The Journal of the Acoustical Society of America

JANUARY 15 2026

PAMGuard: Application software for passive acoustic detection, classification, and localisation of animal sounds

Douglas Gillespie ; Jamie Macaulay ; Michael Oswald ; Marie Roch

J. Acoust. Soc. Am. 159, 437–443 (2026)
https://doi.org/10.1121/10.0042245

JASA ARTICLE

PAMGuard: Application software for passive acoustic detection, classification, and localisation of animal sounds

Douglas Gillespie,[1,a)] Jamie Macaulay,[1] Michael Oswald,[1] and Marie Roch[2]
[1]Sea Mammal Research Unit, Scottish Oceans Institute, University of St Andrews, St Andrews, United Kingdom
[2]Department of Computer Science, San Diego State University, San Diego, California 92182, USA

ABSTRACT:
Detection, classification, and localisation of animal sounds are essential in many ecological studies, including density estimation and behavioural studies. Real-time acoustic processing can also be used in mitigation exercises, with the possibility of curtailing harmful human activities when animals are present. Animal vocalisations vary widely, and there is no single detection algorithm that can robustly detect all sound types. Human-in-the loop analysis is often required to validate algorithm performance and deal with unexpected noise sources such as are often encountered in real-world situations. The PAMGuard software combines advanced automatic analysis algorithms, including AI methods, with interactive visual tools allowing users to develop efficient workflows for both real-time use and for processing archived datasets. A modular framework enables users to configure multiple detectors, classifiers, and localisers suitable for the equipment and species of interest in a particular application. Multiple detectors for different sound types can be run concurrently on the same data. An extensible "plug-in" interface also makes it possible for third parties to independently develop new modules to run within the software framework. Here, we describe the software's core functionality, illustrated using workflows for both real-time and offline use, and present an update on the latest features.
 https://doi.org/10.1121/10.0042245

(Received 12 September 2025; revised 22 December 2025; accepted 29 December 2025; published online 15 January 2026)
[Editor: Kaitlin E. Frasier] Pages: 437–443

I. INTRODUCTION

Sound is produced by many animal species, often as a means of communication and, for some species, echolocation. Detecting these sounds provides researchers with an additional avenue to study animal behaviour (e.g., Gillespie *et al.*, 2021; Macaulay *et al.*, 2022), estimate animal density and distribution (Gordon *et al.*, 2020; Gridley *et al.*, 2020; Westell *et al.*, 2022), efficiently find animals for visual observation (e.g., Brotons *et al.*, 2024; Miller *et al.*, 2016) and to detect the presence of different species in real-time for mitigation of harmful anthropogenic activities (e.g., Palmer *et al.*, 2021a). The broad methodology of using received sounds to study animals is termed passive acoustic monitoring (PAM). PAM can be advantageous in that sounds can be detected when visual observation is not possible: at night, during inclement weather, in dense forest environments, or in the case of marine mammals when they are submerged. Automatic PAM software allows detection of sounds outside the human hearing range, can process many channels of data simultaneously, extract measurements from detected sounds, and can localise sounds in two or three dimensions if a suitable sensor array is deployed. Automation also allows for consistency in analysis and more rapid data processing, particularly important for large datasets, where the possibility of listening to all data is impractical.

Animal sounds, particularly those of marine mammals, are highly variable. Cetacean vocalisations occupy over 13 octaves of the audio spectrum (four orders of magnitude), from infrasonic 15 Hz calls of blue whales to 130 kHz echolocation clicks of harbour porpoises. Similarly, the duration of calls can vary from tens of microseconds for dolphin echolocation clicks to over ten seconds for some baleen whale calls. Soundscapes often contain a multitude of interfering sounds which may come from other animal species, environmental noise (e.g., rain, wind, waves, sediment movement), or anthropogenic activities. In addition, propagation can change the nature of received sounds in many ways, including high-frequency components attenuating faster than low-frequency ones and normal mode dispersion. These complex environments, where both interfering sounds and vocal behaviour may evolve and change over time and space, can lead to errors and misclassifications of sounds when utilising automatic algorithms, especially when the algorithms are used outside of the context or domain for which they were developed or trained.

PAM studies not only target a wide range of species, but often have different study aims, and different hardware, with varying numbers and configurations of hydrophones. Systems might be moored, installed on landers, or towed behind crewed or autonomous vessels. Thus, analysis of PAM data requires software which can handle a large

a)Email: dg50@st-andrews.ac.uk

variety of hardware types and enables the effective use of automated analysis algorithms along with human validation and annotations as necessary.

PAMGuard was developed as a general tool for the detection, classification, and localisation of marine mammal vocalisations, particularly those of cetaceans, although it can also be used in the terrestrial environment. The software combines state-of-the-art automatic algorithms with powerful data visualisation tools to allow a human-in-the-loop validation. Automated algorithms do the bulk of the processing work but allow a human to make final decisions either at the call or encounter level, or to examine a sample of the data to assess detector performance and provide correction factors for parameters derived from the data. PAMGuard algorithms include click detectors (Gillespie *et al.*, 2020), whistle detectors and classifiers (Gillespie *et al.*, 2013), energy detectors (Helble *et al.*, 2012), correlation detectors (Mellinger and Clark, 2000), image-based classifiers (Gillespie, 2004), and localisation algorithms (Gillespie *et al.*, 2020; Gillespie and Macaulay, 2019). The software uses a modular architecture, whereby the user loads modules required to create a custom workflow appropriate for their study goals, hardware, and species of interest. This will typically include acoustic modules, for example, creating a spectrogram, detecting clicks, and running deep learning classifiers, as well as general utilities such as data management, acquisition of navigation [global positioning system (GPS) and automatic identification system (AIS)] and array orientation data, and modules for human data input and annotation.

A graphical user interface (GUI) allows users to configure modules, set up their acoustic workflow, and access data visualisation tools without any need for coding. This combination of a comprehensive visual interface and access to a large number of automated algorithms enables users to efficiently and accurately process large quantities of acoustic data in real-time or post data collection. Programming using Java means that the software can run on a standard consumer desktop or laptop, and makes PAMGuard cross-platform; it will run on most Linux, Apple Macintosh and Microsoft Windows platforms. Currently, installers are provided for Windows and macOS with the Windows version also including libraries to run high-frequency and multi-channel sound acquisition systems. Simple configurations have also been run on low-power advanced RISC maching (ARM) Linux systems on remote moorings (King *et al.*, 2022). The software is targeted at users with a wide range of acoustic expertise. Once configured, a novice user can be easily trained to monitor the software, perform basic quality control tasks and respond to detections. Additional knowledge of bioacoustics is required to select and configure detectors for a specific application, and, as with any toolbox, a higher-level expertise is required for more complex tasks such as three-dimensional (3D) localisation using multi-hydrophone arrays.

The software was first released in 2006 (Gillespie *et al.*, 2008), primarily to address the needs of the hydrocarbon industries to provide mitigation for the harmful effects of noise generated during seismic surveys (Nowacek *et al.*, 2013). At the time of its first release, it contained 29 modules to perform data acquisition, signal processing, detection, and data display tasks and only functioned as a real time detection tool. The number of modules in the software is now 70, and an updated data management and display system allows for offline review of processed data and supports analysis and viewing of data from autonomous acoustic devices such as SoundTraps (www.oceaninstruments.com/) and HydroMoths (www.openacousticdevices.info). In addition, the software supports external plugin modules, which can be developed by third parties and run within PAMGuard with no requirement to modify the core source code. The latest release, Version 2.02.17, contains three major developments: An interface to the Tethys spatial-temporal database for Passive Acoustic Data (Roch *et al.*, 2016), a module that can run artificial intelligence (AI) models, and a batch processing system which automates and parallelises the analysis of multiple datasets, for example, collected with arrays of autonomous recorders. Here, we describe the core functionality and structure of the software alongside these latest additions.

II. IMPLEMENTATION AND ARCHITECTURE

The software can be split into Core functionality, which is used by all modules, and the modules themselves. The Core provides services such as data management, temporal and spatial displays, and configuration management. Modules build on the core functionality to provide signal processing, detection, classification, and localisation tasks, as well as additional bespoke displays as required. Standardisation of the formats and interfaces for these modules' data inputs and outputs makes it possible to interconnect modules to support a passive acoustic data analysis appropriate to specific study goals. Modules use standard classes to provide as much functionality as possible, minimising the amount of new code that needs to be written for each module. New modules can be developed, and their output incorporated into existing displays and data output streams without any need to modify core functionality.

A. GUI framework

The flexible GUI allows users to interact with data both in real time and during offline data analysis (Fig. 1). Many different types of displays may be required for a multi-species study. The "Data Map" provides a high-level overview of when data were recorded and summary output from each module or algorithm. Other types of displays include multiple spectrogram displays for different parts of the spectrum, click displays containing scrolling plots of click bearing against time, lists of detections, maps of the vessel track overlayed with detections, etc. These displays can cover vastly different time periods, from the Data Map showing weeks or years of summarised data, to click waveform displays showing milliseconds of data.

https://doi.org/10.1121/10.0042245

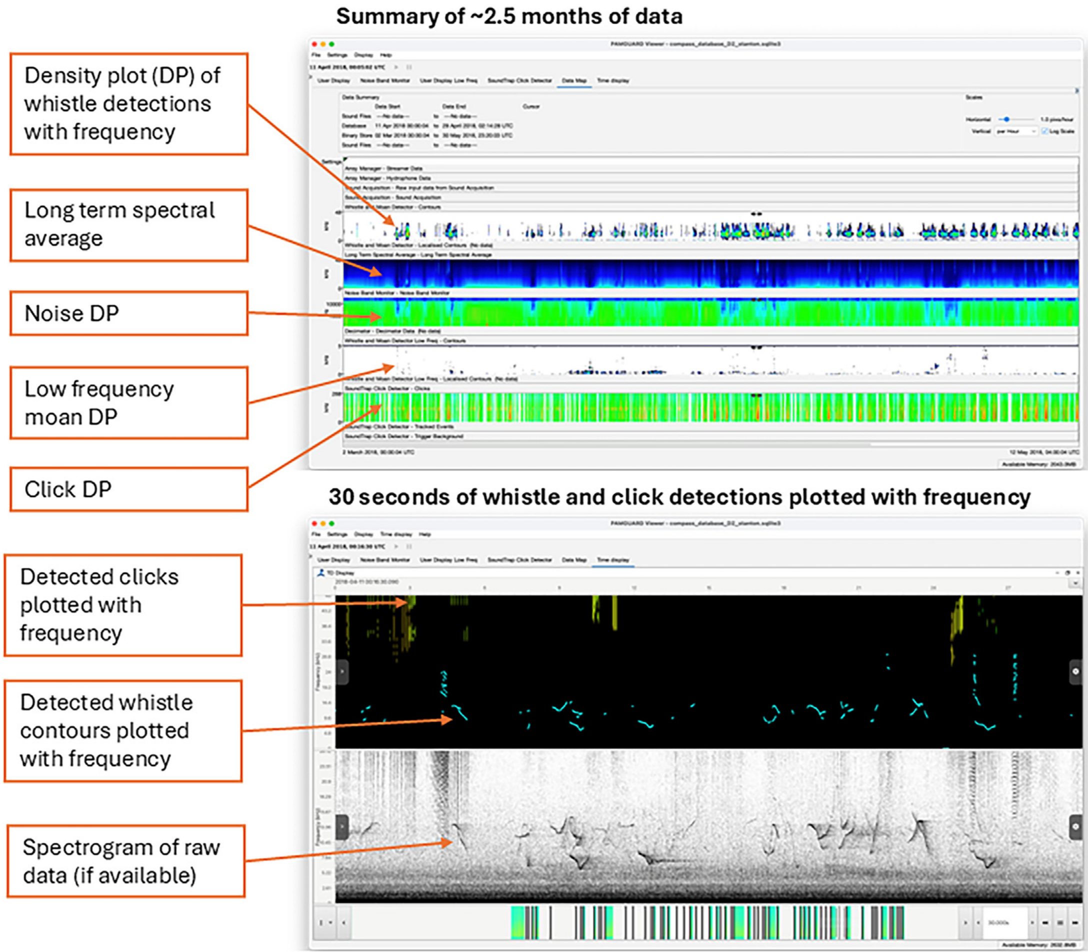

FIG. 1. Example of the GUI configured to show: Top, a data map showing weeks of detection data; Bottom, a frequency time display showing detected dolphin whistles and clicks on the top panel and a spectrogram on the bottom panel.

B. Data management and storage

Modules typically take an input, act on it (e.g., transform, detect, classify, or localise), and produce some type of output. A module's input can be from the outside world (from an audio input device or file, GPS data, depth sensor data, etc.) or from other modules. An observer / observable design pattern (Freeman and Freeman, 2004) was used so that data from one module can be shared by multiple downstream modules: for instance, raw audio time series data can be used directly by the click detector, and also to compute a spectrogram. The spectrogram data might be used both by a display and by a tonal sound detector. Each module processes data in a separate program thread, thereby efficiently using multiple processor cores.

The software uses two forms of external data storage. A SQL database is the preferred option for data that is produced at a relatively low rate and has a fixed record format, such as human annotations or GPS navigation data. However, higher volume data, and data that do not have a fixed record format, such as detected clicks (which are stored with a few hundred samples of raw audio data) and whistles (which are stored with the time-frequency contour of the detected sound), do not store efficiently in a SQL database. For these types of data, we generate bespoke binary data files. Separate binary files are generated for each module, and a new file for each module is generally started once per hour (or once per sound file for post-processing data). A catalogue of files is also created so that data can be located and reloaded efficiently during offline data review.

Detection data can be exported into file formats supported by MATLAB®, R, and Python. Software libraries also exist for all these languages that can read the binary data files directly. These higher-level languages are generally more accessible to many researchers, allowing them to develop workflows and process data beyond what is possible within PAMGuard itself.

C. Module functionality

The standard PAMGuard build includes the core functionality and 70 built in modules. The architecture provides mechanisms to let users interact with modules through menu items and to show their output on standard displays (e.g., maps, spectrograms, plots vs time), where the user can view and annotate detections. Many modules have a logging component which writes to the aforementioned SQL database or binary format files. Finally, modules have a documentation interface, which is used in the software's help system. Modules need only implement the interfaces that are relevant. For example, the fast Fourier transform (FFT) module only uses input and output to other modules (such as displays or detectors), documentation, and user controls for configuring transform parameters, whereas the click detector has its own bespoke display and implements interfaces that allow it to interact with a 3D localisation module.

A list of current modules with details on their available functionality is available in the supplementary material and at https://www.pamguard.org/coremodules.html.

D. Plugin modules

The module architecture is designed to let third parties develop so-called "plugin" modules for their own work. Users can create modules that conform to the Java interface for modules and package their code into a module that PAMGuard will load in the same manner as any other module, thus "plugging in" new functionality. Documents on how to create plug-in modules are on the PAMGuard website. Examples include a module for measuring the inter-pulse interval within sperm whale clicks (Miller *et al.*, 2013), modules for acquiring and processing imaging sonar data (Gillespie *et al.*, 2023), and a tool developed for monitoring the vocal behaviour and welfare of captive cetaceans (Jones *et al.*, 2021).

III. RECENT ADDITIONS

A. Tethys database

PAMGuard requires separate data stores (SQL database and a folder of binary files) for each equipment deployment or cruise. i.e., if someone deployed 20 autonomous recorders on a project, they would have 20 SQL databases and 20 folders of binary data. To facilitate data analysis over large temporal and spatial scales, the software can now export multiple datasets to a single, large scale, temporal spatial database, Tethys, described in Roch *et al.* (2016) and available at https://zenodo.org/records/13626338. This database captures information about characteristics of instrument deployments (e.g., sample rate, channel, calibration) and derived information such as detections of calls or position of calling animals. Access to the database is available via graphical web tools or programmatic interfaces (e.g., MATLAB, R, Java, and Python).

B. AI model support

Many research groups are developing AI-based solutions for animal sound detection, some clearly showing marked improvement over previous, more heuristic methods (e.g., Shiu *et al.*, 2020). However, many of these AI models are inaccessible to most users, existing only as Python code and a trained model file. The PAMGuard Deep Learning module allows users to run deep learning models (e.g., based on Tensorflow or PyTorch) within PAMGuard, making them accessible to non-programmers with minimal work by the model developer.

The deep learning module allows users to place AI detectors and classifiers at the heart of their workflow and integrates with the software's visualisation tools, allowing for manual checking and validation of model output. The deep learning module can accept data from raw audio producers or from detection modules. Running a simple detector as a filter to the input of an AI model can greatly reduce the computational load of the model, e.g., Brinkløv *et al.* (2023) used the click detector module to detect bat calls at a high false positive rate and then classified each call to species or "not a bat call" using a Convolutional Neural Network (CNN) based classifier (Bergler *et al.*, 2019). The initial "simple" call detector stage greatly reduced the volume of data processed by the more computationally intensive CNN model, increasing analysis speed by an order of magnitude.

As well as accepting data from a variety of upstream modules, the deep learning module can also pass results to downstream modules, for example, to localisation modules, allowing users to more accurately acquire positional data for target species.

C. Batch processing

Users often need to process multiple data sets with the same PAMGuard configuration, particularly when arrays of autonomous recorders are deployed. A PAMGuard configuration requires the user to enter the path to the input audio data, the name of the output database and a path to the folder for binary data storage. Editing these three paths, then running the software multiple times, was onerous and prone to human error.

The batch processor module allows further automation of the processing of large data sets. The module takes a single processing configuration and applies it to multiple folders of audio files, creating a SQL database and a set of binary storage files for each. A scheduling manager takes advantage of modern multiple-core processors. It oversees the application of the processing configuration to each of the datasets, permitting a user-configurable maximum number of jobs to execute in parallel. With even moderate processor resources, it is usually reasonable to execute between two and four jobs concurrently, and new jobs are started as current ones complete.

IV. WORKFLOWS AND USAGE EXAMPLES

The software's modular structure makes it extremely flexible, with many possible use-cases and configurations. A common use-case is to track marine mammals in real time (Fig. 2), often used in a mitigation context or for finding animals for visual studies. For example, when tracking sperm whales from a vessel using a towed hydrophone, the click detector automatically detects clicks in real time, calculates a bearing to each click based on time of arrival differences on a pair of hydrophones, and displays each click on a display of bearing against time. The operator then interacts with the scrolling bearing-time display to ignore false detections (e.g., caused by the vessel's propeller) and to select

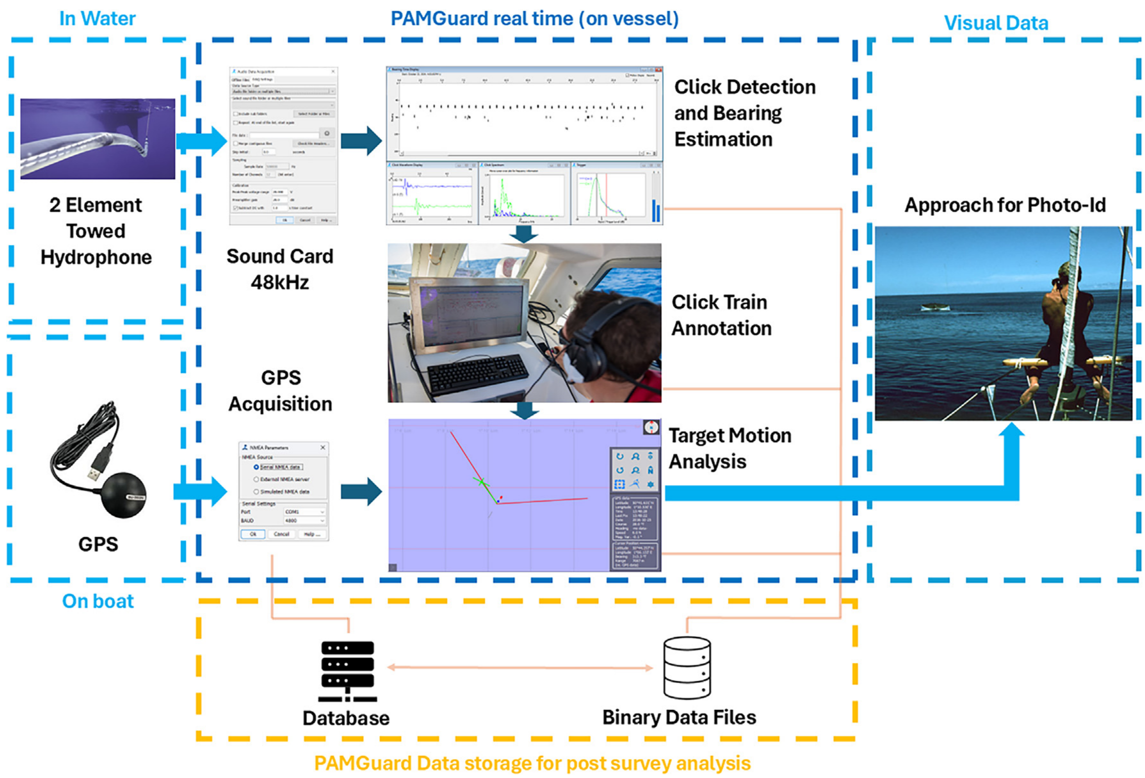

FIG. 2. Real-time workflow: In this example, sperm whale clicks are detected in real time and an operator then selects clicks forming clear click trains, which then go to a target motion localiser. The operator can then make a decision to approach the animal for visual observation or could request the shutdown of a harmful activity.

clicks forming clear lines of bearings (often referred to as a "click train") that can be assigned to individuals or closely spaced groups of whales. As each click is selected, a localisation algorithm combines all of the bearings to clicks within a click train with the vessel's GPS data to estimate the animal's location using target motion analysis (Webber *et al.*, 2022) and displays that location on the PAMGuard map. The operator can then make an operational decision to approach the animal (e.g., for visual study) or request the shutdown of a harmful activity during a mitigation exercise.

During density estimation studies, a similar approach is used to track whales and measure distances from the trackline to vocalising animals in order to estimate survey strip width, or a detection function (Buckland *et al.*, 2001; Webber *et al.*, 2022), although for this application, analysis can be conducted offline.

PAMGuard's semi-automated approach to data analysis can also be applied to post-processing of large datasets, such as acoustic recording devices (e.g., SoundTraps, AudioMoths) or long-term monitoring efforts. For example, Palmer *et al.* (2021b) ran PAMGuard in real time, sampling 12 channels of data at 500 kS/s per channel for a period of 451 days (see Fig. 3) at an operational tidal turbine, streaming raw audio data to an onshore computer via optical fibre from a bespoke acquisition system (Gillespie *et al.*, 2020). By running the software in real time, 1 TB of raw data per day could be discarded, storing only the binary data files, which had less than 1% of the original raw data volume. They then scrolled through the binary data using the software's GUI to identify and mark harbour porpoise encounters, which were subsequently processed with the 3D localisation algorithm to provide tracks of animals. These tracks were then exported to a bespoke MATLAB viewer for a final assessment of collision risk with the turbine rotor.

The level of automation/user interaction in a PAM study varies, from using simple algorithms to detect possible sounds and then using a human expert to annotate and classify species as in Fig. 3, to running a deep learning model that may be highly accurate, requiring only that a human verifies a small subset of data. In both cases, the software's visual displays allow users to check for inconsistencies and validate or mark interesting sections of data across time scales from years to microseconds (see Fig. 1).

Once a researcher is confident that the processing configuration is appropriate for their research question, the libraries and export functionality described in Sec. II B allows them to continue their data analysis outside PAMGuard in whichever external software they please. For example, Rankin *et al.* (2024) describe an open source package written in R (Sakai and Oswald, 2024), which uses PAMGuard detections as input and provides additional species classification pertinent to their study areas.

V. CONCLUSION

The PAMGuard software provides a flexible and accessible toolbox for a variety of PAM tasks in both marine and terrestrial environments. It makes sophisticated detection, classification, and localisation algorithms readily accessible

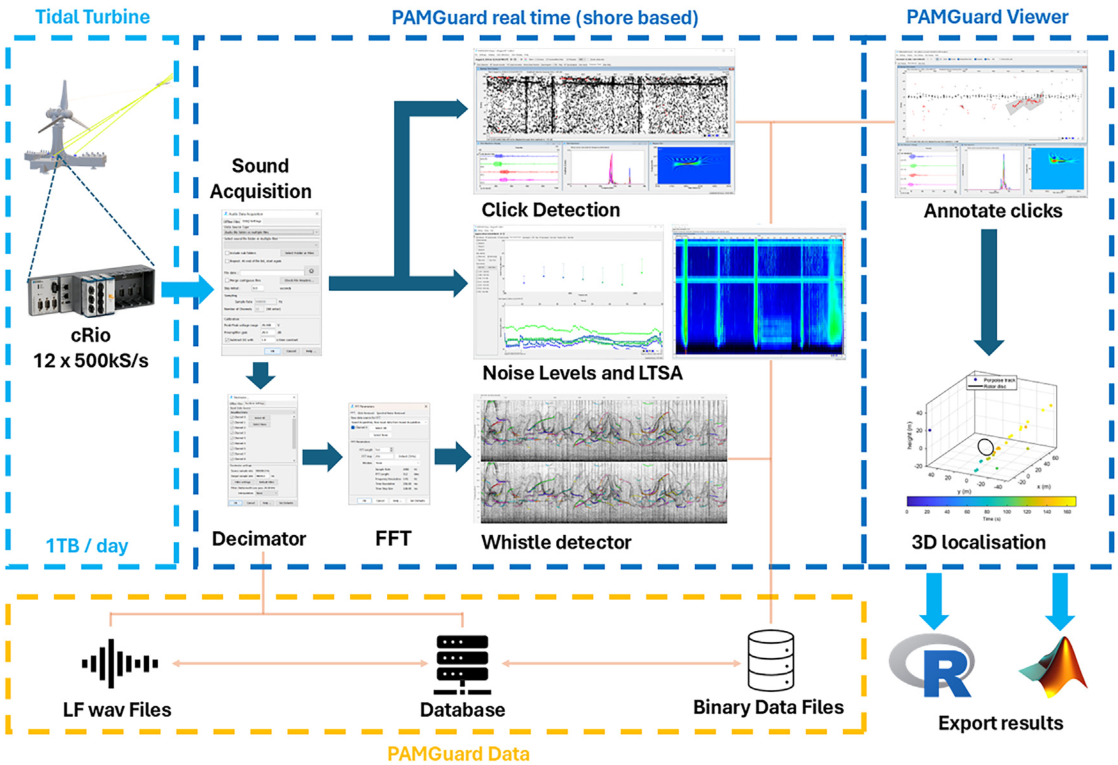

FIG. 3. An acoustic workflow which processed 12 channels of 500 kS/s sampled data for porpoise and dolphin clicks, noise levels and dolphin whistles. The software initially ran in real time, processing 1 TB of acoustic data per day and saving it to the data management system, reducing data volume by orders of magnitude. Data were then reviewed offline, events selected, and further processed using both PAMGuard modules and bespoke MATLAB code.

to non-acoustics specialists with no programming knowledge. Its modular framework enables users to configure a PAM system suitable for the hardware and species of interest to a specific study.

The software maintains a human-in-the-loop approach, though the degree of human effort required for different studies varies widely, from a quick check of a subset of data to a detailed selection of individual sounds. The modular framework makes it possible for developers to incorporate new algorithms into the software framework, and the Deep Learning module allows AI models developed in other languages, such as Python, to be run within PAMGuard without any need to write or modify PAMGuard source code. R, MATLAB®, and Python libraries give access to PAMGuard datafiles, allowing researchers to incorporate PAMGuard into more complex workflows.

SUPPLEMENTARY MATERIAL

See the supplementary material for a list of the 70 PAMGuard modules currently available, along with the data input requirements of each module, its output, data storage, and display options.

ACKNOWLEDGMENTS

Original PAMGuard funding was provided by the Sound and Marine Life OGP Joint Industry Program. Recent developments have been funded by the Bureau of Ocean Energy Management (BOEM) contract 140M0122C0006 and Cooperative Agreement M22AC00019-00 and from the U.S. Navy Living Marine Resources programme N3943020C2202. We are also grateful to two anonymous reviewers for their helpful comments on an earlier draft of this manuscript.

AUTHOR DECLARATIONS

Conflict of Interest

All authors declare that they have no conflicts of interest.

Ethics Approval

Data used in workflow examples were approved by the University of St Andrews School of Biology Ethics Committee (Reference No. SEC18014).

DATA AVAILABILITY

Source code is available under the GPL3 open source licence at https://github.com/PAMGuard/PAMGuard and at https://zenodo.org/records/15593810. Built installers are available for Windows and Mac at www.pamguard.org. Tutorials and links to sample data files are available at https://www.pamguard.org/tutorials.html.

Bergler, C., Schröter, H., Cheng, R. X., Barth, V., Weber, M., Nöth, E., Hofer, H., and Maier, A. (**2019**). "ORCA-SPOT: An automatic killer whale sound detection toolkit using deep learning," Sci. Rep. **9**, 10997.

Brinkløv, S. M. M., Macaulay, J., Bergler, C., Tougaard, J., Beedholm, K., Elmeros, M., and Madsen, P. T. (**2023**). "Open-source workflow approaches to passive acoustic monitoring of bats," Methods Ecol. Evol. **14**, 1747–1763.
Brotons, J.-M., Cerdà, M., Pirotta, E., and Rendell, L. (**2024**). "New data on the distribution of sperm whales in the Western Mediterranean highlights insufficiency of currently proposed protection measures," Aquat. Conserv. Mar. Freshwater Ecosyst. **34**, e70020.
Buckland, S. T., Anderson, D. R., Burnham, K. P., Laake, J. L., Borchers, D. L., and Thomas, L. (**2001**). *Introduction to Distance Sampling: Estimating Abundance of Biological Populations* (Oxford University Press, Oxford, UK).
Freeman, E., and Freeman, E. (**2004**). *Head First Design Patterns*, 1st ed. (O'Reilly, Sebastopol, CA).
Gillespie, D. (**2004**). "Detection and classification of right whale calls using an edge detector operating on a smoothed spectrogram," Can. Acoust. **32**, 39–47, available at https://jcaa.caa-aca.ca/index.php/jcaa/article/view/1586.
Gillespie, D., Caillat, M., Gordon, J., and White, P. (**2013**). "Automatic detection and classification of odontocete whistles," J. Acoust. Soc. Am. **134**, 2427–2437.
Gillespie, D., Hastie, G., Montabaranom, J., Longden, E., Rapson, K., Holoborodko, A., and Sparling, C. (**2023**). "Automated detection and tracking of marine mammals in the vicinity of tidal turbines using multibeam sonar," J. Mech. Sci. Eng. **11**, 2095.
Gillespie, D., and Macaulay, J. (**2019**). "Time of arrival difference estimation for narrow band high frequency echolocation clicks," J. Acoust. Soc. Am. **146**, EL387–EL392.
Gillespie, D., Mellinger, D., Gordon, J., McLaren, D., Redmond, P., McHugh, R., Trinder, P., Deng, X.-Y., and Thode, A. (**2008**). "PAMGUARD: Semiautomated, open source software for real-time acoustic detection and localisation of cetaceans," J. Acoust. Soc. Am. **125**, 2547.
Gillespie, D., Palmer, L., Macaulay, J., Sparling, C., and Hastie, G. (**2020**). "Passive acoustic methods for tracking the 3D movements of small cetaceans around marine structures," PLoS One **15**(5), e0229058.
Gillespie, D., Palmer, L., Macaulay, J., Sparling, C., and Hastie, G. (**2021**). "Harbour porpoises exhibit localized evasion of a tidal turbine," Aquat. Conserv. Mar. Freshwater Ecosyst. **31**, 2459–2468.
Gordon, J., Gillespie, D., Leaper, R., Lee, A., Porter, L., O'Brien, J., Meade, R., Cadhla, O., and Berrow, S. (**2020**). "A first acoustic density estimate for sperm whales in Irish offshore waters," J. Cetacean Res. Manage. **21**, 123–133.
Gridley, T., Martin, M. J., Slater, J., Roux, J., Swift, R., and Elwen, S. H. (**2020**). "Towed passive acoustic monitoring complements visual survey methods for Heaviside's dolphins *Cephalorhynchus heavisidii* in the Namibian Islands Marine Protected Area," Afr. J. Mar. Sci. **42**, 495–506.
Helble, T. A., Ierley, G. R., Gerald, L. D., Roch, M. A., and Hildebrand, J. A. (**2012**). "A generalized power-law detection algorithm for humpback whale vocalizations," J. Acoust. Soc. Am. **131**, 2682–2699.
Jones, B. L., Oswald, M., Tufano, S., Baird, M., Mulsow, J., and Ridgway, S. H. (**2021**). "A system for monitoring acoustics to supplement an animal welfare plan for bottlenose dolphins," J. Zool. Bot. Gard. **2**, 222–233.
King, P., Palmer, K., Tabbutt, S., and Wood, J. (**2022**). *Coastal Acoustic Buoy for Offshore Wind: Project Synthesis* (SMRU Consulting, Friday Harbor, WA).
Macaulay, J., Kingston, A., Coram, A., Oswald, M., Swift, R., Gillespie, D., and Northridge, S. (**2022**). "Passive acoustic tracking of the three-dimensional movements and acoustic behaviour of toothed whales in close proximity to static nets," Methods Ecol. Evol. **13**, 1250–1264.
Mellinger, D. K., and Clark, C. W. (**2000**). "Recognizing transient low-frequency whale sounds by spectrogram correlation," J. Acoust. Soc. Am. **107**, 3518–3529.
Miller, B. S., Calderan, S., Gillespie, D., Weatherup, G., Leaper, R., Collins, K., and Double, M. C. (**2016**). "Software for real-time localization of baleen whale calls using directional sonobuoys: A case study on Antarctic blue whales," J. Acoust. Soc. Am. **139**, EL83–EL89.
Miller, B. S., Growcott, A., Slooten, E., and Dawson, S. M. (**2013**). "Acoustically derived growth rates of sperm whales (*Physeter macrocephalus*) in Kaikoura, New Zealand," J. Acoust. Soc. Am. **134**, 2438–2445.
Nowacek, D. P., Bröker, K., Donovan, G., Gailey, G., Racca, R., Reeves, R. R., Vedenev, A. I., Weller, D. W., and Southall, B. L. (**2013**). "Responsible practices for minimizing and monitoring environmental impacts of marine seismic surveys with an emphasis on marine mammals," Aquat. Mamm. **39**, 356.
Palmer, K., Turner, J., Tabbutt, S., Gillespie, D., Thompson, J., King, P., and Wood, J. (**2021a**). "Evaluation of the Coastal Acoustic Buoy for offshore wind for real time mitigation of North Atlantic right whales," J. Acoust. Soc. Am. **150**, A48–A49.
Palmer, L., Gillespie, D., MacAulay, J. D., Sparling, C. E., Russell, D. J., and Hastie, G. D. (**2021b**). "Harbour porpoise (*Phocoena phocoena*) presence is reduced during tidal turbine operation," Aquat. Conserv. Mar. Freshwater Ecosyst. **31**, 3543–3553.
Rankin, S., Sakai, T., Archer, F. I., Barlow, J., Cholewiak, D., DeAngelis, A. I., McCullough, J. L., Oleson, E. M., Simonis, A., Soldevilla, M. S., and Trickey, J. S. (**2024**). "Open-source machine learning BANTER acoustic classification of beaked whale echolocation pulses," Ecol. Inf. **80**, 102511.
Roch, M. A., Batchelor, H., Baumann-Pickering, S., Berchok, C. L., Cholewiak, D., Fujioka, E., Garland, E. C., Herbert, S., Hildebrand, J., Oleson, E. M., Van Parjis, S., Risch, D., Sirovic, A., and Soldevilla, M. S. (**2016**). "Management of acoustic metadata for bioacoustics," Ecol. Inf. **31**, 122–136.
Sakai, T., and Oswald, M. (**2024**). "PamBinaries: Read and process 'PAMGuard' binary data," https://CRAN.R-project.org/package=PamBinaries (Last viewed 8 January 2026).
Shiu, Y., Palmer, K. J., Roch, M. A., Fleishman, E., Liu, X., Nosal, E.-M., Helble, T., Cholewiak, D., Gillespie, D., and Klinck, H. (**2020**). "Deep neural networks for automated detection of marine mammal species," Sci. Rep. **10**, 607.
Webber, T., Gillespie, D., Lewis, T., Gordon, J., Ruchirabha, T., and Thompson, K. F. (**2022**). "Streamlining analysis methods for large acoustic surveys using automatic detectors with operator validation," Methods Ecol. Evol. **13**, 1765–1777.
Westell, A., Sakai, T., Valtierra, R., Van Parijs, S. M., Cholewiak, D., and DeAngelis, A. (**2022**). "Sperm whale acoustic abundance and dive behaviour in the western North Atlantic," Sci. Rep. **12**, 16821.