to both the climate and ecosystems.[2] The intended aim is that the accounts produced following the SEEA-EA will become a global standard for monitoring environmental change, including any effects upon the economy and, with regular updates, offer continuous insight into the the status and trends of ecosystems, as captured by their extent, condition, and provision of ecosystem services.

The first step in the SEEA-EA, upon which all subsequent steps are based, is the Extent Account. This account quantifies the extent (usually in hectares or km^2) of

[1] https://eur-lex.europa.eu/legal-content/en/TXT/?uri=CELEX:32023R2772

[2] https://seea.un.org/content/how-natural-capital-accounting-can-help-accelerate-finance-nature

each ecosystem type in a study area—called Ecosystem Accounting Area—at one point in time. When multiple time points are available, the SEEA-EA provides a consistent framework to present any changes, represented by tables and maps. The extent account of the SEEA-EA can serve as a template to follow, not only by studies producing accounts, but by any researcher or land manager interested in documenting changes in land cover over time. Several classes of tool are available to calculate land cover changes, each with strengths and weaknesses. Organisations with in-house GIS expertise will develop custom workflows using tool such as ArcGIS, QGIS, or EnSym, while those with in-house programming expertise could develop scripts in languages such as R or Python for automating land cover change assessments. Indeed custom tools for SEEA-EA exist - ARIES creates extent accounts for any EU region using Corine Landover Maps. Here, we introduce a tool for an underserved user group - non technical land managers. Many land managers are not experts in GIS or programming, but have a wealth of habitat maps based on habitat classifications to their region. Our target audience are those land managers, project managers and ecologists who require point and click tools that ingest custom habitat maps allowing the creation of Extent Accounts. We provide a tool following the SEEA-EA approach, without the need for advanced technical expertise. Further, most tools fail to make the operations repeatable, limiting reproducibility (we expand on this key point in the Discussion).

We introduce *ExActR*, an interactive shiny app to facilitate the generation of extent accounts and accompanying visualisations to streamline this process for spatial datasets. The app allows users to upload two or more sets of vectorised spatial data and calculate the increases and decreases in area according to a grouping variable of interest, e.g., habitat type.

Some of the key features of the app are:

- The grouping variable is selected dynamically from the data loaded from each shapefile. These names can be different in each timepoint to help with situations where naming conventions change or the column names of interest are indexed in some way. For example, users can compute the extent account between the `Habitats_2000` column of one dataset and the `habitats_18` column of another dataset.

- While the majority of extent accounts are expected to just have two timepoints (an opening and closing map), it is possible to include multiple timepoints. With n time points, this generates $n-1$ extent accounts, with extent account i showing

the difference from timepoint $i-1$ to i, for $i \in 2, \ldots, n$. Note that the timepoints are assumed to be in chronological order.

- As well as the leaflet maps on the main page, the app provides a tab with an array of plot outputs to visualise changes in extent.
- Using both Shiny's existing functionality and some JavaScript extensions, users can easily copy the table and plot outputs to their clipboard, or bulk export all outputs for the session.

The structure of our paper is as follows. We first provide the overall layout of the Shiny application, as well as a brief overview of the libraries used. We then demonstrate a step-by-step workflow of how the app is utilised for generating extent accounts and give more detail on the computational back-end. We provide three examples of extent accounts to demonstrate features of the app and to show its flexibility when it comes to handling vectorised spatial data. We conclude with a discussion of how the app compares with other methods and plans for future work. Source code for the *ExActR* app, as well as a link to a working demo on an RStudio server, is available on GitHub at `https://github.com/gibbona1/ExActR`.

5.2 Methods

User Interface (UI)

The *ExActR* app (Fig. 5.1) was built in R [207, 228] using the Shiny [33] framework. Shiny is itself a package in R. No knowledge of HTML, CSS, or JavaScript is necessary to build a simple application in Shiny, but small amounts were used here to enhance certain features. One of the many benefits of the Shiny framework is that end-users do not need any knowledge of R programming to interact with the data and produce accounts. Shiny has already been used in developing ecology-related apps and decision-support tools, such as in acoustic annotation of wildlife [79], species-habitat modeling [298], conservation management [193], forest structure assessment [245] and visualising macroalgal canopy photosynthesis levels [119].

ExActR makes use of several open-source R packages [33, 7, 11, 266, 31, 234, 35, 126, 72, 196, 286, 185, 300, 288, 287], of which the most notable are:

- *sf* - Simple features, standardised way to read and manipulate spatial vector data.

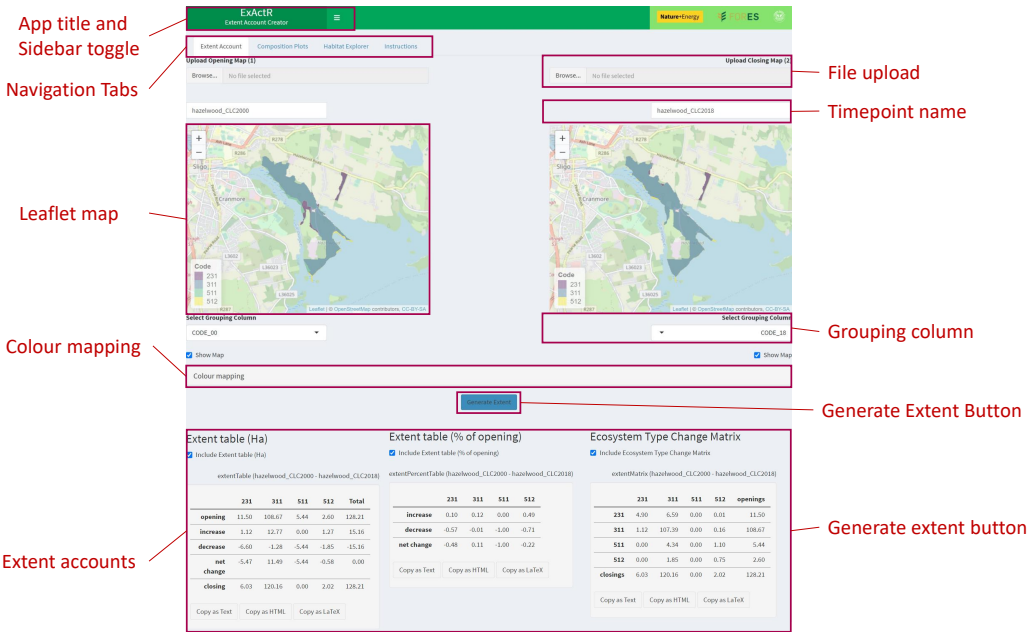


Fig 5.1. Main components of the *ExActR* App User Interface. Each timepoint has a **File Upload** widget and a **Leaflet Map** displaying the spatial data. The **Timepoint Name** can be edited (the default is the shapefile name) and carried down to all outputs. The polygons in the data for each timepoint are coloured by a **Grouping Column**, which will later be used to generate extent. The **Extent accounts** (Main extent table, extent as a percent of opening area and ecosystem type change matrix) are a set of tables and the main desired output of the app. The tables in the diagram are placed side by side to save space but these are stacked vertically in the app, with different timepoints of each table type on the same horizontal if there are multiple. Most settings are contained in the **Sidebar** such as choosing the coordinate reference system, adding/deleting timepoints as well as downloading all outputs in bulk. Users can navigate to other **Tabs** to access other plots, tables and instructions.

- *terra* - Spatial data analysis, used here for intersecting two maps.
- *shinyBS* - This contains extra user-interface (UI) objects such as collapsible panels.
- *shinyjqui* - Allows jQuery UI functionality, primarily used here for resizable plots.
- *shinydashboard* - Dashboard layout. Moving the less-used options to the sidebar reduces clutter in the Shiny app's main body.
- *shinycssloaders* - Illustrates loading for some of the slower plot and table outputs.

- *leaflet* - Interactive visualisation of spatial datasets on maps.
- *viridis* - default colour palette for plots.
- *colourpicker* - choosing colours for the map and plot outputs.
- *ggplot2* - Graphics library for bar charts (`geom_bar`) and sf objects (`geom_sf`).
- *magick* and *xfun* - keep these dimensions of resized images on exporting.
- *dplyr* - data and string manipulation, respectively. Part of tidyverse.
- *xtable* - formatting tables as LaTeX and HTML when exported.

Workflow and computation details

Upon opening the app, users are presented with two upload widgets for map data, with a blank map window under each. Once the spatial data are uploaded, the interactive leaflet maps will be rendered for each. The extent tables below this are similarly hidden until they have data to work on. The **Composition Plots** and **Habitat Explorer** tabs remain blank until data are uploaded and accounts are generated.

The standard use of the app to generate extent accounts is as follows:

1. Upload opening files (requires `.shp`, `.dbf`, `.shx` and `.prj` files uploaded together or in a `.zip` file, or a `.geojson` file). Repeat with closing files. For multiple timepoints, use the sidebar button to add more and upload them in chronological order.
2. Select grouping variable, e.g. habitat classification system, for each time point. These do not have to be the the same name, but should be some ecosystem type to compare areas between time points. Any column from the data can be used as the grouping column and is not specific to those chosen in the case studies below. Moreover, a lookup table can be uploaded to replace the grouping column's codes with corresponding values, allowing users full flexibility in defining and interpreting land cover categories.
3. The colours for each unique value in the list of combined grouping variable values can be (optionally) edited using the colour pickers in the Colour Mapping Panel.
4. Click **Generate Extent**. This is included to stop the accounts from generating for some overly large grouping variables before the user is ready.

5. Wait for and inspect extent table and plot outputs, then copy or export as desired.

The Shiny framework employs reactive programming, allowing user changes in the interface inputs to automatically impact related outputs. This creates a seamless experience for users, as the app updates without the need for manual refreshes when settings are modified. Specifically, generating the Ecosystem Type Change Matrix represents the app's most demanding computational task as it involves intersecting two (possibly large) spatial objects and investigating changed and unchanged regions. The matrix is useful for quantifying changes in land use across multiple ecosystem types at the same time by showing the interaction between each pair of land cover types at opening and closing. Ensuring that this matrix does not refresh with every unrelated input change is crucial for a fluid user experience. The app's back-end features modular code to separate dependencies (user inputs) that influence the interface's outputs. Further details on generating this Ecosystem Type Change Matrix are provided below.

1. Uploaded `sf` objects are read in to `sfs[[i]]` for each timepoint i. `sfs` itself is a list.

2. Each timepoint i uses a grouping variable G_i (a column of `sfs[[i]]`) which is used for comparison across timepoints. Their unique values form a list G for a consistent plot colour map.

3. These `sf` objects can be optionally simplified using the **simplify geometry** option in the sidebar, reducing resolution to speed up computation at the cost of precision.

4. The spatial dataframes `sfs[[i-1]]` and `sfs[[i]]` are converted to spatial vectors using `terra::vect`, intersected using `terra::intersect`, then converted back into an `sf` object, `df_int[[i]]`, for $2 \leq i \leq n$ (see Fig. 5.2). Using `terra` here significantly speeds up generating the intersection compared with just `sf`.

5. The opening-closing pairs (g, g') of all combinations of values from G are then analysed. `df_int[[i]]` is filtered to just those polygons where $G_{i-1} = g$ and $G_i = g'$ The (g, g') entry of the matrix is the total area of these polygons (see Fig. 5.3).

The extent account with opening, additions, deletions and net change are calculated from the Ecosystem Type Change Matrix this way. The table with extent as a percentage of opening value is then calculated. The values default to zero when the opening value for a group is zero.

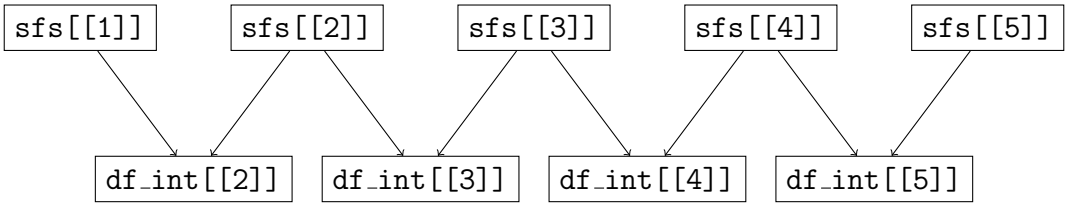


Fig 5.2. Creating sf intersection object. The intersection of maps from two consecutive timepoints takes the same index as the later timepoint, i.

	g_1	g_2	$\cdots$	g_n	Openings
g_1	$A[g_1, g_1]$	$A[g_1, g_2]$	$\cdots$	$A[g_1, g_n]$	$A[g_1, \cdot]$
g_2	$A[g_2, g_1]$	$A[g_2, g_2]$	$\cdots$	$A[g_2, g_n]$	$A[g_2, \cdot]$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
g_n	$A[g_n, g_1]$	$A[g_n, g_2]$	$\cdots$	$A[g_n, g_n]$	$A[g_n, \cdot]$
Closings	$A[\cdot, g_1]$	$A[\cdot, g_2]$	$\cdots$	$A[\cdot, g_n]$	Total Area

Fig 5.3. Ecosystem Type Change (Extent) Matrix diagram. For each pair (g, g') of opening and closing group values respectively the entry $A[g, g']$ in row g and column g' is the change in area from g in opening to g' in closing. The diagonal values $A[g, g]$ are the amounts unchanged for that group. Each row is the unchanged areas plus the reduction in area in that group. Each column is the unchanged areas plus the additions in area to that group. The sum of a row g, $A[g, \cdot]$ will equal the opening extent for that group. Similarly, the sum of a column g, $A[\cdot, g]$ will equal its closing extent. The sum of all openings should equal the sum of all closings, i.e. the total area of the site common to both timepoints.

5.3 Case Studies

5.3.1 Hazelwood (2000-2018, 2 timepoints)

The Hazelwood Demesne is located along the western shore of Lough Gill, Co. Sligo, Ireland[3]. This is a small site with an area of 128 Ha, of which approximately 100 Ha is protected under the Natura 2000 framework[4]. A land cover summary, using CORINE land cover classification, is shown below in Table 5.1, with corresponding maps in Fig. 5.4. Table 5.2 shows the extent accounts for the period.

[3] `https://www.coillte.ie/coillte-nature/ourprojects/restoringhazelwood/`
[4] `https://www.eea.europa.eu/themes/biodiversity/natura-2000`

(a)

Code	Area (Ha)	% Coverage
231	11.50	0.09
311	108.67	0.85
511	5.44	0.04
512	2.60	0.02

(b)

Code	Area (Ha)	% Coverage
231	6.03	0.05
311	120.16	0.94
511	0.00	0.00
512	2.02	0.02

Table 5.1. Hazelwood Land cover in (a) 2000 and (b) 2018. This shows the broadest view of land cover change between the two timepoints. We can see that all land cover types changed somewhat from (a) to (b), with all types decreasing except for code *311 (broad-leaved forest)* which increased during the period. The most notable change is code *511 (water courses)* which has decreased from 5.44 Ha in 2000 (a) to zero Ha in 2018 (b). Without further analysis, we cannot conclude that all of this area converted to code *311* simply because it is the only land cover type with a (net) increase in area.

5.3.2 Dargle (2000-2018, 4 timepoints)

The Dargle catchment area[5] is situated in the southern suburbs of Dublin City and northern County Wicklow in Ireland and is defined by the boundaries prescribed under the Water Framework Directive[6]. The Dargle catchment, covering 17,866 hectares, is ecologically rich, encompassing habitats listed under Annex I of the EU Habitats Directive which cover approximately 25% of the catchment area[7]. It also exhibits a significant presence of Natura 2000 sites. Extent accounts were built for Dargle in [61]. Fig. 5.5 shows some of the plot outputs generated by ExactR while generating the extent accounts. Changes in specific land cover types were more pronounced than in others. Over time, the overall magnitude of land cover changes at this site decreased, indicating increasing landscape stability in the most recent time period. The table outputs themselves are not shown due to size limitations on what can be shown legibly, but are available in Supplemental Information.

5.3.3 Meenadreen (2016-2022, 2 timepoints)

Meenadreen is an onshore wind farm located in Leghowney, South Donegal, Ireland. The wind farm consists of 38 Nordex N90 2.5 MW turbines with a total capacity of 95 MW.

[5] https://wicklowrivers.ie/rivers/dargle-catchment/

[6] https://water.europa.eu/freshwater/europe-freshwater/water-framework-directive

[7] https://www.npws.ie/legislation/eu-directives/habitats-directive

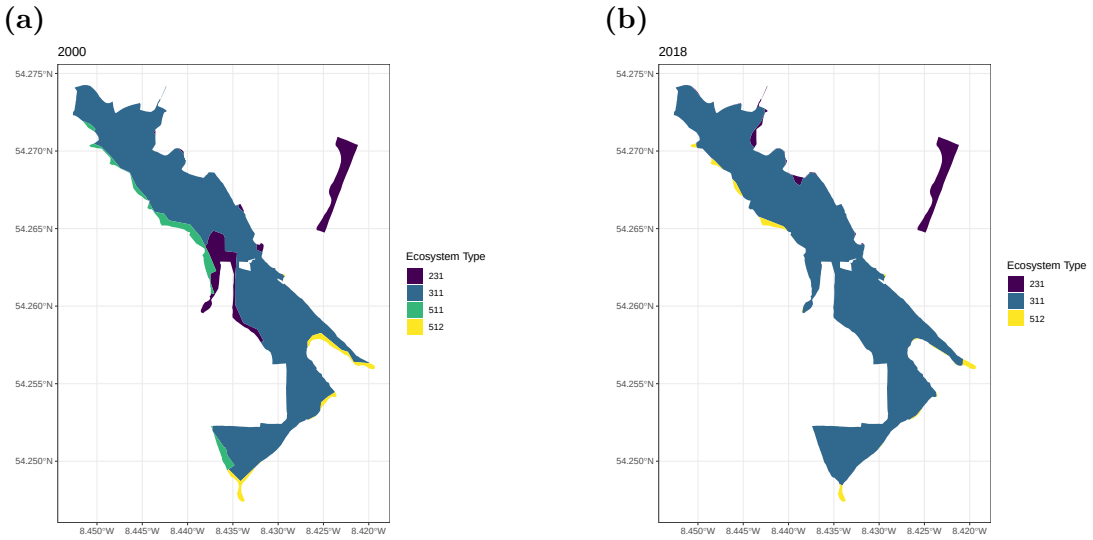


Fig 5.4. Hazelwood Land Cover Maps. The left (a) and right (b) plots show land cover at opening and closing respectively, coloured by CORINE Land Cover code. This smaller site has easily identifiable changes in habitat type of some polygons between opening and closing. A large portion of the habitat has remained unchanged. Code *511 (water courses)* is present on the west side of the opening map is not present at all in the closing map. Both opening and closing are dominated by code *311 (broad-leaved forest)*. Code *231 (pastures)*, and code *512 (water bodies)* are also present. A full collection of plot outputs for Hazelwood is available in Supplemental Information.

It was constructed in 2016 and has been operational since 2017. It is one of Ireland's largest wind farms both in terms of energy production and size, having an area of over 800 Ha.

Data for this study came from the Nature+Energy[160] project, a collaboration between academic researchers and industry partners that aims to measure and enhance biodiversity at onshore wind farms throughout Ireland. Meenadreen was selected due to it being relatively young, meaning we could obtain satellite imagery for early (2016) and post (2022) construction of the wind farm. A segmentation model was used on the GeoTIFF images to classify areas into one of nine habitat types: Artificial (e.g., roads, buildings, turbine hardstands), Conifer, Heathland, Hedgerow, Pasture, Scrub, Semi-Natural Grassland, Shadow (unidentifiable areas) and Water.

Due to the resolution of the image (0.5 m^2) being too high for the app to process for 25 km^2, we selected a 500 m $\times$ 500 m square at the centre of the site, representing $\sim$ 1%

(a) Ecosystem Type Change Matrix

	231	311	511	512	openings
231	4.90	6.59	0.00	0.01	11.50
311	1.12	107.39	0.00	0.16	108.67
511	0.00	4.34	0.00	1.10	5.44
512	0.00	1.85	0.00	0.75	2.60
closings	6.03	120.16	0.00	2.02	128.21

(b) Extent Table

	231	311	511	512	Total
opening	11.50	108.67	5.44	2.60	128.21
increase	1.12	12.77	0.00	1.27	15.16
decrease	-6.60	-1.28	-5.44	-1.85	-15.16
net change	-5.47	11.49	-5.44	-0.58	0.00
closing	6.03	120.16	0.00	2.02	128.21

(c) Extent Percent Table

	231	311	511	512
increase	0.10	0.12	0.00	0.49
decrease	-0.57	-0.01	-1.00	-0.71
net change	-0.48	0.11	-1.00	-0.22

Table 5.2. Hazelwood Extent Account from 2000 to 2018. This collection of tables make up the Ecosystem Extent Account itself. The codes in the margins of each table are CORINE Habitat Classifications. The Ecosystem Type Change Matrix **(a)** shows the changes in each habitat type from opening to closing. This allows users to see land cover change across multiple ecosystem types simultaneously. For example, the sum of all polygons in the intersected `sf` object that where code *512* in the opening column and *311* in the closing column and correspond to 1.85 Ha changing from *512* to *311* between opening and closing. The Extent Table **(b)** is the main component of the account and provides more compact information on the changes in land cover for each ecosystem type. Looking at column *512*, the opening and closing values can be directly seen in *(a)*, while the increase and decrease can be seen by summing the column and row values, respectively, not including the amount unchanged on the diagonal (0.75 Ha) in *(a)*. The Extent Percent Table **(c)** takes a selection of the rows from **(b)** and expresses them as a proportion of opening values for each ecosystem type. Values for **(a)** and **(b)** are in hectares. These tables correspond with the figures shown above in Fig. 5.4. Examining the change in code *511 (water courses)*, we can see from both the tables **(b)** and **(c)** that it has decreased completely to zero by closing (2018). As discussed in Table 5.1 however, we can see from looking at the *511* row of the matrix **(a)** that 4.34 Ha changed to *311 (broad-leaved forest)* and 1.10 Ha changed to *512 (water bodies)*. The matrix therefore gives a more granular picture than the net change in each land cover type during the period.

of the total area covered by the satellite image, as a sample. See Fig. 5.6 for the selected area at each timepoint and some of the visual results generated by the app. The extent account for Meenadreen is available in Supplemental Information.

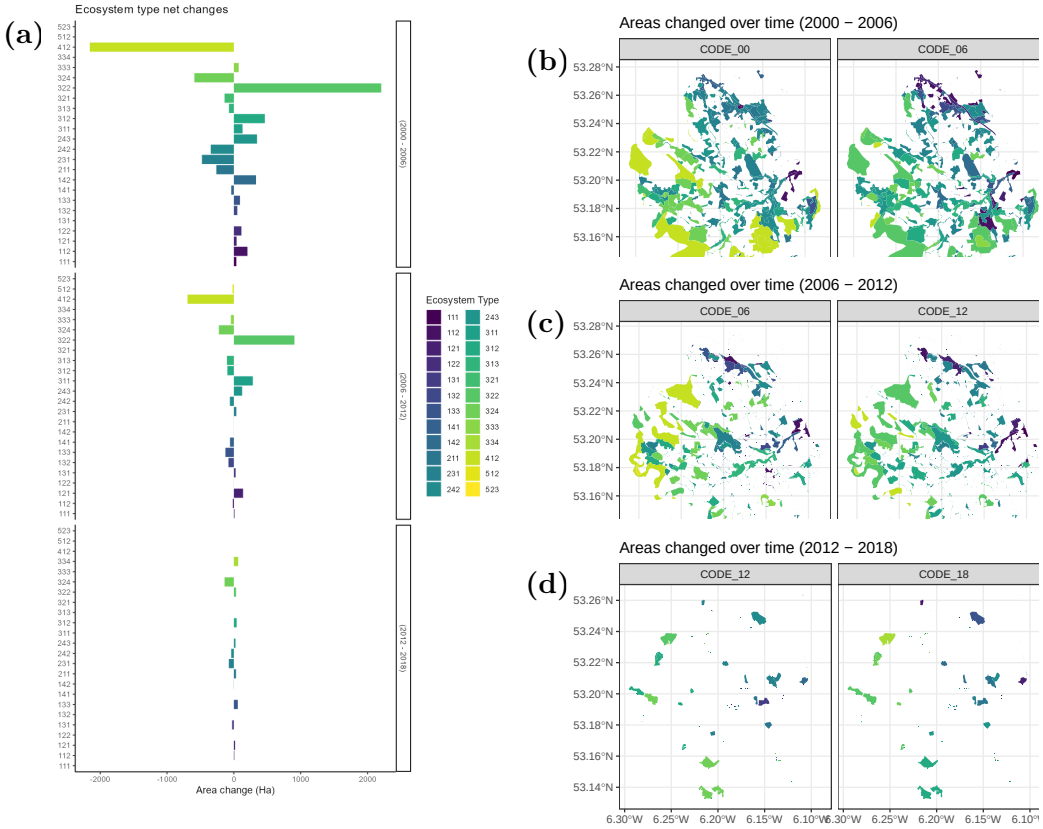


Fig 5.5. Dargle plot outputs. The polygons are coloured by CORINE Land Cover code. Some of the differences are more apparent than others, for example the reduction in code *412 (Peat bogs)* between 2000 and 2006 is much more stark than any change present in code *111 (Continuous urban fabric)* during the same period. The net changes in each ecosystem type can be clearly assessed using the bar charts in (a), where the overall magnitude of changes seems to reduce with each successive account. Given that the opening and closing maps have the same boundary, the sum of all increases and decreases for each account should be zero. The side-by-side plots (b)-(d) show the changed polygons each at opening and closing for each pair of consecutive timepoints. It agrees with (a) in that the magnitude of total area changed is much larger in the first account than the third. Additional outputs for Dargle are available in the supplemental material.

5.4 Discussion

The *ExActR* app allows users to upload two or more geospatial vector datasets and generate ecosystem extent accounts, quantifying land cover change over time. These accounts offer insights into the state and dynamics of focal environments, aligning with

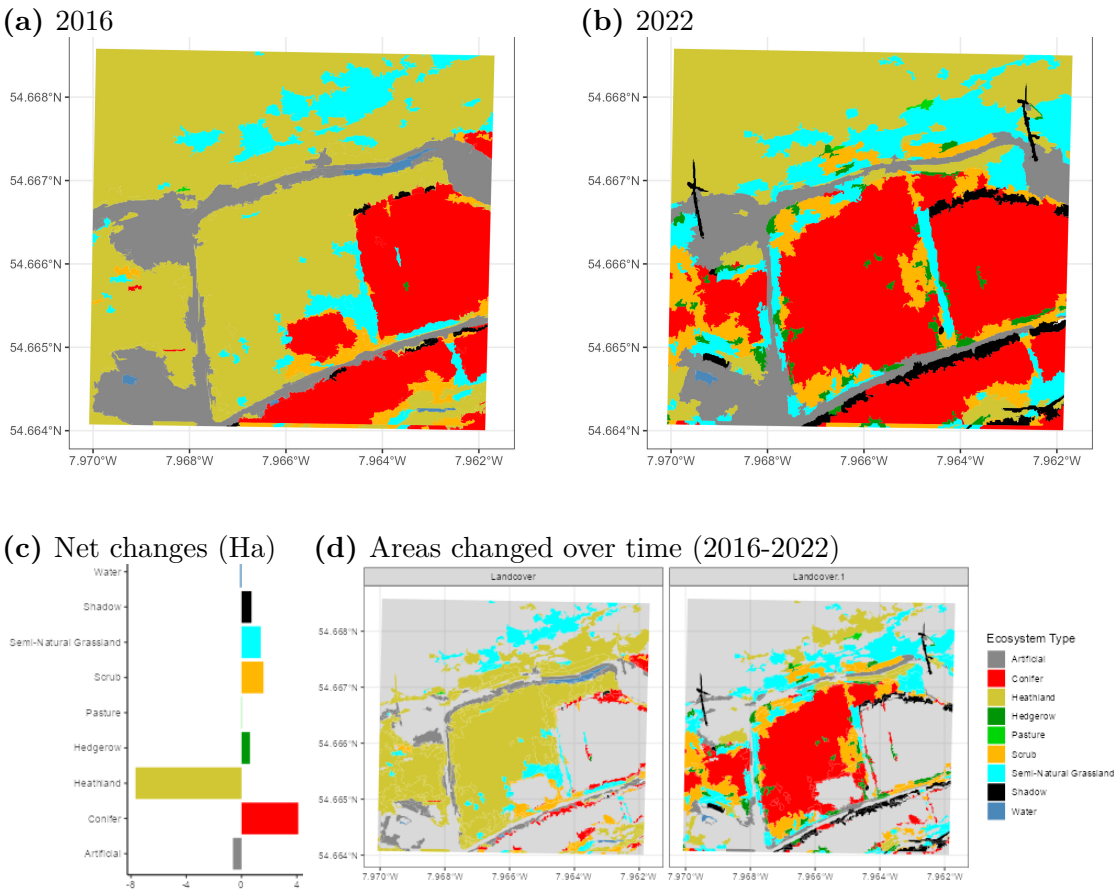


Fig 5.6. Meenadreen plot outputs. Artificial (roads) narrowed after construction, replaced with or returned to primarily heathland, scrub and semi-natural grassland. Hedgerow and Pasture which both had only 0.01 Ha in opening now had 0.61 and 0.03 Ha respectively in closing. Shadow also increased significantly, as evidenced by the shadows cast by the two wind turbines found in this site as well as shadows from conifer, and so the true habitat underneath is unclear. A large conifer plantation in centre of the closing map, replacing primarily heathland, is the most obvious change between the two timepoints. Artificial, Heathland and Water all experienced a net decrease.

conservation efforts. They are the first step in many Ecosystem Accounting pipelines, including the SEEA-EA. Our primary user group is expected to be environmental project managers who are assumed to have limited GIS or programming skills.

The app allows extent accounts to be easily implemented and reproduced, and the reactive elements in the Shiny framework make any changes made automatically carry

down to the connected outputs. Results can be exported in bulk for use in reporting. Together, these features comprise a significant advance on existing tools. For example, GIS workflows offer little reproducibility and require a lot of repetitive manual work. Similarly, Microsoft Excel work is cumbersome and would require one of the other tools to work with spatial data. The EnSym tool, which was used previously for similar work [61], is now available at Data4Nature[8]. However it it not freely accessible. Using R or Python requires familiarity with programming and having a suitable Integrated Development Environment (IDE), an obstacle likely most common among practitioners that may lack the necessary skills[297]. ARIES[9] is perhaps the most valuable resource currently available that does not require other software and is aligned with the SEEA-EA. However, ARIES does not allow users to upload spatial data/polygons of specific boundaries, which is often needed to researchers or practitioners working on specific sites, mostly at small spatial scales. Furthermore, ARIES uses the IUCN Ecosystem Typology classification [127] and does not allow as much flexibility with other typologies of ecosystem types. INCA[10] similarly does not allow the desired level of flexibility as its identified users are "EUROSTAT, Joint Research Centre (JRC) and EU Member States", which mostly work at national or regional level [23]. This last aspect is especially relevant to users who may be interested in using other classifications, such as the CORINE used here or others available for example at a national level. *ExActR* complements the options currently available by providing the following advantages: it can be used without any knowledge of R, GIS, or any other software or programming language; it accommodates different land cover classifications and thus could be used by any country or region with local land cover maps (e.g., annual Land Cover Map of the UK[11]); the code is open-source and can be adapted or modified by skilled users who may be interested in adjusting some of the features to meet their specific goals.

ExActR has some size limitations in terms of the size of spatial data it can handle. The CORINE maps for the entire country of Ireland could be loaded in about a minute, but calculating the extent change (by intersecting the two maps) could not be computed in a reasonable amount of time. Likewise, it took ∼3 minutes to load 25 km^2 land cover maps derived from very-high resolution imagery (0.5 m^2 pixels) and timed out while calculating extent change, highlighting that the minimum mapping area and total area

[8] `https://www.data4nature.com.au`
[9] `https://aries.integratedmodelling.org/aries-for-seea-explorer/`
[10] `https://ecosystem-accounts.jrc.ec.europa.eu/`
[11] `https://www.ceh.ac.uk/data/ukceh-land-cover-maps`

of the map are key factors in the performance of the app. Despite significant speed improvements by using `terra` over `sf` for the intersection step, further work (such as using `RCpp` or similar) is needed to accelerate the process and enable extent calculations for entire countries or larger regions, like Europe, to be computed feasibly. We also explored parallelization options using the `doParallel`, `future`, and `snow` packages in an attempt to further enhance performance. However, these methods did not yield promising results, possibly due of the sequential nature of spatial intersections. However, the use of ecosystem accounts is expanding fast and it is expected to be adopted more also from private businesses [53], which typically operate at small spatial scales. Future development work could also involve expanding support for other file types such as KML, which was developed for Google Earth, as well GeoTIFF, which stores the data in raster format.

The *ExActR* app generates ecosystem extent accounts which, while useful by themselves, are only one part of Ecosystem Accounting. Future work involving the other three ecosystem accounts in the SEEA-EA framework—condition accounts, ecosystem services flow accounts, and monetary ecosystem asset accounts—will help to offer a complete Ecosystem Accounting package for a range of users—ecology practitioners, people working on building accounts at local scales, as well as researchers interested in land cover dynamics.

Author Contributions

AG, FM, CW conceived the ideas and designed methodology; AG led the software development; FM, CW, EK tested the app; FM, CW, EK collected the data; AG, FM, CW analysed the data; AG, FM led the writing of the manuscript. All authors contributed critically to the drafts and gave final approval for publication.

Acknowledgements

We extend our gratitude to Coillte, Ireland's semi-state forestry company, for granting us access to their Hazelwood site via the Coillte Web Viewer App. This dataset was most often used in early development of the app.

We would also like to thank Energia for allowing us the use of data from Meenadreen. This publication is an output of the Nature+Energy Project, funded by Science Founda-

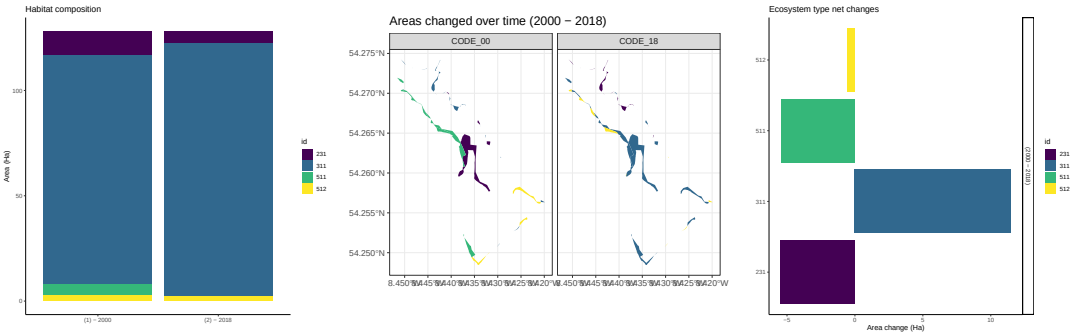
tion Ireland (12/RC/2302_P2), industry partners and MaREI, the SFI Research Centre for Energy, Climate and Marine Research and Innovation.

5.A Supplemental Information

Accessing Data

CORINE data are made available by the European Union through the Copernicus Service [59]. As for the boundaries of Hazelwood, the spatial data are available from Coillte[12]; however, you must have an ArcGIS account to download the data. For the Dargle catchment, the boundaries can be accessed at the EPA Geoportal[13] by navigating to *Water / Water Framework Directive → General information → Catchments Data Package - June 2022*.

To view the collection of *ExActR* outputs for the sites used in the Case Studies, the comparison of Dargle's output with the previously-published accounts, and the spatial data for Meenadreen, these are all available at `https://github.com/gibbona1/extent_app_data`. Below is a selection of the plots and tables generated.

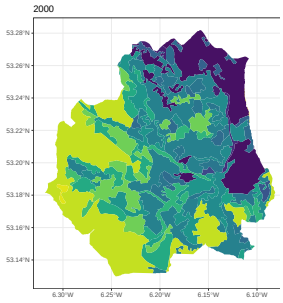


(a) Habitat composition **(b)** Changed Areas from opening to closing **(c)** Net Change in Areas

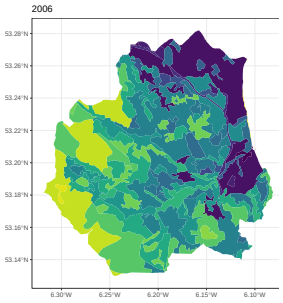
Fig S5.1. Additional plot outputs for Hazelwood.

[12] `https://coillte.maps.arcgis.com/apps/webappviewer/index.html?id=7b05ec6a44a14bd8b523ea1fcb78b4e9`

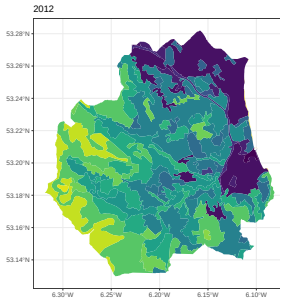
[13] `https://gis.epa.ie/GetData/Download`



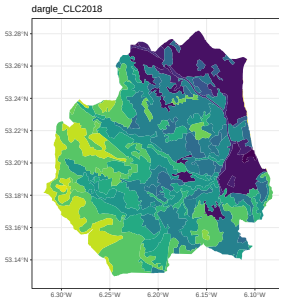
(a) Land cover in 2000



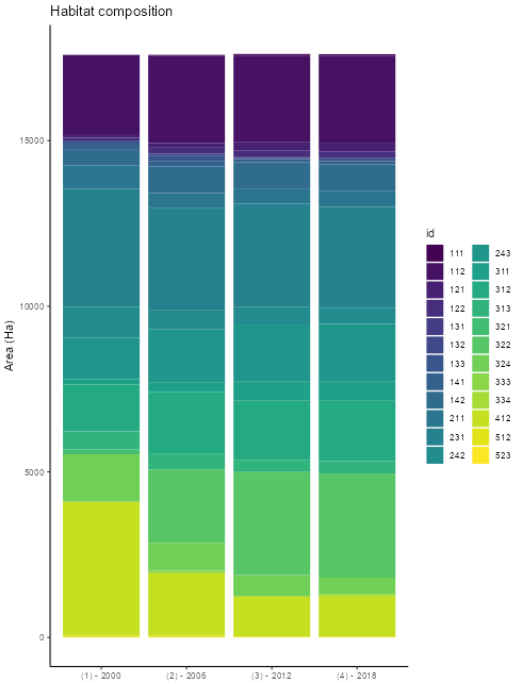
(b) Land cover in 2006



(c) Land cover in 2012



(d) Land cover in 2018



(e) Habitat composition

Fig S5.2. Additional Dargle plot outputs.

(a) Ecosystem Type Change Matrix

	Artificial	Conifer	Heathland	Hedgerow	Pasture	Scrub	Semi-Natural Grassland	Shadow	Water	openings
Artificial	2.41	0.07	0.48	0.09	0.00	0.13	0.39	0.21	0.02	3.79
Conifer	0.02	4.49	0.02	0.06	0.00	0.10	0.10	0.24	0.00	5.03
Heathland	0.65	3.99	5.50	0.29	0.00	1.68	2.22	0.21	0.00	14.54
Hedgerow	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01
Pasture	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01
Scrub	0.07	0.25	0.05	0.07	0.00	0.25	0.05	0.06	0.00	0.80
Semi-Natural Grassland	0.01	0.32	0.74	0.08	0.03	0.20	0.71	0.00	0.00	2.09
Shadow	0.00	0.01	0.00	0.00	0.00	0.01	0.00	0.09	0.00	0.11
Water	0.00	0.01	0.07	0.01	0.00	0.01	0.00	0.02	0.02	0.14
closings	3.17	9.13	6.86	0.61	0.03	2.38	3.48	0.82	0.03	26.51

(b) Extent Table

	Artificial	Conifer	Heathland	Hedgerow	Pasture	Scrub	Semi-Natural Grassland	Shadow	Water	Total
opening	3.79	5.03	14.54	0.01	0.01	0.80	2.09	0.11	0.14	26.51
increase	0.76	4.64	1.37	0.61	0.03	2.13	2.76	0.73	0.02	13.04
decrease	-1.38	-0.54	-9.04	-0.00	-0.01	-0.55	-1.37	-0.01	-0.13	-13.04
net change	-0.63	4.10	-7.67	0.60	0.02	1.58	1.39	0.71	-0.11	0.00
closing	3.17	9.13	6.86	0.61	0.03	2.38	3.48	0.82	0.03	26.51

(c) Extent Percent Table

	Artificial	Conifer	Heathland	Hedgerow	Pasture	Scrub	Semi-Natural Grassland	Shadow	Water
increase	0.20	0.92	0.09	119.78	5.03	2.65	1.32	6.83	0.12
decrease	-0.36	-0.11	-0.62	-0.65	-1.00	-0.69	-0.66	-0.13	-0.90
net change	-0.16	0.82	-0.53	119.13	4.03	1.96	0.66	6.69	-0.77

Table S5.1. Meenadreen Extent Account from 2016 to 2022.

6

Acoustic detection of a rarely vocalising invasive mammal from sparse data

Anthony Gibbons [1], Andrew Parnell [1,2], Ian Donohue [3], Masako Ogasawara [4], Samuel R. P.-J. Ross [5*]

1 Hamilton Institute, Department of Mathematics and Statistics, Maynooth University, Kildare, Ireland
2 School of Mathematics and Statistics, University College Dublin, Dublin, Ireland
3 Zoology, School of Natural Sciences, Trinity College Dublin, Dublin, Ireland
4 Environmental Science and Informatics Section, Core Facilities, Okinawa Institute of Science and Technology Graduate University, Onna, Okinawa, Japan
5 Integrative Community Ecology Unit, Okinawa Institute of Science and Technology Graduate University, Onna, Okinawa, Japan

Abstract

1. Monitoring and limiting the spread of invasive species on islands requires efficient detection and population estimation methods. However, elusive species can be difficult to monitor using traditional methods, making autonomous approaches such as camera trapping and acoustic monitoring increasingly valuable.

2. On the island of Okinawa, a biodiversity hotspot, the small Indian mongoose (*Urva auropunctata*) threatens many native species since its introduction in 1910. Listed among the world's worst invasive species, effective monitoring of *U. auropunctata* in Okinawa is critical. The Okinawa Environmental Observation Network (OKEON) uses camera traps to detect *U. auropunctata*, but success depends on precise placement. Though OKEON also includes a high-resolution acoustic monitoring programme, *U. auropunctata* rarely vocalises, and no audio classification model currently exists for this species.

3. Using sparse *U. auropunctata* vocalisations collected from camera trap videos, we built a lightweight Convolutional Neural Network distilled from a more complex model for classifying contact calls and alarm calls of *U. auropunctata*. Our distilled model, which contains approximately 10 times fewer parameters than the larger model, performed similarly to the full model at detecting these vocalisations from training data, but was considerably faster. The code and model weights are publicly available.

4. We applied the distilled classifier to ∼486 hrs of audio collected over eight years from southern Okinawa, where we successfully detected *U. auropunctata* a handful of times in each year of recording. Our model did not transfer well to unseen data, perhaps owing to the rarity of *U. auropunctata* calls and consequent small training dataset size, limiting its utility for ecological monitoring.

5. *Practical implication.* We demonstrate that sparse audio data from camera trap videos could not effectively train an acoustic classifier for a rarely vocalising invasive mammal, the small Indian mongoose (*U. auropunctata*), in Okinawa. Our classifer had only very limited utility for detecting *U. auropunctata* from passive acoustic monitoring data. Future work further enhancing classifier performance through mix-up and refined model training has the potential to provide actionable insights into the distribution and spread of *U. auropunctata*, aiding targeted

conservation efforts for Okinawa's threatened biodiversity.

Keywords: Invasive Species, Biodiversity, Deep Learning, Passive Acoustic Monitoring, Bioacoustics, Small Indian Mongoose, Okinawa, Yambaru forest

6.1 Introduction

Invasive species significantly threaten biodiversity globally, disrupting ecosystem functions and services through predation, competition, and habitat alteration. Often introduced through human activity, invasive species can establish rapidly, particularly on islands where native species have small populations and limited prior exposure to novel interactions with invasive species [209]. For example, the brown tree snake (*Boiga irregularis*) has driven several bird species to extinction on Guam in the Mariana islands [291], and invasive rats drive declines in tropical island populations by preying on native bird eggs and chicks [96]. At the same time, islands are often hotspots of endemism, but island ecosystems are among the worst affected by human impacts including land use change and invasive species [129, 209]. Accordingly, endemic species on islands face a mounting threat from invasive species [18, see e.g.]. The global economic cost of invasive species damage exceeds US$100 billion annually [46], underscoring the urgency of early detection and effective management strategies, especially for vulnerable island ecosystems. Recent advances in scalable biodiversity monitoring provide promise for detecting invasive species and mitigating their devastating ecological impacts [73, 104].

Passive acoustic monitoring (PAM) has emerged as a powerful tool for assessing biodiversity, offering a non-invasive, scalable, and cost-effective approach to studying sound-producing species. By deploying autonomous microphones, PAM records species vocalisations to infer species presence, diversity, and ecological dynamics for a range of purposes [222]. For example, PAM has been used to monitor bat activity [157], document bird migration [276], track individual bird species population trends [19, 295], reveal freshwater ecosystem dynamics [86] and, notably, for identifying and monitoring invasive species [227]. However, the sheer volume of audio data produced by PAM studies poses its own challenge, particularly when searching for infrequent events or rarely vocalising species [76]. Recent advances in machine learning, particularly deep learning-based image classification models, have gained popularity in bioacoustics for species classification of animal vocalisations. These classifiers vary in complexity, ranging from simple presence/absence detection [157] to those that identify a few [216], dozens

[78, 315], or even hundreds [75, 123] of distinct species. Such pre-trained models can be applied to different situations but often require labelled training data for fine-tuning to specific use cases.

The small Indian mongoose (*Urva auropunctata*, フイリマングース in Japanese) was introduced to Okinawa Island in 1910 by Professor Shozaburo Watase to control venomous pit vipers (*Protobothrops flavoviridis*, ハブ) and agricultural pests. Watase believed strongly in human intervention and had witnessed a mongoose preying on a cobra in Sri Lanka, informing his decision to introduce *U. auropunctata* as an "ecological experiment" [124]. Without contemporary knowledge of ecological balance or consideration of long-term consequences, his introduction of *U. auropunctata* cast a long shadow over Okinawa. Today, the small Indian mongoose is one of the world's worst invasive species [153], and Okinawa is a hotspot for cross-taxon species invasions [41]. In Okinawa, the mongoose began spreading northward, reaching Yambaru forest by the 1990s [303]. Yambaru forest is a biodiversity hotspot, home to a rich fauna and flora including species endemic to Okinawa island [112, 164, 170]. *U. auropunctata* has had devastating ecological consequences for various small mammals, reptiles, amphibians, and insects [57], but particularly for Okinawa's rich avifauna, exacerbating population declines of the critically endangered Okinawa rail (*Gallirallus okinawae*, ヤンバルクイナ), Okinawa woodpecker (*Dendrocopos noguchii*, ノグチゲラ), and Ryukyu robin (*Larvivora komadori*, アカヒゲ) [302], with indications that these species are still declining (see Figure S6.1).

Efforts to mitigate the mongoose's impact began with a removal project in Yambaru forest (2000–2004) and a formal control plan under Japan's Invasive Alien Species Act of 2004 (`https://www.env.go.jp/en/nature/as.html`). Control measures include live traps and bird-excluding kill traps, with mongoose control fences erected in 2005–2006 to isolate a 280 km^2 area in Yambaru for targeted eradication. The Okinawa Prefectural Government and Japan's Ministry of the Environment currently aim for complete eradication of the mongoose inside the fenced area by 2027; *U. auropunctata* capture rates are declining in Yambaru (`https://kyushu.env.go.jp/okinawa/press_00109.html`) and the government recently declared complete extirpation from Amami-Oshima Island, North of Okinawa (`https://www.env.go.jp/en/press/press_03205.html`). Despite these efforts, the mongoose population in Okinawa is still significant, particularly outside of the fence where intervention has been minimal. As such, scalable methods for detection and monitoring are needed to measure the success of future conservation intervention.

Here, we use an eight-year acoustic monitoring dataset from outside of the mongoose control fence in Okinawa to investigate the efficacy of using PAM to monitor the rarely vocalising invasive mongoose. We built a mongoose vocalisation classifier using sparse audio data extracted from camera trap videos, and applied this classifier to audio collected at a single site of the Okinawa Environmental Observation Network (OKEON) [223, 224]. We intend our acoustic classification model to complement traditional *U. auropunctata* population surveys in Okinawa, providing additional evidence for the presence and distribution of *U. auropunctata* in future.

We first provide background on environmental monitoring programme present in Okinawa. We then outline the specifics of our acoustic datasets, both the training data and the large inference dataset. After this, we describe the audio species classification models employed and the distillation method applied. We then provide a detailed description of our experimental design and results, and follow with in-depth discussion of our results, limitations and paths for future work.

6.2 Materials and Methods

Environmental Monitoring Programme

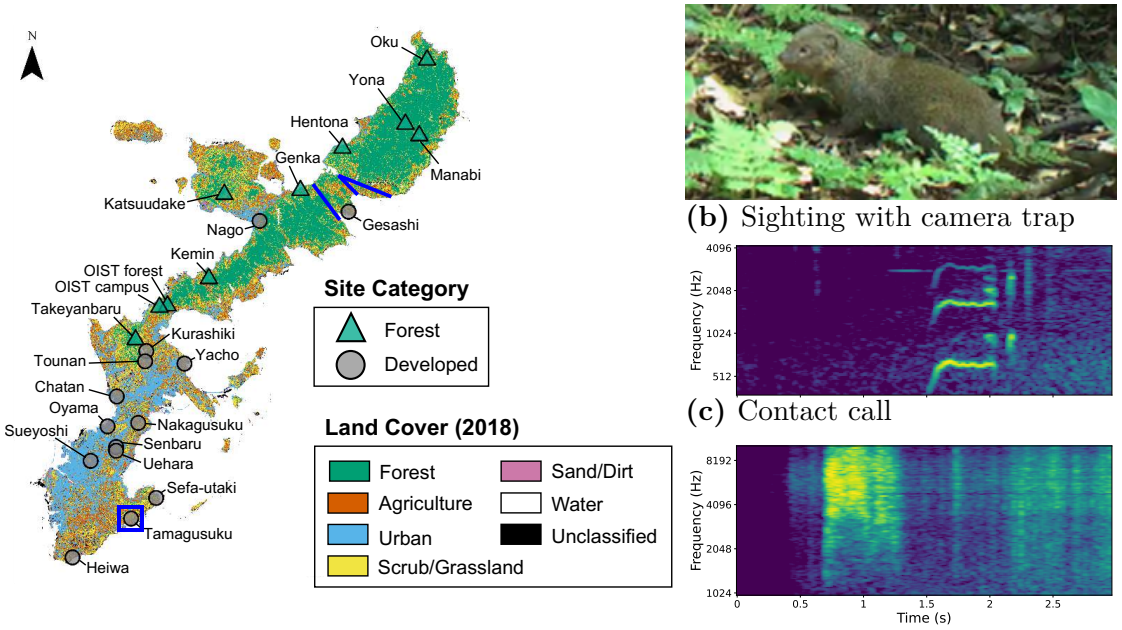
This study uses data from the OKEON (Okinawa Environmental Observation Network) Churamori Project (OKEON 美ら森プロジェクト in Japanese; `https://okeon.unit.oist.jp/`). OKEON aims to monitor terrestrial biodiversity through time at 24 field sites distributed across a range of Okinawa's terrestrial habitats (Figure 6.1a). Each field site deploys modified malaise traps to capture insects, environmental sensors, Song Meter SM4 acoustic recording units (Wildlife Acoustics Inc., Concord, MA, USA), and Hyperfire 2 Covert camera traps (Reconyx Inc., Holmen, WI, USA). Camera traps were placed at a height of ∼1.0 m and are movement-triggered. Acoustic recorders are fixed at ∼1.3 m at each site, facing North, and record with two omnidirectional microphones at default gain (+16 dB) on a duty cycle of 1-min recording, 29-min standby. Recording starts every hour and half-hour, and audio data are saved to SD cards in stereo .WAV format with a sampling rate of 48 kHz, then converted to lossless .flac format. Camera traps and acoustic sensors have been installed since 2017, with previous work in Okinawa demonstrating the efficacy of such remote monitoring techniques for detecting endemic birds, bats, and rats [48, 135, 224, 304], and broadly understanding biodiversity and disturbance [223, 226].

Here, we combine camera trap video (Figure 6.1b) with PAM audio to build a mongoose classifier. Detection of animals using movement-triggered camera traps depends heavily on precise camera placement, whereas PAM sensors cover a much broader detection area with omnidirectional microphones, providing higher relative detection efficiency and probability [56, 40]. However, *U. auropunctata* rarely vocalises—1.28% of 2,500 manually screened *U. auropunctata* video samples included vocalisations (see below)—producing only occasional contact calls, mainly between pups and adults (Figure 6.1c) and alarm calls when fighting conspecifics or capturing prey (Figure 6.1d). Accordingly, initial detection and verification are easier using camera trap data than audio, while audio recordings can provide additional benefits by having wider spatial coverage. To initially screen for mongoose vocalisations for use as training data, we used 4,099 movement-triggered camera trap videos from 2020 and 2022 at 20 OKEON sites. The videos totalled 9.7 hours (5.7 GB), with individual files ranging from 6-10 seconds (mean 8.5 s). Almost all sample files contained *U. auropunctata*.

Data Preparation and Annotation

Approximately 2,500 of 4,099 camera trap videos (Figure 6.2a) were screened manually for *U. auropunctata* vocalisations, yielding 32 examples, which were supplemented with 7 examples from 2025. We also selected an equal number of 'empty' clips from the many confirmed not to contain mongoose audio. Twenty more examples were found via semi-supervised learning (see Supplementary Information for detailed description), for a total of 52 vocalisations from 2022 (Figure 6.2b). Once verified, the audio from the camera trap videos was extracted using MoviePy [319]. Unless otherwise stated, all analysis performed was on the audio data. Another 11 example audio files (mostly alarm calls) were web-scraped from YouTube, Facebook and Xeno-Canto (see Table S6.1). In total, our search and screening process produced 70 examples of mongoose vocalisations. Most mongoose detections were during the hours 07:00-18:00 and vocalisation density was highest during late Summer (∼July-September; Figure 6.2b). *Urva auropunctata* was detected most often at the Tamagusuku OKEON site in the Southeast of Okinawa (Figure 6.2c).

We labelled the 70 mongoose audio clips using NEAL [79], within which we found 134 distinct vocalisations. Due to the small enough of examples for the target species, we applied mix-up augmentation [306]: each mongoose clip in the training set was linearly combined with a randomly selected non-mongoose clip from the inference set, effectively



(a) OKEON sites

(b) Sighting with camera trap

(c) Contact call

(d) Alarm call

Fig 6.1. OKEON, camera trap still and example extracted sounds. (a) Map of OKEON monitoring sites across Okinawa, including land use classification from [224]. Tamagusuku, our focal field site, is highlighted by the blue box, and mongoose control fences are located around the blue lines. (b) Still *U. auropunctata* image from movement-triggered camera trap video. (c,d) Spectrograms of audio from *U. auropunctata* contact (c) and alarm calls (d). Mongoose alarm calls tend to occupy higher frequencies than contact calls; note the differing vertical axes on panels (c) and (d).

doubling the number of mongoose training examples to 244. We took many more examples without mongoose vocalisations from the remaining clips and a sample from the inference dataset to reduce the number of false positives when applying it to the acoustic data. The final classification dataset had 2,005 empty and 244 mongoose examples. The training and test sets were split randomly, stratified by label, with some adjustments: web-scraped files, test files with only one mongoose vocalisation, or examples whose filename also appears in training were moved to the training set. Lastly, additional empty examples from OKEON acoustic data were added to train (1,039) and test (54) to give the model data more similar to what it will encounter at inference time. In order to carry out fine-tuning without overfitting to the test set, the test set was split randomly in two. The final training-validation-test ratio was 0.92:0.04:0.04. The resulting training set contains 3,644 empty and 196 mongoose samples. The validation and test sets each

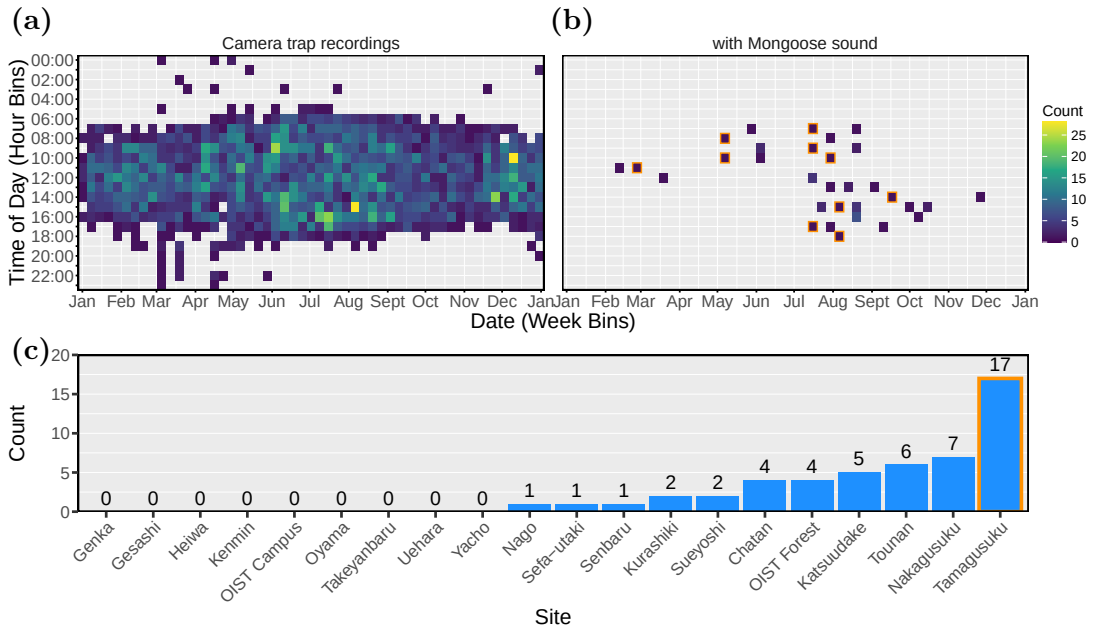


Fig 6.2. Counts of camera trap and acoustic detections by day and hour across all sites. (a,b) The y-axis shows the hour of day (0:00-23:00) and the x-axis is broken into week bins ranging from 1 January to 31 December 2022, inclusive. Brighter colours represent higher validated *U. auropunctata* counts in video (a) and audio (b) from camera traps. Few detections are between 19:00 and 05:00, or in the months November-May. (c) *Urva auropunctata* vocalisation counts from camera trap audio per OKEON site across the sound clips. Tamagusku, outlined in orange in both plots, represents the site with the majority of mongoose vocalisations from camera traps, and thus was selected as our acoustic monitoring case study because of its higher likelihood of containing *Urva auropunctata* vocalisations in scheduled recordings.

include 197 empty and 28 mongoose samples. To classify mongoose vocalisations, we used both VGGish, which looks directly at the audio, and ResNet, which looks at the 256×256 spectrogram images. More details are given in Supplementary Information.

Species Classification Models

Convolutional Neural Networks (CNNs) have become a highly effective tool for image classification over the last decade [136], including for bioacoustic species classification from spectrograms [255]. CNNs provide strong performance, learning task-specific features directly, and are robust to noise and variability in field recordings [152]. Moreover, researchers can leverage transfer learning from pre-trained models to reduce the training

data required to reach adequate performance with their CNNs [312]. We first compared the performance of two CNNs of different complexities (Table 6.1). The larger, teacher model uses transfer learning of two pre-trained CNNs: VGGish and ResNet. VGGish uses raw audio [99] and ResNet uses spectrograms [98], providing complementary but computationally complex information for audio classification. To address the trade-off between generalisation and computational intensity, we also built a smaller, student classifier to be trained using knowledge distillation, where the larger model from the first iteration serves as the teacher [101]. This student CNN takes only spectrograms as input and does not directly use the pre-trained classifiers, but retains much of the teacher's predictive power while reducing the number of parameters and inference time as needed for scaling to large audio datasets (Table 6.1). For further details of teacher and student models, see Supplementary Information.

Table 6.1. Comparison of Models. The teacher model takes in both a 3-channel spectrogram and VGGish embedding as inputs, while the student model uses a single-channel spectrogram. Given its extra preprocessing steps of the ResNet and VGGish runs, the teacher model has over 10 times the parameters, 100 times the multiply-add operations to carry out and is over 10 times larger in storage than the student model. If a student model approaches the teacher's performance, the student model would be advantageous for scaling to large datasets.

Metric	Teacher Model	Student Model	Ratio
Input Shape	[(3, 256, 256), (384,)]	(1, 256, 256)	-
# Parameters	12 060 650	1 090 050	11.06
# Multiply-Adds	2.38×10^9	2.23×10^7	106.73
Total Size (MB)	92.77	8.29	11.19

To evaluate classifier performance, we considered a range of metrics: accuracy (proportion of correct predictions), precision (true positives divided by all predicted positives), recall (true positives divided by all actual positives), F1 score (harmonic mean of precision and recall), and the Area Under the Receiver Operating Characteristic Curve (AUC) score, which plots the true positive rate against the false positive rate at different detection thresholds. Collectively, these metrics allow us a broad understanding of model performance. Both models used the same training hyperparameters: Adam optimiser, learning rate of 0.001, a batch size of 16, and 50 epochs.

In machine learning, a loss function quantifies the discrepancy between a model's predictions and the true outcomes, guiding the optimisation process during training. For species classification, categorical cross entropy measures the difference between the

model's probabilistic output and the true class labels, while Kullback-Leibler (KL) divergence [137] compares two probability distributions (e.g. between two models) and a cosine embedding loss [205] measures the similarity between two embeddings (outputs from earlier layers of the model). The teacher model in this study used cross entropy loss while the student model, employing knowledge distillation, used cross entropy as the target loss and the sum of KL divergence (with temperature $T = 3$) and cosine embedding loss (with margin $m = 0$) as the distillation loss. We used a weighting $\alpha = 0.7$ on distillation loss and $1 - \alpha$ on target loss, typical values for the method.[1] Total loss is the sum of the two. For further details of the loss functions used, see Supplementary Information. A copy of the student model, which we call the vanilla student model, was trained using just categorical cross entropy loss and not by distilling the teacher model. We did this in order to evaluate performance when training from scratch, without taking advantage of learnings from the pre-trained models.

The hardware environment for our model comparison was a local machine with 16 GB of RAM and an Intel(R) Core(TM) i7-1165G7 CPU @ 2.80 GHz, with 4 cores and 8 threads. The operating system was Windows 11 Pro 24H2. The programming environment included Visual Studio Code (version 1.99.3) [39] for prototyping, training, and post-processing, using Python (version 3.9.21) [204] and PyTorch (version 2.6.0) [206] within this environment.

Acoustic Monitoring Case Study

We apply the chosen model to a long-term audio dataset collected as part of the Okinawa Environmental Observation Network's passive acoustic monitoring programme [224]. We chose one OKEON field site, Tamagusuku, as our case study for out-of-sample model inference, owing to the relative number of mongoose vocalisations detected from camera trap audio at this site (Figure 6.2). A single site was selected because the data size, over 150 GB per site across the 24 sites on the island, was too large to process for all sites in the OKEON dataset. Tamagsuku forest is a small disturbed forest surrounded by urban land use (Figure 6.1a). The forest patch has alkaline soil, supporting typical Okinawan flora such as *Ficus microcarpa* (ガジュマル), *F. superba* (アコウ), and *F. virgata* (ハマイヌビワ) which germinate and grow on rocks and large trees. The tree canopy is primarily composed of *Cinnamomum yabunikkei* (ヤブニッケイ) and *Celtis boninensis Koidz* (クワノハエノキ), with other climbing plants such *Flagellaria indica* (トウツルモ

[1] `https://docs.pytorch.org/tutorials/beginner/knowledge_distillation_tutorial.html`

ドキ) and *Ipomoea indica* (ノアサガオ), and palm trees including *Arenga engleri* (クロツグ). Despite the small forest patch size and anthropogenic noise from the surrounding city, Tamagusuku also has a rich soniferous fauna, including resident songbirds such as the Japanese tit (*Parus cinereus*, シジュウカラ), and migratory species like the Japanese paradise flycatcher (*Terpsiphone atrocaudata*, サンコウチョウ). The site is also home to endemic frogs (*e.g.*, *Rhacophorus viridis*, オキナワアオガエル), newts (*e.g.*, *Cynops ensicauda popei*, シュリケンイモリ), and lizards (*e.g.*, *Goniurosaurus kuroiwae*, クロイワトカゲモドキ). Several species of small bats also roost in nearby limestone caves, but do not produce sound in the focal frequency range of our study.

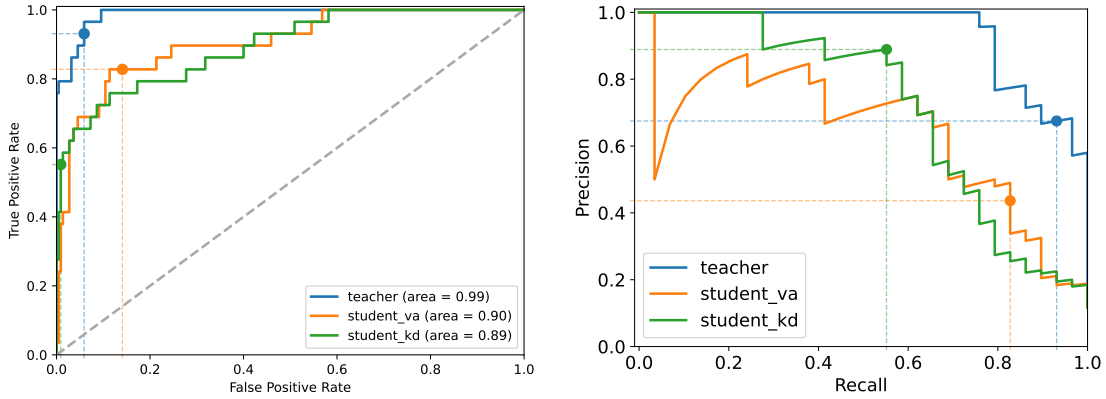
We used audio recordings corresponding to peak mongoose vocalisation periods from preliminary data (Figure 6.2); we targeted 07:00-19:00 each day between 15 May and 14 October from 2017 to 2024. Taking each 1-minute audio file per half hour (25 files/day) each Summer (153 days/year) for eight years resulted in 29,159 min (∼486 hrs) of continuous audio totalling 155 GB compressed to .flac format. We ran the distilled model over this data, and manually verified *Urva auropunctata* detections labelled with > 0.8 confidence.

6.3 Results

Evaluating model performance

The large teacher model was trained on the above dataset and achieved 94.6% accuracy on the test data, using the default threshold of 0.5, with mongoose-specific metrics of 0.61 precision, 0.92 recall, 0.73 F1-score and an area under the ROC curve (AUC) of 0.96. The vanilla student model was trained similarly, using just the spectrograms as inputs and species labels as outputs, and achieved 91.9% accuracy with 0.70, 0.61, 0.65 and 0.92 for precision, recall, F1-score and AUC, respectively. The distilled student model, trained both on the target labels and outputs from the teacher model achieved best validation loss, achieved 92.6% accuracy, with 0.63 precision, 0.78 recall, 0.80 F1-score and 0.99 AUC. Figure 6.3 shows the comparison in more detail.

The distilled model shows significantly better metrics than the vanilla student model, and even surpasses the teacher model at most thresholds. A possible reason for this is that the distillation performs as a form of regularisation, where forcing the student to learn from soft labels could reduce the amount of overfitting the teacher model is experience from its component predtained on images. Coupled with its computational



(a) Receiver Operating Characteristic (ROC) curves. **(b)** Precision-Recall (PR) curves.

Fig 6.3. Performance comparison of teacher and student models. (a) Receiver Operating Characteristic (ROC) curve for each model, showing how well they distinguish between correct and incorrect detections. (b) Precision-Recall (PR) curves, illustrating the trade-off between the two metrics. Curves were created by adjusting the detection threshold from 0 to 1 (default = 0.5, shown at points and coloured dashed lines) and analysing output probabilities for *Urva auropunctata* in the test set. For (a), we calculated true positive and false positive rates, and for (b), we measured precision and recall at each threshold. The grey dashed line on the diagonal in (a) is a random guess, where false positive and false negative rates are expected to be the same at any threshold. **student_va** is the smaller model architecture trained solely on the labelled data, while **student_kd** is the same architecture trained via knowledge distillation using outputs from the **teacher** model.

efficiency 10-fold higher than the teacher model, we decided to apply the distilled model to our full dataset. We first trained the distilled model again on all the data (training, validation, and test) to improve its generalisation. Again, the distilled student model was trained via knowledge distillation to transfer some of the learnings from the large teacher model to a more computationally efficient architecture.

Application to monitoring data

We applied our trained model to the large acoustic dataset from Tamagusuku, spanning 2017 to 2024, to classify mongoose vocalisations. We ran the model on each year of the dataset independently, processing audio recordings to identify mongoose and non-mongoose (empty) vocalisations. Model outputs included considerably more

non-mongoose classifications than predicted mongoose vocalisations, as illustrated by the confidence distribution for the 2022 data in Figure 6.4.

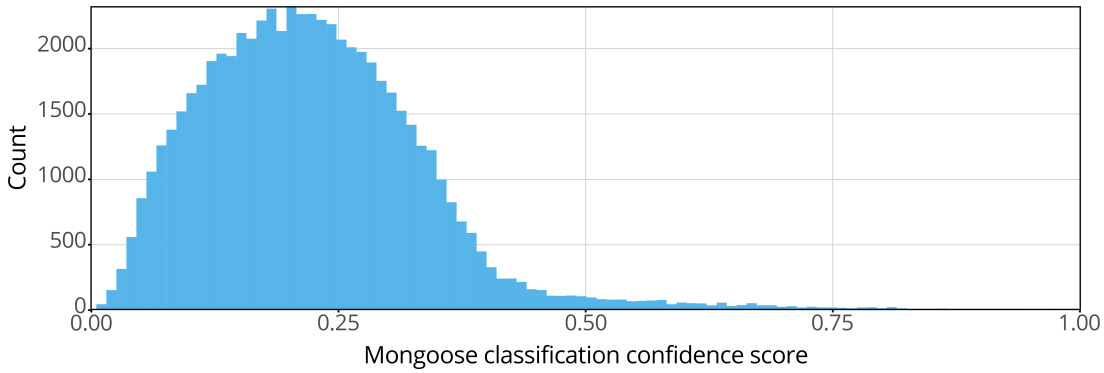


Fig 6.4. Histogram of mongoose classification confidence from 2022 data. The model predictions (confidence scores) are noticeably skewed left, with mostly low-confidence predictions (0.1-0.4 confidence), below the default 0.5 cutoff. The distribution has a sharp decline after 0.4, and a long tail towards 1.0. This is consistent with mongoose being a rarely vocalising species not often present in audio recordings. Of the 62,039 predictions from 2022, only 54 had confidence over 0.8.

We manually verified model predictions above a 0.8 confidence threshold and identified a total of 5 mongoose vocalisations across the entire dataset, with some variation by year (see Table 6.2). Despite the limited number of verified samples among the predictions (0.66% precision across all years), these serve as a supplemental examples for expanding the training set for semi-supervised learning. See Figure S6.4 for a detailed visualization of the model's confident predictions across time.

To analyse the model's behaviour on the spectrograms in the inference data, we examined saliency maps—or class activation maps—to better understand which time-frequency regions and patterns most strongly influence model predictions [309]. These maps are created by computing the gradients of the model's output score with respect to the input spectrogram via backpropagation, thereby identifying the pixels that most strongly influence the prediction. They highlight regions of the spectrogram image that contribute most to classification decision-making, thereby allowing us to interpret whether the model is using meaningful acoustic features or is overfitting to recorder-specific artefacts, background noise, or other spurious signals. By comparing different example mongoose classifications, we can observe the consistency of the model's focus across similar inputs to detect potential sources of misclassification. We show side-by-

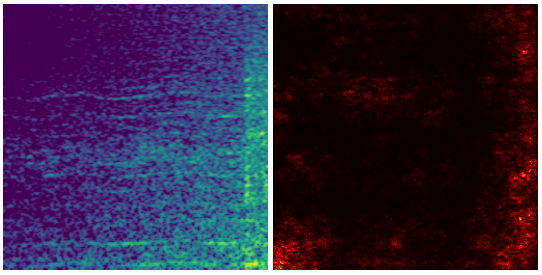
Table 6.2. High-confidence mongoose detections by year. Number of model predictions with >0.8 confidence score per year and verified true positives versus false positives.

Year	Total detections	False Positives	True Positives
2017	114	114	0
2018	170	168	2
2019	74	74	0
2020	114	114	0
2021	129	129	0
2022	54	51	3
2023	77	77	0
2024	32	32	0

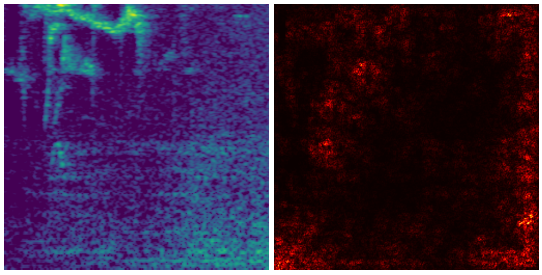
side sample spectrograms and saliency maps for each of a true positive, a false positive, and a true negative example in the inference dataset in Figure 6.5.

6.4 Discussion

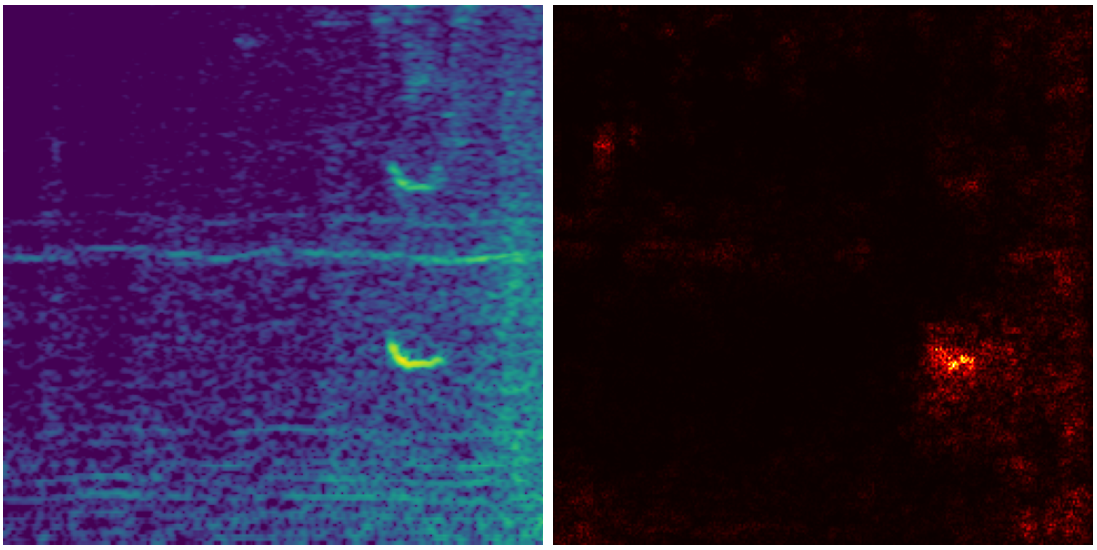
The small Indian mongoose (*Urva auropunctata*) is one of the world's worst invasive species [153] and has been invasive on Okinawa's main island since its deliberate introduction in 1910 [124]. Detecting, monitoring, and ultimately controlling the spread of this species is therefore of considerable importance for Okinawa's endemic wildlife threatened by the mongoose's invasion [57, 302]. We aimed to use audio from camera traps and acoustic recording units in Okinawa as a novel method for detecting and monitoring *U. auropunctata*. This species rarely vocalises, making audio detection a challenge. Indeed, detecting *U. auropunctata* proved challenging, perhaps due to the sparsity of mongoose vocalisations in training and real-world data. To address this, we created a dataset comprising both mongoose vocalisations and negative counter-examples to identify *U. auropunctata* in passive acoustic monitoring data. Using knowledge distillation, we developed a computationally efficient convolutional neural network (CNN) that achieved 92.6% accuracy on the test set. The trained model was applied to a real-world case study—an acoustic dataset spanning eight years—and successfully detected 5 *U. auropunctata* vocalisations. Our approach provides a starting point that has the potential to become a scalable framework for supporting mongoose monitoring and hence conservation efforts to protect Okinawa's unique ecosystems.



(a) Verified empty.



(b) False-positive mongoose.



(c) True positive mongoose.

Fig 6.5. Saliency maps of mongoose classification model. Spectrograms (*viridis* colour palette) and corresponding saliency maps (*hot* palette) of predicted (non)mongoose detections. Regions of high saliency (yellow-white) indicate areas that strongly influenced the model's classification. (a) shows a spectrogram with low confidence, predicted (and verified) to not contain mongoose. (b) shows a high confidence prediction of mongoose, but human verification identified it as a brown-eared bulbul (*Hypsipetes amaurotis*); similar frequency regions in the centre left of the activation map may explain the confusion. (c) shows a verified mongoose prediction with high confidence, with a distinct activated region visible in the saliency map corresponding to the verified mongoose vocalisation.

Despite the promise of our approach and model performance on test data, we identified several limitations which will need to be overcome before the model can be applied in the field. One key issue is the presence of multiple overlapping vocalisations from other (non-target) animals, which can obscure *U. auropunctata* calls and mislead the model. In particular, the presence of cicada choruses during the Summer months poses a challenge

for detecting and interpreting acoustic signals due to increased background noise [226], and many *U. auropunctata* false positives were detected when both cicada choruses and a non-target species signal were present.Additionally, the model's performance may be constrained by the limited availability of high-quality, labelled training data. To address these challenges, alternative approaches could be explored, such as building separate classifiers for *U. auropunctata* alarm and contact calls, as well as countertraining with distinct categories for tricky signals such as the large-billed crow (*Corvus macrorhynchos*, ハシブトガラス), brown-eared bulbul (*Hypsipetes amaurotis*, シロガシラ) and Japanese bush warbler (*Horornis diphone*, ウグイス). More advanced techniques such as deep metric learning may also be useful to further distinguish different signal classes [183], while simulating soundscapes could expand the limited *U. auropunctata* training set using synthetic samples [284, 78].

To enhance the model's performance further, validated samples (both true positive *U. auropunctata* samples and false positives as counter-examples) identified in the large Tamagusuku dataset can be used along with the already labelled dataset for this paper to train a more robust model. This has the dual benefit of both providing both more data and data that is specific to the acoustic recorders used, and so the model should be better able to detect *U. auropunctata* in acoustic recordings given this extra training. By repeating this with each site in the OKEON dataset, island-wide insights could be gained into mongoose activity. However, transferability to other sites may not be a given because of the varied soundscape present in Okinawa.

6.5 Conclusion

Our study demonstrated a field audio-based computational approach for monitoring the small Indian mongoose (*Urva auropunctata*) in Okinawa. By leveraging sparse vocalisations extracted from camera trap videos collected across the island as part of an ongoing environmental monitoring initiative, we trained a lightweight CNN model to detect *U. auropunctata* sounds. By distilling from a larger model to a smaller model, we achieved comparable (even superior) performance while significantly reducing computational overhead. When applied to ∼486 hours of audio data collected over eight years in Southern Okinawa, the trained classifier could identify *U. auropunctata* vocalisations, but with high false-positive error rates. Our approach shows some potential in combining sparse audio data with deep learning models to monitor invasive mammals, potentially offering a scalable tool to support conservation efforts aimed at protecting Okinawa's threatened

ecosystems. Future applications of this classifier to additional sites could further refine our understanding of the spatial distribution of *U. auropunctata* and inform targeted management interventions.

Author Contributions

A. Gibbons: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing - Original Draft, Data Curation. **I. Donohue:** Writing - Review & Editing, Supervision. **A. Parnell:** Writing - Review & Editing, Supervision, Funding acquisition. **M. Ogasawara:** Resources, Data Curation, Writing - Review & Editing, Funding acquisition. **S.R.P.-J. Ross:** Conceptualization, Methodology, Validation, Investigation, Writing - Original Draft, Visualization, Project administration.

Acknowledgments

We thank Kenneth Dudley, Mayuko Suwabe, Masashi Yoshimura, Takuma Yoshida, Toshihiro Kinjo, Shinji Iriyama, Ten Fujikawa, Seiichiro Nakagawa, and Shoko Suzuki from the Environmental Science and Informatics Section of Core Facilities at OIST. We also thank Takumi Akasaka, Sara Keen, and Dave Armitage for early discussion, and Nick Friedman for data included in Figure S6.1. This work was supported by JST Grant Number JPMJPF2205 and the Nature+Energy Project, funded by Research Ireland (12/RC/2302 P2), industry partners and MaREI, the Research Ireland Research Centre for Energy, Climate and Marine Research and Innovation. SRPJR was supported by subsidy funding to OIST.

Conflict of Interest Statement

The authors have no conflicts of interest to declare.

Data and Code Availability

Code and documentation for preprocessing the data, training the species classifiers, inference on large audio dataset, along with the model weights, and post-processing scripts for generating the plots used in this paper are available at `https://github.com/gibbona1/OIST_mongoose`. All OKEON data, including the raw audio and labelled camera trap data used in this study, are archived with the Okinawa Institute of Science and Technol-

ogy Graduate University's high-performance computing center and are available from the Environmental Science and Informatics Section of Core Facilities at OIST upon request.

6.A Supplementary Information

Figures

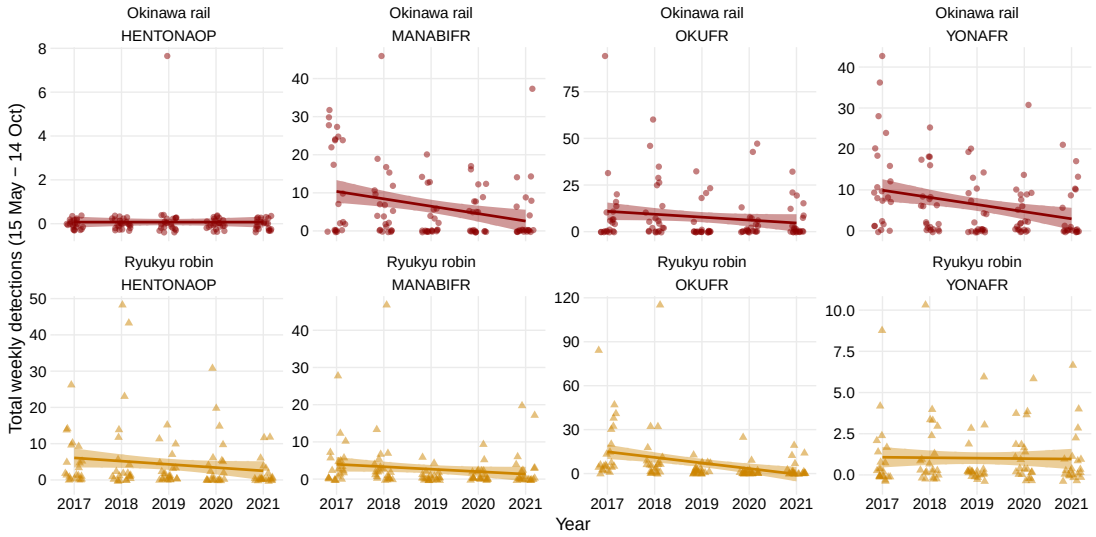


Fig S6.1. Detection trends for Yambaru bird species.

(a) Tamagusuku canopy. **(b)** Tamagusuku forest floor.

Fig S6.2. Tamagusuku site. Photo credit: Amy Morrell

(a) Camera Trap. **(b)** Acoustic Recorder.

Fig S6.3. OKEON recording equipment.

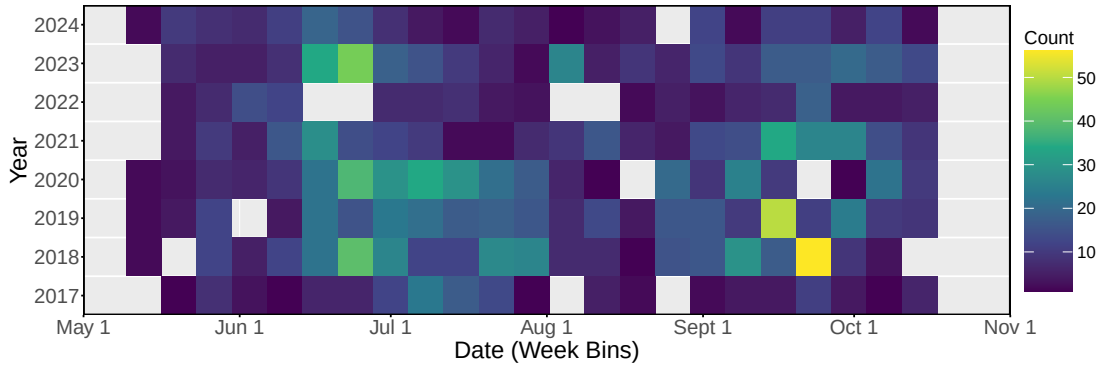
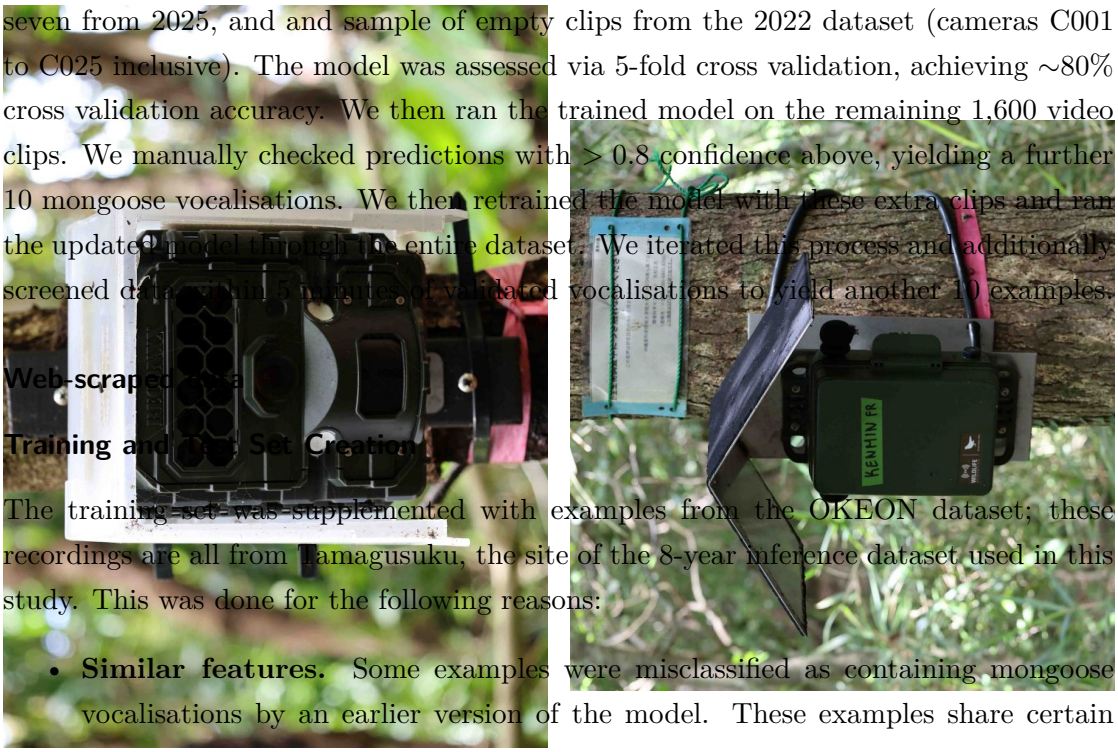


Fig S6.4. Counts of confident model predictions over the dataset length.

Expanding Dataset via Semi-supervised Learning

We built a feature extractor for semi-supervised learning to find more mongoose vocalisations in the camera trap data. This involved taking the outputs from VGGish and ResNet and training a lightGBM classifier [307] to distinguish mongoose from non-mongoose clips. The dataset used consisted of the 32 mongoose clips from 2022 and seven from 2025, and and sample of empty clips from the 2022 dataset (cameras C001 to C025 inclusive). The model was assessed via 5-fold cross validation, achieving ∼80% cross validation accuracy. We then ran the trained model on the remaining 1,600 video clips. We manually checked predictions with > 0.8 confidence above, yielding a further 10 mongoose vocalisations. We then retrained the model with these extra clips and ran the updated model through the entire dataset. We iterated this process and additionally screened data to obtain 5 minutes of validated vocalisations to yield another 10 examples.



Web-scraped data

Training and Test Set Creation

The training set was supplemented with examples from the OKEON dataset; these recordings are all from Tamagusuku, the site of the 8-year inference dataset used in this study. This was done for the following reasons:

- **Similar features.** Some examples were misclassified as containing mongoose vocalisations by an earlier version of the model. These examples share certain

Filename	URL	Length (s)
short1.wav	https://youtu.be/AE9BEdvXfF8	18
short2.wav	https://youtu.be/rtN1NoXM3Cc	7
short3.wav	https://youtu.be/-mOD-sDuWS4	10
video1.wav	https://youtu.be/V2QP1bQSSc0	9
video2.wav	https://youtu.be/jdvEn9VxyQg	15
video3.wav	https://youtu.be/ch5-exHYlG4	34
video4.wav	https://youtu.be/2smtcRvugnU	82
video5.wav	https://youtu.be/ISWCrjGZVjw	67
video6.wav	https://facebook.com/share/v/1BN4bVzecR	15
XC578808.wav	https://xeno-canto.org/578808	153
XC156050.wav	https://xeno-canto.org/156050	109
XC156097.wav	https://xeno-canto.org/156097	71
XC156111.wav	https://xeno-canto.org/156111	23
XC191039.wav	https://xeno-canto.org/191039	66
XC202788.wav	https://xeno-canto.org/202788	6
XC202790.wav	https://xeno-canto.org/202790	16
XC236034.wav	https://xeno-canto.org/236034	20
XC286152.wav	https://xeno-canto.org/286152	17
XC474444.wav	https://xeno-canto.org/474444	59
XC578817.wav	https://xeno-canto.org/578817	3
XC884889.wav	https://xeno-canto.org/884889	56
XC984255.wav	https://xeno-canto.org/984255	42
XC984256.wav	https://xeno-canto.org/984256	43

Table S6.1. Web scraped audio details. The web scraped audio was extracted from Xeno-Canto recordings and YouTube and Facebook videos. The two Xeno-Canto files beginning '98' are of the Crab-eating Mongoose recorded in Taiwan, but the vocalisations were deemed to be similar enough to the species of interest, *Urva auropunctata*. The remaining Xeno-Canto files were bird species used as counter-examples for training. The majority of the mongoose recordings are alarm calls, while short1 and video6 contain contact calls. In the final training set, these web-scraped examples constitute nearly half of the mongoose examples (these were excluded from the validation and test sets), underscoring their utility in enhancing the dataset's diversity and improving model robustness.

acoustic features, e.g. bird species vocalising in similar frequency ranges to mongoose, hence the confusion, and so are valuable counter-examples for training the chosen model.

- **Reducing domain shift.** PAM dataset examples should more closely represent the large dataset we run inference on than camera trap audio. In particular, the camera trap audio has more limited spatial coverage than those collected by the the acoustic recorders.
- **Aligning distribution.** The model, while trained on a roughly 16:1 empty:mongoose ratio, could differ noticeably from the large inference data. The inference data may have much less than 1% that contain mongoose vocalisations. Adding high-quality negative samples should align more features with empty clips, reducing false positives.

Once the training data was constructed, examples were set to be 3 seconds from the annotation start, which biases towards the mongoose vocalisation being at the beginning of the examples. Clips were randomly 'rolled' so the vocalisation(s) present occur at different points in the clip. Average waveforms were analysed for stereo audio files. The signal was converted to a mel-spectrogram with `n_fft`=4096, `hop_length`=512, `n_mels`=256, `power`=4.0 and cropped to 400 Hz–9 kHz. The output spectrograms were cropped to 256×256 single-channel (grayscale) and repeated to three channels for input to ResNet. To reduce background noise, each spectrogram had the mean of each frequency (y-axis) bin subtracted across the entire row. Finally, the spectrograms were normalised to be in the range [0,1] and then normalised again with the means and standard deviations from ImageNet dataset, to conform with the input format the pre-trained ResNet expects. VGGish uses the 3-second signal array and its sample rate as inputs.

Classification Model Construction

The initial model for mongoose classification made use of two pre-trained CNNs. The first is VGGish [99], a VGG-like architecture trained on AudioSet [74]. It takes raw 3-second mono audio and its sample rate as input and generates feature embeddings for each second, which are then flattened to a single vector of length $128 \times 3 = 384$. Concurrently, a ResNet model [98], pre-trained on ImageNet [43], takes in a 256×256×3 image and outputs an embedding of length 1000. All but the final layer of the ResNet model were frozen; the model can update the unfrozen layer during training. The embeddings from both model outputs are concatenated to a vector of length 1384 and passed through 3 fully connected (FC) layers, with dropout with probability 0.5 applied after each to reduce overfitting. Using two feature extractors trained on different datasets and modalities (images for ResNet, audio for VGGish) provides complementary insights from both

models. During training, the VGGish embeddings and mel spectrograms can be saved after their initial generation in the first epoch and loaded in future epochs, reducing training time. The ResNet embedding changes since the final layer is trainable, but an earlier layer could technically have been saved.

Our target dataset consists of over ∼486 hours of data, requiring ∼583,000 spectrograms (and waveform arrays for VGGish) to be computed and run through the model. A smaller CNN, while faster than the above, may not be able to generalise given the relatively small size of the mongoose audio dataset. To address these computational constraints, while also taking advantage of the learnings of the two pre-trained models, the second iteration of the classifier is a smaller CNN which takes in only spectrograms and does not make direct use of the pre-trained classifiers. Using knowledge distillation, where the larger model from the first iteration serves as the teacher [101] for the smaller student model, the training objective combines cross-entropy loss, as in the teacher model, with Kullback-Leibler (KL) divergence-based distillation loss to align the student model's softmax outputs with the teacher's. Since the output has only two classes, this may not be enough to transfer much of the learnings from teacher to student. The distilled student model was trained additionally with a cosine embedding loss aligning outputs from an earlier layer (with dimension 256) in both models. This enables the smaller model to retain much of the teacher's predictive power while reducing parameters and inference time. A comparison of the teacher and smaller student model used in this paper is given in Table 6.1.

7 Conclusion

This thesis has focused on the development of new ML techniques and analysis tools for monitoring biodiversity, namely audio annotation software, generative models, bird and invasive mammal species classifiers via sound, and an application to quantify land cover change. In this concluding chapter, we summarise the contributions made by each research chapter in this thesis, consider the limitations of our research, and identify promising areas for future work in the field.

7.1 Chapter Summaries

Chapter 3 of thesis introduces NEAL, an open-source Shiny R application that improves the efficiency and consistency of audio data annotation, with a focus on bird vocalisations but extendable to other species. By enabling precise annotations - capturing time, frequency and call type - NEAL supports the generation of multiple label sets simultaneously, serving as robust human-verified targets for ML applications. Its no-code, interactive interface empowers domain experts without programming knowledge, while its modular design and open-source nature foster community-driven innovation and fea-

ture expansion, such as integrating environmental data or improving visualisation tools and computational efficiency.

Chapter 4 contributes to the thesis by demonstrating the utility of generative AI, namely Stable Diffusion, in enhancing bird species audio classification through high-quality synthetic spectrogram augmentation. The synthetic samples generated by the Stable Diffusion model were of better quality when compared with those generated by an ACGAN. An ensemble model trained on a combination of real and synthetic data performed better on both validation and human-labelled test data than those trained with real data alone, overcoming data collection constraints common in bioacoustics.

Chapter 5 introduces ExActR, an open-source Shiny R application that enhances ecosystem extent accounting by enabling environmental project managers with limited GIS or programming skills to quantify land cover changes using geospatial vector datasets. By offering a no-code, reproducible, and flexible workflow, ExActR was built in alignent with the SEEA-EA framework and addresses the needs of practitioners working at local scales. ExActR contributes to the thesis's broader aim of developing user-friendly bioacoustic and ecological tools, with potential to expand into a comprehensive Ecosystem Accounting package.

In Chapter 6, the final research chapter, we investigate building an acoustic species classifier for the small Indian mongoose (*Urva auropunctata*), an invasive mammal which threatens many endemic species on Okinawa island, Japan. This mammal rarely vocalises, complicating the development of an audio classification model. We demonstrate that sparse audio data from camera trap videos can effectively train an acoustic classifier and distil to a smaller model to speed up inference on larger datasets such as long-term acoustic monitoring at OKEON. We show the utility of our classifier for detecting *U. auropunctata* and propose future work to apply this classifier to additional locations across the island, aiding targeted conservation efforts for Okinawa's threatened biodiversity.

7.2 Limitations and Future Work

This thesis has advanced ecological research through the development of innovative tools and methodologies for bioacoustics and ecosystem accounting, yet several limitations and opportunities for future work remain. These span the datasets, methodologies, and applications explored, offering pathways to enhance the robustness, scalability, and impact of the proposed solutions.

Chapters 3 and 4 use acoustic data collected using SongMeter Minibat recorders. A

smaller dataset was collected by simultaneously deploying a Minibat and Audiomoth at the same location for approximately four weeks, but was not used for any of the work in this thesis. Recent research [115] highlights differences in recorder performance which may drift over time, suggesting potential biases when using multiple devices across sites or over the course of a long-term study. A promising direction is to employ a Variational Autoencoder (VAE), like that used in Chapter 4, to transfer features between recordings from different devices. This could improve the quality of data from cheaper devices or reduce bias in multi-recorder setups by standardising acoustic features. This would minimise systematic differences in model predictions that arise solely from hardware variation and give a more objective analysis of bioacoustic data. Future work should investigate the existing dataset of simultaneous recordings and, if promising, expand to more sites and time horizons.

The bioacoustic studies in this thesis relied on recorders placed at considerable distances from one another, which provided coarse insights into species distributions. However, subtle spatial variations in species presence within smaller areas (e.g., < 1 km) remain under-explored. Investigating fine-scale differences at a single site could reveal microhabitat preferences or behavioural patterns, enhancing conservation strategies. Future research should deploy denser recorder networks within sites to capture high-resolution spatial data, coupled with advanced spatial analysis techniques, such as geo statistical modelling, to map species distributions and inform site-specific management plans.

During the course of the work carried out on this thesis, in particular the acoustic recorders deployed at wind farms, a substantial ultrasonic dataset of bat vocalisations was also collected. Furthermore, a proof of concept bat species classifier (six bat species, birds, and noise) was developed. However, the classifier's application was limited to general species identification rather than addressing targeted ecological questions. Future work could train more sophisticated models using this dataset to investigate specific research topics, such as bat feeding behaviour, investigating moth presence, or spatial differences in bat activity across habitats.

While the ExActR app, discussed in Chapter 5, successfully generates ecosystem extent accounts, extending the Shiny framework to support condition accounts—the second pillar of the SEEA-EA framework—presents significant challenges. Condition accounts requires significantly more flexibility to accommodate diverse ecological indicators of interest, which are more complex to standardise than extent calculations. The need for user-defined parameters and data inputs complicates end-to-end automation in

a no-code interface. Future development should focus on modular Shiny components that allow users to select and customise condition indicators (such as soil carbon, tree cover and water quality) while maintaining accessibility. Collaboration with SEEA-EA practitioners could guide the prioritisation of key metrics, enabling a comprehensive ecosystem accounting package.

The generative AI study in Chapter 4 demonstrated that the Stable Diffusion model produced higher-quality synthetic spectrograms than the ACGAN, improving classifier performance, but at a higher computational cost. This trade-off limits scalability for resource-constrained settings. A potential middle ground could involve the single pass nature of GANs with the quality enhancements of the VQ-VAE from the Stable Diffusion model to reduce computational demands.

The mongoose classification model in Chapter 6 showed promise but was limited by the training dataset's size and diversity of samples. Expanding the model to encompass the full OKEON dataset, which may include additional verified examples across the time of the project, would significantly increase training data volume and improve model robustness. This enhancement could support island-wide mongoose tracking efforts, aiding invasive species management. Future work should prioritise data curation to incorporate verified samples, refine the model with techniques mentioned in the chapter (specifically mix-up and knowledge distillation), and collaborate with local conservationists to align outputs with tracking objectives.

Bibliography

[1] Martín Abadi et al. "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems". In: *arXiv preprint arXiv:1603.04467* (2015). Software available from tensorflow.org. URL: https://arxiv.org/abs/1603.04467.

[2] Garrick Aden-Buie. *shinyThings: Reusable Shiny Modules and Other Shiny Things*. R package version 0.4.0. 2022. URL: https://github.com/gadenbuie/shinyThings.

[3] Erhan Akbal, Sengul Dogan, and Turker Tuncer. "An automated multi-species bioacoustics sound classification method based on a nonlinear pattern: Twine-pat". In: *Ecological Informatics* 68 (2022), p. 101529. ISSN: 1574-9541. DOI: 10.1016/j.ecoinf.2021.101529.

[4] J. Allen. "Short term spectral analysis, synthesis, and modification by discrete Fourier transform". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 25.3 (1977), pp. 235–238. DOI: 10.1109/TASSP.1977.1162950.

[5] Soghra Andaryani et al. "The utility of a hybrid GEOMOD-Markov Chain model of land-use change in the context of highly water-demanding agriculture in a semi-arid region". In: *Ecological Informatics* 64 (2021), p. 101332. ISSN: 1574-9541. DOI: https://doi.org/10.1016/j.ecoinf.2021.101332.

[6] Vincent Arnaud et al. "Improving the workflow to crack Small, Unbalanced, Noisy, but Genuine (SUNG) datasets in bioacoustics: the case of bonobo calls". In: *bioRxiv* (2022). DOI: 10.1101/2022.06.26.497684.

[7] Dean Attali. *shinyjs: Easily Improve the User Experience of Your Shiny Apps in Seconds*. R package version 2.1.0. 2021. URL: https://CRAN.R-project.org/package=shinyjs.

[8] Audacity Team. *Audacity®*. https://www.audacityteam.org/. Accessed: 2022-06-15. 2022.

[9] Kenneth J. Bagstad et al. "Towards ecosystem accounts for Rwanda: Tracking 25 years of change in flows and potential supply of ecosystem services". In: *People and Nature* 2.1 (2020), pp. 163–188. DOI: https://doi.org/10.1002/pan3.10062.

[10] Behnaz Bahmei, Elina Birmingham, and Siamak Arzanpour. "CNN-RNN and Data Augmentation Using Deep Convolutional Generative Adversarial Network for Environmental Sound Classification". In: *IEEE Signal Processing Letters* 29 (2022), pp. 682–686. DOI: 10.1109/LSP.2022.3150258.

[11] Eric Bailey. *shinyBS: Twitter Bootstrap Components for Shiny*. R package version 0.61.1. 2022. URL: https://CRAN.R-project.org/package=shinyBS.

[12] Ed Baker and Sarah Vincent. "A deafening silence: a lack of data and reproducibility in published bioacoustics research?" In: *Biodiversity Data Journal* 7 (2019), e36783. ISSN: 1314-2836. DOI: 10.3897/BDJ.7.e36783.

[13] Mustafa Baniodeh et al. *Scaling Laws of Motion Forecasting and Planning – A Technical Report*. 2025. DOI: 10.48550/ARXIV.2506.08228.

[14] Baringa Partners LLP. *Wind for a Euro: Cost-benefit analysis of wind energy in Ireland 2000-2020*. https://www.baringa.com/globalassets/insights/low-carbon-futures/our-market-and-policy-studies-in-ireland/wind-for-a-euro-cost-benefit-analysis-of-wind-energy-in-ireland-2000-2020/baringa_wind_for_a_euro_report_jan_2019.pdf. Accessed: 2025-07-25. Jan. 2019.

[15] Shane Barratt and Rishi Sharma. *A Note on the Inception Score*. 2018. arXiv: 1801.01973 [stat.ML]. URL: https://arxiv.org/abs/1801.01973.

[16] Yves Bas, Didier Bas, and Jean-François Julien. "Tadarida: A toolbox for animal detection on acoustic recordings". In: *Journal of open research software* 5.1 (2017), pp. 6–6. DOI: 10.5334/jors.154.

[17] Mark F. Baumgartner et al. "Persistent near real-time passive acoustic monitoring for baleen whales from a moored buoy: System description and evaluation". In: *Methods in Ecology and Evolution* 10.9 (2019), pp. 1476–1489. DOI: https://doi.org/10.1111/2041-210X.13244.

[18] Paulo A. V. Borges et al. "Invasibility and species richness of island endemic arthropods: a general model of endemic vs. exotic species". In: *Journal of Biogeography* 33.1 (2006), pp. 169–187. DOI: 10.1111/j.1365-2699.2005.01324.x.

[19] Abraham L. Borker et al. "Vocal Activity as a Low Cost and Scalable Index of Seabird Colony Size". In: *Conservation Biology* 28.4 (Mar. 2014), pp. 1100–1108. ISSN: 1523-1739. DOI: 10.1111/cobi.12264.

[20] Francisco J Bravo Sanchez et al. "Bioacoustic classification of avian calls from raw sound waveforms with an open-source deep learning architecture". en. In: *Sci. Rep.* 11.1 (Aug. 2021), p. 15733. DOI: 10.1038/s41598-021-95076-6.

[21] E. O. Brigham and R. E. Morrow. "The fast Fourier transform". In: *IEEE Spectrum* 4.12 (1967), pp. 63–70. DOI: 10.1109/MSPEC.1967.5217220.

[22] Marco Brunoldi et al. "A Permanent Automated Real-Time Passive Acoustic Monitoring System for Bottlenose Dolphin Conservation in the Mediterranean Sea". In: *PLOS ONE* 11.1 (Jan. 2016), pp. 1–35. DOI: 10.1371/journal.pone.0145362.

[23] Marcel Buchhorn et al. "Establishing a reference tool for ecosystem accounting in Europe, based on the INCA methodology". In: *One Ecosystem* 7 (2022), e85389. DOI: 10.3897/oneeco.7.e85389.

[24] K Camphuysen et al. *Towards Standardised Seabirds at Sea Census Techniques in Connection with Environmental Impact Assessments for Offshore Wind Farms in the UK*. Tech. rep. COWRIE-BAM-02-2002. Royal Nether-

lands Institute for Sea Research (NIOZ), Apr. 2004. DOI: 10.13140/RG.2.1.2230.0244.

[25] C. Cannam, C. Landone, and M. Sandler. "Sonic Visualiser: An Open Source Application for Viewing, Analysing, and Annotating Music Audio Files". In: *Proceedings of the ACM Multimedia 2010 International Conference*. Firenze, Italy, Oct. 2010, pp. 1467–1468.

[26] Heqin Cao et al. "Wavelet Analysis Reveals Phenology Mismatch between Leaf Phenology of Temperate Forest Plants and the Siberian Roe Deer Molting under Global Warming". In: *Remote Sensing* 14.16 (2022). ISSN: 2072-4292. DOI: 10.3390/rs14163901.

[27] Thomas Capelle, Dominic Rampas, and Rahul Vigneswaran. *tcapelle/Diffusion-Models-pytorch.*, Mar. 20, 2023. URL: https://github.com/tcapelle/Diffusion-Models-pytorch.

[28] Bradley J. Cardinale et al. "Biodiversity loss and its impact on humanity". In: *Nature* 486.7401 (June 2012), pp. 59–67. ISSN: 1476-4687. DOI: 10.1038/nature11148.

[29] Pierre Chambon et al. *Adapting Pretrained Vision-Language Foundational Models to Medical Imaging Domains*. 2022. arXiv: 2210.04133 [cs.CV]. URL: https://arxiv.org/abs/2210.04133.

[30] Winston Chang. *shinythemes: Themes for Shiny*. R package version 1.2.0. 2021. URL: https://CRAN.R-project.org/package=shinythemes.

[31] Winston Chang and Barbara Borges Ribeiro. *shinydashboard: Create Dashboards with 'Shiny'*. R package version 0.7.2. 2021. URL: https://CRAN.R-project.org/package=shinydashboard.

[32] Winston Chang, Javier Luraschi, and Timothy Mastny. *profvis: Interactive Visualizations for Profiling R Code*. R package version 0.3.7. 2020. URL: https://CRAN.R-project.org/package=profvis.

[33] Winston Chang et al. *shiny: Web Application Framework for R*. R package version 1.7.1. 2021. URL: https://CRAN.R-project.org/package=shiny.

[34] Hangting Chen et al. "Long-term scalogram integrated with an iterative data augmentation scheme for acoustic scene classification". In: *The Journal of the Acoustical Society of America* 149.6 (June 2021), pp. 4198–4213. ISSN: 1520-8524. DOI: 10.1121/10.0005202.

[35] Joe Cheng et al. *leaflet: Create Interactive Web Maps with the JavaScript 'Leaflet' Library*. R package version 2.2.1. 2023. URL: https://CRAN.R-project.org/package=leaflet.

[36] Daniel Y. Choi, Thomas W. Wittig, and Bryan M. Kluever. "An evaluation of bird and bat mortality at wind turbines in the Northeastern United States". In: *PLOS ONE* 15.8 (Aug. 2020), pp. 1–22. DOI: 10.1371/journal.pone.0238034.

[37] François Chollet et al. *Keras*. https://keras.io. Software available from keras.io. 2015.

[38] Sarah J. Converse, Brett T. McClintock, and Paul B. Conn. "Special Feature: Linking capture–recapture and movement". In: *Ecology* 103.10 (2022), e3770. DOI: https://doi.org/10.1002/ecy.3770.

[39] Microsoft Corporation. *Visual Studio Code*. https://code.visualstudio.com/. Version 1.93.1. 2024.

[40] Anne-Sophie Crunchant et al. "Listening and watching: Do camera traps or acoustic sensors more efficiently detect wild chimpanzees in an open habitat?" In: *Methods in Ecology and Evolution* 11.4 (2020), pp. 542–552. DOI: 10.1111/2041-210X.13362.

[41] Wayne Dawson et al. "Global hotspots and correlates of alien species richness across taxonomic groups". In: *Nature Ecology & Evolution* 1.7 (June 2017). ISSN: 2397-334X. DOI: 10.1038/s41559-017-0186.

[42] Jia Deng et al. "ImageNet: A large-scale hierarchical image database". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[43] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[44] Department of the Environment, Climate and Communications; Department of the Taoiseach. *Climate action plan 2019*. June 2019. URL: https://www.gov.ie/en/publication/ccb2e0-the-climate-action-plan-2019/.

[45] Mia A. Derhé et al. "Measuring the success of reforestation for restoring biodiversity and ecosystem functioning". In: *Journal of Applied Ecology* 53.6 (2016), pp. 1714–1724. DOI: https://doi.org/10.1111/1365-2664.12728.

[46] Christophe Diagne et al. "High and rising economic costs of biological invasions worldwide". In: *Nature* 592.7855 (Mar. 2021), pp. 571–576. ISSN: 1476-4687. DOI: 10.1038/s41586-021-03405-6.

[47] Andrew Digby et al. "A practical comparison of manual and autonomous methods for acoustic monitoring". In: *Methods in Ecology and Evolution* 4.7 (2013), pp. 675–683. URL: https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12060.

[48] Vladimir Dinets et al. "Acoustic Detection of an Unknown Bat Species in Okinawa". In: *Mammal Study* 45.4 (Oct. 2020). ISSN: 1343-4152. DOI: 10.3106/ms2019-0077.

[49] Chris Donahue, Julian McAuley, and Miller Puckette. *Adversarial Audio Synthesis*. 2019. arXiv: 1802.04208 [cs.SD]. URL: https://arxiv.org/abs/1802.04208.

[50] Lisa W. Drew. "Are We Losing the Science of Taxonomy?: As need grows, numbers and training are failing to keep up." In: *BioScience* 61.12 (Dec. 2011), pp. 942–946. ISSN: 0006-3568. DOI: 10.1525/bio.2011.61.12.4.

[51] ALLAN L. DREWITT and ROWENA H. W. LANGSTON. "Assessing the impacts of wind farms on birds". In: *Ibis* 148.s1 (2006), pp. 29–42. DOI: 10.1111/j.1474-919X.2006.00516.x.

[52] Dryad. *Dryad Digital Repository*. https://datadryad.org. Accessed: March 26, 2025. 2025.

[53] Bram Edens et al. “Establishing the SEEA Ecosystem Accounting as a global standard”. In: *Ecosystem Services* 54 (2022), p. 101413. ISSN: 2212-0416. DOI: https://doi.org/10.1016/j.ecoser.2022.101413.

[54] Joris P.C. Eekhout et al. “The impact of reservoir construction and changes in land use and climate on ecosystem services in a large Mediterranean catchment”. In: *Journal of Hydrology* 590 (2020), p. 125208. ISSN: 0022-1694. DOI: https://doi.org/10.1016/j.jhydrol.2020.125208.

[55] Benjamin Elizalde et al. “CLAP: Learning Audio Concepts From Natural Language Supervision”. In: *arXiv preprint arXiv:2211.06687* (2023).

[56] Hiroto Enari et al. “An evaluation of the efficiency of passive acoustic monitoring in detecting deer and primates in comparison with camera traps”. In: *Ecological Indicators* 98 (2019), pp. 753–762. ISSN: 1470-160X. DOI: 10.1016/j.ecolind.2018.11.062.

[57] Environmental Risk Research Center, National Institute for Environmental Studies, Japan. *Invasive Species of Japan Database: Small Indian mongoose (Urva auropunctata)*. https://www.nies.go.jp/biodiversity/invasive/DB/detail/10210e.html. Accessed: 2025-05-07. 2025.

[58] European Commission. *Directive 2009/28/EC of the European Parliament and of the Council on the promotion of the use of energy from renewable sources*. https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32009L0028&from=en. Accessed: 2025-07-25. 2009.

[59] European Environment Agency. *CORINE Land Cover 2018 (vector), Europe, 6-yearly - version 2020_20u1, May 2020*. en. 2019. DOI: 10.2909/71C95A07-E296-44FC-B22B-415F42ACFDF0.

[60] European Organization For Nuclear Research and OpenAIRE. *Zenodo*. en. 2013. DOI: 10.25495/7GXK-RD71. URL: https://www.zenodo.org/.

[61] Catherine Anne Farrell et al. “Applying the System of Environmental Economic Accounting-Ecosystem Accounting (SEEA-EA) framework at catchment scale to develop ecosystem extent and condition accounts”. In: *One Ecosystem* 6 (2021), e65582. DOI: 10.3897/oneeco.6.e65582.

[62] André C. Ferreira et al. “Deep learning-based methods for individual recognition in small birds”. In: *Methods in Ecology and Evolution* 11.9 (2020), pp. 1072–1085. DOI: 10.1111/2041-210X.13436.

[63] Pablo Fierro et al. “Rainbow Trout diets and macroinvertebrates assemblages responses from watersheds dominated by native and exotic plantations”. In: *Ecological Indicators* 60 (2016), pp. 655–667. ISSN: 1470-160X. DOI: https://doi.org/10.1016/j.ecolind.2015.08.018.

[64] Sam Firke. *janitor: Simple Tools for Examining and Cleaning Dirty Data*. R package version 2.1.0. 2021. URL: https://CRAN.R-project.org/package=janitor.

[65] Mikael Lytzau Forup and Jane Memmott. “The restoration of plant–pollinator interactions in hay meadows”. In: *Restoration Ecology* 13.2 (2005), pp. 265–274.

[66] Python Software Foundation. *Python*. Version 3.10.12. 2023. URL: https://www.python.org/.

[67] Yixing Fu et al. “Classification of birdsong spectrograms based on DR-ACGAN and dynamic convolution”. In: *Ecological Informatics* 77 (2023), p. 102250. ISSN: 1574-9541. DOI: 10.1016/j.ecoinf.2023.102250.

[68] Yukio Fukuzawa et al. “Koe: Web-based software to classify acoustic units and analyse sequence structure in animal vocalizations”. In: *Methods in Ecology and Evolution* 11.3 (2020), pp. 431–441. DOI: https://doi.org/10.1111/2041-210X.13336.

[69] Frank Fundel, Daniel A. Braun, and Sebastian Gottwald. “Automatic bat call classification using transformer networks”. In: *Ecological Informatics* (2023). ISSN: 1574-9541. DOI: https://doi.org/10.1016/j.ecoinf.2023.102288.

[70] Gupta G et al. “Comparing recurrent convolutional neural networks for large scale bird species classification”. In: *Sci Rep. 2021 Aug 24;11(1):17085* 10 (2021). DOI: 1038/s41598-021-96446-w.

[71] Daniel García et al. “Common birds combine pest control and seed dispersal in apple orchards through a hybrid interaction network”. In: *Agriculture, Ecosystems & Environment* 365 (2024), p. 108927. ISSN: 0167-8809. DOI: 10.1016/j.agee.2024.108927.

[72] Garnier et al. *viridis - Colorblind-Friendly Color Maps for R*. R package version 0.6.2. 2021. DOI: 10.5281/zenodo.4679424. URL: https://sjmgarnier.github.io/viridis/.

[73] A. Gasc et al. “Cricket calling communities as an indicator of the invasive ant Wasmannia auropunctata in an insular biodiversity hotspot”. In: *Biological Invasions* 20.5 (Nov. 2017), pp. 1099–1111. ISSN: 1573-1464. DOI: 10.1007/s10530-017-1612-0.

[74] Jort F. Gemmeke et al. “Audio Set: An ontology and human-labeled dataset for audio events”. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017, pp. 776–780. DOI: 10.1109/ICASSP.2017.7952261.

[75] Burooj Ghani et al. “Global birdsong embeddings enable superior transfer learning for bioacoustic classification”. In: *Scientific Reports* 13.1 (2023), p. 22876. DOI: 10.1038/s41598-023-49989-z.

[76] Rory Gibb et al. “Emerging opportunities and challenges for passive acoustics in ecological assessment and monitoring”. In: *Methods in Ecology and Evolution* 10.2 (Nov. 2018). Ed. by Luca Börger, pp. 169–185. ISSN: 2041-210X. DOI: 10.1111/2041-210x.13101.

[77] Rory Gibb et al. “Emerging opportunities and challenges for passive acoustics in ecological assessment and monitoring”. In: *Methods in Ecology and Evolution* 10.2 (2019), pp. 169–185. DOI: 10.1111/2041-210X.13101.

[78] Anthony Gibbons et al. *Generative AI-based data augmentation for improved bioacoustic classification in noisy environments*. 2024. URL: https://arxiv.org/abs/2412.01530.

[79] Anthony Gibbons et al. “NEAL: an open-source tool for audio annotation”. In: *PeerJ* (2023). DOI: 10.7717/peerj.15913.

[80] Gillian Gilbert, Andrew Stanbury, and Lesley Lewis. "Birds of Conservation Concern in Ireland 4:2020-2026". In: *Irish Birds Number 43 2021* 43 (Aug. 2021), pp. 1–22.

[81] Douglas Gillespie et al. "Passive acoustic methods for tracking the 3D movements of small cetaceans around marine structures". In: *PLOS ONE* (May 2020), pp. 1–16. DOI: 10.1371/journal.pone.0229058.

[82] Hervé Goëau et al. "LifeCLEF Bird Identification Task 2016: The arrival of Deep learning". In: *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation Forum*. Vol. CEUR Workshop Proceedings. 1609. Évora, Portugal, Sept. 2016, pp. 440–449. URL: https://hal.science/hal-01373779.

[83] Ian J. Goodfellow et al. *Generative Adversarial Networks*. 2014. arXiv: 1406.2661 [stat.ML]. URL: https://arxiv.org/abs/1406.2661.

[84] Courtney E. Gorman et al. "A decision methodology for site-level ecosystem accounting". In: *Journal of Environmental Management* 366 (2024), p. 121814. ISSN: 0301-4797. DOI: https://doi.org/10.1016/j.jenvman.2024.121814. URL: https://www.sciencedirect.com/science/article/pii/S0301479724018000.

[85] Carolyn-Monika Görres and David Chesmore. "Active sound production of scarab beetle larvae opens up new possibilities for species-specific pest monitoring in soils". en. In: *Sci. Rep.* 9.1 (July 2019), p. 10115.

[86] Benjamin L. Gottesman et al. "Acoustic monitoring reveals diversity and surprising dynamics in tropical freshwater soundscapes". In: *Freshwater Biology* 65.1 (2020), pp. 117–132. DOI: 10.1111/fwb.13096.

[87] Jannis Gottwald et al. "Introduction of an automatic and open-source radiotracking system for small animals". In: *Methods in Ecology and Evolution* 10.12 (2019), pp. 2163–2172. DOI: https://doi.org/10.1111/2041-210X.13294.

[88] Jianping Gou et al. "Knowledge Distillation: A Survey". In: *International Journal of Computer Vision* 129.6 (June 2021), pp. 1789–1819. ISSN: 1573-1405. DOI: 10.1007/s11263-021-01453-z.

[89] Yves Grandvalet and Yoshua Bengio. "Semi-supervised Learning by Entropy Minimization". In: *Advances in Neural Information Processing Systems*. Ed. by L. Saul, Y. Weiss, and L. Bottou. Vol. 17. MIT Press, 2004. URL: https://proceedings.neurips.cc/paper_files/paper/2004/file/96f2b50b5d3613adf9c27049b2a888c7-Paper.pdf.

[90] David Granjon. *shinydashboardPlus: Add More 'AdminLTE2' Components to 'shinydashboard'*. R package version 2.0.3. 2021. URL: https://CRAN.R-project.org/package=shinydashboardPlus.

[91] Daniel W. Griffin and Jae S. Lim. "Signal estimation from modified short-time Fourier transform". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 32.2 (1984), pp. 236–243.

[92] Gudgud96. *frechet-audio-distance*. https://github.com/gudgud96/frechet-audio-distance. GitHub repository (main branch implementation of FAD). 2023.

[93] Axel-Christian Guei et al. "ECOGEN: Bird sounds generation using deep learning". In: *Methods in Ecology and Evolution* 15.1 (2024), pp. 69–79. DOI: https://doi.org/10.1111/2041-210X.14239.

[94] Nick M. Haddad et al. "Habitat fragmentation and its lasting impact on Earth's ecosystems". In: *Science Advances* 1.2 (2015), e1500052. DOI: 10.1126/sciadv.1500052.

[95] Stiele V. Hagens, Anthony R. Rendall, and Desley A. Whisson. "Passive acoustic surveys for predicting species' distributions: Optimising detection probability". In: *PLOS ONE* 13.7 (July 2018), pp. 1–16. DOI: 10.1371/journal.pone.0199396. URL: https://doi.org/10.1371/journal.pone.0199396.

[96] Grant A. Harper and Nancy Bunbury. "Invasive rats on tropical islands: Their population biology and impacts on native species". In: *Global Ecology and Conservation* 3 (2015), pp. 607–627. ISSN: 2351-9894. DOI: 10.1016/j.gecco.2015.02.010.

[97] Larry G Harris and Megan C Tyrrell. "Changing community states in the Gulf of Maine: synergism between invaders, overfishing and climate change". In: *Biological Invasions* 3 (2001), pp. 9–21.

[98] Kaiming He et al. *Deep Residual Learning for Image Recognition*. 2015. arXiv: 1512.03385 [cs.CV]. URL: https://arxiv.org/abs/1512.03385.

[99] Shawn Hershey et al. "CNN architectures for large-scale audio classification". In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017, pp. 131–135. DOI: 10.1109/ICASSP.2017.7952132.

[100] Martin Heusel et al. "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/8a1d694707eb0fefe65871369074926d-Paper.pdf.

[101] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. *Distilling the Knowledge in a Neural Network*. 2015. URL: https://arxiv.org/abs/1503.02531.

[102] Jonathan Ho, Ajay Jain, and Pieter Abbeel. *Denoising Diffusion Probabilistic Models*. 2020. arXiv: 2006.11239 [cs.LG]. URL: https://arxiv.org/abs/2006.11239.

[103] Jonathan M. Hoekstra et al. "Confronting a biome crisis: global disparities of habitat loss and protection". In: *Ecology Letters* 8.1 (2005), pp. 23–29. DOI: https://doi.org/10.1111/j.1461-0248.2004.00686.x.

[104] Daniel F Hofstadter et al. "Arresting the spread of invasive species in continental systems". In: *Frontiers in Ecology and the Environment* 20.5 (Jan. 2022), pp. 278–284. ISSN: 1540-9309. DOI: 10.1002/fee.2458.

[105] Torsten Hothorn and Achim Zeileis. "partykit: A Modular Toolkit for Recursive Partytioning in R". In: *Journal of Machine Learning Research* 16 (2015), pp. 3905–3909. URL: https://jmlr.org/papers/v16/hothorn15a.html.

[106] Jeremy Howard et al. *fastai/course22-web.*, June 24, 2024. URL: https://github.com/fastai/course22-web.

[107] Chunbo Huang et al. “How is biodiversity changing in response to ecological restoration in terrestrial ecosystems? A meta-analysis in China”. In: *Science of The Total Environment* 650 (2019), pp. 1–9. ISSN: 0048-9697. DOI: https://doi.org/10.1016/j.scitotenv.2018.08.320.

[108] Evi M.C. Huijben, Josien P.W. Pluim, and Maureen A.J.M. van Eijnatten. “Denoising diffusion probabilistic models for addressing data limitations in chest X-ray classification”. In: *Informatics in Medicine Unlocked* 50 (2024), p. 101575. ISSN: 2352-9148. DOI: 10.1016/j.imu.2024.101575.

[109] J. D. Hunter. “Matplotlib: A 2D graphics environment”. In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: 10.1109/MCSE.2007.55.

[110] GitHub Inc. *GitHub*. https://github.com. Accessed: March 26, 2025. 2025.

[111] Irish Wind Energy Association. *Environmental Impacts*. Accessed: 2025-08-06. 2019. URL: https://www.iwea.ie.

[112] Yosiaki Itô, Kuniharu Miyagi, and Hidetoshi Ota. “Imminent extinction crisis among the endemic species of the forests of Yanbaru, Okinawa, Japan”. In: *Oryx* 34.4 (Oct. 2000), pp. 305–316. ISSN: 1365-3008. DOI: 10.1046/j.1365-3008.2000.00136.x.

[113] Maes J et al. *Mapping and Assessment of Ecosystems and their Services: An EU ecosystem assessment*. Tech. rep. KJ-NA-30161-EN-N (online),KJ-NB-30161-EN-N (supplement). ISBN: 978-92-76-17833-0 (online),978-92-76-22954-4 (supplement). Luxembourg (Luxembourg): European Union, 2020. DOI: 10.2760/519233.

[114] Samuel M. Jantz et al. “Future habitat loss and extinctions driven by land-use change in biodiversity hotspots under four scenarios of climate-change mitigation”. In: *Conservation Biology* 29.4 (2015), pp. 1122–1131. DOI: https://doi.org/10.1111/cobi.12549.

[115] David Jarrett et al. “Mitigating bias in long-term terrestrial ecoacoustic studies”. In: *Journal of Applied Ecology* 62.4 (Feb. 2025), pp. 761–772. ISSN: 1365-2664. DOI: 10.1111/1365-2664.70000. URL: http://dx.doi.org/10.1111/1365-2664.70000.

[116] Lorène Jeantet and Emmanuel Dufourq. "Improving deep learning acoustic classifiers with contextual information for wildlife monitoring". In: *Ecological Informatics* (2023). ISSN: 1574-9541. DOI: https://doi.org/10.1016/j.ecoinf.2023.102256.

[117] Ahmadreza Jeddi, Mohammad Javad Shafiee, and Alexander Wong. *A Simple Fine-tuning Is All You Need: Towards Robust Deep Learning Via Adversarial Fine-tuning*. 2020. arXiv: 2012.13628 [cs.CV]. URL: https://arxiv.org/abs/2012.13628.

[118] THIERRY AUBIN JEROME SUEUR and CAROLINE SIMONIS. "SEEWAVE, A FREE MODULAR TOOL FOR SOUND ANALYSIS AND SYNTHESIS". In: *Bioacoustics* 18.2 (2008), pp. 213–226. DOI: 10.1080/09524622.2008.9753600.

[119] Mark P. Johnson. "PhycoCanopy: An R Shiny tool for exploring primary production in macroalgal canopies". In: *Methods in Ecology and Evolution* 13.5 (2022), pp. 963–968. DOI: https://doi.org/10.1111/2041-210X.13843.

[120] Gareth Jones et al. "Carpe noctem: the importance of bats as bioindicators". In: *Endangered species research* 8.1-2 (2009), pp. 93–115.

[121] K. Lisa Yang Center for Conservation Bioacoustics at the Cornell Lab of Ornithology. *Raven Pro: Interactive Sound Analysis Software (Version 1.6.5)*. Computer software. Ithaca, NY, 2024. URL: https://www.ravensoundsoftware.com/.

[122] Emmanuel Kahembwe and Subramanian Ramamoorthy. "Lower dimensional kernels for video discriminators". In: *Neural Networks* 132 (2020), pp. 506–520. ISSN: 0893-6080. DOI: 10.1016/j.neunet.2020.09.016.

[123] Stefan Kahl et al. "BirdNET: A deep learning solution for avian diversity monitoring". In: *Ecological Informatics* 61 (2021), p. 101236. DOI: https://doi.org/10.1016/j.ecoinf.2021.101236.

[124] Yukibumi Kaneko. "Historical investigation into articles on alien species introduced around 1910, particularly Shozaburou Watase's introduction of the mongoose to the Okinawa Islands, Japan". In: *Honyurui Kagaku (Mammalian Science)* 61.2 (2021), pp. 129–160. DOI: 10.11238/mammalianscience.61.129.

[125] Jared Kaplan et al. *Scaling Laws for Neural Language Models*. 2020. DOI: 10.48550/ARXIV.2001.08361. URL: https://arxiv.org/abs/2001.08361.

[126] Bhaskar Karambelkar and Barret Schloerke. *leaflet.extras: Extra Functionality for 'leaflet' Package*. R package version 1.0.0. 2018. URL: https://CRAN.R-project.org/package=leaflet.extras.

[127] David A. Keith et al. "A function-based typology for Earth's ecosystems". In: *Nature* 610.7932 (Oct. 2022), pp. 513–518. ISSN: 1476-4687. DOI: 10.1038/s41586-022-05318-4.

[128] Brian Keville. *A Spatial Analysis of Ireland's Future Onshore Wind Energy Potential*. PowerPoint presentation at Irish Wind Energy Association (IWEA) Spring Conference 2019. Accessed: 2025-08-06. 2019. URL: https://windenergyireland.com/images/Article_files/Brian_Keville_Powerpoint_slides_-_IWEA_Spring_2019.pdf.

[129] Gerold Kier et al. "A global assessment of endemism and species richness across island and mainland regions". In: *Proceedings of the National Academy of Sciences* 106.23 (2009), pp. 9322–9327. DOI: 10.1073/pnas.0810306106.

[130] Kevin Kilgour et al. "Fréchet Audio Distance: A Reference-Free Metric for Evaluating Music Enhancement Algorithms". In: *arXiv preprint arXiv:1812.08466* (2019).

[131] Gwantae Kim, David K. Han, and Hanseok Ko. *SpecMix : A Mixed Sample Data Augmentation method for Training withTime-Frequency Domain Features*. 2021. arXiv: 2108.03020 [cs.SD]. URL: https://arxiv.org/abs/2108.03020.

[132] Emma King et al. “Ecosystem accounting for biodiversity surrounding windfarms”. In: *Journal of Environmental Management* 391 (2025), p. 126618. ISSN: 0301-4797. DOI: https://doi.org/10.1016/j.jenvman.2025.126618. URL: https://www.sciencedirect.com/science/article/pii/S0301479725025940.

[133] Diederik P Kingma and Max Welling. *Auto-Encoding Variational Bayes*. 2022. arXiv: 1312.6114 [stat.ML]. URL: https://arxiv.org/abs/1312.6114.

[134] Holger Klinck et al. *BirdCLEF 2024*. https://kaggle.com/competitions/birdclef-2024. Kaggle. 2024.

[135] Shun Kobayashi et al. “Mammal and bird species using cavities among forests of different ages on Okinawajima Island in the Ryukyu Archipelago, Japan”. In: *Journal of Forest Research* 27.6 (Apr. 2022), pp. 439–449. ISSN: 1610-7403. DOI: 10.1080/13416979.2022.2060553.

[136] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems*. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc., 2012. URL: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[137] S. Kullback and R. A. Leibler. “On Information and Sufficiency”. In: *The Annals of Mathematical Statistics* 22.1 (Mar. 1951), pp. 79–86. ISSN: 0003-4851. DOI: 10.1214/aoms/1177729694.

[138] Kundan Kumar et al. *MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis*. 2019. arXiv: 1910.06711 [eess.AS]. URL: https://arxiv.org/abs/1910.06711.

[139] Doris Lagos-Kutz et al. “The Soybean Aphid Suction Trap Network: Sampling the Aerobiological “Soup””. In: *American Entomologist* 66.1 (Mar. 2020), pp. 48–55. ISSN: 1046-2821. DOI: 10.1093/ae/tmaa009.

[140] Renaud Lahaye et al. “DNA barcoding the floras of biodiversity hotspots”. In: *Proceedings of the National Academy of Sciences* 105.8 (2008), pp. 2923–2928. DOI: 10.1073/pnas.0709936105.

[141] Daniel Langenkämper et al. “On the impact of Citizen Science-derived data quality on deep learning based classification in marine images”. In: *PLOS ONE* 14.6 (June 2019), pp. 1–16. DOI: 10.1371/journal.pone.0218086. URL: https://doi.org/10.1371/journal.pone.0218086.

[142] Frank Wugt Larsen et al. “Birds as biodiversity surrogates: will supplementing birds with other taxa improve effectiveness?” In: *Journal of Applied Ecology* 49.2 (2012), pp. 349–356. DOI: 10.1111/j.1365-2664.2011.02094.x.

[143] Sabine Lattemann and Thomas Höpner. “Environmental impact and impact assessment of seawater desalination”. In: *Desalination* 220.1 (2008). European Desalination Society and Center for Research and Technology Hellas (CERTH), Sani Resort 22 –25 April 2007, Halkidiki, Greece, pp. 1–15. ISSN: 0011-9164. DOI: https://doi.org/10.1016/j.desal.2007.03.009. URL: https://www.sciencedirect.com/science/article/pii/S0011916407006005.

[144] Patrik Lauha et al. “Domain-specific neural networks improve automated bird sound recognition already with small amount of local data”. In: *Methods in Ecology and Evolution* 13.12 (2022), pp. 2799–2810. DOI: 10.1111/2041-210X.14003.

[145] Y. Lecun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324. DOI: 10.1109/5.726791.

[146] Jongpil Lee et al. *Raw Waveform-based Audio Classification Using Sample-level CNN Architectures*. 2017. arXiv: 1712.00866 [cs.SD]. URL: https://arxiv.org/abs/1712.00866.

[147] Tsai-Wei Li and Gwo-Chuan Lee. “Performance Analysis of Fine-tune Transferred Deep Learning”. In: *2021 IEEE 3rd Eurasia Conference on*

IOT, Communication and Engineering (ECICE). 2021, pp. 315–319. DOI: 10.1109/ECICE52819.2021.9645649.

[148] Uwe Ligges et al. *tuneR: Analysis of Music and Speech*. 2018. URL: https://CRAN.R-project.org/package=tuneR.

[149] Kai-Hsiang Lin et al. "Improving faster-than-real-time human acoustic event detection by saliency-maximized audio visualization". In: *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2012, pp. 2277–2280. DOI: 10.1109/ICASSP.2012.6288368.

[150] Pavel Linhart et al. "The potential for acoustic individual identification in mammals". In: *Mammalian Biology* 102.3 (June 2022), pp. 667–683. ISSN: 1618-1476. DOI: 10.1007/s42991-021-00222-2.

[151] Tyler Littlefield and Colin Fay. *keys: Keyboard Shortcuts for 'shiny'*. R package version 0.1.1. 2021. URL: https://CRAN.R-project.org/package=keys.

[152] Vincent Lostanlen et al. "Robust sound event detection in bioacoustic sensor networks". In: *PLOS ONE* 14.10 (Oct. 2019), pp. 1–31. DOI: 10.1371/journal.pone.0214168. URL: https://doi.org/10.1371/journal.pone.0214168.

[153] Sarah Lowe et al. *100 of the world's worst invasive alien species: a selection from the global invasive species database*. Vol. 12. Invasive Species Specialist Group Auckland, 2000. URL: https://portals.iucn.org/library/sites/library/files/documents/2000-126.pdf.

[154] Chen-Yi Lu, Dan Jeric Arcega Rustia, and Ta-Te Lin. "Generative Adversarial Network Based Image Augmentation for Insect Pest Classification Enhancement". In: *IFAC-PapersOnLine* 52.30 (2019). 6th IFAC Conference on Sensing, Control and Automation Technologies for Agriculture AGRICONTROL 2019, pp. 1–5. ISSN: 2405-8963. DOI: 10.1016/j.ifacol.2019.12.406.

[155] Andreas Lugmayr et al. *RePaint: Inpainting using Denoising Diffusion Probabilistic Models*. 2022. arXiv: 2201.09865 [cs.CV]. URL: https://arxiv.org/abs/2201.09865.

[156] Xuenan Ma et al. "Assessing the relative contributions, combined effects and multiscale uncertainty of future land use and climate change on water-related ecosystem services in Southwest China using a novel integrated modelling framework". In: *Sustainable Cities and Society* 106 (2024), p. 105400. ISSN: 2210-6707. DOI: https://doi.org/10.1016/j.scs.2024.105400.

[157] Oisin Mac Aodha et al. "Bat detective—Deep learning tools for bat acoustic signal detection". In: *PLoS computational biology* 14.3 (2018), e1005995. DOI: 10.1371/journal.pcbi.1005995.

[158] Oisin Mac Aodha et al. "Bat detective—Deep learning tools for bat acoustic signal detection". In: *PLOS Computational Biology* 14.3 (Mar. 2018). Ed. by Brock Fenton, e1005995. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1005995.

[159] Shyam Madhusudhana. *shyamblast/Koogu: v0.7.2*. 2023. DOI: 10.5281/ZENODO.8254287. URL: https://zenodo.org/record/8254287.

[160] MaREI, the SFI Research Centre for Energy, Climate and Marine. *Nature + Energy*. 2024. URL: https://www.marei.ie/project/natureenergy/.

[161] Stephen Marsland et al. "AviaNZ: A future-proofed program for annotation and recognition of animal sounds in long-time field recordings". In: *Methods in Ecology and Evolution* 10 (8 2019), pp. 1189–1195. ISSN: 2041210X. DOI: 10.1111/2041-210X.13213.

[162] V. Martínez-López et al. "Uncovering mechanisms of bird seed dispersal in semiarid environments to help to restore them". In: *Ecosphere* 10.4 (2019), e02673. DOI: 10.1002/ecs2.2673.

[163] Brian McFee et al. *librosa/librosa: 0.10.2.post1*. Version 0.10.2.post1. May 2024. DOI: 10.5281/zenodo.11192913.

[164] D. W. McWhirter et al. "A check-list of the birds of Okinawa Prefecture with notes on recent status including hypothetical records". In: *Bulletin of Okinawa Prefectural Museum* 22 (1996), pp. 33–152. URL: https://okimu.jp/sp/userfiles/files/page/museum/issue/bulletin/kiyou22/22-3.pdf.

[165] A. Mesaros et al. “DCASE 2017 Challenge Setup: Tasks, Datasets and Baseline System”. In: *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017)*. Nov. 2017, pp. 85–92. URL: https://dcase.community/documents/workshop2017/proceedings/DCASE2017Workshop_Mesaros_100.pdf.

[166] Oliver C. Metcalf et al. “Optimizing tropical forest bird surveys using passive acoustic monitoring and high temporal resolution sampling”. In: *Remote Sensing in Ecology and Conservation* (July 2021). Ed. by Nathalie Pettorelli and Christos Astaras, pp. 45–56. ISSN: 2056-3485. DOI: 10.1002/rse2.227.

[167] Midjourney, Inc. *Midjourney*. Date Accessed: 2024-10-16. 2024. URL: https://www.midjourney.com.

[168] A.M. Milanelli et al. “Temporal patterns in the soundscape of the port area in an urban estuary”. In: *Estuarine, Coastal and Shelf Science* 297 (Feb. 2024). ISSN: 0272-7714. DOI: 10.1016/j.ecss.2023.108596.

[169] Millennium Ecosystem Assessment Business and Industry Synthesis Team. *Ecosystems and Human Well-being: Opportunities and Challenges for Business and Industry*. https://www.millenniumassessment.org/documents/document.353.aspx.pdf. Accessed: 2025-07-25. 2005.

[170] Asako Miyamoto, Syoko Tamanaha, and Yuya Watari. “Landscape features of endangered Ryukyu long-furred rat (Diplothrix legata) roadkill sites in Yambaru, Okinawa-jima Island”. In: *Journal of Forest Research* 26.3 (Feb. 2021), pp. 201–207. ISSN: 1610-7403. DOI: 10.1080/13416979.2021.1887437.

[171] Alessandro Mondanaro et al. “Modelling reveals the effect of climate and land use change on Madagascar’s chameleons fauna”. In: *Communications Biology* 7.1 (July 2024), p. 889. ISSN: 2399-3642. DOI: 10.1038/s42003-024-06597-5.

[172] M.M. Morgan and J. Braasch. “Long-term deep learning-facilitated environmental acoustic monitoring in the Capital Region of New York State”. In: *Ecological Informatics* 61 (2021), p. 101242. ISSN: 1574-9541. DOI: https:

//doi.org/10.1016/j.ecoinf.2021.101242. URL: https://www.sciencedirect.com/science/article/pii/S1574954121000339.

[173] James AJ Mortimer and Terry C Greene. "Investigating bird call identification uncertainty using data from processed audio recordings". In: *New Zealand journal of ecology* 41.1 (2017), pp. 126–133.

[174] Ilyass Moummad, Romain Serizel, and Nicolas Farrugia. *Self-Supervised Learning for Few-Shot Bird Sound Classification*. 2024. arXiv: 2312.15824 [cs.SD]. URL: https://arxiv.org/abs/2312.15824.

[175] Ilyass Moummad et al. *Mixture of Mixups for Multi-label Classification of Rare Anuran Sounds*. 2024. eprint: arXiv:2403.09598.

[176] Wenjie Mu et al. "Environmental sound classification using temporal-frequency attention based convolutional neural network". In: *Scientific Reports* 11.1 (Nov. 2021), p. 21552. ISSN: 2045-2322. DOI: 10.1038/s41598-021-01045-4.

[177] Taslim Murad, Sarwan Ali, and Murray Patterson. "Exploring the potential of GANs in biological sequence analysis". en. In: *Biology (Basel)* 12.6 (2023). DOI: 10.3390%2Fbiology12060854.

[178] Elon Musk. *Interview on X Broadcast*. https://x.com/i/broadcasts/1yNxagNdkMlGj. Accessed: 2025-07-17. 2025.

[179] Andrew Ng. *MLOps: From Model-centric to Data-centric AI*. June 2021. URL: https://www.deeplearning.ai/wp-content/uploads/2021/06/MLOps-From-Model-centric-to-Data-centric-AI.pdf.

[180] Joseph M. Northrup et al. "Synergistic effects of climate and land-use change influence broad-scale avian population declines". In: *Global Change Biology* 25.5 (2019), pp. 1561–1575. DOI: https://doi.org/10.1111/gcb.14571.

[181] Stavros Ntalampiras. "Bird species identification via transfer learning from music genres". In: *Ecological Informatics* 44 (2018), pp. 76–81. ISSN: 1574-9541. DOI: https://doi.org/10.1016/j.ecoinf.2018.01.006. URL: https://www.sciencedirect.com/science/article/pii/S1574954117302467.

[182] NVIDIA, Péter Vingelmann, and Frank H.P. Fitzek. *CUDA, release: 12.2*. 2023. URL: https://developer.nvidia.com/cuda-toolkit.

[183] Yurika Oba and Hideyuki Doi. "Accelerating ecosystem monitoring through computer vision with deep metric learning". In: *Ecological Complexity* 62 (2025), p. 101124. ISSN: 1476-945X. DOI: 10.1016/j.ecocom.2025.101124.

[184] Augustus Odena, Christopher Olah, and Jonathon Shlens. *Conditional Image Synthesis With Auxiliary Classifier GANs*. 2017. arXiv: 1610.09585 [stat.ML]. URL: https://arxiv.org/abs/1610.09585.

[185] Jeroen Ooms. *magick: Advanced Graphics and Image-Processing in R*. R package version 2.8.1. 2023. URL: https://CRAN.R-project.org/package=magick.

[186] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. *Neural Discrete Representation Learning*. 2018. arXiv: 1711.00937 [cs.LG]. URL: https://arxiv.org/abs/1711.00937.

[187] Julie N. Oswald et al. "A Collection of Best Practices for the Collection and Analysis of Bioacoustic Data". In: *Applied Sciences* 12.23 (2022). ISSN: 2076-3417. DOI: 10.3390/app122312046. URL: https://www.mdpi.com/2076-3417/12/23/12046.

[188] A. F. Owens et al. "Automated detection of Bornean white-bearded gibbon (Hylobates albibarbis) vocalisations using an open-source framework for deep learning". In: *bioRxiv* (2024). DOI: 10.1101/2024.04.15.589517.

[189] Morgane Papin et al. "Acoustic localization at large scales: a promising method for grey wolf monitoring". In: *Frontiers in Zoology* (Apr. 2018). ISSN: 1742-9994. DOI: 10.1186/s12983-018-0260-2.

[190] Hyunsin Park and Chang D. Yoo. "CNN-Based Learnable Gammatone Filterbank and Equal-Loudness Normalization for Environmental Sound Classification". In: *IEEE Signal Processing Letters* 27 (2020), pp. 411–415. DOI: 10.1109/LSP.2020.2975422.

[191] Taesung Park et al. *Semantic Image Synthesis with Spatially-Adaptive Normalization*. 2019. arXiv: 1903.07291 [cs.CV]. URL: https://arxiv.org/abs/1903.07291.

[192] Dawn Parry et al. “Latent Diffusion Model Based Spectrogram Augmentation Techniques for Whale Call Detection in Novel Environments”. In: *OCEANS 2024 - Halifax*. 2024, pp. 1–7. DOI: 10.1109/OCEANS55160.2024.10754260.

[193] Luz Pascal et al. “A Shiny r app to solve the problem of when to stop managing or surveying species under imperfect detection”. In: *Methods in Ecology and Evolution* 11.12 (2020), pp. 1707–1715. DOI: https://doi.org/10.1111/2041-210X.13501.

[194] Adam Paszke et al. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*. Ed. by H. Wallach et al. Software available from pytorch.org. 2019, pp. 8024–8035. URL: https://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[195] Dwarkesh Patel. *Can synthetic data unlock AI recursive self-improvement? — Mark Zuckerberg*. https://www.youtube.com/watch?v=9TU0XjJqp0g. Accessed: 2025-07-17. Apr. 2024.

[196] Edzer Pebesma. “Simple Features for R: Standardized Support for Spatial Vector Data”. In: *The R Journal* 10.1 (2018), pp. 439–446. DOI: 10.32614/RJ-2018-009. URL: https://doi.org/10.32614/RJ-2018-009.

[197] Thomas Lin Pedersen et al. *shinyFiles: A Server-Side File System Viewer for Shiny*. R package version 0.9.1. 2021. URL: https://CRAN.R-project.org/package=shinyFiles.

[198] Victor Perrier, Fanny Meyer, and David Granjon. *shinyWidgets: Custom Inputs Widgets for Shiny*. R package version 0.6.3. 2022. URL: https://CRAN.R-project.org/package=shinyWidgets.

[199] Marta Picciulin et al. “A Fish and Dolphin Biophony in the Boat Noise-Dominated Soundscape of the Cres-Lošinj Archipelago (Croatia)”. In: *Journal of Marine Science and Engineering* 2 (Feb. 2022). ISSN: 2077-1312. DOI: 10.3390/jmse10020300.

[200] EirGrid Plc and SONI Ltd. *Tomorrow's Energy Scenarios 2023 Final Report*. https://cms.eirgrid.ie/sites/default/files/publications/TES-2023-Final-Full-Report.pdf. Accessed: 2025-07-25. Jan. 2024.

[201] Kit S. Prendergast et al. "The relative performance of sampling methods for native bees: an empirical test and review of the literature". In: *Ecosphere* 11.5 (2020), e03076. DOI: https://doi.org/10.1002/ecs2.3076.

[202] Tharaka S. Priyadarshana et al. "Crop and landscape heterogeneity increase biodiversity in agricultural landscapes: A global review and meta-analysis". In: *Ecology Letters* 27.3 (2024). e14412 ELE-01112-2023.R2, e14412. DOI: https://doi.org/10.1111/ele.14412.

[203] R.L. Putland et al. "A hidden Markov model approach to indicate Bryde's whale acoustics". In: *Ecological Indicators* (2018), pp. 479–487. ISSN: 1470-160X. DOI: https://doi.org/10.1016/j.ecolind.2017.09.025.

[204] Python Software Foundation. *Python*. Version 3.9.21. 2024. URL: https://www.python.org.

[205] PyTorch Core Team. *torch.nn.CosineEmbeddingLoss*. https://docs.pytorch.org/docs/stable/generated/torch.nn.CosineEmbeddingLoss.html. Accessed: 2025-05-09. 2024.

[206] PyTorch Team. *PyTorch*. Version 2.6.0. 2025. URL: https://pytorch.org.

[207] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2023. URL: https://www.R-project.org/.

[208] Aditya Ramesh et al. *Zero-Shot Text-to-Image Generation*. 2021. arXiv: 2102.12092 [cs.CV]. URL: https://arxiv.org/abs/2102.12092.

[209] Jamie K. Reaser et al. "Ecological and socioeconomic impacts of invasive alien species in island ecosystems". In: *Environmental Conservation* 34.2 (2007), pp. 98–111. DOI: 10.1017/S0376892907003815.

[210] Joseph Redmon et al. "You only look once: Unified, real-time object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 779–788.

[211] Yao Ren et al. “A Framework for Bioacoustic Vocalization Analysis Using Hidden Markov Models”. In: *Algorithms* 2.4 (2009), pp. 1410–1428. ISSN: 1999-4893. DOI: 10.3390/a2041410. URL: https://www.mdpi.com/1999-4893/2/4/1410.

[212] Suzanne M. Richardson et al. “Peaks in bat activity at turbines and the implications for mitigating the impact of wind energy developments on bats”. In: *Scientific Reports* 11.1 (Feb. 2021), p. 3636. ISSN: 2045-2322. DOI: 10.1038/s41598-021-82014-9. URL: https://doi.org/10.1038/s41598-021-82014-9.

[213] D Risch et al. “Minke whale acoustic behavior and multi-year seasonal and diel vocalization patterns in Massachusetts Bay, USA”. In: *Marine Ecology Progress Series* (Aug. 2013), pp. 279–295. ISSN: 1616-1599. DOI: 10.3354/meps10426.

[214] Francisco Rivas et al. “Rthoptera: Standardised Insect Bioacoustics in R”. In: *bioRxiv* (2024). DOI: 10.1101/2024.11.04.621915.

[215] Andrea Ravignani Rodrigo Manriquez P Sonja A. Kotz and Bart de Boer. “Bioacoustic classification of a small dataset of mammalian vocalisations using deep learning”. In: *Bioacoustics* 33.4 (2024), pp. 354–371. DOI: 10.1080/09524622.2024.2354468.

[216] Andrea Ravignani Rodrigo Manriquez P Sonja A. Kotz and Bart de Boer. “Bioacoustic classification of a small dataset of mammalian vocalisations using deep learning”. In: *Bioacoustics* 33.4 (2024), pp. 354–371. DOI: 10.1080/09524622.2024.2354468.

[217] Tracey L. Rogers et al. “Density Can Be Misleading for Low-Density Species: Benefits of Passive Acoustic Monitoring”. In: *PLOS ONE* 8.1 (Jan. 2013), pp. 1–11. DOI: 10.1371/journal.pone.0052542. URL: https://doi.org/10.1371/journal.pone.0052542.

[218] Robin Rombach et al. *High-Resolution Image Synthesis with Latent Diffusion Models*. 2022. arXiv: 2112.10752 [cs.CV]. URL: https://arxiv.org/abs/2112.10752.

[219] Jesse C. Ross and Paul E. Allen. "Random Forest for improved analysis efficiency in passive acoustic monitoring". In: *Ecological Informatics* 21 (2014). Ecological Acoustics, pp. 34–39. ISSN: 1574-9541. DOI: https://doi.org/10.1016/j.ecoinf.2013.12.002. URL: https://www.sciencedirect.com/science/article/pii/S1574954113001234.

[220] Samuel R. P. -J. Ross et al. "Listening to ecosystems: data-rich acoustic monitoring through landscape-scale sensor networks". In: *Ecological Research* 33.1 (2018), pp. 135–147. DOI: 10.1007/s11284-017-1509-5.

[221] Samuel R. P.-J. Ross et al. "Divergent ecological responses to typhoon disturbance revealed via landscape-scale acoustic monitoring". In: *Global Change Biology* 30.1 (2024). e17067 GCB-23-1585.R1, e17067. DOI: https://doi.org/10.1111/gcb.17067. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/gcb.17067. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/gcb.17067.

[222] Samuel R. P.-J. Ross et al. "Passive acoustic monitoring provides a fresh perspective on fundamental ecological questions". In: *Functional Ecology* 37.4 (2023), pp. 959–975. DOI: 10.1111/1365-2435.14275.

[223] Samuel R. P.-J. Ross et al. "Divergent ecological responses to typhoon disturbance revealed via landscape-scale acoustic monitoring". In: *Global Change Biology* 30.1 (Dec. 2023). ISSN: 1365-2486. DOI: 10.1111/gcb.17067.

[224] Samuel R. P.-J. Ross et al. "Listening to ecosystems: data-rich acoustic monitoring through landscape-scale sensor networks". In: *Ecological Research* 33.1 (Nov. 2017), pp. 135–147. ISSN: 1440-1703. DOI: 10.1007/s11284-017-1509-5.

[225] Samuel R.P-J. Ross et al. "Utility of acoustic indices for ecological monitoring in complex sonic environments". In: *Ecological Indicators* 121 (2021), p. 107114. ISSN: 1470-160X. DOI: https://doi.org/10.1016/j.ecolind.2020.107114. URL: https://www.sciencedirect.com/science/article/pii/S1470160X20310530.

[226] Samuel R.P-J. Ross et al. "Utility of acoustic indices for ecological monitoring in complex sonic environments". In: *Ecological Indicators* 121 (Feb. 2021), p. 107114. ISSN: 1470-160X. DOI: 10.1016/j.ecolind.2020.107114.

[227] Rodney A. Rountree and Francis Juanes. "Potential of passive acoustic recording for monitoring invasive species: freshwater drum invasion of the Hudson River via the New York canal system". In: *Biological Invasions* 19.7 (Mar. 2017), pp. 2075–2088. ISSN: 1573-1464. DOI: 10.1007/s10530-017-1419-z.

[228] RStudio Team. *RStudio: Integrated Development Environment for R*. RStudio, PBC. Boston, MA, 2020. URL: http://www.rstudio.com/.

[229] Zachary J. Ruff et al. "PNW-Cnet v4: Automated species identification for passive acoustic monitoring". In: *SoftwareX* 23 (July 2023). ISSN: 2352-7110. DOI: 10.1016/j.softx.2023.101473.

[230] Cate Ryan et al. "Ecosystem integrity of active sand dunes: A case study to implement and test the SEEA-EA global standard, from Aotearoa New Zealand". In: *Ecological Indicators* 149 (2023), p. 110172. ISSN: 1470-160X. DOI: https://doi.org/10.1016/j.ecolind.2023.110172.

[231] Seyedmorteza Sadat et al. *CADS: Unleashing the Diversity of Diffusion Models through Condition-Annealed Sampling*. 2024. arXiv: 2310.17347 [cs.CV]. URL: https://arxiv.org/abs/2310.17347.

[232] Masaki Saito et al. "Train Sparsely, Generate Densely: Memory-Efficient Unsupervised Training of High-Resolution Temporal GAN". In: *International Journal of Computer Vision* 128.10 (2020), pp. 2586–2606. ISSN: 1573-1405. DOI: 10.1007/s11263-020-01333-y.

[233] Justin Salamon et al. "Fusing shallow and deep learning for bioacoustic bird species classification". In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017, pp. 141–145. DOI: 10.1109/ICASSP.2017.7952134.

[234] Andras Sali and Dean Attali. *shinycssloaders: Add Loading Animations to a 'shiny' Output While It's Recalculating*. R package version 1.0.0. 2020. URL: https://CRAN.R-project.org/package=shinycssloaders.

[235] Tim Salimans et al. *Improved Techniques for Training GANs*. 2016. arXiv: 1606.03498 [cs.LG]. URL: https://arxiv.org/abs/1606.03498.

[236] Lida Sanchez, Christian R. Moreno, and Emanuel C. Mora. "Echolocation calls ofNatalus primus(Chiroptera: Natalidae): Implications for conservation monitoring of this species". In: *Cogent Biology* (Jan. 2017). Ed. by Hynek Burda. ISSN: 2331-2025. DOI: 10.1080/23312025.2017.1355027.

[237] Mark Sandler et al. *MobileNetV2: Inverted Residuals and Linear Bottlenecks*. 2019. arXiv: 1801.04381 [cs.CV]. URL: https://arxiv.org/abs/1801.04381.

[238] Karthik V. Sarma et al. "Harnessing clinical annotations to improve deep learning performance in prostate segmentation". In: *PLOS ONE* 16.6 (June 2021), pp. 1–15. DOI: 10.1371/journal.pone.0253829. URL: https://doi.org/10.1371/journal.pone.0253829.

[239] Michele Scarpiniti, Raffaele Parisi, and Yong-Cheol Lee. "A Scalogram-Based CNN Approach for Audio Classification in Construction Sites". In: *Applied Sciences* 14.1 (2024). ISSN: 2076-3417. DOI: 10.3390/app14010090. URL: https://www.mdpi.com/2076-3417/14/1/90.

[240] Barret Schloerke. *reactlog: Reactivity Visualizer for 'shiny'*. R package version 1.1.0. 2020. URL: https://CRAN.R-project.org/package=reactlog.

[241] Esther Sebastián-González et al. "Bioacoustics for species management: two case studies with a Hawaiian forest bird". In: *Ecology and Evolution* 5.20 (2015), pp. 4696–4705. DOI: https://doi.org/10.1002/ece3.1743.

[242] Taylor R. Shirley and Adam K. Janke. "Ring-necked pheasant nest site selection in a landscape with high adoption of fall-seeded cover crops". In: *Wildlife Society Bulletin* 47.1 (2023), e1394. DOI: https://doi.org/10.1002/wsb.1394.

[243] Connor Shorten and Taghi M. Khoshgoftaar. "A survey on Image Data Augmentation for Deep Learning". In: *Journal of Big Data* 6.1 (July 2019), p. 60. ISSN: 2196-1115. DOI: 10.1186/s40537-019-0197-0.

[244] Bruno Silva et al. "soundClass: An automatic sound classification tool for biodiversity monitoring using machine learning". In: *Methods in Ecology and Evolution* (Aug. 2022), pp. 2356–2362. ISSN: 2041-210X. DOI: 10.1111/2041-210x.13964.

[245] Carlos Alberto Silva et al. "treetop: A Shiny-based application and R package for extracting forest information from LiDAR data for ecologists and conservationists". In: *Methods in Ecology and Evolution* 13.6 (2022), pp. 1164–1176. DOI: https://doi.org/10.1111/2041-210X.13830.

[246] Pedro Silva. *imola: CSS Layouts (Grid and Flexbox) Implementation for R/Shiny*. R package version 0.3.2. 2021. URL: https://CRAN.R-project.org/package=imola.

[247] Karen Simonyan and Andrew Zisserman. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2014. DOI: 10.48550/ARXIV.1409.1556.

[248] Karen Simonyan and Andrew Zisserman. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015. arXiv: 1409.1556 [cs.CV]. URL: https://arxiv.org/abs/1409.1556.

[249] Robert Dargavel Smith et al. *teticio/audio-diffusion*. Sept. 25, 2024. URL: https://github.com/teticio/audio-diffusion.

[250] Jascha Sohl-Dickstein et al. *Deep Unsupervised Learning using Nonequilibrium Thermodynamics*. 2015. arXiv: 1503.03585 [cs.LG]. URL: https://arxiv.org/abs/1503.03585.

[251] Amit Kumar Srivastava et al. "Winter wheat yield prediction using convolutional neural networks from environmental and phenological data". In: *Scientific Reports* 12.1 (Feb. 2022), p. 3215. ISSN: 2045-2322. DOI: 10.1038/s41598-022-06249-w. URL: https://doi.org/10.1038/s41598-022-06249-w.

[252] S. S. Stevens, J. Volkmann, and E. B. Newman. "A Scale for the Measurement of the Psychological Magnitude Pitch". In: *The Journal of the Acoustical Society of America* 8.3 (Jan. 1937), pp. 185–190. ISSN: 0001-4966. DOI: 10.1121/1.1915893.

[253] Jane C. Stout et al. *SIMBIOSYS: Sectoral Impacts on Biodiversity and Ecosystem Services*. STRIVE Report Series No.115. Accessed: 2025-07-25. Environmental Protection Agency, Ireland, 2013. URL: https://www.epa.ie/publications/research/biodiversity/STRIVE_115_SIMBIOSYS-Sectoral-Impacts-on-Biodiversity-and-Ecosystem-Services.pdf.

[254] Dan Stowell. "Computational bioacoustics with deep learning: a review and roadmap". In: *PeerJ* 10 (Mar. 2022), e13152. DOI: 10.7717/peerj.13152.

[255] Dan Stowell. "Computational bioacoustics with deep learning: a review and roadmap". In: *PeerJ* 10 (Mar. 2022), e13152. ISSN: 2167-8359. DOI: 10.7717/peerj.13152.

[256] Dan Stowell and Mark D. Plumbley. "Automatic large-scale classification of bird sounds is strongly improved by unsupervised feature learning". In: *PeerJ* 2 (July 2014), e488. ISSN: 2167-8359. DOI: 10.7717/peerj.488.

[257] Dan Stowell et al. "Automatic acoustic detection of birds through deep learning: The first Bird Audio Detection challenge". In: *Methods in Ecology and Evolution* 10.3 (2019), pp. 368–380. DOI: https://doi.org/10.1111/2041-210X.13103.

[258] Dan Stowell et al. "Detection and classification of acoustic scenes and events". In: *IEEE Transactions on Multimedia* 17.10 (2015), pp. 1733–1746.

[259] J. Sueur, T. Aubin, and C. Simonis. "Seewave: a free modular tool for sound analysis and synthesis". In: *Bioacoustics* 18 (2008), pp. 213–226. URL: https://www.tandfonline.com/doi/abs/10.1080/09524622.2008.9753600.

[260] Larissa Sayuri Moreira Sugai et al. "Terrestrial Passive Acoustic Monitoring: Review and Perspectives". In: *BioScience* 69.1 (Nov. 2018), pp. 15–25. ISSN: 0006-3568. DOI: 10.1093/biosci/biy147.

[261] Yuren Sun et al. "Classification of animal sounds in a hyperdiverse rainforest using convolutional neural networks with data augmentation". In: *Ecological Indicators* (Dec. 2022). ISSN: 1470-160X. DOI: 10.1016/j.ecolind.2022.109621.

[262] Sustainable Energy Authority of Ireland. *Energy in Ireland 2018 Report*. https://www.seai.ie/sites/default/files/publications/Energy-in-Ireland-2018.pdf. Accessed: 2025-07-25. Dec. 2018.

[263] Sustainable Energy Authority of Ireland. *Wind Energy Roadmap 2011-2050*. https://www.seai.ie/sites/default/files/publications/Wind_Energy_Roadmap_2011-2050.pdf. Accessed: 2025-07-25. 2011.

[264] Richard S. Sutton. *The Bitter Lesson*. http://www.incompleteideas.net/IncIdeas/BitterLesson.html. Accessed: 2025-07-17. Mar. 2019.

[265] Laszlo Talas et al. "CamoGAN: Evolving optimum camouflage with Generative Adversarial Networks". In: *Methods in Ecology and Evolution* 11.2 (2020), pp. 240–247. DOI: 10.1111/2041-210X.13334.

[266] Yang Tang. *shinyjqui: 'jQuery UI' Interactions and Effects for Shiny*. R package version 0.4.1. 2022. URL: https://CRAN.R-project.org/package=shinyjqui.

[267] PyTorch Team. *PyTorch*. Version 2.0.1+cu117. 2023. URL: https://pytorch.org/.

[268] TEEB - The Economics of Ecosystems and Biodiversity. *The Economics of Ecosystems and Biodiversity: Mainstreaming the Economics of Nature: A Synthesis of the Approach, Conclusions and Recommendations of TEEB*. https://library.wur.nl/WebQuery/titel/1948271. Accessed: 2025-07-25. Bonn, Oct. 2010.

[269] Chris B. Thaxter et al. "Bird and bat species' global vulnerability to collision mortality at wind farms revealed through a trait-based assessment". In: *Proceedings of the Royal Society B: Biological Sciences* 284.1862 (Sept. 2017), p. 20170829. ISSN: 1471-2954. DOI: 10.1098/rspb.2017.0829. URL: http://dx.doi.org/10.1098/rspb.2017.0829.

[270] Mark Thomas et al. *Marine Mammal Species Classification using Convolutional Neural Networks and a Novel Acoustic Representation*. 2019. DOI: 10.48550/ARXIV.1907.13188. URL: https://arxiv.org/abs/1907.13188.

[271] Alicia Toon, Lyn G. Cook, and Michael D. Crisp. "Evolutionary consequences of shifts to bird-pollination in the Australian pea-flowered legumes (Mirbelieae and Bossiaeeae)". In: *BMC Evolutionary Biology* 14.1 (Mar. 2014), p. 43. ISSN: 1471-2148. DOI: 10.1186/1471-2148-14-43.

[272] Martino Trapanotto et al. "Convolutional Neural Networks for the Identification of African Lions from Individual Vocalizations". In: *Journal of Imaging* (2022). ISSN: 2313-433X. DOI: 10.3390/jimaging8040096.

[273] Yongyut Trisurat, Piyathip Eawpanich, and Risto Kalliola. "Integrating land use and climate change scenarios and models into assessment of forested watershed services in Southern Thailand". In: *Environmental Research* 147 (2016), pp. 611–620. ISSN: 0013-9351. DOI: https://doi.org/10.1016/j.envres.2016.02.019.

[274] *System of Environmental Economic Accounting: Ecosystem Accounting*. https://seea.un.org/ecosystem-accounting. Date Accessed: 2023-02-07.

[275] Benigno Uria et al. *Neural Autoregressive Distribution Estimation*. 2016. arXiv: 1605.02226 [cs.LG]. URL: https://arxiv.org/abs/1605.02226.

[276] Benjamin M. Van Doren et al. "Nighthawk: Acoustic monitoring of nocturnal bird migration in the Americas". In: *Methods in Ecology and Evolution* 15.2 (Dec. 2023), pp. 329–344. ISSN: 2041-210X. DOI: 10.1111/2041-210x.14272.

[277] R. Varun Prakash et al. "Multi-level LSTM framework with hybrid sonic features for human-animal conflict evasion". In: *The Visual Computer* (Aug. 2024). ISSN: 1432-2315. DOI: 10.1007/s00371-024-03588-9.

[278] B. Vimal et al. "MFCC Based Audio Classification Using Machine Learning". In: *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. 2021, pp. 1–4. DOI: 10.1109/ICCCNT51525.2021.9579881.

[279] Pascal Vincent et al. “Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion”. In: *Journal of Machine Learning Research* 11.110 (2010), pp. 3371–3408. URL: http://jmlr.org/papers/v11/vincent10a.html.

[280] Jia Wang and Shigeru Tabeta. “Four-channel generative adversarial networks can predict the distribution of reef-associated fish in the South and East China Seas”. In: *Ecological Informatics* 78 (2023), p. 102321. ISSN: 1574-9541. DOI: 10.1016/j.ecoinf.2023.102321.

[281] Mou Wang et al. “Hybrid Constant-Q Transform Based CNN Ensemble for Acoustic Scene Classification”. In: *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. 2019, pp. 1511–1516. DOI: 10.1109/APSIPAASC47483.2019.9023236.

[282] Victoria E. Warren et al. “Passive Acoustic Monitoring Reveals Spatio-Temporal Distributions of Antarctic and Pygmy Blue Whales Around Central New Zealand”. In: *Frontiers in Marine Science* 7 (2021). ISSN: 2296-7745. DOI: 10.3389/fmars.2020.575257.

[283] Tom Weiss et al. “Guided Diffusion for Inverse Molecular Design”. In: *ChemRxiv* (2023). This content is a preprint and has not been peer-reviewed. DOI: 10.26434/chemrxiv-2023-z8ltp.

[284] Matthew J. Weldy et al. “Simulated soundscapes and transfer learning boost the performance of acoustic classifiers under data scarcity”. In: *Methods in Ecology and Evolution* n/a.n/a (June 2025). DOI: https://doi.org/10.1111/2041-210X.70089.

[285] Yuval Werber et al. “BATScan: A radar classification tool reveals large-scale bat migration patterns”. In: *Methods in Ecology and Evolution* 14.7 (2023), pp. 1764–1779. DOI: 10.1111/2041-210X.14125.

[286] Hadley Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016. ISBN: 978-3-319-24277-4. URL: https://ggplot2.tidyverse.org.

[287] Hadley Wickham. *stringr: Simple, Consistent Wrappers for Common String Operations*. R package version 1.4.0. 2019. URL: https://CRAN.R-project.org/package=stringr.

[288] Hadley Wickham et al. *dplyr: A Grammar of Data Manipulation*. R package version 1.0.7. 2021. URL: https://CRAN.R-project.org/package=dplyr.

[289] Wildlife Acoustics. *Kaleidoscope Pro: Bioacoustic Analysis Software*. Computer software. Version 5.4.8, Accessed: 25 March 2025. 2025. URL: https://www.wildlifeacoustics.com/products/kaleidoscope-pro.

[290] Wildlife Acoustics. *Song Meter Mini Bat*. Date Accessed: 2024-10-16. 2024. URL: https://www.wildlifeacoustics.com/.

[291] Gary J. Wiles et al. "Impacts of the Brown Tree Snake: Patterns of Decline and Species Persistence in Guam's Avifauna". In: *Conservation Biology* 17.5 (2003), pp. 1350–1360. DOI: 10.1046/j.1523-1739.2003.01526.x.

[292] Emma M Williams, Colin F J O'Donnell, and Doug P Armstrong. "Cost-benefit analysis of acoustic recorders as a solution to sampling challenges experienced monitoring cryptic species". en. In: *Ecol. Evol.* 8.13 (July 2018), pp. 6839–6848.

[293] Wind Energy Ireland. *Future Onshore Wind Study Report*. Tech. rep. Accessed: 2025-08-06. Wind Energy Ireland, Apr. 2025. URL: https://windenergyireland.com/images/files/241008-future-onshore-wind-study-report-f1-20250423.pdf.

[294] Wind Energy Ireland. *Protecting Consumers: Our onshore wind energy opportunity*. Tech. rep. Accessed: 2025-08-06. Wind Energy Ireland, Apr. 2025. URL: https://windenergyireland.com/images/files/241008-future-onshore-wind-study-report-f1-20250423.pdf.

[295] Connor M Wood et al. "Early detection of rapid Barred Owl population growth within the range of the California Spotted Owl advises the Precautionary Principle". In: *The Condor* 122.1 (Jan. 2020). ISSN: 1938-5129. DOI: 10.1093/condor/duz058.

[296] Heather Wood, Regina Lindborg, and Simon Jakobsson. "European Union tree density limits do not reflect bat diversity in wood-pastures". In: *Biological Conservation* 210 (2017), pp. 60–71. ISSN: 0006-3207. DOI: https://doi.org/10.1016/j.biocon.2017.04.001.

[297] Kevin A. Wood, Richard A. Stillman, and John D. Goss-Custard. "Co-creation of individual-based models by practitioners and modellers to inform environmental decision-making". In: *Journal of Applied Ecology* 52.4 (2015), pp. 810–815. DOI: https://doi.org/10.1111/1365-2664.12419.

[298] Lyndsie S. Wszola et al. "Translating statistical species-habitat models to interactive decision support tools". In: *PLoS ONE* 12 (12 Dec. 2017). ISSN: 19326203. DOI: 10.1371/journal.pone.0188244.

[299] Te-Lin Wu, Shao-Hua Sun, and Du Ang. *clvrai/ACGAN-PyTorch*. Mar. 29, 2018. URL: https://github.com/clvrai/ACGAN-PyTorch.

[300] Yihui Xie. *xfun: Supporting Functions for Packages Maintained by 'Yihui Xie'*. R package version 0.41. 2023. URL: https://CRAN.R-project.org/package=xfun.

[301] Yihui Xie, Joe Cheng, and Xianying Tan. *DT: A Wrapper of the JavaScript Library 'DataTables'*. R package version 0.20. 2021. URL: https://CRAN.R-project.org/package=DT.

[302] Tsutomu Yagihashi et al. "Eradication of the mongoose is crucial for the conservation of three endemic bird species in Yambaru, Okinawa Island, Japan". In: *Biological Invasions* 23.7 (Apr. 2021), pp. 2249–2260. ISSN: 1573-1464. DOI: 10.1007/s10530-021-02503-w.

[303] Fumio Yamada and Ken Sugimura. "Negative Impact of an Invasive Small Indian Mongoose *Herpestes javanicus* on Native Wildlife Species and Evaluation of a Control Project in Amami-Ohshima and Okinawa Islands, Japan". In: *Global Environmental Research* 8.2 (Jan. 2004), pp. 117–124. URL: https://airies.wikiplus.net/journal_08-1eng.html.

[304] Fumio Yamada et al. “Rediscovery After Thirty Years Since the Last Capture of the Critically Endangered Okinawa Spiny Rat *Tokudaia muenninki* in the Northern Part of Okinawa Island”. In: *Mammal Study* 35.4 (Dec. 2010), pp. 243–255. ISSN: 1348-6160. DOI: 10.3106/041.035.0404.

[305] Myat Su Yin et al. “A Lightweight Deep Learning Approach to Mosquito Classification from Wingbeat Sounds”. In: *Proceedings of the Conference on Information Technology for Social Good*. GoodIT ’21. Roma, Italy: Association for Computing Machinery, 2021, pp. 37–42. ISBN: 9781450384780. DOI: 10.1145/3462203.3475908.

[306] Hongyi Zhang et al. *mixup: Beyond Empirical Risk Minimization*. 2018. arXiv: 1710.09412 [cs.LG]. URL: https://arxiv.org/abs/1710.09412.

[307] Huan Zhang, Si Si, and Cho-Jui Hsieh. *GPU-acceleration for Large-scale Tree Boosting*. 2017. URL: https://arxiv.org/abs/1706.08359.

[308] Qiang Zhang et al. “Generating Adversarial Examples in Audio Classification with Generative Adversarial Network”. In: *2022 7th International Conference on Image, Vision and Computing (ICIVC)*. 2022, pp. 848–853. DOI: 10.1109/ICIVC55077.2022.9886154.

[309] Tao Zhang et al. “Acoustic scene classification based on Mel spectrogram decomposition and model merging”. In: *Applied Acoustics* 182 (2021), p. 108258. ISSN: 0003-682X. DOI: https://doi.org/10.1016/j.apacoust.2021.108258.

[310] Zhichao Zhang et al. *Deep Convolutional Neural Network with Mixup for Environmental Sound Classification*. 2018. arXiv: 1808.08405 [cs.SD]. URL: https://arxiv.org/abs/1808.08405.

[311] Xuzhe Zhao et al. “Additive, antagonistic and synergistic interactions of future climate and land use change on Theaceae species assemblages in China”. In: *Global Ecology and Conservation* 48 (2023), e02750. ISSN: 2351-9894. DOI: https://doi.org/10.1016/j.gecco.2023.e02750.

[312] Zehui Zhao et al. "A comparison review of transfer learning and self-supervised learning: Definitions, applications, advantages and limitations". In: *Expert Systems with Applications* 242 (2024), p. 122807. ISSN: 0957-4174. DOI: 10.1016/j.eswa.2023.122807.

[313] Hristo Zhivomirov. "On the Development of STFT-analysis and ISTFT-synthesis Routines and their Practical Implementation". In: *TEM Journal* 8.1 (2019), pp. 56–64. URL: https://www.temjournal.com/content/81/TEMJournalFebruary2019_56_64.pdf.

[314] Ming Zhong et al. "Beluga whale acoustic signal classification using deep learning neural network models". In: *The Journal of the Acoustical Society of America* 147.3 (Mar. 2020), pp. 1834–1841. ISSN: 0001-4966. DOI: 10.1121/10.0000921.

[315] Ming Zhong et al. "Multispecies bioacoustic classification using transfer learning of deep convolutional neural networks with pseudo-labeling". In: *Applied Acoustics* 166 (2020), p. 107375. ISSN: 0003-682X. DOI: https://doi.org/10.1016/j.apacoust.2020.107375.

[316] George Zhou, Yunchan Chen, and Candace Chien. "On the analysis of data augmentation methods for spectral imaged based heart sound classification using convolutional neural networks". In: *BMC Medical Informatics and Decision Making* 22.1 (Aug. 2022), p. 226. ISSN: 1472-6947. DOI: 10.1186/s12911-022-01942-2.

[317] Z.H. Zhou. *Ensemble Methods: Foundations and Algorithms*. CHAPMAN & HALL/CRC MACHINE LEA. Taylor & Francis, 2012. ISBN: 9781439830031. URL: https://books.google.ie/books?id=BDB50Ev2ur4C.

[318] Sándor Zsebők et al. "Automatic bird song and syllable segmentation with an open-source deep-learning object detection method – a case study in the Collared Flycatcher". In: *Ornis Hungarica* 27.2 (2019), pp. 59–66. DOI: doi:10.2478/orhu-2019-0015.

[319] Rémy Zulko et al. *MoviePy: Video editing with Python*. Version 2.0. 2025. URL: https://github.com/Zulko/moviepy.