Check for updates

METHOD ARTICLE

REVISED An open reproducible framework for CNN-based cetacean vocalization detection in passive acoustic monitoring

[version 2; peer review: 2 approved with reservations]

Rocco De Marco

1. Institute of Biological Resources and Marine Biotechnology (IRBIM), National Research Council (CNR), Ancona, Largo Fiera della Pesca, 2, 60126, Italy

v2 **First published:** 05 Jul 2026, **6**:232
https://doi.org/10.12688/openreseurope.24335.1
Latest published: 06 Oct 2026, **6**:232
https://doi.org/10.12688/openreseurope.24335.2

Abstract

Background

Passive acoustic monitoring (PAM) combined with deep learning is increasingly used to detect cetacean vocalizations. The workflow from raw audio to a trained detector — annotation, segmentation, spectrogram parameterization, image processing, and model evaluation — is seldom accompanied by the complete set of configuration parameters, code, and data needed for independent reproduction, limiting methodological comparability and systematic benchmarking.

Methods

This article presents ai-pam-pipeline, an open-source six-stage framework that formalizes the complete CNN-based detection workflow, with every parameter defined in a single YAML configuration file ensuring exact experimental reproducibility. The framework separates methodological structure from computational implementation, allowing single-channel, spectrogram-based classification workflows to be adapted to other species and vocalization types through configuration alone, without code modification. Validation was conducted on bottlenose dolphin (*Tursiops truncatus*) recordings through two complementary experiments: (A) a binary whistle detector trained on controlled-environment data and evaluated on an independent open-ocean dataset, and (B) a five-class vocalization classifier demonstrating

Open Peer Review

Approval Status ? ?

	1	2
version 2 (revision) 06 Oct 2026		
version 1 05 Jul 2026	? view	? view

1. **Mayu Ogawa**, Japan Agency for Marine-Earth Science and Technology, Yokosuka, Japan

2. **Jamie Macaulay**, University of St Andrews, St Andrews, UK

Any reports and responses or comments on the article can be found at the end of the article.

framework scalability.

Results

The binary detector achieved macro F1 = 0.960 under ten-fold cross-validation with file-level grouped splitting. Cross-domain evaluation yielded a precision of 0.999 and a false discovery rate of 0.001 (macro F1 = 0.744); precision-recall analysis shows that recall rises to 0.76 at a lower decision threshold while precision remains above 0.95, indicating partial but operationally useful generalization. The multiclass extension achieved macro F1 = 0.835 across five vocalization categories; inter-class confusion between echolocation click trains and burst-pulse sounds reflects biological signal overlap rather than classifier failure.

Conclusions

The framework provides a reproducible and documented baseline for CNN-based cetacean detection in PAM. The complete toolkit is released under the Apache 2.0 license, with training and evaluation datasets made openly available through separate public repositories. Together, these resources enable exact replication of the reported analyses and support adaptation to new monitoring contexts.

Plain Language summary

Dolphins and other cetaceans are increasingly monitored by recording the sounds they make underwater, a technique called passive acoustic monitoring. Modern systems use artificial intelligence to automatically detect dolphin whistles in these recordings, but the way each research group prepares its data and trains its models is rarely described in full, which makes results hard to reproduce or compare. This paper introduces ai-pam-pipeline, a freely available software framework that organizes the whole process — from labeling recordings to training and testing a detector — into six clearly defined steps, all controlled by a single settings file. Because every choice is written down in that file, anyone can reproduce the exact same results or adapt the workflow to a different species. We tested the framework on bottlenose dolphin recordings: it detected whistles accurately, and when applied to a completely separate recording made in another ocean, it never produced a false alarm. The framework is openly released so other researchers can build on it.

Keywords

bioacoustics, passive acoustic monitoring, convolutional neural network, cetacean whistle detection, spectrogram preprocessing, reproducible pipeline

Corresponding author: Rocco De Marco (rocco.demarco@cnr.it)

Author roles: De Marco R: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: This project has received funding from the European Commission LIFE Programme under grant agreement No [LIFE19 NAT/IT/000942](project name LIFE DELFI).
The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Copyright: © 2026 De Marco R. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: De Marco R. **An open reproducible framework for CNN-based cetacean vocalization detection in passive acoustic monitoring [version 2; peer review: 2 approved with reservations]** Open Research Europe 2026, **6**:232 https://doi.org/10.12688/openreseurope.24335.2

First published: 05 Jul 2026, **6**:232 https://doi.org/10.12688/openreseurope.24335.1

REVISED **Amendments from Version 1**

The current revision addresses every point raised by both reviewers. The principal changes are:
1) Data leakage corrected (Reviewer 2, Major 1): cross-validation redesigned with StratifiedGroupKFold grouping by source audio file; all experiments re-run; all numbers updated throughout.

2) Precision-recall curves added (Reviewer 2, Major 2): new §3.3.3 with threshold sweep analysis and Figure 3.

3) Claims moderated throughout (both reviewers): "never," "confirming generalization," and the recall/FDR priority contradiction resolved.

4) Literature substantially expanded (Reviewer 2, Major 4 and Minor 8–10): ten new references added, including metadata standards (Tethys, ANSI/ASA S3/SC1.7), training frameworks (Koogu, Ketos, ANIMAL-SPOT, PAMGuard, DeepAcoustics), and missing whistle detection studies (Li et al. 2020, Li et al. 2023, White et al. 2022).

5) Figure 3 replaced (both reviewers): the erroneously retained N_fft boxplot has been replaced with the correct precision-recall analysis; Figure 4 now shows the updated confusion matrix for Experiment B.

6) Repository updated to v1.1.0: includes the grouped cross-validation fix (group_by: "stem" as default), C-spectrogram engine, and updated documentation.

Any further responses from the reviewers can be found at the end of the article

1. Introduction

1.1 Cetacean bioacoustics and passive acoustic monitoring

Over the past few decades, Passive Acoustic Monitoring (PAM) has become a widely-used, non-invasive tool for monitoring acoustically active species over time,[1] especially when environmental conditions make visual sightings logistically impossible. As researchers increasingly deploy PAM across multiple sites and extended timeframes, they generate massive datasets that require highly advanced analytical frameworks. Monitoring cetaceans[2] is one of the most challenging applications of this technology. Marine mammals produce a wide range of vocalizations across wide frequency bands, sometimes in habitats heavily disrupted by intense human noise. Because of these environmental complexities, maintaining high accuracy in population monitoring supports reliable long-term assessment.

Growing human impacts on marine ecosystems have intensified the need for comprehensive, large-scale conservation efforts. Since marine mammals rely almost entirely on sound to communicate, navigate, and forage, PAM serves as the foundation for studying these vulnerable species. Visual surveys are constrained by weather, daylight, and observer availability, while PAM can enable continuous, unattended recording enabling researchers to scale their monitoring efforts efficiently across both coastal waters and remote offshore environments.[3] On a technical level, underwater microphones called hydrophones capture these vocalizations by converting pressure changes into electrical signals. When paired with low-noise preamplifiers, these systems generally operate at sampling rates up to 200 kHz. PAM networks are expanding to support a range of management objectives, including estimating population density, assessing behavioral response to anthropogenic noise, and – as in the present work – reducing harmful interaction between cetaceans and commercial fisheries.[4]

Once the audio is digitized at high frequencies, operators can either monitor the signals in real time for immediate detection or archive the recordings for later study. Both approaches rely on a mix of manual audio-visual inspection and advanced automated workflows.[5] For tonal vocalizations such as whistles, the dominant analytical representation is the spectrogram. Generated via the Short-Time Fourier Transform (STFT), the spectrogram converts raw time-domain signals into a two-dimensional time-frequency image. This visual format is ideal for identifying tonal vocalizations such as whistles, which appear as smooth, continuous, frequency-modulated lines that stand out clearly against sharp, impulsive sounds like echolocation clicks or background static.[5] The field has steadily shifted away from manual spectrogram reviews toward automated detection. Early digital signal processing (DSP) techniques—such as template matching,[6] active contour methods,[7] and entropy-based detectors[8] —contributed to automated whistle identification. Over the past decade, deep learning approaches – particularly with convolutional neural networks (CNN) applied to spectrogram images – have achieved detection and classification performance that consistently exceed earlier DSP methods on published benchmarks, though conventional approaches remain widely deployed in operational systems.

1.2 Deep learning for cetacean PAM: state of the art

Convolutional neural networks are now a widely used architecture for detecting cetacean vocalizations via spectrograms, and have outperformed traditional methods in several tasks. Jiang et al.[9] presented a two-stage pipeline in which an initial

CNN detects if a brief spectrogram frame contains a whistle, and a second network classifies the sound by species, distinguishing between killer whales and long-finned pilot whales. This has been obtained utilizing small grayscale spectrogram tiles; they achieved 97% detection and 95% classification accuracy. This success showed that compact, single-channel inputs can effectively capture the geometric shape of whistle contours, provided that the frequency limits and frame durations are carefully selected. While Jiang et al.[9] focused on pre-rendered spectrogram images, Shiu et al.[10] showed that these models could adapt to varying recording conditions by using a different input format. Instead of treating the spectrogram as a standard visual image, two-second audio clips were extracted, and the Short-Time Fourier Transform (STFT) is calculated using a 128 ms Hann window with a 50 ms step size. This process generated a matrix of time-frequency magnitude values, which they normalized using the squared Frobenius norm and passed directly into the CNN as a numerical tensor. When tested on North Atlantic right whale upcalls between 39 Hz and 358 Hz, their model delivered significantly better precision-recall metrics and cut false positives per hour by orders of magnitude compared to existing baselines. The model maintained its high performance on entirely new recordings from different times and locations, successfully passing the real-world stress test that operational passive acoustic monitoring systems regularly encounter.

Li L et al.[11] utilized a depthwise-separable CNN to classify multiple types of whistles from Indo-Pacific bottlenose dolphins (Tursiops aduncus). They used three-channel spectrogram inputs combined with a comprehensive set of time-domain data augmentations, including pitch-shifting, time-stretching, noise mixing, temporal shifting, and gain adjustments. Classification accuracy increased from 93.7% on the original samples to 98.53%, proving that data augmentation can effectively offset the challenges of small training sets and unbalanced classes. In this study, the authors admitted that their chosen STFT parameters might not represent the ideal spectrogram configuration for whistle classification, treating the spectrogram generation as an arbitrary choice rather than an optimized variable.[11]

Li P et al.[12] addressed the annotation bottleneck by training a deep network for whistle contour extraction on samples synthesized from background recordings, requiring minimal manual annotation, and evaluated it on common (*Delphinus* spp.) and bottlenose dolphin whistles. In a follow-up study, the same group[13] removed the need for human-annotated training data by training on pseudo-labels generated by existing whistle extractors, including a sequential Monte Carlo tracker, and evaluated the method on five odontocete species. White et al.[14] extended CNN detection beyond whistles to four sound-source classes (delphinid tonal calls, biological clicks, vessel noise, and ambient noise), stacking spectrograms computed at three FFT sizes (1024, 2048, and 4096 points) as the RGB channels of a single EfficientNet-B0 input.

Passing from simple binary detection to tracing the exact paths of vocalizations, researchers like Jin et al.[15] have reframed the challenge as a semantic segmentation task. Their model processed high-resolution grayscale spectrograms to generate pixel-level masks of whistles, which were then refined using morphological post-processing to map out centerlines and untangle overlapping calls. Despite evaluating their model on noisy coastal recordings filled with snapping shrimp interference, they achieved a precision of 0.96, an accuracy of 0.89, and an F-score of 0.86. These results show that dense, high-resolution inputs are essential when a study needs to preserve precise contour curves and thickness for subsequent bioacoustic analyses. Taking a different approach, Nur Korkmaz et al.[16] investigated the potential of transfer learning for dolphin whistle detection. Single-channel grayscale spectrograms have been duplicated into a three-channel format, and a pre-trained VGG16 network using automated hyperparameter optimization has been used. This method achieved a 92.3% mean accuracy on a substantial, annotated dataset, easily outperforming traditional detectors while keeping the initial preprocessing straightforward. Their work underscores the practical benefit of adapting established computer-vision models when annotated underwater audio datasets are scarce. Based on these studies, the way input data is represented is just as critical to performance as the neural network design itself. Small, single-channel tiles work well for basic detection and generalize efficiently with minimal computing power, three-channel inputs allow researchers to leverage powerful pre-trained models, and high-resolution imagery preserves fine pixel-level details of sound contours. Preprocessing choices such as frequency filtering, STFT configuration, and image scaling are therefore not incidental: they shape what the network learns and how well it transfers. While this reality is widely accepted in theory, the field still lacks a rigorous, quantitative evaluation of how these preprocessing choices interact within a unified, reproducible pipeline for cetacean monitoring.[17] Table 1 presents a summary of these cited studies.

A second, related challenge concerns the annotation of training data. The studies reviewed above employ diverse and largely undocumented labeling conventions: event boundaries, minimum duration thresholds, and the criteria for distinguishing whistle from click train or burst pulse are rarely made explicit. This fragmentation has been identified as a systemic limitation of bioacoustic machine learning[17] and has prompted community efforts to standardize acoustic metadata — notably the Tethys database framework,[18] the emerging ANSI/ASA S3/SC1.7 standard for bioacoustic metadata,[19] and broader PAM data management guidelines.[20] On the training-framework side, several open-source toolkits have appeared: Koogu,[21] Ketos,[22] and ANIMAL-SPOT[23] for bioacoustic deep learning; DeepAcoustics for

object-detection-based whistle identification[24]; and the PAMGuard platform for real-time detection, classification, and localization.[25] Inter-annotator variability in marine PAM contexts has been shown to depend on both vocalization type and signal-to-noise ratio,[26] introducing a source of performance variance that is rarely quantified in comparative studies. The proposed pipeline addresses this gap by formalizing an annotation protocol with explicit decision criteria for each label class (Section 2.2) with a unified, formally defined framework that makes every preprocessing and evaluation decision explicit, reproducible, and systematically configurable.

1.3 Objectives and contributions

Objective 1 — A formally described, six-stage methodological framework. The proposed pipeline formalizes each stage of the CNN-based detection workflow — from annotation protocols and segmentation strategies through spectrogram parameterization and preprocessing to cross-validation design and performance reporting — making every decision point explicit and reproducible. Each stage generalizes to other taxa and vocalization types through YAML-based configuration.

Objective 2 — A full open-source implementation: ai-pam-pipeline. The toolkit provides a command-line interface whose entire parameter space is defined by a single YAML experiment file, ensuring that any reported result can be exactly regenerated. A plug-in architecture allows the CNN model to be substituted without modifying the training loop. Full technical documentation is provided as Extended Data.

Objective 3 — End-to-end validation on two complementary tasks. The framework is demonstrated on (A) a binary whistle detector for *T. truncatus* with cross-domain evaluation on an independent open-ocean dataset, and (B) a five-class vocalization classifier on the same training set, exercising the multiclass configuration and the scroll augmentation mechanism. Together, the two experiments show all six pipeline stages operating as an integrated, reproducible workflow.

2. Materials and methods

2.1 Overview and design principles

Implementing deep learning in Passive Acoustic Monitoring (PAM) requires a coordinated sequence of interdependent steps. The process begins with raw data analysis, in which PAM operators screen recordings to extract, identify, and classify individual signals. These signals are then converted into spectrogram images, a step that involves choosing among a wide range of parameters — color scales, image resolution, and STFT settings — each of which can meaningfully affect the outcome. The resulting images may further undergo preprocessing or augmentation to compensate for data scarcity, followed by visual inspection to verify their quality. This stage alone can be time-consuming and

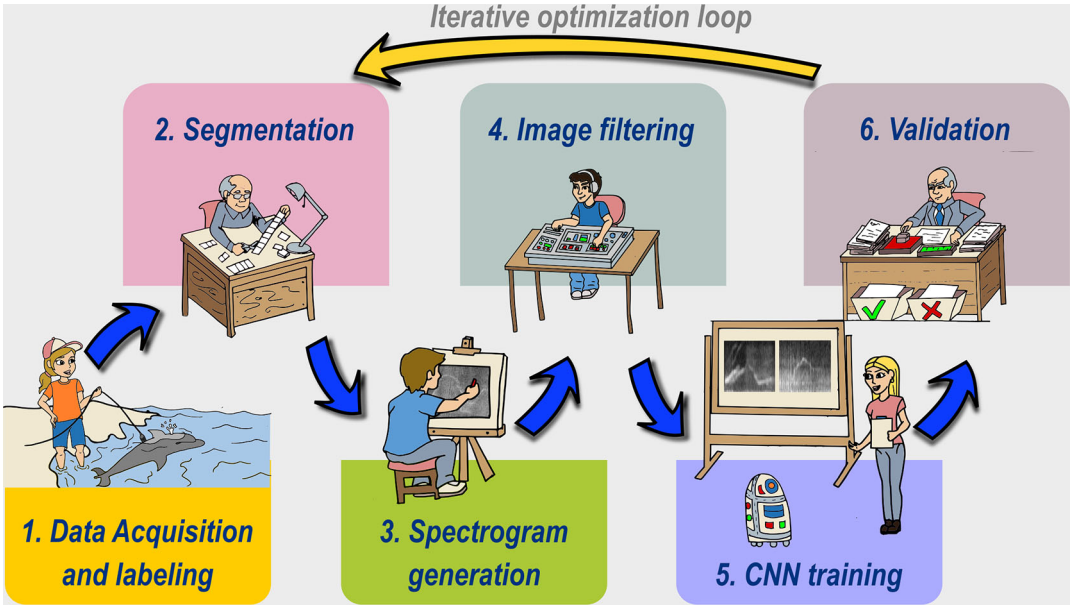

Figure 1. Six-stage pipeline block diagram of ai-pam-pipeline. Module names are annotated for each stage. Architecture-level labels use general PAM terminology; inline annotations report the cetacean validation parameters adopted in Section 3 as an example instantiation.

prone to undetected inconsistencies in training data composition. Once the image dataset is assembled, model training can proceed.

Although individual processing stages are described to varying degrees in the literature, the complete sequence is less often documented as a single executable and reproducible workflow, even though existing toolkits cover substantial parts of it. Yet undocumented workflows directly undermine reproducibility: CNNs are inherently stochastic and sensitive to parameter choices, so even minor untracked variations can produce divergent outcomes. The framework proposed here addresses this by formalizing the entire workflow into six integrated stages (Figure 1), all governed by a single configuration file. The sequence spans from raw data annotation (Stage 1) to final performance evaluation (Stage 6), with intermediate steps covering audio segmentation into discrete windows (Stage 2), spectrogram generation (Stage 3), optional image filtering for feature enhancement (Stage 4), and CNN model training (Stage 5).

2.2 Stage 1 — Audio acquisition and PAM annotation (Figure 1)

In PAM applications, data acquisition at monitoring sites is typically carried out using autonomous devices equipped with a hydrophone, which converts acoustic waves into an electrical signal, followed by a preamplifier, an analog-to-digital converter (ADC), and a recording system. Depending on the target species and the type of vocalization of interest (e.g., tonal or pulsed signals), the relevant signal bandwidth may range from a few hertz up to approximately 25 kHz, as is typical for bottlenose dolphin whistles, and extend beyond 160 kHz for high-frequency echolocation clicks. Consequently, the sampling rate must be selected accordingly, in compliance with the Nyquist criterion, typically adopting values of 96, 192, or 384 kS/s. The acquired data are usually stored on integrated memory media, in WAV format or in lossless compressed formats such as FLAC. In long-term monitoring deployments, recordings are typically partitioned into fixed-duration segments (e.g., 1, 5, 30, or 60 minutes). It is considered best practice to adopt a standardized naming convention that includes a timestamp in a convenient format, such as YYYYMMDD_hhmmss. When multiple recording units are deployed, prepending a unique station ID (e.g., HYD1_) and appending the sampling rate (e.g., _192) is highly recommended to ensure traceability. Such structured filenames are fundamental for enabling downstream automated processing and metadata extraction. The acquired data subsequently undergo extensive processing aimed at detecting vocalizations, which may be performed through visual and acoustic inspection by trained personnel and/or with the support of detection software such as PAMGuard.[25] This annotation process, in which segments of recordings are labeled by associating specific tags with particular vocalizations, remains — as previously highlighted — non-standardized.[26] In the case of *Tursiops truncatus*, there is not even a universally accepted classification of the different vocalization types. The proposed approach consists of leveraging the tagging functionalities available in the open-source software Audacity (or similar tools). The user can graphically select portions of the audio signal containing specific sounds and assign them a label. Labels may overlap in the case of superimposed sounds. The final output is a tab-separated values (TSV) file containing three main fields: t_{start}, t_{end}, and *label*. The first two represent the temporal offsets, expressed in seconds as floating-point values, relative to the beginning of the recording. Audacity supports a variety of audio formats (e.g., WAV, FLAC), while labels are exported as plain text (TXT) files. To facilitate automated pairing within the pipeline, it is strongly recommended that label files maintain the same stem as the source audio file (e.g., HYD1_20200101_100025_192.txt). In this work, a simplified tagging scheme was adopted for the general detection of whistles produced by *Tursiops truncatus*:

W (whistle): tonal, continuous, frequency-modulated signal with a smooth contour in the spectrogram.

MW (multiple whistle): segments containing overlapping whistles.

ECT (click train): regular series of broadband pulses, typically appearing as vertical striations.[27]

BPS (burst pulse sound): rapid bursts of irregular pulses.[27]

FB (feeding buzzes): a rapid sequence of broadband pulses with progressively decreasing inter-pulse intervals, typically reaching <5 ms at peak rate. On the spectrogram they appear as a dense band of vertical striations, with total event duration generally between 0.5 and 3 seconds.[27] In the Oltremare recordings, feeding buzzes are associated with foraging behavior, though the annotation criterion is the acoustic structure rather than the behavioral context.

NOISE: vessel interference, flow noise, and non-target biological sounds.

The same approach can be extended by adopting a more articulated labeling scheme or by applying the same principles to other species. Although a standardized classification framework would be desirable, in its absence, it is essential to

maintain a consistent labeling criterion across all stages of the processing pipeline. The establishment of a universally shared annotation scheme would significantly increase methodological consistency and facilitate inter-calibration across diverse research efforts. For example, a classification approach similar to the FAO alpha-3 codes, employed for fish species classification, could be adopted.

The temporal boundaries of each annotation follow class-specific criteria. For whistles (W), a new annotation begins when the fundamental frequency contour shows a gap exceeding 50 ms or a discontinuity in frequency trajectory. For echolocation click trains (ECT), the annotation spans from the first to the last click in a continuous train, with a maximum permitted inter-click gap of 200 ms. For burst-pulse sounds (BPS), boundaries correspond to the onset and offset of the visible pulse train. These criteria follow the species-specific guidelines established for the Oltremare corpus (Screpanti et al.[28]) and are documented in the Reference Manual (Extended Data, S4.2).

An advantage of maintaining paired audio and label files is the ability to extract specific tagged and classified fragments automatically, ensuring consistency and replicability across large datasets. It could also facilitate the systematic creation of specialized databases, where vocalizations are indexed by type and optionally integrated with metadata (date, time, and environmental parameters).

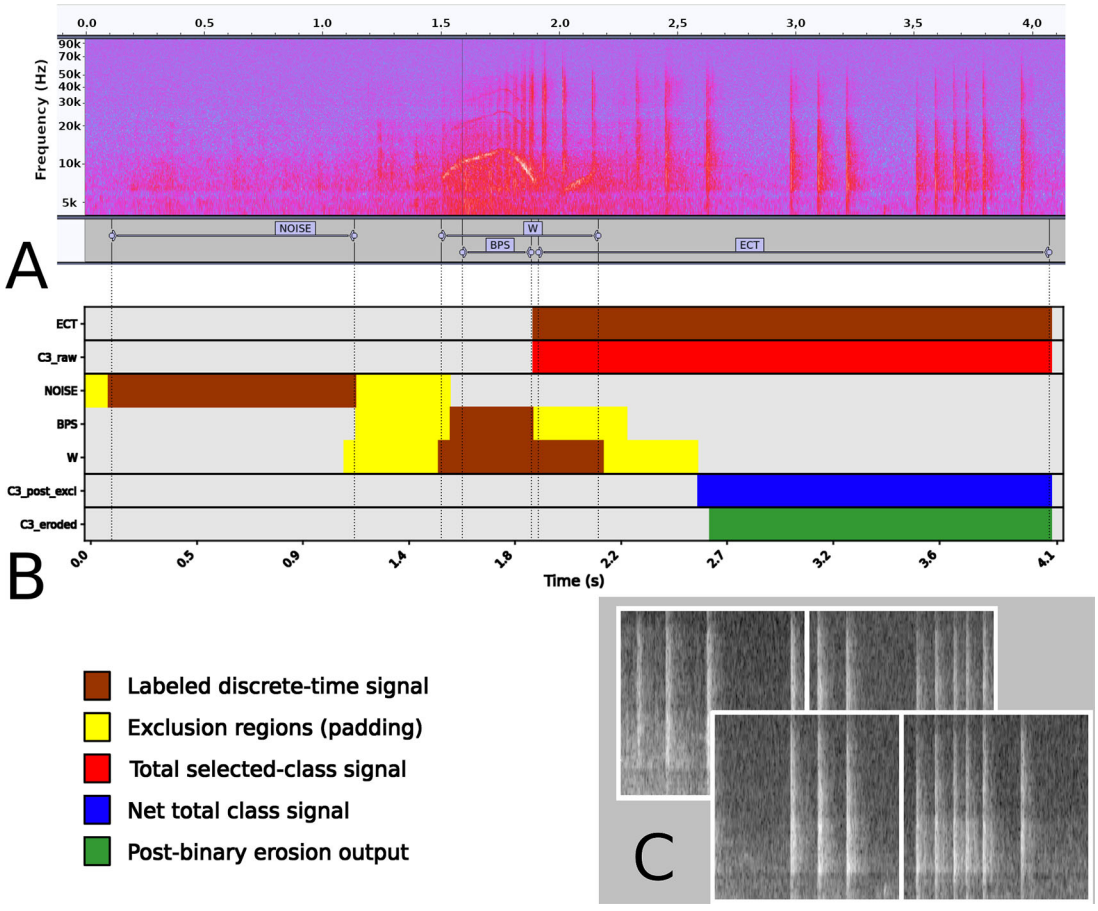

Figure 2. Multi-step segmentation and exclusion logic for class-specific sample extraction from a dense multi-label acoustic recording. (A) Raw Audacity label file: four annotation classes (NOISE, BPS, W, ECT) are present and temporally overlapping in a ~4 s segment. (B) Construction of the usable ECT (C3) signal: the raw ECT span (C3_raw) is reduced by exclusion zones derived from all overlapping non-target annotations (yellow), yielding the net usable signal (C3_post_excl, blue) and its morphologically eroded version (C3_eroded, green), which constitutes the final sampling domain for spectrogram window extraction. (C) Representative 224 × 224 spectrogram tiles extracted from the eroded span, showing uncontaminated echolocation click trains. Although the example is drawn from a T. truncatus recording, the exclusion and erosion procedure is identical for any class in any annotated PAM dataset.

Table 1. Summary of CNN-based cetacean vocalization studies reviewed in Section 1.2.

Study	Species/call type	Input repr.	Size	Ch.	Task	Preprocessing treatment
Jiang [9]	*Orcinus orca*/ *Globicephala melas*, whistle	Grayscale image	~180 × 120	1	Det. + Class.	Fixed band-limiting & frame duration.
Shiu [10]	*Eubalaena glacialis*, upcalls	Num. matrix	40 × 40	1	Binary Det.	Fixed STFT (128 ms Hann, 50 ms adv.)
Li L [11]	*Tursiops aduncus*, whistles	RGB image	200 × 200	3	Multiclass	Fixed STFT; acknowledged as non-optimal.
Li P [12]	*Delphinus* spp./ *Tursiops truncatus*, whistles	Spectrogram patches	50 × 50	1	Contour extr.	Fixed STFT (8 ms Hamming, 2 ms adv.); training samples synthesized from background noise
Li P [13]	Five odontocete species, whistles	Grayscale patches	64 × 64	1	Contour tracking	Same STFT as[12]; trained on pseudo-labels, no human annotation
White [14]	Multiple marine sources	Stacked multi-resolution spectrograms	224 × 224	3	Multiclass (4 cl.)	Three FFT sizes stacked as channels; Hann, 75% overlap
Jin [15]	*Tursiops truncatus, whistles*	Grayscale image	1000 × 1000	1	Sem. Segm.	Fixed high-res for contour fidelity.
Korkmaz [16]	*Tursiops truncatus*, whistles	RGB image	224 × 224	3	Binary Det.	Minimal and fixed for VGG16.

Abbreviations: Det. (Detection), Class. (Classification), Sem. Segm. (Semantic Segmentation), Cont. Extr. (Contour Extraction).

2.3 Stage 2 — Audio handling, segmentation and windowing

Inference models based on neural networks require input data with strictly defined dimensions. Whether the architecture operates directly on the raw acoustic signal or on spectrogram representations, a fixed temporal observation window must be established. A wider window can capture finer acoustic details and potentially improve detection resolution, though this comes at the cost of reduced system responsiveness in real-time applications and greater computational overhead at inference time. Narrower windows, conversely, allow faster response and lower resource consumption, but may fail to capture the contour features that define a vocalization, with a corresponding effect on model metrics. In the work of Scaradozzi et al., which used the pipeline described here for the detection of *T. truncatus* whistles, a 0.8-second frame proved effective for capturing the bioacoustic modulation of interest.[29] For continuous temporal analysis, overlapping consecutive windows is standard practice, preventing signals from being split across boundaries and subsequently misclassified. That study employed a 50% overlap, generating a new observation window every 0.4 seconds. These values are adopted throughout the present work as the cetacean validation parameters; other taxa or vocalization types may require different window lengths, all of which are configurable in the YAML experiment file (e.g., pipeline.stage_2.as_len, pipeline.stage_2.as_overlap). Species with longer stereotyped vocalizations — such as the song units of mysticetes or the multi-second call sequences of some odontocetes — are accommodated by increasing as_len accordingly, with no further modification to the pipeline. The choice of window size and overlap also has direct implications for real-time deployments, particularly on edge devices with constrained computational resources.[30] When a tagged signal is shorter than the defined window, *ai-pam-pipeline* automatically centers the acoustic event within the frame to preserve spatial consistency for the CNN.

Isolating signals by class and rigorously excluding target-class events from the background class are essential to prevent inconsistency during training. Without this, the CNN risks learning to distinguish not between pure background and the target bioacoustic feature, but between categories that are acoustically contaminated by one another. The problem is especially acute in multi-class settings, where training samples labeled for one vocalization type must not contain

fragments of another: such contamination introduces ambiguity directly into the feature space. To address this, the library implements an automated padding mechanism that expands exclusion zones around each annotation boundary as a function of the sliding window step; the technical details of this logic are provided in the Extended Data.

In PAM recordings where multiple vocalization types co-occur simultaneously, this exclusion and erosion logic becomes the primary means of obtaining acoustically unambiguous training samples. The mechanism is fully general: any annotation class whose temporal extent overlaps with the target class generates an exclusion zone, and the morphologically eroded residual of the target span defines the sampling domain for that class. The design applies identically regardless of species or vocalization type, provided the annotations follow the label-file convention described in Section 2.2.

Figure 2 illustrates this process for an exemplary extraction from a multi-label recording. The target species is *Tursiops truncatus* and the class of interest is the echolocation click train (ECT), but the same logic applies to any acoustic class in any annotated PAM dataset. Panel A shows the raw Audacity label file, in which NOISE, BPS, W, and ECT annotations are present and temporally overlapping. Panel B traces the construction of the final usable ECT signal: the raw ECT span (C3_raw) is first reduced by the exclusion zones generated from the overlapping NOISE, BPS, and W annotations (C3_post_excl), after which a one-dimensional binary erosion (C3_eroded) is applied using a structuring element of length equal to the minimum duration expressed in frames. The operation acts on the binary timeline representation, where active frames denote the presence of the target class after exclusion filtering. The effect is inherently contractive: by shrinking each contiguous active segment at both boundaries, it eliminates residual fragments that contain insufficient signal. Panel C shows representative spectrogram tiles extracted from the eroded span, each containing an unambiguous click train free of contamination from adjacent vocalizations. The exclusion zone width is parametrically controlled by the analysis window length and step size, as detailed in the Extended Data.

An additional mechanism for expanding the training set is scroll augmentation: for a given annotated event, the analysis window is slid from the position where only the minimum required vocalization duration appears at the right edge, generating multiple time-shifted copies of the same event. This is the primary data augmentation strategy implemented in *ai-pam-pipeline*, activated via the YAML parameter pipeline.stage_2.augmentation_scroll. Image processing filters (Section 2.5) can also serve as a basis for augmentation: applying one or more transformations to a clean spectrogram — for instance, adding synthetic pink noise at varying intensity levels to a whistle image — produces training samples spanning a range of apparent signal-to-noise ratios. This approach is fully supported within the framework: running Stages 2–4 with different filter configurations and merging the resulting spectrogram directories before training is sufficient to compose a multi-condition dataset without any modification to the pipeline. Practical guidance for windowing parameter selection is provided in the Reference Manual (Extended Data, S4.3–S4.5).

2.4 Stage 3 — Spectrogram generation

As summarized in Table 1, the most widely adopted methodological approach involves the use of Convolutional Neural Networks (CNNs) that process spectrograms as input. Spectrogram images are generated via the discrete STFT, defined as:

$$X_{m,\omega} = \sum_{n=-\infty}^{\infty} x(n)w(n-mH)e^{-j\omega n}$$

where $w(n)$ represents the window function, m is the discrete time index, and H is the hop size. In this framework, a Hanning window is applied to minimize spectral leakage. The hop size H is dynamically calculated based on the chosen FFT length (N_{fft}) and the overlap percentage (ov), following the relation

$$H = N_{fft} \times (1 - ov)$$

To align the representation with the logarithmic perception of sound intensity and to compress the dynamic range for the CNN, the magnitude of the STFT is converted to a decibel (dB) scale:

$$S_{m,k} = 20\log_{10}|X_{m,k}|$$

The spectrogram image generation process must allow for the systematic and uniform adjustment of core acoustic parameters, such as the FFT size (N_{fft}) and the Hanning window overlap. Additionally, it must support customizable visualization settings, including output image dimensions (width and height) and the selection between grayscale representations or specific predefined colormaps. In the experiments reported in Section 3, the grayscale mode is used

throughout with an image size of 224×224 pixels. This differs from the 300×150 format adopted in the reference architecture,[29] reflecting an updated preprocessing choice evaluated as part of the present framework.

While band-pass filtering in the time domain is a common practice, the proposed library performs frequency selection directly on the generated spectrogram. This spectrogram cropping restricts the analysis to a specific frequency range $[f_{\min}, f_{\max}]$, effectively discarding components outside the bandwidth of interest. In the case of *T. truncatus* whistle detection, a frequency crop in the range of 3–25 kHz is generally employed, as determined by the species' whistle bandwidth. This range encompasses the most discriminative frequency-modulated contours of the species while optimizing the input dimensionality for the CNN. The image dimensions (width_px, height_px) are likewise decoupled from the window length: a longer analysis window produces a wider raw spectrogram that is resampled to the same fixed pixel grid, preserving CNN input consistency regardless of species. By restricting the vertical axis of the spectrogram image to these limits, the pipeline ensures that the model focuses on the biologically relevant signal features, improving both classification accuracy and computational efficiency during inference. The heterogeneity of spectrogram formats across the literature (Table 1) underscores the value of recording every STFT setting explicitly in the configuration file, so that the representation used in any given study can be reproduced without ambiguity. The adoption of a single-channel grayscale representation provides distinct computational advantages, particularly for deployment in resource-constrained environments. By reducing the input dimensionality from the three channels required by standard RGB images to a single data stream, the model minimizes memory footprint and processing overhead. Criteria for selecting FFT parameters for new datasets and species are documented in the Reference Manual (Extended Data, S5.1).

2.5 Stage 4 — Spectrogram image preprocessing

While raw spectrograms provide a fundamental time-frequency representation, they often suffer from non-uniform background noise, low contrast, and interference from anthropogenic sources. Preprocessing the spectrogram can substantially improve the signal-to-noise ratio of target vocalizations — particularly the frequency-modulated contours of dolphin whistles — making it easier for the model to extract the relevant features and learn to recognize them reliably.[31] This step is optional in the proposed pipeline, but if it is used, it must be applied with strict consistency: the same transformations that are applied to the training data must be applied identically to every input processed by the inferential model during detection and classification.[32] In real-time deployments, the computational cost of these filters also deserves careful consideration, as it directly affects system responsiveness. In PAM applications, filtering is commonly applied to the raw acoustic signal[14] to attenuate background noise or bring out vocalization features that would otherwise be difficult to detect. Conversely, applying filters to the spectrogram image does not involve the original signal, but rather operates directly on the time-frequency representation generated via STFT before it is fed into the neural network. While this image-based approach is widely established in computer vision — for instance, in the enhancement of underwater photography[33] — has only recentely begun to be explored in cetacean vocalization detection.

The toolkit supports a set of standard image-domain filters from OpenCV — including morphological operators (Top-Hat), edge detectors (Laplacian, Canny, Sobel, Scharr), and adaptive contrast enhancement (CLAHE) — that can be applied individually or in combination to the raw spectrogram before CNN input. A global polarity inversion is also available. The complete filter specifications are documented in the Extended Data.

Beyond image filtering, Stage 4 includes a normalization step applicable to each spectrogram image prior to CNN input. Per-image normalization rescales the pixel intensity values to the range $[0, 1]$ independently for each sample, compensating for global amplitude variability across recording sessions, hydrophone sensitivities, and environmental conditions. This operation ensures that the CNN receives input distributions of consistent dynamic range, regardless of the absolute signal level in the original recording, and is a prerequisite for stable gradient-based optimization. In all experiments reported in Section 3, per-image normalization is enabled via the YAML parameter pipeline.stage_4.image_normalization: true and constitutes the only active Stage 4 operation; image-domain filters are not applied in this work.

The toolkit introduced in this study provides a dedicated script, ai_pam_preprocessing.py, designed to execute the operations detailed in Stages 2 through 4 as a unified, streamlined workflow. This script directly interfaces with raw audio recordings and their associated label files to generate spectrograms, optionally filtered, according to the specific experimental parameters encoded in the configuration file described in the following section.

2.6 Stage 5 — CNN training with k-fold cross-validation

As established in Section 1.2, CNNs operating on spectrogram images are a widely used approach for automated vocalization detection in PAM. The proposed framework is implemented for CNNs but is not restricted to them: the plug-in architecture convention — any Python file exposing a build_model(input_shape, n_classes, kernel_size) function — allows recurrent (RNN, LSTM), hybrid (CRNN), or transformer-based (ViT) architectures to be substituted without

modifying the training loop. For models requiring sequential inputs, the toolkit produces time-series spectrogram extracts by modulating the frame overlap, as described in Section 2.3.

The present study utilizes a baseline CNN architecture,[29] specifically engineered for efficiency and deployment on resource-constrained hardware,[30] implemented using the TensorFlow framework.[32] The network consists of three convolutional blocks with an increasing number of filters (32, 64, and 128, respectively); each block incorporates a Rectified Linear Unit (ReLU) activation and a 2×2 max-pooling layer. The architecture concludes with a fully connected layer of 128 neurons and a sigmoid activation function for the binary output layer. For multiclass tasks, the sigmoid output is replaced by a softmax activation over N classes, and the loss function switches from binary to categorical cross-entropy; the model file is passed via the YAML key pipeline.stage_5.model.architecture without other changes to the loop.

The model takes as input single-channel spectrogram images, with 224×224 pixel dimensions consistent with the spectrogram generation parameters described in Section 2.4. Training uses the Adam optimizer with an initial learning rate of 0.0001. The loss function is binary cross-entropy for the binary classification task and categorical cross-entropy in the multiclass setting, reflecting the sigmoid and softmax output activations respectively.

The training process uses a batch size of 32 and a maximum of 100 epochs. A linear learning-rate warmup is optionally applied during the initial epochs to improve optimization stability. Model performance is evaluated using accuracy during training and complemented by additional metrics (e.g., macro F1-score) computed on the held-out data. The training process incorporates an early stopping mechanism with a patience of 10 epochs and a 5-epoch warmup period— designed to prevent premature termination during initial loss plateaus—while ensuring the restoration of the best-performing weights at the end of the session.

To ensure reproducibility, all random processes are controlled via fixed seeds, and model initialization is re-seeded at each fold.

To conclude the model preparation phase, the proposed toolkit provides a dedicated script, ai_pam_train.py, which automates the management of k-fold cross-validation, individual model training, and in-domain validation, ultimately generating the performance metrics discussed in Section 2.7 and managing the training directives (epochs, early stopping, batch size, etc.). Training hyperparameter recommendations are provided in the Reference Manual (Extended Data, S7.4).

2.7 Stage 6 — Model evaluation and performance measurement

2.7.1 In-domain cross-validation

Once the spectrogram image dataset has been assembled, the framework evaluates model performance through stratified k-fold cross-validation. The full dataset is partitioned into k non-overlapping subsets (folds) of approximately equal size. In each of the k iterations, one fold serves as the held-out test set while the remaining $k-1$ folds are used for training, including an in-training validation split that governs early stopping. This procedure is repeated until every fold has served once as the test set, yielding k independent performance estimates. In the present study, $k = 10$ and partitioning is performed by random assignment with a fixed seed (seed 42), ensuring full reproducibility.

To prevent data leakage from temporally overlapping windows, the cross-validation folds are constructed using StratifiedGroupKFold (scikit-learn), with the source audio filename as the grouping key (group_by: 'stem' in the YAML configuration). This ensures that all windows derived from the same recording — including those generated by the 50% temporal overlap or by scroll augmentation — remain within the same fold, providing a conservative estimate of in-domain performance. A diagnostic log reports the window-to-group ratio per class and verifies that no group spans multiple folds (see Extended Data, S7.3.1).

For binary classification, the predicted class is determined by applying a fixed threshold θ to the sigmoid output: a sample is assigned to the positive class (vocalization present) if the network output exceeds θ, and to the negative class otherwise. In the present study the decision threshold is fixed at $\theta = 0.5$ for the primary evaluation. The sensitivity of cross-domain metrics to threshold choice is examined in Section 3.3.3.

2.7.2 Binary classification metrics

Performance is summarized by five complementary metrics computed per fold and then averaged across folds:

- Accuracy $= \frac{(TP+TN)}{(TP+TN+FP+FN)}$
- Precision $= \frac{TP}{(TP+FP)}$
- Recall $= \frac{TP}{(TP+FN)}$
- F1-score $= 2 \cdot Precision \cdot \frac{Recall}{(Precision+Recall)}$
- False Discovery Rate $(FDR) = \frac{FP}{(FP+TP)}$

Recall and F1-score are the primary metrics of interest in PAM detection tasks, however which metric should be prioritized depends on the deployment context: for surveys and density estimation, missed detections are usually the more costly error, whereas for the deterrence application targeted here false alarms carry a direct operational cost (Sections 3.2 and 4.2).

2.7.3 Multiclass evaluation metrics

For classification tasks involving $N > 2$ classes, the sigmoid output layer is replaced by a softmax activation, and a sample is assigned to the class with the highest predicted probability. Performance is assessed through per-class and aggregate metrics.

Per-class Precision, Recall, and F1. Each metric is computed independently for each class k:

$$Recall_k = \frac{TP_k}{TP_k + FN_k}$$

$$Precision_k = \frac{TP_k}{TP_k + FP_k}$$

$$F1_k = 2 \cdot \frac{Precision_k \cdot Recall_k}{Precision_k + Recall_k}$$

where TP_k, FP_k, and FN_k are the true positives, false positives, and false negatives for class k, respectively. Per-class recall is particularly diagnostic: in the presence of class imbalance, a model may achieve high overall accuracy while suppressing minority classes entirely ($Recall_k \to 0$), a failure mode that is invisible to aggregate metrics alone.

Macro-averaged F1. The unweighted mean F1 across all N classes:

$$Macro - F1 = \frac{1}{N} \sum_{k=1}^{N} F1_k$$

Macro F1 is sensitive to minority-class performance and constitutes the primary aggregate metric when all classes are of equal operational importance, regardless of their support in the training set.

Weighted-averaged F1. The support-weighted mean:

$$Weighted - F1 = \frac{\sum_{k=1}^{N} n_k \cdot F1_k}{\sum_{k=1}^{N} n_k}$$

where n_k is the number of true instances of class k. Weighted F1 reflects aggregate performance on the training distribution and may be misleadingly high when the dominant class performs well while minority classes are suppressed; it is reported for reference alongside Macro F1.

Confusion matrix. The full $N \times N$ matrix C, where C_{ij} is the number of instances of true class i predicted as class j. Normalized by true class (row-normalized), the confusion matrix identifies systematic confusions between acoustically similar classes and distinguishes genuine misclassification from class suppression due to imbalance.

The practical effect of unbalanced training data and annotation incompleteness on these metrics is illustrated in Section 3.4, where a known-imbalanced multiclass configuration on the Oltremare dataset produces the metric patterns described above.

2.7.4 Cross-validation design for multiclass tasks

For multiclass tasks with unequal class frequencies, stratified k-fold cross-validation is recommended to ensure each fold contains a representative distribution of all classes. For the binary balanced design in Experiment A (Section 3.2), standard and stratified k-fold are equivalent. The stratified design is adopted in Experiment B (Section 3.4).

2.7.5 Cross-dataset generalization

In-domain cross-validation provides an estimate of performance on held-out data drawn from the same recording session and acoustic environment as the training set. This upper bound is an optimistic predictor of deployment performance, since real-world PAM scenarios routinely involve equipment, sites, and species that differ from the training conditions. The framework therefore separates in-domain validation from cross-dataset testing as two distinct evaluation stages. Cross-dataset testing is carried out using ai_pam_crosstest.py, which loads a set of trained fold models and evaluates them on an entirely independent spectrogram dataset.

The evaluation was conducted using the *all* mode: performance metrics are computed independently for each of the k fold models, providing a granular distribution of results.

2.8 Iterative use of the framework

Although the framework is organized into six formal stages, its execution is intentionally iterative rather than strictly linear. When validation performance is unsatisfactory, the experiment can be reconfigured by modifying the YAML file and re-running the pipeline from Stage 2 onward, or, when needed, from any later stage downstream of the unchanged source recordings. In practice, this allows segmentation parameters, spectrogram settings, preprocessing filters, sampling policies, or training hyperparameters to be adjusted without revisiting the original recordings or repeating the manual

Table 2. Reproducibility reference for Experiment A: YAML parameter paths and values used in all training runs.

YAML path	Value (Exp. A)	Description
pipeline.stage_2.as_len	0.8	Analysis window length (s)
pipeline.stage_2.as_overlap	0.5	Window overlap fraction
pipeline.stage_3.fft_len	512	FFT window size
pipeline.stage_3.fft_overlap	0.5	STFT overlap
pipeline.stage_3.image.width_px	224	Spectrogram width (px)
pipeline.stage_3.image.height_px	224	Spectrogram height (px)
pipeline.stage_4.image_normalization	true	Per-image normalization
pipeline.stage_5.model.architecture	baseline_cnn.py	CNN model file
pipeline.stage_5.model.kernel_size	3×3	Convolutional kernel size
pipeline.stage_5.cross_validation.max_samples per class	4700	Samples per class (balanced)
pipeline.stage_5.cross_validation.k_folds	10	Cross-validation folds
pipeline.stage_5.cross_validation.group_by	stem	Fold grouping key (leakage protection)
pipeline.stage_5.seed	42	Random seed
pipeline.stage_6.evaluation.threshold	0.5	Binary class. Sigmoid θ

Table 3. Experiment A performance summary, mean ± SD across ten folds. In-domain: held-out Oltremare data. Cross-domain: DCLDE 2022, evaluated independently with all ten fold models. Precision, recall and FDR refer to the whistle class; macro F1 averages both classes.

Metric	In-domain (Oltremare)	Cross-domain (DCLDE 2022)
Accuracy	**0.960 ± 0.017**	**0.760 ± 0.041**
Precision	**0.969 ± 0.015**	**0.999 ± 0.002**
Recall	**0.950 ± 0.028**	**0.520 ± 0.083**
Macro F1	**0.960 ± 0.017**	**0.744 ± 0.049**
FDR	**0.032 ± 0.015**	**0.001 ± 0.002**

Table 4. Per-class metrics for Experiment A, aggregated across ten folds. C0 = non-whistle (NOISE, FB, BPS, ECT); C1 = whistle (W, MW).

Dataset	Class	Precision	Recall	F1
Oltremare (in-domain)	**C0 (non-whistle)**	**0.952**	**0.969**	**0.960**
	C1 (whistle)	**0.969**	**0.950**	**0.959**
DCLDE 2022 (cross-domain)	**C0 (non-whistle)**	**0.678**	**1.000**	**0.807**
	C1(whistle)	**0.999**	**0.520**	**0.681**

labeling process. Since annotation is one of the most demanding and time-consuming steps in PAM workflows, preserving the labeled source material while enabling controlled reprocessing is a major operational advantage.

3. Use cases

3.1 Datasets

3.1.1 Oltremare dataset (in-domain)

Acoustic recordings were collected at the Oltremare marine park (Riccione, Italy) in November 2021, as described in Screpanti et al.[28] Seven free-swimming *Tursiops truncatus* were recorded using a UREC U384K autonomous recorder (Dodotronic/Nauta) equipped with a SQ26–05 hydrophone at a sampling rate of 192 kHz. Annotation was performed by a trained PAM operator following the labeling protocol described in Section 2.2, yielding 3,214 whistle segments (label W), 233 multiple whistle segments (label MW) and 2,545 non-whistle segments (labels NOISE, ECT, BPS, FB). Each annotated segment yields multiple spectrogram windows depending on its duration and the configured overlap. Since the primary focus of the labeling performed on the dataset was the detection of whistles, the other vocalizations (ECT, BPS, FB, and NOISE) were only labeled to a limited extent. In particular, a very small number of feeding buzzes were labeled (only 308 segments). The complete annotated dataset is publicly available.[34]

3.1.2 DCLDE 2022 cross-test dataset

For cross-domain evaluation of experiment A, the DCLDE Oahu dataset was used, which concerns acoustic data collected through a multi-channel towed array hydrophone during the Hawaiian Islands Cetacean and Ecosystem Assessment Survey (HICEAS) in 2017.[35] A single recording containing vocalizations of *T. truncatus* (Acoustic detection number 46) was identified in the dataset, dated July 15, 2017, from 11:09 PM to 11:24 PM (UTC). A specific track (n. 3) was extracted for testing purposes and annotated independently following the same protocol as the Oltremare dataset, yielding 458 whistle segments. This process resulted in 715 whistle spectrograms (positive class) and 1,054 non-whistle spectrograms (negative class). From this pool, 500 spectrograms were randomly sampled for each class— utilizing a fixed seed of 42—to establish a balanced test set for the cross-domain performance evaluation. No DCLDE material was used during training; the dataset is used exclusively for cross-domain evaluation.

3.2 Validation

Two experiments validate the framework on the Oltremare dataset. Both share the same baseline CNN architecture and training protocol; they differ in the number of classes, the spectrogram frequency range, and the augmentation strategy, all of which are controlled through the YAML configuration file without modifying any code.

Experiment A demonstrates the binary whistle detector through all six pipeline stages, including cross-domain evaluation. The positive class (C1) groups whistle and multiple-whistle annotations (W, MW); the negative class (C0) groups all co-occurring non-whistle vocalizations — background noise, feeding buzzes, burst-pulse sounds, and echolocation click trains (NOISE, FB, BPS, ECT). Defining C0 from the full range of non-target vocalizations, rather than from background noise alone, trains the detector to reject the acoustic events that co-occur with whistles in real recordings, which reduces the false-positive rate under operational conditions. In this operational context, a false positive carries a direct intervention cost, so minimizing the false discovery rate on unseen data is the primary design objective; recall is a secondary metric that quantifies detector sensitivity. Audio was segmented into 0.8 s windows with 50% overlap (Stage 2). Each window was rendered as a 224 × 224 grayscale spectrogram over 3–25 kHz with a 512-sample FFT and 50% STFT overlap (Stage 3), followed by per-image intensity normalization (Stage 4); no image-domain filters were applied. The model was trained under ten-fold stratified group cross-validation, balanced at 4,700 samples per class with seed 42 (Stage 5). Stage 6 evaluated each of the ten fold models independently on the DCLDE 2022 cross-domain test set.

Experiment B demonstrates the multiclass configuration of the same framework. Five mutually exclusive classes were defined from the Oltremare annotations: C0 (NOISE), C1 (W/MW), C2 (BPS), C3 (ECT), and C4 (FB). The frequency crop was widened to 3–96 kHz to capture the broadband structure of impulsive vocalizations. Because the feeding-buzz class (C4) had substantially fewer annotations than the others (~600 segments, from 308 single annotations), scroll augmentation was enabled for C4 alone, translating the analysis window in 0.1 s steps to generate multiple time-shifted copies of each event (augmentation_scroll: true, scroll_step: 0.1). The resulting dataset totaled C0: 8,154; C1: 5,834; C2: 1,792; C3: 2,406; C4: 3,694 windows. Training used 2,000 randomly sampled images per class (C2: 1,792, limited by annotation density) under ten-fold stratified group cross-validation. Stage 6 was not extended to an external dataset; performance is reported for in-domain held-out data only, as the purpose of Experiment B is to demonstrate framework scalability rather than to characterize cross-domain behavior.

Table 2 lists the YAML key parameters for experiment A.

3.3 Experiment A

3.3.1 In-domain performance (Oltremare)

Performance was assessed at two levels. In-domain ten-fold cross-validation on the Oltremare dataset estimates how the detector generalizes to held-out data from the same recording environment. Cross-domain evaluation on DCLDE 2022 — recordings made with different equipment, in the Pacific Ocean, with free-ranging animals — estimates how the same models behave under the domain shift that operational PAM systems routinely encounter. Table 3 reports summary statistics for both levels; Table 4 gives the per-class breakdown.

3.3.2 Cross-domain performance (DCLDE 2022)

On the DCLDE 2022 dataset, the ten fold models maintained near-perfect precision: the mean was 0.999 (± 0.002) with a false discovery rate of 0.001 (± 0.002). Nine of the ten folds produced no false positives; a single fold produced one false positive out of 500 negative test windows (Table 3). Recall dropped to 0.520 (± 0.083), meaning that roughly half of the DCLDE whistles were detected, while virtually every window the models flagged as a whistle was a true whistle.

This combination — maximal precision with reduced recall — is the expected signature of domain shift. The models were trained on close-range recordings of seven captive animals in a reverberant basin and tested on free-ranging dolphins recorded at sea with a towed array. The whistles that were not recovered are those whose spectrographic appearance departs most from the training distribution. For an edge-deployed deterrent system, where a false positive triggers an unnecessary intervention, a false discovery rate below 0.1% on entirely unseen data is the operationally critical result. The balanced composition of the test set (500 whistle and 500 non-whistle windows) means that these metrics characterize the detector's discrimination ability under controlled conditions; they do not directly predict the false-alarm rate in operational deployment, where non-target audio vastly outnumbers target vocalizations.

3.3.3 Threshold sensitivity

To assess whether the cross-domain recall of 0.520 at $\theta = 0.50$ reflects a fundamental model limitation or an effect of the decision threshold, the DCLDE evaluation was repeated at 19 threshold values from 0.05 to 0.95 (Figure 3). At $\theta = 0.05$, mean recall rose to 0.755 while precision remained above 0.95 (FDR = 0.046), indicating that a substantial fraction of DCLDE whistles produce sigmoid outputs above zero but below the default threshold.

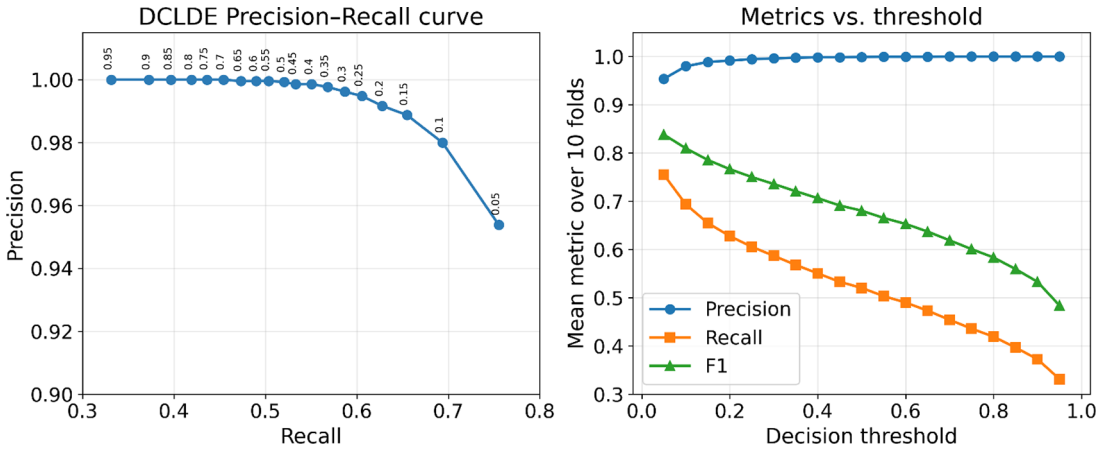

Figure 3. Precision-recall analysis for Experiment A cross-domain evaluation (DCLDE 2022 test set, grouped models, 10 folds). Left panel: mean precision plotted against mean recall across threshold values from θ = 0.05 to θ = 0.95 (step 0.05). Right panel: mean precision, recall, and F1 as a function of decision threshold θ. The dashed vertical line marks θ = 0.50.

The precision-recall curve (Figure 3, left panel) shows that precision remains above 0.99 for recall values up to ≈ 0.63 (θ = 0.20), then gradually decreases while staying above 0.95 up to recall ≈ 0.76. Beyond this point, the trade-off steepens.

The choice of operating point is application-dependent. For deterrence systems where each false positive carries an operational cost, the default θ = 0.50 maintains near-perfect precision. For survey-oriented monitoring where detection coverage is prioritized, a lower threshold recovers three quarters of the whistles with fewer than 5% false positives. The optimal threshold for a specific deployment should be calibrated on a held-out dataset representative of the target environment, rather than selected from the evaluation data used here.

3.4 Experiment B — Multiclass vocalization classification

Experiment B evaluates the same framework in a five-class configuration, demonstrating that the pipeline scales from binary detection to multiclass classification through configuration alone. Because the purpose is to show framework scalability, evaluation is limited to in-domain cross-validation; a cross-domain test was not conducted for this experiment.

3.4.1 In-domain performance

The multiclass model reached a macro F1 of 0.835 (± 0.026) across the ten stratified folds (Table 5). The noise (C0) and whistle (C1) classes retained near-identical performance to the binary setting, with F1 values of 0.930 and 0.930 respectively, confirming that expanding the task to five classes does not degrade whistle detection. The feeding-buzz class (C4) achieved an F1 of 0.808 (± 0.041) despite only ~600 original annotated segments; scroll augmentation generated 3,694 training windows from this material, providing sufficient class representation for stable training.

Lower performance was observed for *Burst Pulse Sounds* (C2: $F_1 = 70.9\% \pm 5.9\%$) and the *Echolocation Click Trains* (C3: $F_1 = 79.9\% \pm 3.3\%$). The confusion matrix (Figure 4) provides further insight into these results, revealing significant inter-class confusion between ECT and BPS. To a lesser extent, a similar pattern of misclassification is observable between BPS and Feeding Buzzes.

With the exception of a few cases involving Feeding Buzzes, other vocalizations were rarely misclassified as Whistles. This underlines the high specificity of the *Whistle* class, likely due to a more refined labeling process and a more distinctive spectral signature in the training dataset.

4. Discussion

4.1 Methodological considerations

CNN-based detection of cetacean vocalizations involves a large number of configurable parameters[36] at every stage of the workflow, from annotation conventions to spectrogram rendering and training hyperparameters. The value of a structured

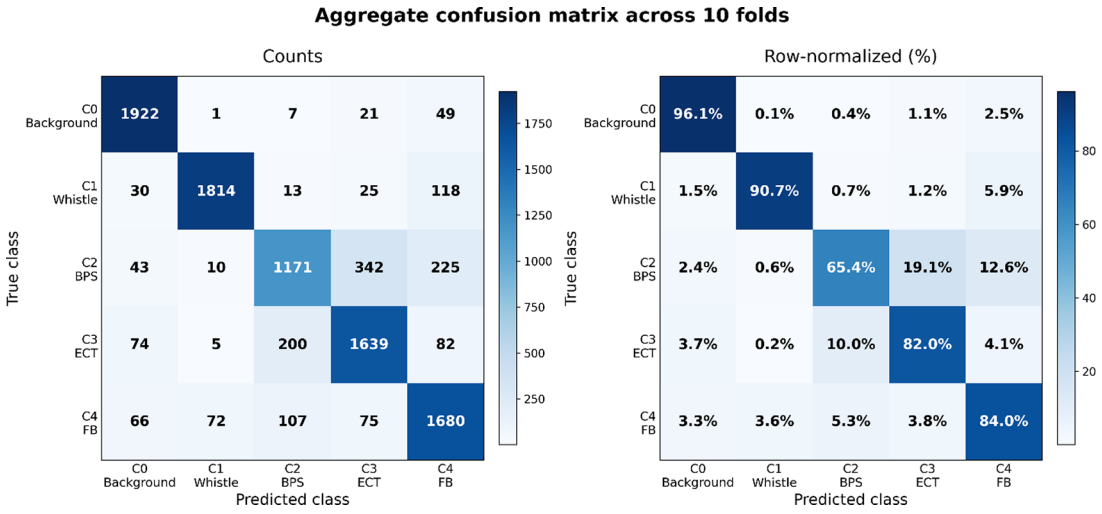

Figure 4. Normalized confusion matrix for Experiment B (five-class Oltremare multiclass demonstration). Rows represent true classes; columns represent predicted classes. Values are the sum of true-class instances assigned to each predicted class, across the 10 stratified folds. On right: Percentage values, normalized per row. Diagonal percentage entries correspond to per-class recall.

pipeline lies in making each of these choices explicit and reproducible, so that their individual and combined effects on detection performance can be studied systematically.

Two design choices in these experiments illustrate how the framework supports deliberate, documented decisions. In Experiment A, the negative class was assembled from the full set of non-whistle vocalizations rather than from background noise alone, making the training task harder but more representative of operational conditions. In Experiment B, scroll augmentation was applied selectively to the data-scarce feeding-buzz class, generating sufficient training material without artificially inflating the other classes. Both choices are encoded in the configuration file and reproduce exactly; neither required modifying any code. In a field where preprocessing decisions are often left implicit, recording them in a machine-readable form is itself a methodological contribution.

During peer review, it was identified that the original cross-validation partitioning operated at the window level, allowing temporally overlapping windows from the same recording to appear in both training and test partitions. This was corrected by implementing grouped cross-validation (StratifiedGroupKFold), with the source audio filename as the grouping key. A leakage diagnostic showed that 97.3% of C0 windows and 69.0% of C1 windows originated from annotated segments longer than the 0.8 s analysis window, producing thousands of overlapping window pairs. Despite this substantial exposure, the corrected in-domain F1 decreased by less than one percentage point (Exp A: −0.8 pp; Exp B: −0.6 pp), indicating that the CNN learned general spectro-temporal features rather than memorizing specific window content. The inter-fold standard deviation increased by a factor of approximately four (Exp A: from 0.004 to 0.017), reflecting the true variability across recording sessions. Running the same experiment with both grouped and ungrouped folds provides an independent diagnostic of CNN feature extraction quality — a use of the framework's configurable group_by parameter that extends beyond simple leakage prevention.

Table 5. Multiclass applicative demonstration (Oltremare, Experiment B): per-class and aggregate performance metrics, mean ± SD across 10 stratified grouped folds. Values are in percent.

Class	Label	Precision	Recall	F1
C0	NOISE	90.2 ± 3.6	96.1 ± 2.0	93.0 ± 1.5
C1	W/MW	95.6 ± 4.7	90.7 ± 3.1	93.0 ± 3.5
C2	BPS	78.6 ± 6.0	65.4 ± 8.7	70.9 ± 5.9
C3	ECT	78.5 ± 5.6	81.9 ± 6.0	79.9 ± 3.3
C4	FB	78.1 ± 4.7	84.1 ± 6.7	80.8 ± 4.1
Macro (unweighted)		84.2 ± 2.3	83.6 ± 2.4	83.5 ± 2.6

The YAML configuration should be understood in the context of existing metadata and framework standards. Training toolkits including koogu,[21] Ketos,[22] AnimalSpot,[23] and the PAMGuard[25] deep learning module cover model building and inference. Community efforts to standardize acoustic metadata — Tethys,[18] the ANSI/ASA S3/SC1.7 standard,[19,21] and PAM data management guidelines[20] — operate at the annotation and archive level. The ai-pam-pipeline configuration file complements these by recording the full chain of executable parameters — from annotation conventions through spectrogram rendering to cross-validation grouping and cross-domain evaluation — in a single human-readable file that is automatically archived alongside every output. The framework produces standard Keras model files that can be converted to TensorFlow Lite, ONNX, or other inference formats for deployment on external platforms including PAMGuard's deep learning module. Version 1.1 also includes a C-spectrogram engine that reproduces the exact spectrogram computation of a target edge device, ensuring training-serving consistency for embedded deployments (see Extended Data, S5.3).

4.2 Generalization gap and deployment implications

The separation between in-domain and cross-domain results is the central methodological point of Experiment A. In-domain cross-validation alone would suggest a near-solved problem, with a macro F1 of 0.960. The cross-domain evaluation (F1 of 0.744) reveals the more realistic figure: the same models recover roughly half the whistles in a genuinely independent recording. A pipeline that stops at in-domain validation would systematically overstate deployment performance; the gap measured here — approximately 22 percentage points of macro F1 — quantifies how large that overstatement can be.

The precision-recall analysis (Figure 3) reveals that the cross-domain recall at the default threshold understates the model's detection capacity. At $\theta = 0.05$, recall rises to 0.755 while precision remains above 0.95, indicating that a large fraction of DCLDE whistles activate the detector but produce output probabilities below the default threshold. The operating point should therefore be selected based on the acceptable trade-off between false alarms and missed detections for the target application.

In the dimension that matters most for this detector, the cross-domain result is strong. The mean FDR across folds was below 0.1%. For an edge-deployed deterrent system, where every false positive triggers an unnecessary intervention, this outcome is preferable to a higher-recall detector with non-zero false alarms. The balanced-set metrics reported here characterize the detector's discrimination ability under controlled conditions. In operational deployment, where non-target audio vastly outnumbers vocalizations, metrics such as false positives per hour of continuous recording would provide a more direct estimate of system performance. Such deployment-oriented evaluation is planned as a next step.

The domain shift between training and test environments is unusually large: controlled marine-park basin versus open-ocean towed array in a different ocean basin. That the detector maintains meaningful discrimination under these conditions is an encouraging result in its own right. Systematic investigation of how individual parameter choices affect cross-domain performance — including an N_fft sweep across multiple values — is enabled by the framework's design and is the subject of ongoing work. A comparison of the CNN detector against established classical approaches on the same datasets is planned as future work.

The framework is what makes this measurement routine. Because every parameter lives in a single configuration file, a practitioner who wants to reduce the generalization gap can modify the training set composition, the spectrogram band, or the class definitions, and re-run only the affected pipeline stages without revisiting annotation or rewriting code. Cross-domain evaluation becomes a standard step of the workflow rather than an exceptional one.

4.3 Limitations

Several limitations bound the scope of these results. First, both experiments use a single species (*T. truncatus*) and a single training site (Oltremare); the pipeline is designed to generalize through reconfiguration, but that generalization is asserted rather than demonstrated here. The training recordings come from a semi-enclosed marine-park basin with seven resident animals at close range; this controlled environment provides high signal-to-noise ratio but does not represent the acoustic variability of open-ocean conditions.

Second, the cross-domain evaluation used a single DCLDE recording track. It therefore demonstrates transfer to one independent recording but does not characterize performance variability across multiple sites, recording systems, seasons, or acoustic conditions. Cross-domain performance was measured on a balanced subsample. Deployment-oriented metrics such as false positives per hour were not computed. Third, the multiclass results (Experiment B) are purely in-domain; the ECT/BPS confusion pattern described in Section 3.4.1 may change substantially on recordings with different signal-to-noise ratios or individual variation. Finally, despite fixed seeds, small run-to-run numerical

differences can arise from multi-threaded CPU floating-point operations; exact bitwise reproduction may require fixing the hardware thread configuration in addition to the seed.

The framework is demonstrated here using single-channel, spectrogram-based CNN classification of *Tursiops truncatus* vocalizations. However, its YAML-based configuration supports application to other species and acoustically comparable vocalization types without changes to the pipeline code, by adjusting parameters such as class definitions, analysis-window duration, frequency range, STFT settings, image dimensions, and preprocessing operations. Its model plug-in interface also accommodates compatible architectures, including recurrent neural networks (RNNs), long short-term memory networks (LSTMs), convolutional recurrent neural networks (CRNNs), and vision transformers (ViTs), although these alternatives were not evaluated in the present study. Object-detection architectures, pixel-level segmentation methods, multichannel processing, and click detection operating directly on time-domain signals remain outside the framework's current scope.

5. Conclusions

This study presented ai-pam-pipeline, an open-source six-stage framework that formalizes the complete workflow from raw audio annotation to cross-domain CNN evaluation, with every parameter defined in a single YAML configuration file. Demonstrated on *Tursiops truncatus* vocalizations, a binary whistle detector achieved a macro F1 of 0.960 in-domain (ten-fold grouped cross-validation) and, on an independent open-ocean dataset, a precision of 0.999 with a macro F1 of 0.744. Precision-recall analysis showed that recall rises to 0.76 at a lower decision threshold while precision remains above 0.95, indicating that the model's cross-domain discrimination extends beyond the default operating point. A five-class classifier demonstrated scalability, achieving a macro F1 of 0.835 in-domain. The toolkit is released under the Apache 2.0 license (v1.1.0) and archived on Zenodo, enabling exact reproduction of these results and adaptation to other species through configuration alone. The framework was developed within the LIFE DELFI project for operational edge-deployed deterrent systems, and is offered to the broader PAM community as a foundation for reproducible detector development.

Software availability

Source code is available on GitHub: https://github.com/roccodm/ai-pam-pipeline

Archived version: De Marco R. *ai-pam-pipeline framework, v1.1.0.* Zenodo, 2026. https://doi.org/10.5281/zenodo.19860736[37]

License: Apache 2.0.

Ethics and consent

Ethical approval and consent were not required for this study. The acoustic recordings used in the experiments were collected as part of the LIFE DELFI project (LIFE19 NAT/IT/000942) and are publicly available. Data acquisition was conducted entirely in passive mode: a hydrophone recorder was positioned unobtrusively within the facility, out of direct view of the animals, and no interaction with, handling of, or intervention upon the recorded individuals took place at any stage. The dolphins were free-swimming in their normal environment throughout the recording sessions. No animal experimentation, invasive procedures, or behavioral manipulation of any kind were performed.

Software Implementation and CLI Usage

The following ten steps reproduce Experiment A (binary whistle detection, $N_{\text{fft}} = 512$) from a clean system with no prior installation. Estimated time: 15–20 minutes for installation and dataset preparation; training time depends on hardware (GPU recommended).

Step 1 — Download the toolkit.

Download the versioned source archive from Zenodo:

```
https://doi.org/10.5281/zenodo.19860736
```

Step 2 — Extract and enter the directory.

```
unzip ai-pam-pipeline-v1.0.zip cd ai-pam-pipeline
```

Step 3 — Create a virtual environment.

```
python -m venv.venv source.venv/bin/activate # Linux/macOS# .venv\Scripts
\activate # Windows
```

Step 4 — Install dependencies.

```
pip install -r requirements.txt
```

Core dependencies include TensorFlow, scipy, Pillow, matplotlib, and soundfile. GPU acceleration is available if a CUDA-compatible device is present and the appropriate TensorFlow GPU build is installed (see Section S1).

Step 5 — Download the Oltremare dataset.

Download the annotated Oltremare dataset from SEANOE:

```
https://doi.org/10.17882/109081
```

Step 6 — Place audio files.

Copy all `.wav` recording files into the `data/audio/` directory:

```
ai-pam-pipeline/
└── data/
    └── audio/
        ├── 20211120_102109_192.wav
        └── …
```

Step 7 — Place label files.

Copy the corresponding Audacity label files (`.txt`) into `data/labels/`. Each label file must share the same stem as its audio file:

```
ai-pam-pipeline/
└── data/
    └── labels/
        ├── 20211120_102109_192.txt
        └── …
```

Step 8 — Run preprocessing (Stages 1–4).

```
python ai_pam_preprocessing.py conf/binary_cnn.yaml
```

This command reads the configuration file `conf/binary_cnn.yaml`, processes all paired audio/label files, and exports segmented spectrogram images to `outputs/spectrograms/`. A summary report is printed to the console on completion.

Step 9 — Run training (Stages 5–6).

```
python ai_pam_train.py conf/binary_cnn.yaml -v
```

Step 10 — Inspect results.

Open the training summary report: `outputs/report/train/summary_report.md`.

This file contains cross-fold mean ± SD for all metrics, per-fold detail, and a fold-level confusion matrix. Individual `.keras` model files and `_metrics.json` records are available in `outputs/models/` for further analysis or cross-domain evaluation.

Data and software availability

Source data

- Oltremare annotated dataset used for model training: Di Nardo F, De Marco R, Scaradozzi D. Bottlenose dolphin vocalizations in controlled environments: a dataset for behavioral classification. SEANOE, 2025. https://doi.org/10.17882/109081[34]

- DCLDE 2022 Oahu towed array dataset used for cross-domain evaluation: NOAA Pacific Islands Fisheries Science Center. Hawaiian Islands Cetacean and Ecosystem Assessment Survey (HICEAS) towed array data. NOAA National Centers for Environmental Information, 2022. https://doi.org/10.25921/e12p-gj65[35]

Data are available under the terms of the Creative Commons Attribution 4.0 International license (CC BY 4.0).

Extended data

Technical documentation: De Marco, R. (2026). ai-pam-pipeline — Reference Manual (1.0.0). Zenodo. https://doi.org/10.5281/zenodo.20697433[38]

Data are available under the terms of the Creative Commons Attribution 4.0 International license (CC BY 4.0).

Acknowledgments

The author acknowledges the CNR-IRBIM HPC service, with special thanks to Fabrizio Moro and Paolo Scarpini, and the CINECA award under the ISCRA initiative (Italian SuperComputing Resource Allocation, project id: HP10CL9TFX), for the availability of high-performance computing resources and support.

The author wishes to express his special gratitude to Vanessa De Marco for her artistic contribution in designing Figure 1.

The author used AI-assisted tools during manuscript preparation and software development. Claude Sonnet 4.6, Google Gemini Flash 3.5 and Grammarly were used for language refinement of selected passages to improve clarity and readability. Claude Sonnet 4.6 was also used to assist in refactoring the framework codebase, specifically in reorganizing the original monolithic Python implementation into a modular multi-file architecture and in generating inline code documentation. All scientific content, experimental design, data analysis, and conclusions are the sole responsibility of the author. All AI-generated outputs were reviewed, verified, and revised by the author prior to inclusion.

Bibliography

1. Gibb R, Browning E, Glover-Kapfer P, *et al.*: **Emerging opportunities and challenges for passive acoustics in ecological assessment and monitoring.** *Methods Ecol. Evol.* 2019; **10**: 169–185.
Publisher Full Text

2. Mellinger DK, Stafford KM, Moore SE, *et al.*: **An overview of fixed passive acoustic observation methods for cetaceans.** *Oceanography.* 2007; **20**(4): 36–45.
Publisher Full Text

3. Van Parijs SM, Clark CW, Sousa-Lima RS, *et al.*: **Management and research applications of real-time and archival passive acoustic sensors over varying temporal and spatial scales.** *Mar. Ecol. Prog. Ser.* 2009; **395**: 21–36.
Publisher Full Text

4. Gonzalvo J, Carpentieri P: *Depredation by marine mammals in fishing gear — a review of the Mediterranean Sea, Black Sea and contiguous Atlantic area. FAO Studies and Reviews No. 102.* Rome: FAO; 2023.
Publisher Full Text

5. Zimmer WMX: *Passive acoustic monitoring of cetaceans.* Cambridge: Cambridge University Press; 2011.
Publisher Full Text

6. Gillespie D, Caillat M, Gordon J, *et al.*: **Automatic detection and classification of odontocete whistles.** *J. Acoust. Soc. Am.* 2013; **134**(3): 2427–2437.
PubMed Abstract | Publisher Full Text

7. Serra O, Martins F, Padovese LR: **Active contour-based detection of estuarine dolphin whistles in spectrogram images.** *Eco. Inform.* 2020; **55**: 101036.
Publisher Full Text

8. Siddagangaiah S, Chen CF, Hu WC, *et al.*: **Automatic detection of dolphin whistles and clicks based on entropy approach.** *Ecol. Indic.* 2020; **117**: 106559.
Publisher Full Text

9. Jiang J, Bu L, Duan F, *et al.*: **Whistle detection and classification for whales based on convolutional neural networks.** *Appl. Acoust.* 2019; **150**: 169–178.
Publisher Full Text

10. Shiu Y, Palmer KJ, Roch MA, *et al.*: **Deep neural networks for automated detection of marine mammal species.** *Sci. Rep.* 2020; **10**: 607.
PubMed Abstract | Publisher Full Text | Free Full Text

11. Li L, Qiao G, Liu S, *et al.*: **Automated classification of *Tursiops aduncus* whistles based on a depth-wise separable convolutional neural network and data augmentation.** *J. Acoust. Soc. Am.* 2021; **150**(5): 3861–3873.
PubMed Abstract | Publisher Full Text

12. Li P, *et al.*: **Learning Deep Models from Synthetic Data for Extracting Dolphin Whistle Contours**. 2020 *International Joint Conference on Neural Networks (IJCNN).* Glasgow, UK, **2020**; 1–10.
Publisher Full Text

13. Li P, Liu X, Klinck H, *et al.*: **Using deep learning to track time × frequency whistle contours of toothed whales without human-annotated training data.** *J. Acoust. Soc. Am.* 1 July 2023; **154**(1): 502–517.
PubMed Abstract | Publisher Full Text

14. White EL, White PR, Bull JM, *et al.*: **More than a whistle: automated detection of marine sound sources with a convolutional neural**

network. *Front Mar Sci.* 2022; **9**: 879145.
Publisher Full Text

15. Jin C, Kim M, Jang S, *et al.*: **Semantic segmentation-based whistle extraction of Indo-Pacific bottlenose dolphin.** *Ecol. Indic.* 2022; **137**: 108792.
Publisher Full Text

16. Nur Korkmaz B, Diamant R, Danino G, *et al.*: **Automated detection of dolphin whistles with convolutional networks and transfer learning.** *Front Artif Intell.* 2023; **6**: 1099022.
PubMed Abstract | Publisher Full Text | Free Full Text

17. Stowell D: **Computational bioacoustics with deep learning: a review and roadmap.** *PeerJ.* 2022; **10**: e13152.
PubMed Abstract | Publisher Full Text | Free Full Text

18. Roch MA, Batchelor H, Baumann-Pickering S, *et al.*: **Management of acoustic metadata for bioacoustics.** *Ecol Inform.* 2016; **31**: 122–136.
Publisher Full Text

19. Roch MA, Baumann-Pickering S: **Standard for acoustic metadata for passive acoustic monitoring.** *J Acoust Soc Am.* 2019; **146**: 2935.
Publisher Full Text

20. Wall CC, Haver SM, Hatch LT, *et al.*: **The next wave of passive acoustic data management: how centralized access can enhance science.** *Front Mar Sci.* 2021; **8**: 703682.
Publisher Full Text

21. Madhusudhana S: **Koogu (v0.7.2) [software].** *Zenodo.* 2023.
Publisher Full Text

22. Kirsebom OS, Frazão F, Padovese B, *et al.*: **Ketos — a deep learning package for creating acoustic detectors and classifiers.** *J Acoust Soc Am.* 2021; **150**(4_Supplement): A164.
Publisher Full Text

23. Bergler C, Smeele SQ, Tyndel SA, *et al.*: **ANIMAL-SPOT enables animal-independent signal detection and classification using deep learning.** *Sci Rep.* 2022; **12**: 21966.
PubMed Abstract | Publisher Full Text | Free Full Text

24. Sugarman PC, Ferguson EL, Alongi GC, *et al.*: **Effects of network selection and acoustic environment on bounding-box object detection of delphinid whistles using a deep learning tool.** *J Acoust Soc Am.* 2025; **157**(6): 4613–4627.
PubMed Abstract | Publisher Full Text

25. Gillespie D, Macaulay J, Oswald M, *et al.*: **PAMGuard: application software for passive acoustic detection, classification, and localisation of animal sounds.** *J Acoust Soc Am.* 2026; **159**(1): 437–443.
Publisher Full Text

26. Nguyen Hong Duc P, Torterotot M, Samaran F, *et al.*: **Assessing inter-annotator agreement from collaborative annotation campaign in marine bioacoustics.** *Eco. Inform.* 2021; **61**: 101185.
Publisher Full Text

27. Luís AR, Couchinho MN, dos Santos ME: **A quantitative analysis of pulsed signals emitted by wild bottlenose dolphins.** *PLoS ONE.* 2016; **11**(7): e0157781.
PubMed Abstract | Publisher Full Text | Free Full Text

28. Screpanti L, Di Nardo F, De Marco R, *et al.*: **Acoustic analysis of bottlenose dolphin vocalizations for behavioral classification in controlled settings.** *PLoS ONE.* 2025; **20**.
PubMed Abstract | Publisher Full Text | Free Full Text

29. Scaradozzi D, De Marco R, Li Veli D, *et al.*: **Convolutional neural networks for enhancing detection of dolphin whistles in a dense acoustic environment.** *IEEE Access.* 2024; **12**: 127141–127148.
Publisher Full Text

30. De Marco R, Di Nardo F, Rongoni A, *et al.*: **Real-time dolphin whistle detection on Raspberry Pi Zero 2 W with a TFLite convolutional neural network.** *Robotics.* 2025; **14**(5): 67.
Publisher Full Text

31. Jing J, Liu S, Wang G, *et al.*: **Recent advances on image edge detection: a comprehensive review.** *Neurocomputing.* 2022; **503**: 259–271.
Publisher Full Text

32. Baylor D, Breck E, Cheng HT, *et al.*: **TFX: a TensorFlow-based production-scale machine learning platform.** *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* Halifax; 2017; pp. 1387–1395.
Publisher Full Text

33. Ulutas G, Ustubioglu B: **Underwater image enhancement using contrast limited adaptive histogram equalization and layered difference representation.** *Multimed. Tools Appl.* 2021; **80**: 15067–15091.
Publisher Full Text

34. Di Nardo F, De Marco R, Scaradozzi D: Bottlenose dolphin vocalizations in controlled environments: a dataset for behavioral classification. [dataset]. SEANOE; 2025.
Publisher Full Text

35. NOAA Pacific Islands Fisheries Science Center: Hawaiian Islands Cetacean and Ecosystem Assessment Survey (HICEAS) towed array data. [dataset]. NOAA National Centers for Environmental Information; 2022.
Publisher Full Text

36. Gundersen OE, Kjensmo S: **State of the art: reproducibility in artificial intelligence.** *Proceedings of the AAAI Conference on Artificial Intelligence.* 2018; vol. **32**(1): pp. 1644–1651.
Publisher Full Text

37. De Marco R: *ai-pam-pipeline framework, v1.0.0 [software].* Zenodo; 2026.
Publisher Full Text

38. De Marco R: **ai-pam-pipeline framework, v1.1.0 [software].** *Zenodo.* 2026.
Publisher Full Text

Open Peer Review

Current Peer Review Status: ? ?

Version 1

Reviewer Report 01 September 2026

https://doi.org/10.21956/openreseurope.26352.r78481

© **2026 Macaulay J.** This is an open access peer review report distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

?

Jamie Macaulay
[1] University of St Andrews, St Andrews, UK
[2] University of St Andrews, St Andrews, UK

Overall
The author has created a Python package for training deep learning models, in which a single YAML file defines the metadata for an experiment, including annotation handling and pre-processing steps. YAML is relatively simple and human-readable, which makes changing these initial steps straightforward.

The article is generally understandable and very well written (the author notes that AI tools were involved in its preparation). It is detailed and accessible and so, as well as reporting the author's work, would make a good learning resource for students. The author also spends time on cross-domain validation, which is generally lacking in the current literature, although the results do indicate that the classifier is not particularly well suited to the domain shift that was evaluated (it was a pretty large domain shift from captive animals in a tank to wild animals though).

There are several issues that need to be addressed: sweeping statements about PAM that are not supported by evidence, the framing of the contribution relative to existing work, referencing of past work, a figure that does not match its caption, and the assessment of classifier efficacy across decision thresholds. Most importantly, it is not clear from the manuscript that the cross-validation folds were grouped in a way that prevents overlapping and augmented copies of the same annotated event appearing in both the training and test sets; if they were not, the headline in-domain result is inflated. The author should address these points before the paper is finalised.

The paper is suitable for Open Research Europe; it is a good engineering write-up rather than a significant methodological advance and with the changes proposed below it will be suitable for indexing.

Major comments
1. **Grouping of samples across cross-validation folds.** Section 2.3 segments the audio into

0.8 s windows with 50% overlap, and Section 2.3 and Section 3.2 describe scroll augmentation generating multiple time-shifted copies of each annotated event. Section 2.7.1 then states that fold partitioning is "performed by random assignment with a fixed seed (seed 42)", with no mention of grouping by annotation, event, or source file. Is therefore the case that overlapping windows and augmented copies derived from the same vocalization can be distributed across different folds, so that near duplicates of test samples appear in training? If that is the case, the reported in-domain macro F1 of 0.966 is unrealistic and is not a valid estimate of held-out performance; it may also account for some of the apparent size of the in-domain to cross-domain gap. Please state at what level the splits are made. If splitting is at the window level, the in-domain cross-validation should be re-run with grouping applied at the annotation-event level (and preferably at the file level), and the revised figures reported. Given that reproducibility is the central claim of the paper, this needs to be resolved and documented in the configuration schema.

2. **Fixed decision threshold and the absence of precision–recall curves.** The decision threshold is fixed at $\theta = 0.5$ throughout (Section 2.7.1, Table 2) and is described as "the balanced operating point of the sigmoid output", but no justification is given for using this value on the cross-domain data and no precision recall curve is reported. This matters because the paper's central conclusion rests on the cross-domain recall of 0.545: as it stands, we cannot distinguish a model that generalises poorly from a model that is being read at an unsuitable operating point. Precision–recall curves for both the in-domain and DCLDE evaluations should be shown. If recall at a lower threshold rises substantially while precision remains high, that changes the interpretation of the whole of Section 4.2.
3. **The zero false-positive result is not operationally informative as presented.** The cross-domain test set is a balanced sample of 500 positive and 500 negative windows (Section 3.1.2). In real PAM deployments the negatives outnumber the positives by orders of magnitude, so a per-window false discovery rate measured on a balanced set says very little about the rate of false alarms that an operator would actually experience. A far more informative statistic is the number of false positives per hour of recording, which is a convention in the detection literature the author cites. This can be obtained by running the detector over a longer continuous section of the DCLDE data rather than a balanced subsample. I agree with Reviewer 1 that the wording should be softened, but I would go further: the issue is the choice of metric, not only the phrasing. The claim in the Plain Language Summary that the detector "never produced a false alarm" should be removed.
4. **Positioning of the YAML configuration relative to existing standards and frameworks.** The author is right that a good metadata scheme for the training-to-prediction workflow is important; however, the importance and novelty of this contribution are overstated. Metadata standards for annotations already exist, for example Tethys and the recent ANSI/ASA S3/SC1.7 standard for acoustic metadata, as do training frameworks for bioacoustics such as Koogu, Ketos and AnimalSpot, all of which define their own metadata conventions, some more comprehensive than others. PAMGuard, which is mentioned in the manuscript but not cited, also has a standard for describing the pre-processing steps attached to a model, and Raven is widely used for manual annotation with its own text-based format. There is nothing inherently wrong with the existence of multiple standards, but that is what this YAML schema is, another one, rather than something new. I would suggest this is reframed accordingly, with the existing work cited and the specific advantages of the proposed schema stated explicitly. Allowing users to import or convert annotations from other formats would be a valuable addition and would considerably increase the practical reach of the toolkit.

5. **Scope of the framework - what it does and does not accommodate.** The abstract states that the framework can be adapted to different species, vocalization types and acoustic environments through configuration alone. In practice the design is clearly built around single-channel CNN classification of spectrogram images, and it is not obvious how it would extend to several approaches that are in common use. Object-detection architectures such as YOLO produce multiple bounding boxes per segment rather than one label per window; segmentation approaches generate a mask that is then post-processed, for example by graph search, to extract contours (some of this has been referenced); multi-channel input is used routinely in towed-array and localisation work; and click-based detection, including AI-based click train classifiers, does not operate on spectrogram images at all. I am not suggesting the author implement these, but claims should be narrowed to what the framework actually supports, with the out-of-scope cases named. The paper also says very little about how trained models and their pre-processing parameters are exported for use in other software; given that the stated aim is to serve the wider PAM community, a short section on interoperability with existing software would strengthen it.
6. **Figure 3 does not match its caption, and contains an unclaimed result.** Figure 3 is captioned as a normalized confusion matrix for Experiment B, but shows box plots of F1 and recall against FFT length for both datasets. The confusion matrix referred to in Sections 3.4.1 and 4 is therefore missing and needs to be added, and the FFT-length figure needs its own caption. This should have been caught by a proofread. It's also of note that the FFT result is perhaps one of the most interesting finding in the paper but is currently just ignored?? An FFT length of 256 appears to give markedly better cross-domain F1 and recall than 512 or 1024, while the in-domain results are essentially flat, which is exactly the kind of quantitative evidence on how pre-processing choices affect detection performance that the Introduction says the field lacks. I would encourage the author to promote this to a proper analysis with methods, results and discussion.
7. **Missing literature and unsupported generalisations.** The Introduction needs more references and contains several sweeping statements that should be removed or evidenced; specific instances are listed in the minor comments below. Two examples stand out. First, the claim that missing whistles are the more consequential error (Section 2.7.2) is contradicted in Section 4.2, where zero false positives is argued to be the operationally critical result. Much of PAM is context dependent; the performance required for real-time mitigation, density estimation and behavioural study are all different and the manuscript should say which context it is optimising for and be consistent about it. Reviewer 1 makes the same point. Second, several statements about what the field currently does are asserted without support; the relevant frameworks and detection approaches listed above should be cited rather than passed over.
8. **Optional: comparison against a classical whistle detection algorithm.** This is not required for publication, but a comparison against a well-established non-AI whistle detector, for example the PAMGuard Whistle and Moan detector, on the same in-domain and DCLDE material would be a valuable addition. My expectation is that it would be less accurate in domain but more stable across the domain shift, which would be an informative result either way and would inform the generalisation question the paper raises.

Minor comments

Suggested references are neither mandatory nor exhaustive, but the author does need to add more references and examples in the Introduction. Quoted manuscript text is given in italics.

Section 1.1

1. The following statement needs rewording:
"Passive Acoustic Monitoring (PAM) has become a widely-used, non-invasive tool for tracking acoustically active species over time"
PAM is used for spatial and temporal analysis of species. "Tracking" suggests animal movement, which PAM can be used for but rarely is.

2. Please rephrase:
"Marine mammals produce a variegated repertoire of vocalizations"

3. Change "frequently" to "sometimes":
"wide frequency bands, frequently in habitats heavily disrupted by intense human noise"

4. Please rephrase:
"Visual surveys are limited by weather conditions and human bias"
PAM is also subject to human bias in annotation, and drone surveys combined with AI can take humans largely out of the loop.

5. Too narrow a focus
"The push is to expand PAM networks to map cetacean distributions and behaviors, which is particularly vital for minimizing harmful interactions with commercial fisheries."
Why single out bycatch here? These data are useful for a whole host of purposes: monitoring population levels, studying the impacts of windfarm construction and fossil fuel development, supporting the implementation of MPAs, and so on. PAM for bycatch study is relatively undeveloped, albeit with a lot of potential.

6. Not true and should be corrected:
"whether analyzed by humans or algorithms, the processing pipeline relies on a shared visual foundation: the spectrogram."
A great deal of PAM does not involve a spectrogram. The PAMGuard click detector module is widely used in marine bioacoustics, and C-PODs and F-PODs are ubiquitous and store no raw audio or spectral data at all.

7. This needs evidence or should be softened:
"Today, however, deep learning approaches have largely taken over, with convolutional neural networks (CNNs) analyzing spectrogram images directly to deliver far superior detection accuracy."
Have they? Where is the evidence or the references? Deep learning shows a lot of promise, but many researchers still use older non-AI approaches, partly because of software integration and partly because the domain shift issue with AI is such a problem. Heuristic approaches tend to be more stable or at least can be modified more easily.

Section 1.2

8. There are a number of relevant whistle detection and classification papers not mentioned here, including Li et al. (2020), Li et al. (2023) and White et al. (2022); see the reference list below. If the author takes up the comparison with classical algorithms suggested in major comment 8, Gillespie et al. (2013) and Gruden and White (2020) should also be cited.

9. On inter-annotator variability:
"Inter-annotator variability in marine PAM contexts has been shown to depend on both vocalization type and signal-to-noise ratio, introducing a source of performance variance that is rarely quantified in comparative studies. The proposed pipeline addresses this gap by formalizing an annotation protocol with explicit decision criteria for each label class."
This is a long-recognised problem for which metadata standards have already been developed. Please rephrase and add references; the ANSI/ASA standard, Roch et al. (2016) and Wall et al. (2021) are the obvious starting points.

Section 2.1

10. This statement requires evidence:
"In practice, the full pipeline is rarely described in its entirety, let alone standardized"
Sweeping generalisations require evidence, and again, other frameworks already exist that do much of this.

Section 2.2

11. PAMGuard is referred to here but not cited. Please add Gillespie et al. (2026).

12. I agree with Reviewer 1 that the criteria for event boundaries, minimum duration and the maximum permitted gap for grouping consecutive signals into a single annotation need to be stated. This is particularly important for click trains and consecutive whistles, and it bears directly on the class confusion discussed in Section 3.4.1.

Section 2.5

13. On image-domain pre processing steps:
"it remains relatively unexplored within the context of cetacean vocalization detection."
I am not sure this is true; other frameworks use these techniques. See, for example, the DeepAcoustics work of Sugarman et al. (2025).

Section 2.

14. The following overstates the position:
"CNNs operating on spectrogram images represent the current standard for automated vocalization detection in PAM."
The current standard is a mix of approaches, and spectrogram CNNs are only one part of it. Section 2.7

15. On the fixed threshold — see major comment 2:
"decision threshold is fixed at $\theta = 0.5$, the balanced operating point of the sigmoid output."
Precision–recall plots against threshold would be informative here.

Section 3.1.1

16 It would be worth stating explicitly in the Limitations that the training material comes from

seven animals in a marine park basin, which is a reverberant, close-range, low-diversity acoustic environment. This is a fairly extreme starting point for a domain shift comparison, and saying so would help the reader interpret the size of the gap reported in Section 4.2; it is closer to a worst case than a typical one...

Section 3.4.1

17. The figure is incorrect but the idea that lower FFT's somehow make the domain shift less of an issue is intriguing and fits exactly with the theme of this paper - how pre-processing steps are important...it needs to be discussed in text. See longer comment above.

Section 4.2

*18. quantifies how large that overstatement can be. Y*es, but also the author should give themselves some credit here - the domain shift from a marine park to the Pacific ocean is pretty huge!

Some suggested references

These are suggestions rather than requirements, and the list is not exhaustive.

Metadata standards
Acoustical Society of America and American National Standards Institute (2026) S3/SC1.7-2026: Standard for Acoustic Metadata for Passive Acoustic Monitoring. Washington DC: American National Standards Institute. Committee: M. A. Roch (chair), S. Baumann-Pickering, D. Cholewiak, S. Guan, J. Kanes, K. H. Kim, H. Klinck, X. Mouy, A. Širović and C. C. Wall.

Roch, M. A., Batchelor, H., Baumann-Pickering, S., Berchok, C. L., Cholewiak, D., Fujioka, E., Garland, E. C., Herbert, S., Hildebrand, J. A., Oleson, E. M., Van Parijs, S. M., Risch, D. and Širović, A. (2016) 'Management of acoustic metadata for bioacoustics', Ecological Informatics, 31, pp. 122–136.

Wall, C. C., Haver, S. M., Hatch, L. T., Miksis-Olds, J., Bochenek, R., Dziak, R. P. and Gedamke, J. (2021) 'The next wave of passive acoustic data management: how centralized access can enhance science', Frontiers in Marine Science, 8, 703682. Available at: https://doi.org/10.3389/fmars.2021.703682.

Software and frameworks

Gillespie, D. et al. (2026) 'PAMGuard: application software for passive acoustic detection, classification, and localisation of animal sounds', The Journal of the Acoustical Society of America, 159(1), pp. 437–443. Available at: https://doi.org/10.1121/10.0042245.

Sugarman, P. C., Ferguson, E. L., Alongi, G. C., Pettis Schallert, J. and Lyn, H. (2025) 'Effects of network selection and acoustic environment on bounding-box object detection of delphinid whistles using a deep learning tool', The Journal of the Acoustical Society of America, 157(6), pp. 4613–4627. Available at: https://doi.org/10.1121/10.0036942.

Whistle detection and classification

Li, P. et al. (2020) 'Learning deep models from synthetic data for extracting dolphin whistle contours'. Available at: https://doi.org/10.48550/ARXIV.2005.08894.

Li, P. et al. (2023) 'Using deep learning to track time × frequency whistle contours of toothed whales without human-annotated training data', The Journal of the Acoustical Society of America, 154(1), pp. 502–517. Available at: https://doi.org/10.1121/10.0020274.

White, E. L. et al. (2022) 'More than a whistle: automated detection of marine sound sources with a convolutional neural network', Frontiers in Marine Science, 9, 879145. Available at: https://doi.org/10.3389/fmars.2022.879145.

Classical detection algorithms

Gillespie, D. et al. (2013) 'Automatic detection and classification of odontocete whistles', The Journal of the Acoustical Society of America, 134(3), pp. 2427–2437. Available at: https://doi.org/10.1121/1.4816555.

Gruden, P. and White, P. R. (2020) 'Automated extraction of dolphin whistles — a sequential Monte Carlo probability hypothesis density approach', The Journal of the Acoustical Society of America, 148(5), pp. 3014–3026. Available at: https://doi.org/10.1121/10.0002257.

References

1. M. A. Roch (chair), S. Baumann-Pickering, D. Cholewiak, S. Guan, J. Kanes, K. H. Kim, H. Klinck, X. Mouy, A. Širović and C. C. Wall: Standard for Acoustic Metadata for Passive Acoustic Monitoring. *https://webstore.ansi.org/standards/asa/asaansis3sc12026*.
2. Roch, M. A., Batchelor, H., Baumann-Pickering, S., Berchok, C. L., Cholewiak, D., Fujioka, E., Garland, E. C., Herbert, S., Hildebrand, J. A., Oleson, E. M., Van Parijs, S. M., Risch, D. and Širović, A.: Management of acoustic metadata for bioacoustics. *https://www.sciencedirect.com/science/article/pii/S1574954115001983*.
3. Wall C, Haver S, Hatch L, Miksis-Olds J, et al.: The Next Wave of Passive Acoustic Data Management: How Centralized Access Can Enhance Science. *Frontiers in Marine Science*. 2021; **8**. Publisher Full Text
4. Sugarman P, Ferguson E, Alongi G, Schallert J, et al.: Effects of network selection and acoustic environment on bounding-box object detection of delphinid whistles using a deep learning tool. *The Journal of the Acoustical Society of America*. 2025; **157** (6): 4613-4627 Publisher Full Text
5. Gillespie D, Macaulay J, Oswald M, Roch M: PAMGuard: Application software for passive acoustic detection, classification, and localisation of animal sounds. *The Journal of the Acoustical Society of America*. 2026; **159** (1): 437-443 Publisher Full Text
6. Li, P. et al.: Learning deep models from synthetic data for extracting dolphin whistle contours. *https://www.researchgate.net/publication/347020144_Learning_Deep_Models_from_Synthetic_Data_for_Extracting_Dolph*
.
7. Li P, Liu X, Klinck H, Gruden P, et al.: Using deep learning to track time × frequency whistle contours of toothed whales without human-annotated training data. *The Journal of the Acoustical Society of America*. 2023; **154** (1): 502-517 Publisher Full Text
8. White E, White P, Bull J, Risch D, et al.: More than a whistle: Automated detection of marine sound sources with a convolutional neural network. *Frontiers in Marine Science*. 2022; **9**. Publisher

Full Text
9. Gillespie D, Caillat M, Gordon J, White P: Automatic detection and classification of odontocete whistles. *The Journal of the Acoustical Society of America*. 2013; **134** (3): 2427-2437 Publisher Full Text
10. Gruden P, White P: Automated extraction of dolphin whistles—A sequential Monte Carlo probability hypothesis density approach. *The Journal of the Acoustical Society of America*. 2020; **148** (5): 3014-3026 Publisher Full Text

Is the rationale for developing the new method (or application) clearly explained?
Yes

Is the description of the method technically sound?
Yes

Are sufficient details provided to allow replication of the method development and its use by others?
Yes

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?
Yes

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?
Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Experienced with multiple types of PAM, PAMGuard developer, bioacoustic researcher, bycatch researcher.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 24 Sep 2026

Rocco De Marco

I wish to express my sincere gratitude to both reviewers for the exceptional quality and depth of their evaluation.
Dr. Macaulay's review identified a data leakage issue in the cross-validation design, which has been corrected both in an updated software release (v1.1.0) and in all reported results. Both reviews reflect a level of professional commitment and domain expertise, and the revised manuscript is substantially stronger as a direct result of their work. Their feedback has been formative, not merely corrective. I thank the reviewer for the thorough and constructive evaluation, and for the recognition that the work addresses a genuine gap in cross-domain validation practice. The characterization of the domain shift as "pretty large"

is accurate and is now explicitly acknowledged in the revised Discussion (§4.2): the training and test environments — a controlled marine-park basin versus an open-ocean towed array — represent an unusually demanding generalization test, which makes the near-perfect precision on unseen data a meaningful result even with reduced recall. According with the reviewer's assessment, the framework's value lies in making documented, reproducible engineering decisions routine rather than exceptional. All issues identified — sweeping statements, literature gaps, the incorrect figure, threshold analysis, and most importantly the cross-validation grouping — have been addressed in the revision, as detailed below. An updated version of the article has just been submitted; I do not know the timeline for its release.

Major comments
M1 - reviewer: Grouping of samples across cross-validation folds. Section 2.3 segments the audio into 0.8 s windows with 50% overlap, and Section 2.3 and Section 3.2 describe scroll augmentation generating multiple time-shifted copies of each annotated event. Section 2.7.1 then states that fold partitioning is “performed by random assignment with a fixed seed (seed 42)”, with no mention of grouping by annotation, event, or source file. Is therefore the case that overlapping windows and augmented copies derived from the same vocalization can be distributed across different folds, so that near duplicates of test samples appear in training? If that is the case, the reported in-domain macro F1 of 0.966 is unrealistic and is not a valid estimate of held-out performance; it may also account for some of the apparent size of the in-domain to cross-domain gap. Please state at what level the splits are made. If splitting is at the window level, the in-domain cross-validation should be re-run with grouping applied at the annotation-event level (and preferably at the file level), and the revised figures reported. Given that reproducibility is the central claim of the paper, this needs to be resolved and documented in the configuration schema.

Author answer: The reviewer is correct, and I am grateful for this identification, which has materially improved the reliability of the reported metrics. In the original implementation, fold partitioning was performed at the window level, which allowed overlapping and augmented copies of the same source event to appear in both training and test sets. I have corrected this by implementing `StratifiedGroupKFold` (scikit-learn), which groups all windows derived from the same source audio file into the same fold, ensuring zero overlap between train and test at the recording level. A leakage diagnostic confirmed that 97.3% of C0 windows and 69.0% of C1 windows originated from annotated segments longer than the 0.8 s analysis window, producing thousands of overlapping window pairs — yet the corrected results show a modest F1 decrease: - Experiment A: Macro F1 from 0.968 ± 0.004 to 0.960 ± 0.017 (−0.8 pp)
- Experiment B: Macro F1 from 0.841 ± 0.016 to 0.835 ± 0.026 (−0.6 pp) The small magnitude of the correction, despite the large leakage exposure, indicates that the CNN learned general spectro-temporal features rather than memorizing specific windows. This is consistent with the three max-pooling layers in the architecture, which confer intrinsic invariance to the sub-second temporal shifts between overlapping windows. The inter-fold standard deviation increased by a factor of approximately four (Exp A: from 0.004 to 0.017), reflecting the true variability across recording sessions that the legacy split had artificially suppressed. Running the same experiment with both grouped and ungrouped folds also serves as an independent diagnostic of CNN feature extraction quality: a large difference would indicate overfitting to

window-level patterns, while a small difference — as observed here — confirms that the model generalizes from acoustic features rather than memorized content. This paired comparison is a use of the framework's configurable `group_by` parameter that extends beyond simple leakage prevention. Cross-domain results were recomputed with the corrected models: Macro F1 = 0.744 ± 0.049, precision = 0.999 ± 0.002. The cross-domain evaluation was never affected by the leakage (physically separate datasets) but has been re-run for completeness. The grouping parameter `group_by` has been added to the YAML configuration schema (default: `"stem"`) and is documented in the updated Reference Manual (Extended Data, S7.3.1). The software repository has been updated to v1.1.0 (DOI: 10.5281/zenodo.19860736). All tables and text have been updated with the corrected figures.

M2 - reviewer: Fixed decision threshold and the absence of precision–recall curves. The decision threshold is fixed at θ = 0.5 throughout (Section 2.7.1, Table 2) and is described as "the balanced operating point of the sigmoid output", but no justification is given for using this value on the cross-domain data and no precision recall curve is reported. This matters because the paper's central conclusion rests on the cross-domain recall of 0.545: as it stands, we cannot distinguish a model that generalises poorly from a model that is being read at an unsuitable operating point. Precision–recall curves for both the in-domain and DCLDE evaluations should be shown. If recall at a lower threshold rises substantially while precision remains high, that changes the interpretation of the whole of Section 4.2.

Author answer: I agree that precision-recall analysis is essential for interpreting the cross-domain results, and I thank the reviewer for this guidance. I have generated PR curves by sweeping θ from 0.05 to 0.95 on the DCLDE cross-domain test set using the corrected (grouped) models. The results (new Figure 3) show that the model's cross-domain discrimination is substantially better than the default operating point suggests: at θ = 0.05, recall rises from 0.520 to 0.755 while mean precision remains above 0.95 (FDR = 0.046). This indicates that a large proportion of DCLDE whistles produce sigmoid outputs well above zero but below the default 0.50 threshold. The choice of operating point is now presented as application-dependent: for deterrence systems, θ = 0.50 maintains near-perfect precision; for survey monitoring, a lower threshold recovers three quarters of the whistles with fewer than 5% false positives. I note that the optimal threshold for a specific deployment should be selected on a held-out calibration set rather than on the test data used here. The PR analysis has been added as a new subsection (§3.3.3) with Figure 3, and discussed in §4.2.

M3 - reviewer: The zero false-positive result is not operationally informative as presented. The cross-domain test set is a balanced sample of 500 positive and 500 negative windows (Section 3.1.2). In real PAM deployments the negatives outnumber the positives by orders of magnitude, so a per-window false discovery rate measured on a balanced set says very little about the rate of false alarms that an operator would actually experience. A far more informative statistic is the number of false positives per hour of recording, which is a convention in the detection literature the author cites. This can be obtained by running the detector over a longer continuous section of the DCLDE data rather than a balanced subsample. I agree with Reviewer 1 that the wording should be softened, but I would go further: the issue is the choice of metric, not only the phrasing. The claim in the Plain Language Summary that the detector "never produced a false alarm" should be removed.

Author answer: The reviewer makes a valid distinction between evaluation metrics for controlled comparison and metrics for deployment estimation. I agree that the per-window FDR on a balanced test set does not directly predict the false alarm rate in operational PAM. The balanced evaluation in this study is designed as a controlled framework validation — comparing model performance under standardized conditions — rather than as a deployment simulation. The revised text clarifies this distinction explicitly. The Discussion (§4.2) now states that the balanced-set metrics characterize the detector's discrimination ability under controlled conditions and that deployment metrics such as false positives per hour of continuous recording would require evaluation on unsubsampled data, which is planned as part of future deployment validation. The new PR curve analysis (Figure 3) partially addresses this concern by showing how precision degrades as the operating point shifts toward higher recall, providing practitioners with the information needed to estimate false-alarm behavior at any operating point.

M4 - reviewer: Positioning of the YAML configuration relative to existing standards and frameworks. The author is right that a good metadata scheme for the training-to-prediction workflow is important; however, the importance and novelty of this contribution are overstated. Metadata standards for annotations already exist, for example Tethys and the recent ANSI/ASA S3/SC1.7 standard for acoustic metadata, as do training frameworks for bioacoustics such as Koogu, Ketos and AnimalSpot, all of which define their own metadata conventions, some more comprehensive than others. PAMGuard, which is mentioned in the manuscript but not cited, also has a standard for describing the pre-processing steps attached to a model, and Raven is widely used for manual annotation with its own text-based format. There is nothing inherently wrong with the existence of multiple standards, but that is what this YAML schema is, another one, rather than something new. I would suggest this is reframed accordingly, with the existing work cited and the specific advantages of the proposed schema stated explicitly. Allowing users to import or convert annotations from other formats would be a valuable addition and would considerably increase the practical reach of the toolkit.

Author answer: I thank the reviewer for these references, which have significantly improved the positioning of this work. The Introduction (§1.2) and Discussion (§4.1) have been revised to cite the Tethys database framework [18], the emerging ANSI/ASA S3/SC1.7 standard [19], PAM data management guidelines (Wall et al. [20]), and the training frameworks Koogu [21], Ketos [22], ANIMAL-SPOT [23], DeepAcoustics [24], and PAMGuard [25]. The YAML configuration is now presented not as a novel metadata concept but as a practical, self-contained implementation that bundles preprocessing, training, and evaluation parameters into a single human-readable file, complementing rather than replacing existing standards. The specific advantages stated explicitly are: (a) full workflow coverage from annotation to cross-domain evaluation in one file; (b) automatic archiving of the executed configuration alongside outputs via `run_config.yaml`; (c) direct executability without external schema validation; and (d) integration of experimental-integrity features such as leakage-safe grouped cross-validation (`group_by`) and hardware-identical spectrogram rendering (`c_spectrogram`) within the same configuration schema.

M5 - reviewer: Scope of the framework - what it does and does not accommodate. The

abstract states that the framework can be adapted to different species, vocalization types and acoustic environments through configuration alone. In practice the design is clearly built around single-channel CNN classification of spectrogram images, and it is not obvious how it would extend to several approaches that are in common use. Object-detection architectures such as YOLO produce multiple bounding boxes per segment rather than one label per window; segmentation approaches generate a mask that is then post-processed, for example by graph search, to extract contours (some of this has been referenced); multi-channel input is used routinely in towed-array and localisation work; and click-based detection, including AI-based click train classifiers, does not operate on spectrogram images at all. I am not suggesting the author implement these, but claims should be narrowed to what the framework actually supports, with the out-of-scope cases named. The paper also says very little about how trained models and their pre-processing parameters are exported for use in other software; given that the stated aim is to serve the wider PAM community, a short section on interoperability with existing software would strengthen it.

Author answer: I agree. The Abstract and §1.3 have been revised to state that the framework supports single-channel spectrogram-based CNN classification, binary and multiclass, with configurable frequency range, temporal windowing, and image preprocessing. Object-detection architectures (e.g., YOLO), pixel-level segmentation, multi-channel array processing, and click-based detection on time-domain signals are now explicitly named as outside the current scope in §4.3. Regarding model export and interoperability: a note has been added to §4.1 stating that the framework produces standard Keras model files convertible to TensorFlow Lite, ONNX, or other inference formats for deployment on external platforms including PAMGuard's deep learning module. The v1.1 release also includes a C-spectrogram engine for training-serving consistency on edge devices (Extended Data, S5.3).

M6 - reviewer: Figure 3 does not match its caption, and contains an unclaimed result. Figure 3 is captioned as a normalized confusion matrix for Experiment B, but shows box plots of F1 and recall against FFT length for both datasets. The confusion matrix referred to in Sections 3.4.1 and 4 is therefore missing and needs to be added, and the FFT-length figure needs its own caption. This should have been caught by a proofread. It's also of note that the FFT result is perhaps one of the most interesting finding in the paper but is currently just ignored?? An FFT length of 256 appears to give markedly better cross-domain F1 and recall than 512 or 1024, while the in-domain results are essentially flat, which is exactly the kind of quantitative evidence on how pre-processing choices affect detection performance that the Introduction says the field lacks. I would encourage the author to promote this to a proper analysis with methods, results and discussion.

Author answer: The reviewer is correct that (original) Figure 3 was erroneously retained from a preliminary version. It has been replaced with the precision-recall analysis for Experiment A. I appreciate the reviewer's recognition of the FFT-length result, and I agree that it deserves the rigorous treatment the reviewer describes. That investigation has been developed as a dedicated empirical study with a systematic Nfft sweep across five values (128–2048 samples), overlap sensitivity check, and paired statistical testing (Wilcoxon signed-rank with Bonferroni correction). A preprint is now available: De Marco R. et al. "Is FFT window length a neutral preprocessing choice in CNN-based dolphin whistle detection?

A cross-domain sensitivity analysis." bioRxiv, 2026. https://doi.org/10.64898/2026.05.01.721665. I chose to separate it from the present Method Article to maintain a clear distinction between the framework description and the empirical findings it enables.

M7 - reviewer: Missing literature and unsupported generalisations. The Introduction needs more references and contains several sweeping statements that should be removed or evidenced; specific instances are listed in the minor comments below. Two examples stand out. First, the claim that missing whistles are the more consequential error (Section 2.7.2) is contradicted in Section 4.2, where zero false positives is argued to be the operationally critical result. Much of PAM is context dependent; the performance required for real-time mitigation, density estimation and behavioural study are all different and the manuscript should say which context it is optimising for and be consistent about it. Reviewer 1 makes the same point. Second, several statements about what the field currently does are asserted without support; the relevant frameworks and detection approaches listed above should be cited rather than passed over.

Author answer: I have thoroughly revised the Introduction to remove unsupported generalizations and add the missing references. Specific changes include:
- §1.1: "tracking" changed in "monitoring"; spectrogram as universal foundation corrected to acknowledge non-spectrogram approaches; "deep learning has largely taken over" to balanced statement; PAM applications broadened
- §1.2: added Li et al. (2020) [12], Li et al. (2023) [13], and moved White et al. (2022) [14] into the review; cited Tethys [18], ANSI/ASA S3/SC1.7 [19], Wall et al. [20], Gillespie et al. [25], and Sugarman et al. [24] for DeepAcoustics
- §2.1: "rarely described, let alone standardized" softened and cited with existing frameworks [21–25]
- §2.7.2: removed the claim that missed whistles are "the more consequential error"; replaced with context-dependent framing consistent with §4.2
- Table 1 has been expanded with the three new studies and all species names have been standardized with full scientific binomials in italics.

M8 - reviewer: Optional: comparison against a classical whistle detection algorithm. This is not required for publication, but a comparison against a well-established non-AI whistle detector, for example the PAMGuard Whistle and Moan detector, on the same in-domain and DCLDE material would be a valuable addition. My expectation is that it would be less accurate in domain but more stable across the domain shift, which would be an informative result either way and would inform the generalisation question the paper raises.

Author answer: I agree that a direct comparison against the PAMGuard Whistle and Moan Detector would be a valuable contribution, and I thank the reviewer for framing this as optional. The practical barriers to a rigorous automated comparison are non-trivial: PAMGuard's batch processing mode requires a GUI-generated configuration file, and its output format requires dedicated parsing to produce per-window detection metrics comparable with those reported here. Configuring the WMD parameters to a fair operating point for the DCLDE dataset (without inadvertently tuning to the test data) would itself

require a systematic evaluation protocol. I would respectfully suggest that implementing a standardized command-line interface or API for PAMGuard's core detectors would substantially lower the barrier to such automated benchmarking, and could enable reproducible cross-validation studies that are currently difficult to conduct at scale. In the meantime, I note that Nur Korkmaz et al. [16] have already compared a CNN approach against a PAMGuard baseline on whistle detection, providing partial context. I have added a note in §4.2 identifying a CNN-versus-classical-detector comparison as a planned next step, and I hope the framework presented here — with its standardized cross-domain evaluation protocol — will serve as useful infrastructure for such comparisons when they are conducted.

Minor comments
reviewer: Suggested references are neither mandatory nor exhaustive, but the author does need to add more references and examples in the Introduction. Quoted manuscript text is given in italics.

Author answer: I thank the reviewer for these valuable suggestions. I have expanded the Introduction by adding the suggested references and integrating further examples to provide a more comprehensive context for the manuscript.

Reviewer: m1. Section 1.1. The following statement needs rewording:
“Passive Acoustic Monitoring (PAM) has become a widely-used, non-invasive tool for tracking acoustically active species over time”
PAM is used for spatial and temporal analysis of species. “Tracking” suggests animal movement, which PAM can be used for but rarely is.
Author answer: I agree with the reviewer that "tracking" can imply following individual animal movements. I have replaced "tracking" with "monitoring" to describe the temporal and spatial assessment of species presence accurately.

Reviewer:m2. Please rephrase:
“Marine mammals produce a variegated repertoire of vocalizations”
Author answer: I have rephrased "a variegated repertoire of vocalizations" to "a wide range of vocalizations" for better stylistic clarity.

Reviewer:m3. Change “frequently” to “sometimes”: “wide frequency bands, frequently in habitats heavily disrupted by intense human noise”
Author answer: I have revised the sentence, replacing "frequently" with "often" (and softening the assertion) to better reflect the variability of acoustic masking across different habitats.

Reviewer:m4. Please rephrase:
“Visual surveys are limited by weather conditions and human bias” PAM is also subject to human bias in annotation, and drone surveys combined with AI can take humans largely out of the loop.
Author answer: I agree with this perspective. I have updated Section 1.1 to acknowledge that PAM and human annotation protocols are also subject to observer bias, providing a more balanced contrast with visual survey methodologies.

Reviewer:m5. Too narrow a focus
“The push is to expand PAM networks to map cetacean distributions and behaviors, which is particularly vital for minimizing harmful interactions with commercial fisheries.”
Why single out bycatch here? These data are useful for a whole host of purposes: monitoring population levels, studying the impacts of windfarm construction and fossil fuel development, supporting the implementation of MPAs, and so on. PAM for bycatch study is relatively undeveloped, albeit with a lot of potential.
Author answer: I agree that focusing solely on bycatch mitigation was too narrow. I have broadened the statement by introducing "for example" and incorporating other major applications of PAM data, including population monitoring, environmental impact assessments for offshore developments, and supporting marine protected areas (MPAs).

Reviewer:m6. Not true and should be corrected:
“whether analyzed by humans or algorithms, the processing pipeline relies on a shared visual foundation: the spectrogram.”
A great deal of PAM does not involve a spectrogram. The PAMGuard click detector module is widely used in marine bioacoustics, and C-PODs and F-PODs are ubiquitous and store no raw audio or spectral data at all.
Author answer: I appreciate this important correction. I have reworded the passage to clarify that spectrogram-based processing applies specifically to time-frequency analysis workflows, explicitly noting non-spectrogram approaches.

Reviewer:m7. This needs evidence or should be softened:
“Today, however, deep learning approaches have largely taken over, with convolutional neural networks (CNNs) analyzing spectrogram images directly to deliver far superior detection accuracy.”
Have they? Where is the evidence or the references? Deep learning shows a lot of promise, but many researchers still use older non-AI approaches, partly because of software integration and partly because the domain shift issue with AI is such a problem. Heuristic approaches tend to be more stable or at least can be modified more easily.
Author answer: I have softened this statement to offer a more balanced overview. The revised text notes that while deep learning approaches have shown great promise and rapid adoption, classical and heuristic algorithms remain widely used due to software integration stability and domain shift challenges.

Reviewer:m8. Section 1.2
There are a number of relevant whistle detection and classification papers not mentioned here, including Li et al. (2020), Li et al. (2023) and White et al. (2022); see the reference list below. If the author takes up the comparison with classical algorithms suggested in major comment 8, Gillespie et al. (2013) and Gruden and White (2020) should also be cited.
Author answer: I thank the reviewer for pointing out these relevant works. I have incorporated citations to Li et al. (2020), Li et al. (2023), White et al. (2022), Gillespie et al. (2013), and Gruden & White (2020) into Section 1.2 and updated Table 1 accordingly.

Reviewer:m9. On inter-annotator variability:
“Inter-annotator variability in marine PAM contexts has been shown to depend on both

vocalization type and signal-to-noise ratio, introducing a source of performance variance that is rarely quantified in comparative studies. The proposed pipeline addresses this gap by formalizing an annotation protocol with explicit decision criteria for each label class."
This is a long-recognised problem for which metadata standards have already been developed. Please rephrase and add references; the ANSI/ASA standard, Roch et al. (2016) and Wall et al. (2021) are the obvious starting points.
Author answer: I have revised this section to contextualize inter-annotator variability within existing standardization efforts, explicitly citing established metadata frameworks and relevant literature, including the ANSI/ASA standard, Roch et al. (2016), and Wall et al. (2021).

Reviewer:m10. Section 2.1
This statement requires evidence: "In practice, the full pipeline is rarely described in its entirety, let alone standardized"
Sweeping generalisations require evidence, and again, other frameworks already exist that do much of this.
Author answer: I have softened the generalized statement regarding pipeline descriptions and added references to existing PAM software frameworks to better acknowledge prior standardization efforts in the field.

Reviewer:m11. Section 2.2
PAMGuard is referred to here but not cited. Please add Gillespie et al. (2026).
Author answer: I have added the formal citation for PAMGuard (Gillespie et al.) in Section 2.2.

Reviewer:m12. I agree with Reviewer 1 that the criteria for event boundaries, minimum duration and the maximum permitted gap for grouping consecutive signals into a single annotation need to be stated. This is particularly important for click trains and consecutive whistles, and it bears directly on the class confusion discussed in Section 3.4.1.
Author answer: I have updated Section 2.2 to explicitly detail the annotation boundary criteria, including minimum event duration, maximum allowable gap for grouping consecutive emissions, and decision rules for distinguishing individual call events.

Reviewer:m13. Section 2.5
On image-domain pre processing steps:
"it remains relatively unexplored within the context of cetacean vocalization detection."
I am not sure this is true; other frameworks use these techniques. See, for example, the DeepAcoustics work of Sugarman et al. (2025).
Author answer: I have rephrased this statement to soften the claim, acknowledging previous applications of image-domain pre-processing in marine bioacoustics and citing relevant literature such as Sugarman et al.

Reviewer:m14. The following overstates the position:
"CNNs operating on spectrogram images represent the current standard for automated vocalization detection in PAM."
The current standard is a mix of approaches, and spectrogram CNNs are only one part of it.
Author answer: I have modified the phrasing from "represent the current standard" to

"represent a widely used approach" to avoid overstating the predominance of CNN-based spectrogram models.

Reviewer:m15. Section 2.7
On the fixed threshold — see major comment 2: “decision threshold is fixed at θ = 0.5, the balanced operating point of the sigmoid output.”
Precision–recall plots against threshold would be informative here.
Author answer: I agree that evaluating performance across varying decision thresholds adds useful insight. Precision–Recall curves across operating thresholds have been generated and included as Figure 3 in the revised manuscript.

Reviewer:m16 Setion 3.1.1
It would be worth stating explicitly in the Limitations that the training material comes from seven animals in a marine park basin, which is a reverberant, close-range, low-diversity acoustic environment. This is a fairly extreme starting point for a domain shift comparison, and saying so would help the reader interpret the size of the gap reported in Section 4.2; it is closer to a worst case than a typical one...
Author answer: I have explicitly detailed this limitation in Section 4.3, noting that the training dataset originates from seven animals in a reverberant marine park basin, thereby framing this setup as a challenging, worst-case scenario for testing cross-domain transferability.

Reviewer:m17. Setion 3.4.1
The figure is incorrect but the idea that lower FFT’s somehow make the domain shift less of an issue is intriguing and fits exactly with the theme of this paper - how pre-processing steps are important...it needs to be discussed in text. See longer comment above.
Author answer: I have expanded the text in Section 3.4.1 to discuss the role of FFT window size Nfft on domain generalization, highlighting this finding as a promising topic for dedicated future investigation

Reviewer:m18. Section 4.2
quantifies how large that overstatement can be. Yes, but also the author should give themselves some credit here - the domain shift from a marine park to the Pacific ocean is pretty huge!
Author answer: I thank the reviewer for this insightful comment. I have updated Section 4.2 to explicitly highlight that shifting directly from a controlled, reverberant facility to an open oceanic environment represents an exceptionally stringent evaluation of domain generalization

Competing Interests: No competing interests were disclosed.

Reviewer Report 21 August 2026

https://doi.org/10.21956/openreseurope.26352.r77868

© 2026 Ogawa M. This is an open access peer review report distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

?

Mayu Ogawa
[1] Japan Agency for Marine-Earth Science and Technology, Yokosuka, Japan
[2] Japan Agency for Marine-Earth Science and Technology, Yokosuka, Japan

General Comments
This study presents ai-pam-pipeline, an open-source framework that organizes CNN-based cetacean acoustic detection into six integrated stages, from annotation to model evaluation. Major parameters can be managed through a single YAML configuration file, and the *Software Implementation and CLI Usage* section provides clear instructions for reproducing the analyses. This design is useful for improving reproducibility and methodological comparability in the development of CNN-based detectors for passive acoustic monitoring.
However, given the decrease in performance between the in-domain and cross-domain evaluations, the claims regarding generalization should be interpreted more cautiously. In addition, the manuscript should more clearly distinguish between the range of species, vocalization types, and acoustic environments that have been empirically evaluated in this study and those that the framework is designed or expected to accommodate. This clarification would better define the practical significance and scope of the proposed framework.

Major comments

1. The framework allows users to configure many parameters, including window length, overlap, FFT size, frequency range, augmentation, and classification threshold. However, the manuscript provides limited guidance on how these values should be selected and compared when applying the framework to new datasets or species. More explicit practical guidance on parameter-selection procedures and decision criteria would strengthen both the methodological standardization and usability of the framework.

2. The statement “it never produced a false alarm” is too strong. The absence of false positives depends on the size and composition of the evaluated test set, and the external evaluation in this study was based on a limited dataset. I therefore recommend avoiding the word “never” and instead stating that “no false positives were observed in the evaluated external test set.”

3. The Abstract states that the workflow from raw audio to model evaluation is rarely documented in sufficient detail for reproducibility or comparison. However, spectrogram parameterization, image processing, and model evaluation are described to varying degrees in several of the peer-reviewed studies cited in the manuscript. I therefore suggest moderating this statement and clarifying that the main limitation may not be the absence of methodological descriptions themselves, but whether the complete workflow can be independently reproduced through sufficiently detailed reporting and the availability of associated data, annotations, configuration files, and code.

4. In Section 2.2, W, ECT, and BPS are mainly defined using observable acoustic characteristics, whereas FB is described as a “dense vertical texture associated with foraging activity.” If

annotation is based on acoustic recordings alone, it may be difficult to determine whether a signal is actually associated with foraging behavior. Because this section aims to provide reproducible annotation criteria, I recommend defining FB primarily using directly observable acoustic features, such as spectrogram structure, pulse rate, inter-pulse interval, and event duration, rather than its presumed behavioral function.

5. Although standardizing the annotation protocol is an important objective of the framework, the criteria used to define the temporal boundaries of individual annotation events are not sufficiently clear. This may be particularly important for repeated signals such as echolocation click trains or consecutive whistles, where annotators may differ in how they define the start and end of an event or how much temporal gap is allowed within a single annotation. To improve reproducibility, I recommend providing clearer criteria for event boundaries, minimum duration, and the allowable temporal gap for grouping consecutive signals into the same annotation event.

6. The statement “confirming generalization to unseen acoustic conditions” appears too strong. Macro F1 decreased from 0.966 to 0.759 and Recall from 0.956 to 0.545, while the external evaluation was based on a single recording track. Please moderate the wording regarding cross-domain generalization to better reflect the performance observed in the external dataset.

7. The Abstract states that the framework can be adapted to different species, vocalization types, and acoustic environments. However, as also acknowledged in the Limitations, the present study evaluated only *Tursiops truncatus*, and applicability to other species has not been empirically demonstrated. I recommend moderating these statements, particularly regarding applicability to other species, and clearly distinguishing between demonstrated applicability and broader applications that require future validation.

8. The Methods state that missed whistles are the more consequential error and that Recall and F1 are the primary metrics. However, the Discussion states that zero false positives are preferable even with lower Recall. Please clarify the intended operational objective and ensure that the prioritization of errors is applied consistently throughout the manuscript.

9. The manuscript highlights that users can modify parameters such as training-set composition, spectrogram frequency range, and class definitions and then rerun the pipeline to address the generalization gap. However, the present study stops at demonstrating the gap and does not show how such modifications affect cross-domain performance. If feasible, a simple before-and-after example based on one parameter change would help illustrate the practical value of the framework for iterative detector improvement.

10. Figure 3 appears inconsistent with both its caption and the corresponding Results text. The caption describes a normalized confusion matrix for Experiment B, whereas the displayed figure appears to show F1 and Recall across different FFT sizes. Please verify and correct the figure, caption, and corresponding text. If the FFT-size analysis is intended, please describe the analysis and results explicitly in the Methods and Results.

Minor comments

1. In Section 1.1, the statement that maintaining high accuracy in population monitoring is "vital for effective marine conservation" may imply a more direct relationship between detector performance and conservation outcomes than intended. A more cautious expression, such as "supports reliable monitoring for marine conservation," may be appropriate.

2. Section 1.1 emphasizes the use of PAM networks for reducing harmful interactions with commercial fisheries. However, PAM has a broad range of applications, including distribution and occurrence monitoring, density estimation, behavioral studies, anthropogenic noise assessment, and environmental impact assessment associated with marine development. I therefore suggest using a less exclusive expression such as "for example," or briefly mentioning additional applications.

3. The phrase "track marine life continuously" may be interpreted as continuous tracking of individual animals. If the intended meaning is continuous acoustic monitoring over extended periods, a more precise expression such as "continuously monitor acoustic occurrence over extended periods" may be preferable.

4. The Species/call type column in Table 1 appears to mix scientific names, common names, and call-type descriptions. Please standardize the terminology and consistently italicize scientific names, including *Tursiops aduncus* and *Tursiops truncatus*.

5. In several parts of the Methods, it is not always clear whether a processing choice is a fixed feature of the framework or a parameter selected specifically for the *Tursiops truncatus* case study. For example, the statement that "a Hanning window is applied" could be interpreted as a fixed framework setting. Please distinguish more clearly between general framework functionality and case-specific parameter choices, such as image size, frequency range, and FFT size.

6. Please clarify how the negative windows were generated. Were all valid non-target segments outside the W/MW annotations used as negative samples, or were negative windows selected only from the annotated NOISE, FB, BPS, and ECT segments?

7. Table 5 includes a separate Label column, whereas Table 4 presents only the class identifiers C0 and C1. For consistency and readability, I suggest adding a Label column to Table 4 and indicating the corresponding class labels.

8. The Conclusions report the cross-domain performance of Experiment A, including FDR = 0.000 and macro F1 = 0.759. However, an important finding of this study is the decrease in performance from the in-domain to the cross-domain evaluation. Reporting both results would more clearly communicate the observed generalization gap.

Is the rationale for developing the new method (or application) clearly explained?

Yes

Is the description of the method technically sound?

Yes

Are sufficient details provided to allow replication of the method development and its use by others?
Yes

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?
Yes

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?
Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Marine bioacoustics, with a particular focus on PAM and acoustic data analysis of marine mammals, including small cetaceans. My research includes the analysis of PAM datasets and the application of machine-learning methods to acoustic detection and analysis.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 24 Sep 2026

Rocco De Marco

I wish to express my sincere gratitude to both reviewers for the exceptional quality and depth of their evaluation.
Dr. Ogawa's careful reading identified important areas where the manuscript's claims exceeded the supporting evidence, and provided precise guidance on how to align the narrative with the experimental results. Both reviews reflect a level of professional commitment and domain expertise, and the revised manuscript is substantially stronger as a direct result of their work. Their feedback has been formative, not merely corrective. I thank the reviewer for recognizing the framework's contribution to reproducibility and methodological comparability, and for the positive assessment of the "Software Implementation and CLI Usage" section. Both observations regarding cross-domain generalization claims and the distinction between demonstrated and expected applicability are well taken, and have been addressed systematically throughout the revision: all generalization claims have been moderated to reflect the evidence, and the scope of empirical validation (single species, single training site) is now clearly distinguished from the framework's design flexibility. The detailed responses below describe each change. An updated version of the article has just been submitted; I do not know the timeline for its release.

Major comments M1 - reviewer: The framework allows users to configure many parameters, including window length, overlap, FFT size, frequency range, augmentation, and classification threshold. However, the manuscript provides limited guidance on how

these values should be selected and compared when applying the framework to new datasets or species. More explicit practical guidance on parameter-selection procedures and decision criteria would strengthen both the methodological standardization and usability of the framework.

Author answer: I thank the reviewer for this observation. Detailed practical guidance for parameter selection — including recommended ranges, decision criteria for new datasets and species, and worked examples — is provided in the Reference Manual deposited as Extended Data (De Marco, 2026, DOI: 10.5281/zenodo.20697433, updated to v1.1). The manual covers parameter selection for all six pipeline stages, including FFT size (S5.1), windowing (S4.3–S4.5), frequency crop (S5.1), normalization mode (S3.4), cross-validation design (S7.3), and the new grouped cross-validation feature (S7.3.1). To improve discoverability, I have added explicit cross-references to the relevant manual sections in the Methods text (§2.3, §2.4, §2.6) and a brief paragraph in §4.1 summarizing the key parameter-selection principles.

M2 - reviewer: The statement “it never produced a false alarm” is too strong. The absence of false positives depends on the size and composition of the evaluated test set, and the external evaluation in this study was based on a limited dataset. I therefore recommend avoiding the word “never” and instead stating that “no false positives were observed in the evaluated external test set.”

Author answer:
I agree. Following the correction of the cross-validation fold assignment (see response to Reviewer 2, Major 1), the cross-domain false discovery rate with the corrected models is 0.001 ± 0.002, with nine of ten folds producing no false positives and one fold producing two. All instances of "never" and "zero false positives" have been replaced with precise quantitative statements throughout the manuscript and Plain Language Summary.

M3 - reviewer: The Abstract states that the workflow from raw audio to model evaluation is rarely documented in sufficient detail for reproducibility or comparison. However, spectrogram parameterization, image processing, and model evaluation are described to varying degrees in several of the peer-reviewed studies cited in the manuscript. I therefore suggest moderating this statement and clarifying that the main limitation may not be the absence of methodological descriptions themselves, but whether the complete workflow can be independently reproduced through sufficiently detailed reporting and the availability of associated data, annotations, configuration files, and code.

Author answer:
I agree and have revised the Abstract to clarify that the limitation is not the absence of methodological descriptions per se, but whether the complete workflow is accompanied by the configuration files, code, and data needed for independent reproduction.

M4 - reviewer: In Section 2.2, W, ECT, and BPS are mainly defined using observable acoustic characteristics, whereas FB is described as a “dense vertical texture associated with foraging activity.” If annotation is based on acoustic recordings alone, it may be difficult to determine whether a signal is actually associated with foraging behavior. Because this section aims to

provide reproducible annotation criteria, I recommend defining FB primarily using directly observable acoustic features, such as spectrogram structure, pulse rate, inter-pulse interval, and event duration, rather than its presumed behavioral function.

Author answer: I agree. The feeding buzz definition in §2.2 has been revised to emphasize spectrographic characteristics: a rapid sequence of broadband pulses with progressively decreasing inter-pulse intervals, appearing as a dense band of vertical striations on the spectrogram. The behavioral association with foraging is retained as contextual information but is no longer the primary defining criterion.

M5 - reviewer: Although standardizing the annotation protocol is an important objective of the framework, the criteria used to define the temporal boundaries of individual annotation events are not sufficiently clear. This may be particularly important for repeated signals such as echolocation click trains or consecutive whistles, where annotators may differ in how they define the start and end of an event or how much temporal gap is allowed within a single annotation. To improve reproducibility, I recommend providing clearer criteria for event boundaries, minimum duration, and the allowable temporal gap for grouping consecutive signals into the same annotation event.

Author answer: I have added a paragraph in §2.2 specifying boundary criteria for each vocalization type: gap-based segmentation for whistles, maximum inter-click gap for echolocation click trains, and onset/offset of visible pulse trains for burst-pulse sounds. These criteria are documented both in the revised manuscript and in the Reference Manual (Extended Data,S4.2).

M6 - reviewer: The statement “confirming generalization to unseen acoustic conditions” appears too strong. Macro F1 decreased from 0.966 to 0.759 and Recall from 0.956 to 0.545, while the external evaluation was based on a single recording track. Please moderate the wording regarding cross-domain generalization to better reflect the performance observed in the external dataset.

Author answer: I agree and have revised all statements regarding cross-domain performance. The revised manuscript states that the detector "maintains high precision on unseen data while recall decreases substantially, indicating partial but operationally useful generalization." The new precision-recall analysis (Figure 3) further contextualizes this: at a lower decision threshold ($\theta = 0.05$), recall rises to 0.755 while precision remains above 0.95, indicating that the model captures a larger portion of the cross-domain whistles than the default operating point suggests.

M7 - reviewer: The Abstract states that the framework can be adapted to different species, vocalization types, and acoustic environments. However, as also acknowledged in the Limitations, the present study evaluated only Tursiops truncatus, and applicability to other species has not been empirically demonstrated. I recommend moderating these statements, particularly regarding applicability to other species, and clearly distinguishing between demonstrated applicability and broader applications that require future validation.

Author answer: I agree and have revised the Abstract and §1.3 to clearly distinguish

between the framework's design flexibility (configurable for any species through YAML parameters) and its empirical validation (limited to *T. truncatus* in the present study). The Limitations section (§4.3) now explicitly states that applicability to other species requires new annotated material and parameter adjustment, and has not been demonstrated here.

M8 - reviewer: The Methods state that missed whistles are the more consequential error and that Recall and F1 are the primary metrics. However, the Discussion states that zero false positives are preferable even with lower Recall. Please clarify the intended operational objective and ensure that the prioritization of errors is applied consistently throughout the manuscript.

Author answer: Both reviewers identify this inconsistency, which I have now resolved.
The revised §2.7.2 states that which metric should be prioritized depends on the deployment context: for surveys and density estimation, missed detections are usually the more costly error, whereas for the deterrence application targeted here, false alarms carry a direct operational cost.
The new precision-recall analysis (Figure 3) makes this trade-off explicit and quantifiable: at $\theta = 0.50$ the detector prioritizes precision (0.999); at $\theta = 0.05$ it recovers 75.5% of whistles with precision above 0.95. The choice of operating point is now presented as application-dependent throughout the manuscript.

M9 - reviewer: The manuscript highlights that users can modify parameters such as training-set composition, spectrogram frequency range, and class definitions and then rerun the pipeline to address the generalization gap. However, the present study stops at demonstrating the gap and does not show how such modifications affect cross-domain performance. If feasible, a simple before-and-after example based on one parameter change would help illustrate the practical value of the framework for iterative detector improvement.

Author answer: I agree that such an example would strengthen the demonstration. A systematic study of this kind, investigating the effect of FFT window length on cross-domain generalization across five values and multiple test conditions, has been developed as a dedicated empirical study and is available as a preprint: De Marco R. et al. "Is FFT window length a neutral preprocessing choice in CNN-based dolphin whistle detection? A cross-domain sensitivity analysis." bioRxiv, 2026. https://doi.org/10.64898/2026.05.01.721665. Including such an analysis in the present Method Article would shift its focus from framework description to empirical parameter optimization, which I believe is better served by that separate publication. A forward reference has been added in §4.2.

M10 - reviewer: Figure 3 appears inconsistent with both its caption and the corresponding Results text. The caption describes a normalized confusion matrix for Experiment B, whereas the displayed figure appears to show F1 and Recall across different FFT sizes. Please verify and correct the figure, caption, and corresponding text. If the FFT-size analysis is intended, please describe the analysis and results explicitly in the Methods and Results.

Author answer: The reviewer is correct. Figure 3 in the submitted manuscript was

erroneously retained from a preliminary version of the work and does not belong to the present paper. It has been replaced with the precision-recall analysis for Experiment A (cross-domain, DCLDE 2022). The confusion matrix for Experiment B is now Figure 4. I apologize for the oversight.

Minor comments

m1 - reviewer: In Section 1.1, the statement that maintaining high accuracy in population monitoring is “vital for effective marine conservation” may imply a more direct relationship between detector performance and conservation outcomes than intended. A more cautious expression, such as “supports reliable monitoring for marine conservation,” may be appropriate.
Author answer: I Agree, the phrase has been reformulated with: "supports reliable long-term assessment"

m2 - reviewer: Section 1.1 emphasizes the use of PAM networks for reducing harmful interactions with commercial fisheries. However, PAM has a broad range of applications, including distribution and occurrence monitoring, density estimation, behavioral studies, anthropogenic noise assessment, and environmental impact assessment associated with marine development. I therefore suggest using a less exclusive expression such as “for example,” or briefly mentioning additional applications.

Author answer: I agree with the reviewer that PAM has a much broader scope of application. I have revised Section 1.1 by introducing the qualifier “for example” and broadening the context to explicitly mention additional applications, including density estimation, occurrence monitoring, and assessing responses to anthropogenic noise.

m3 - reviewer: The phrase “track marine life continuously” may be interpreted as continuous tracking of individual animals. If the intended meaning is continuous acoustic monitoring over extended periods, a more precise expression such as “continuously monitor acoustic occurrence over extended periods” may be preferable.

Author answer: I thank the reviewer for pointing out this potential ambiguity. To avoid implying the movement tracking of individual animals, I have replaced “track marine life continuously” with “enable continuous, unattended recording over extended periods”.

m4 - reviewer: The Species/call type column in Table 1 appears to mix scientific names, common names, and call-type descriptions. Please standardize the terminology and consistently italicize scientific names, including Tursiops aduncus and Tursiops truncatus.

Author answer: I agree with this recommendation and have standardized the terminology in Table 1. All species names have been updated to full binomial nomenclature in italics (e.g., Tursiops aduncus and Tursiops truncatus), and the call-type descriptions have been standardized across the entire column.

m5 - reviewer: In several parts of the Methods, it is not always clear whether a processing choice is a fixed feature of the framework or a parameter selected specifically for the

Tursiops truncatus case study. For example, the statement that “a Hanning window is applied” could be interpreted as a fixed framework setting. Please distinguish more clearly between general framework functionality and case-specific parameter choices, such as image size, frequency range, and FFT size.

Author answer: I appreciate this constructive feedback. I have revised the Methods section to clearly differentiate between the core, configurable functionality of the general framework and the case-specific parameter selections chosen for the Tursiops truncatus case study (such as FFT size, window function, frequency range, and image dimensions).

m6 - reviewer: Please clarify how the negative windows were generated. Were all valid non-target segments outside the W/MW annotations used as negative samples, or were negative windows selected only from the annotated NOISE, FB, BPS, and ECT segments?

Author answer: I have updated §3.2 to explicitly describe the sampling and generation procedure for negative windows. The text now clearly details the exact composition of the negative dataset (C0) used in the training and evaluation pipelines.

m7 - reviewer: Table 5 includes a separate Label column, whereas Table 4 presents only the class identifiers C0 and C1. For consistency and readability, I suggest adding a Label column to Table 4 and indicating the corresponding class labels.

Author answer: I agree that consistency across tables enhances readability. Table 4 has been updated to include a "Label" column clearly mapping the class identifiers (C0 = non-whistle, C1 = whistle), aligning its format with Table 5.

m8 - reviewer: The Conclusions report the cross-domain performance of Experiment A, including FDR = 0.000 and macro F1 = 0.759. However, an important finding of this study is the decrease in performance from the in-domain to the cross-domain evaluation. Reporting both results would more clearly communicate the observed generalization gap.

Author answer: I agree with the reviewer that presenting both in-domain and cross-domain metrics provides a clearer picture of the observed generalization gap. The Conclusions section has been updated to explicitly report both the in-domain F1-score (0.960) and the cross-domain F1-score (0.744).

Competing Interests: No competing interests were disclosed.