Automated Detection and Classification of Cetacean Acoustic Signals

Neural network assisted pipelines for real time alert systems, long term surveys, and communication modelling

Paul Best

Université de Toulon

A project submitted for the degree of

Doctor of Philosophy

2022

Contents

List of Acronyms

AE Auto-Encoder

ANN Artificial Neural Network

ARU autonomous recording unit

ASHA Async Successive Halving Algorithm

ASV Autonomous Surface Vehicule

AUC Area Under the ROC Curve

Bm Balaenoptera Musculus

BCE Binary Cross Entropy

Bp Balaenoptera Physalus

CE Cross-Entropy

CNN Convolutionnal Neural Network

DAC Digital Analog Converter

DBSCAN Density Based Spatial Clustering of Applications with Noise

DCLDE Detection Classification Localisation and Density Estimation of marine mammals

DEC Deep Embedded Clustering

DFT Discrete Fourier Transform

EM Expectation Maximisation

FFT Fast Fourier Transform

GMM Gaussian Mixture Model

GPU Graphical Processing Unit

HMM Hidden Markov Model

ICI Inter CLick Interval

IIC Invariant Information Clustering

IIR infinite impulse response

INI Inter Note Interval

IPI Inter Pulse Interval

KDE Kernel Density Estimate

KL Kullback-Leibler

LTSA Long Term Spectral Average

mAP mean Average Precision

MCU Microcontroller Unit

MFCC Mel Frequency Cepstral Coefficients

MI Mutual Information

MSE Mean Square Error

NAS Network-Attached Storage

NMI Normalized Mutual Information

NRKW Northern Resident killer whales

NRW Northern Right Whale

PAM Passive Acoustic Monitoring

PCEN Per-Channel Energy Normalisation

PLC Power Law Coefficient

PR Precision Recall

ReLU Rectified Linear Unit

RNN Recurent Neural Networks

ROC Reveiving Operating Characteristics

SGD Stochastic Gradient Descent

SOOS Southern Ocean Observing System

SOTA State Of The Art

SNR Signal to Noise Ratio

SSL Self Supervised Learning

STFT Short Term Fourier Transform

SVM Support Vector Machine

TDOA Time Difference Of Arrival

TK Teager-Kaiser

UDA Unsupervised Data Augmentation

UI User Interface

UMAP Uniform Manifold Approximation and Projection

VGG Visual Geometry Group

1
Introduction

Contents

1.1 Cetaceans and acoustics

Despite a first appearance on land, some mammalian species are now commonly found in planet Earth's oceans, forming the marine mammals group. Families evolving from 3 distinct orders have physiologically evolved to thrive in the marine environment: Pinnipeds (*Carnivora*, e.g. seals and walruses), Sirenians (*Afrotheria*, e.g. manatees), and Cetaceans (*Cetartiodactyla*, e.g. whales and dolphins). The following thesis will focus on the latter. Cetaceans are classified into two suborders: Odontocetes (toothed cetaceans, such as dolphins, orcas or sperm whales) and Mysticetes (baleen cetaceans, such as blue whales or humpback whales, see Fig. 1.1).

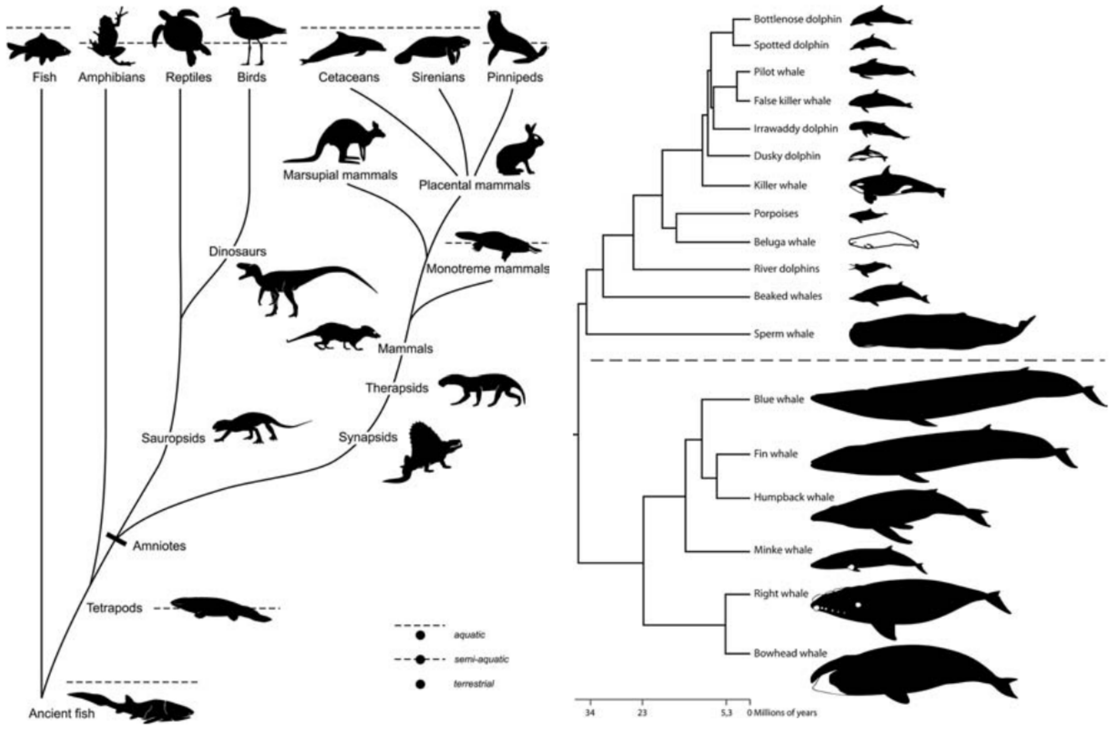


Figure 1.1: (left) Evolution of the marine mammals. (right) Cetacean evolutionary relationships (top: Odontocetes, bottom: Mysticetes). Both figures are taken from Whitehead and Rendell [205].

Returning to the sea some 50 million years ago [205], cetaceans now show a complete adaptation to their marine environment, with their powerful flukes, streamlined body, and nostrils displaced on top of their head (allowing for efficient breathing while swimming). Another important adaptation, especially relevant to this study, is the development of their acoustic capabilities, both as emitters and as receivers. Indeed, light typically fades out after a few dozen meters in water, which makes of vision a quite limited sense. In contrast, the higher density of water (compared to air) makes sound travel faster and further. Cetaceans make use of this property to communicate and/or echolocate up to great distances.

Blue whale calls can be heard 200km away [176], and sperm whales are able to detect a 1m object at 470m [55].

1.1.1 Echolocation

One of the uses cetaceans make of underwater acoustics is echolocation. Alike an active sonar, emitting a sound and measuring how it bounces back to you (its echo) allows to sense distance from surrounding objects, their shape [140], or even their texture [78] (Fig. 1.2). Bats use echolocation to navigate and hunt in dark caves, odontocetes use echolocation in a similar way underwater.

Short impulse like sounds commonly named ‘clicks’ (transitory waves) are mostly associated with echolocation purposes [7]. However, there is not one single type of click used for echolocation: it might coincide with habitats and feeding behaviours [100]. Using short duration clicks, more can be sent in a small period of time without them mixing up, thus increasing the potential temporal resolution of the echolocation. This is typically suited for hunting at high speeds, like small odontocetes do. On the other hand, clicks at lower frequencies will travel further, and thus would be more suited for hunting from long distances like sperm whales do (extremely high Kogia clicks go against this hypothesis).

Finally, despite the old consensus that only odontocetes echolocate with their high frequency clicks, new studies suggest that mysticetes might also make use of their low frequency signals as sonars [126].

1.1.2 Communication

The second major use of sound by cetaceans is communication, a broad concept that can be divided into two main categories: song and social communication systems [92].

Song

The term song has been first used for cetacean signals by Payne and McVay [148], listening to humpback whales whose vocalisations met the following definition: "a series of notes, generally of more than one type, uttered in succession and

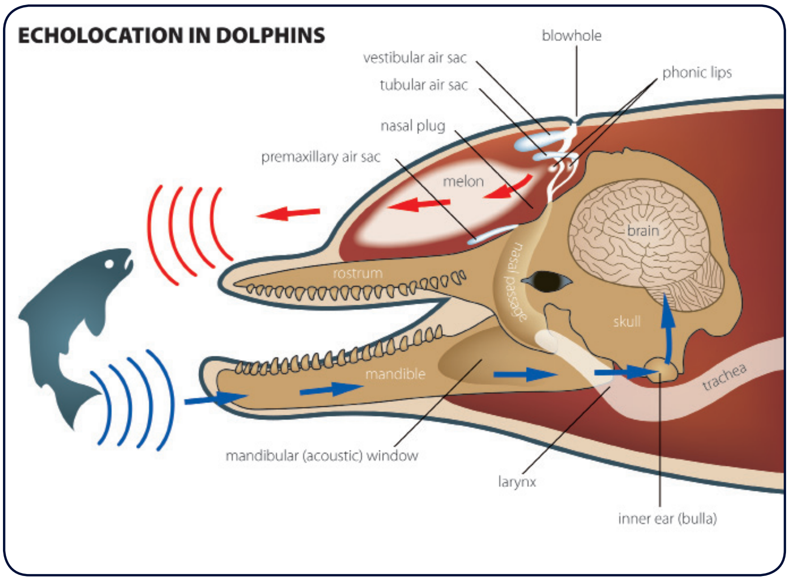


Figure 1.2: Illustration of the dolphin echolocation mechanism for hunting purposes (image credit: Uko Gorter - American Cetacean Society).

so related as to form a recognisable sequence or pattern in time". Similarly to bird songs, they have shown a role in reproductive behaviours: mostly males are observed singing, during the reproductive season, potentially to attract females, fend off other males, or a combination of both [39, 179]. Songs usually come in strictly patterned sequences, shared by whole species or communities [205]. Among cetaceans, they have yet been observed only in mysticetes, with the most renowned one probably being the humpback whale song.

Social communication

On the other hand, communication is also observed in odontocetes social groups. Alike in songs, these signals are patterned vocalisations, some of which being identified in discrete categories [65, 203]. However, they are not restricted to reproductive contexts, and appear in relatively less deterministic sequences. The term song therefore seems less appropriate for this phenomenon, which is rather associated with social bonding functions [170, 66].

In most cases, these vocal signals occur with tonal, whistled or pulsed calls. Their associated categories ('call type') are commonly defined by characteristics on their time / frequency contour. As an exception, sperm whales produce clicks in stereotyped rhythmic sequences (named codas) that were also attributed to communication purposes [203]. It is however not excluded that other odontocetes use clicks as means of communication, but no similar stereotyped sequences have yet been observed among them.

1.1.3 Culture

The term culture is often encountered when describing cetacean communication systems [59, 69, 158]. It seems appropriate to describe the vocal divergences observed between cetacean communities. In a broad sense, culture is defined as 'behaviour or information shared within a community, that is acquired from conspecifics through some form of social learning' [205]. In cetaceans, it takes form as specialisation in diets or hunting techniques (e.g. with orcas) or as specific vocal patterns. For instance, sperm whale codas [158] (Fig. 1.3), orca stereotyped calls [41], humpback whale songs [70] (Fig. 1.6) or fin whale pulse sequences [29], are all community specific, some evolving through the years, and thus are described as cultural phenomenons. This is only possible thanks to the vocal production learning capacity that cetaceans demonstrate [92], a relatively rare characteristic among mammals.

1.1.4 Human activity impacts

In the twentieth century, close to 3 million large whales were caught by whalers [163]. Seeing some whale species coming close to extinction has motivated a large majority of the international community to cease commercial whaling in the late 20th century. However, cetaceans are still heavily impacted by human marine activities in numerous ways (Fig. 1.4). We will focus here on the ones related to acoustics.

There exist a wide variety of anthropogenic acoustic disturbances in the marine environment, which has triggered the development of a new fields of research focusing solely on ambient noise levels [127]. Marine traffic, seismic surveys using airguns (often to search for oil patches), pile driving (for marine constructions such as offshore wind turbines), military sonars and explosive tests are the most widespread, with several consequences on cetaceans.

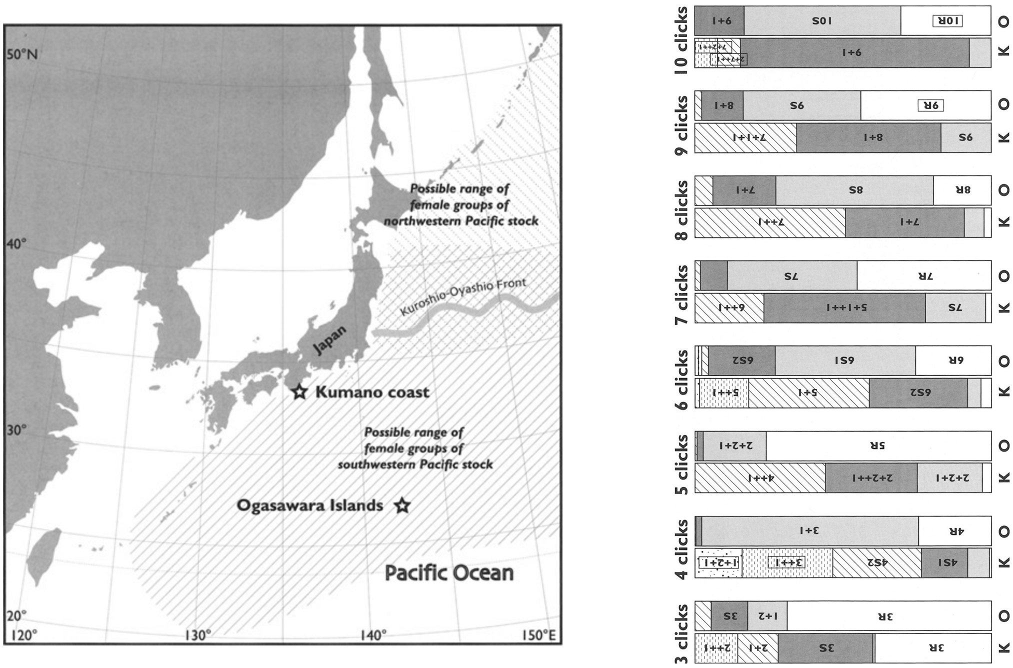


Figure 1.3: (left) Location of two sperm whale communities. (right) Differences in codas patterns (dialects) between the two communities (K: Kumano coast, O: Ogasawara Islands) . Figures are taken from Amano et al. [5]

We hear better in a silent environment. This implies the first consequence of acoustic disturbances: acoustic masking. With increasing ambient noise levels, the hearing capacities of cetaceans decrease, thus hindering their ability to communicate, hunt, and navigate [48]. More generally, dense marine traffic has also been shown to cause stress to some cetaceans species [165].

The second main consequence is acoustic impairment: temporary or permanent injuries of the hearing apparatus. Powerful sounds such as those emitted by airguns or military tests have been shown to cause deafness in some cetaceans, sometimes leading to mass strandings [47].

Eventually, arising from a dense marine traffic, presumably combined with disorientation due to acoustic masking, the collision problem has also attracted the attention of the cetacean conservation community. Especially affecting large mysticetes (e.g. fin whales or right whales), records of death from collisions with boats show a significant impact on whale populations [164], motivating measures to mitigate collision risks.

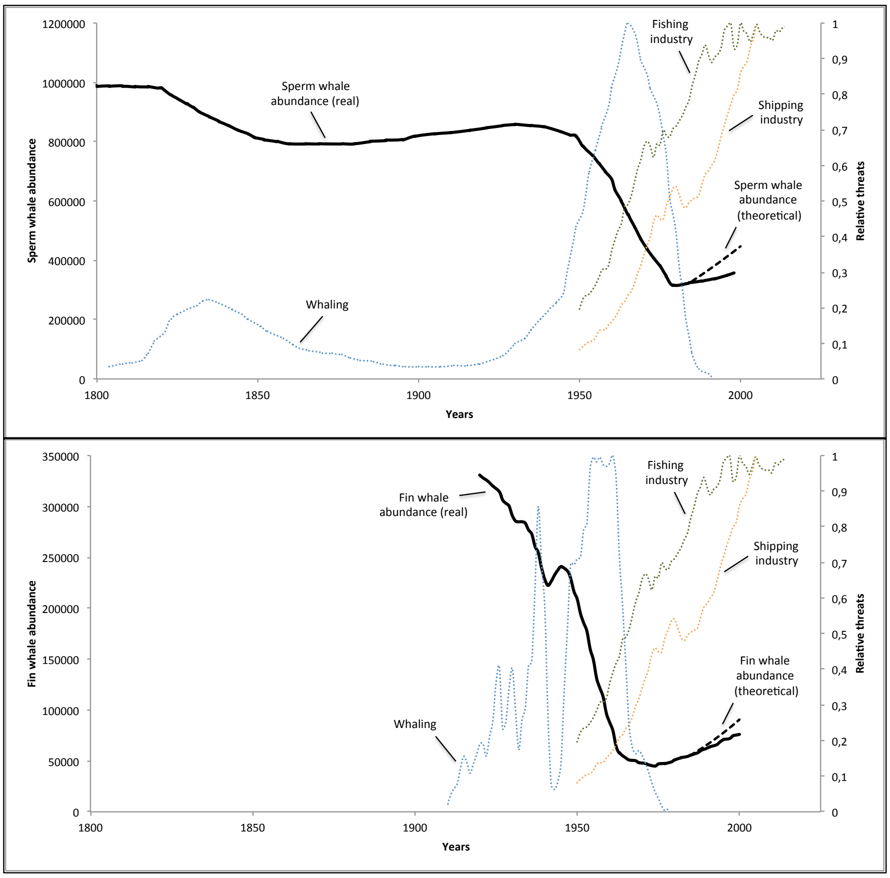


Figure 1.4: Evolution of the worldwide sperm (top) and fin (bottom) whale populations and the main human-induced direct mortality threats. The threats are expressed in relative value. This figure is taken from Sèbe [172].

1.2 Passive Acoustic Monitoring of cetaceans

To reveal the aforementioned complexity and diversity of cetacean's uses of acoustics, scientists also have put forward their hearing sense. Passive Acoustic Monitoring (PAM) is a field of bioacoustic studies that combines several scientific and technical domains, from electronics for recording hardware, to signal processing and statistical analysis. The term passive refers to the notion of listening from a distance, without interfering with the animals, as opposed to active sonar systems or attaching acoustic tags to the animals. The analysis of the cetaceans acoustic activity is providing important insights on their behaviour, population dynamics, social structures or even physiology.

1.2.1 Comparison of acoustic and visual surveys

Besides PAM, visual surveys is the second main approach to the biological study of cetaceans. Each comes with its pros and cons. The acoustic approach enables long term surveys at relatively low costs: placing a fixed antenna allows to monitor biological activities for several consecutive months, requiring human intervention only for the installation and extraction of the recording system. In contrast, the visual approach demands a continuous human implication throughout the survey, in the relatively inaccessible marine environment.

In terms of detection capacities, cetaceans can be heard from great distances (up to 200km for the blue whale [176]), even during deep dives, while they can be visually detected from relatively short distances (around 1km, depending on weather conditions) only when surfacing (less than a third of the time for sperm whales [202]). However, species had first to be classified visually before we could learn on their associated acoustic behaviour, and photo identification is still to this day the only reliable way to recognise individuals. Moreover, the observation of group sizes, behaviour, and body conditions still mostly relies on vision. The two approaches thus really are complementary.

1.2.2 Antenna types

PAM starts by placing hydrophone(s) (underwater microphones) to listen or record the acoustic environment. They can be fixed on the sea floor (bottom mounted), to a buoy (sonobuoy), to a cable towed by a boat (towed array), or directly to the hull of a boat or Autonomous Surface Vehicule (ASV) (Fig. 1.5). When recording with multiple synchronised hydrophones, one can also triangulate (infer the position of) sound sources, by measuring their Time Difference Of Arrivals (TDOAs) for instance. The types of recording devices, their placement in the water column, and their layout between each other have crucial impacts on the yielded recordings, facilitating or not the following signal processing analysis.

For that matter, when implementing acoustic recording systems, one has to make compromises. Indeed, the functioning time of a recorder is limited by its

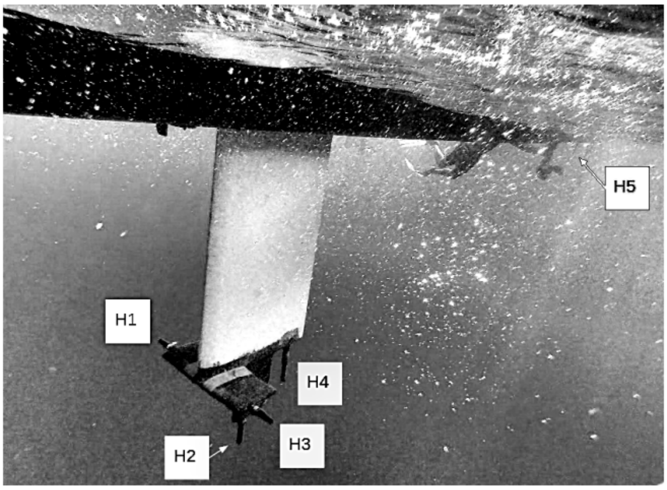


Figure 1.5: Example of a multi-hydrophone antenna mounted on an ASV, taken from Poupard et al. [153].

available resources (battery power and data storage). On the other hand, settings that allow for a more detailed view of the acoustic scene (increased number of channels, sampling frequency and/or the byte depth) also imply a higher rate of consumption of these resources.

1.2.3 PAM for biological studies

The first step of the analysis of acoustic signals typically comes down to the detection and classification of cetacean vocalisations. The amount of detection through time in long term surveys already provides significant information on the animals' lives. From these, one can infer population density [192] and seasonal or dial presence patterns [155]. When combining several antennas, these statistics can also be spacialised.

The analysis of the detected signals can then bring further knowledge on the recorded animals, such as community membership, current behaviour (hunting, socialising, courting), and individual characteristics (sexual maturity, body size for sperm whales [155]). These measures can themselves be put in a space-time perspective, potentially revealing patterns. In this way, PAM becomes useful to cetacean behavioural biology and stock structure assessment.

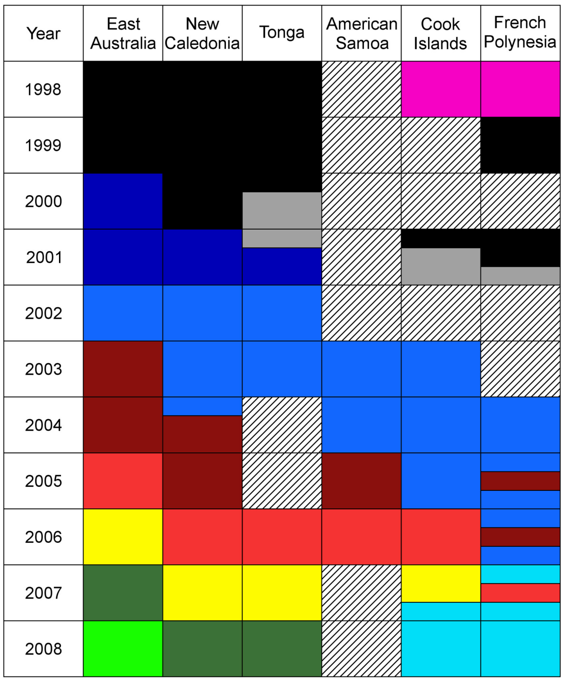


Figure 1.6: Example of passive acoustic monitoring findings: long term cultural transmission of humpback whale songs eastward through the Southern Pacific Ocean (each colour represents identified song types). Taken from Garland et al. [70].

A second field PAM provides to is the study of animal communication systems. Cetaceans indeed represent a significant part among the vocal learning species (along with birds, bats, seals, elephants, mice and primates). Identifying patterned sequences and associating them with species, communities and/or behaviours yields exemplary data on the development of vocal interaction in the animal kingdom [64]. Moreover, acoustic behaviour studies revealing cultural differences has provided knowledge on population dynamics [139] (Fig. 1.6) and social structures [72]. There is therefore a great diversity of biological wonders that PAM contributes to unveil.

1.2.4 Cetacean conservation

Some may question the amount of effort put into cetacean biology studies, considering that knowledge of nature is not in itself a sufficient driver. In that regard, it is to be kept in mind that cetaceans occupy the top of the ocean's food web, and therefore are significant regulators of their ecosystem as a whole. Moreover, the oceanic ecosystem is not only an important provider of food to humans, but also

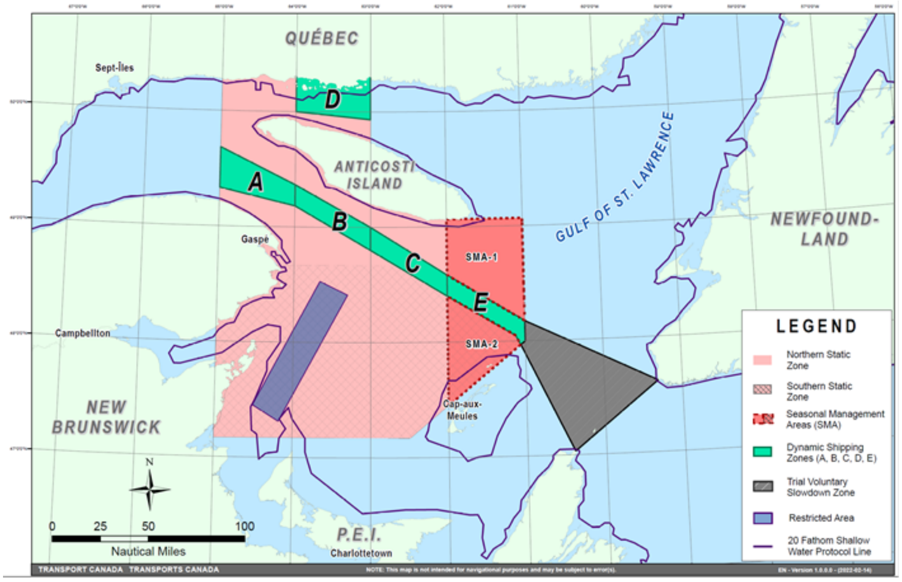

QUÉBEC
Sept-Îles
D
A
ANTICOSTI ISLAND
B
Gaspé
C
SMA-1
E
SMA-2
GULF OF ST. LAWRENCE
NEWFOUND-LAND
Campbellton
NEW BRUNSWICK
Cap-aux-Meules
N
0
25
50
100
Nautical Miles
P.E.I.
Charlottetown
LEGEND
Northern Static Zone
Southern Static Zone
Seasonal Management Areas (SMA)
Dynamic Shipping Zones (A, B, C, D, E)
Trial Voluntary Slowdown Zone
Restricted Area
20 Fathom Shallow Water Protocol Line
TRANSPORT CANADA TRANSPORTS CANADA
NOTE: This map is not intended for navigational purposes and may be subject to error(s).
EN - Version 1.0.0.0 - (2022-02-14)

Figure 1.7: Example of conservation measure in the Gulf of St. Lawrence in Canada [28]. Reduced speed zones are put in place all year round (red) and seasonally (green) to protect North Atlantic Right Whales.

crucial to breathe (it is responsible for around 70% of the atmosphere's oxygen production [79]). This field of study thus matters not only for the knowledge of planet earth's animal kingdom, but simply to our long term survival.

As stated previously, human activities heavily impact cetacean species, putting some of them close to extinction [104]. Therefore, it seems relevant that we learn how to mitigate this impact and work on cetacean conservation policies. Some regulation measures have already been put in place, e.g. the Marine Mammal Protection Act [62], speed regulations [61] (Fig. 1.7), and the definition of marine mammal sanctuaries [136, 189]. Monitoring the efficiency and/or need for regulations, as well as maximising their relevance (e.g. habitats and/or seasons of importance) can only be done via the knowledge of the animals, thus justifying their study.

1.3 Neural Networks and PAM

1.3.1 Automated PAM before Neural Networks

To carry out the aforementioned long term cetacean surveys, acoustic detections are needed. This process can be done by manually inspecting signals, especially their time / frequency representation (spectrograms). However, this is very costly in human efforts, which motivated the development of automatic detection mechanisms. With such systems in hand, researchers can seamlessly process months of data to yield results such as spatio-temporal presence statistics.

The development of detection systems has long been done with handcrafted algorithms [73]. They can be sufficient for some use cases, but often come quite limited, as the variety of sounds to detect and potential noises increase. Analysing long streams of data across recording devices and antenna locations demands highly robust detection systems, for which handcrafted algorithms remain unsatisfactory.

As an analogy, let us consider our ability to recognise our kin by the sound of their voice. Formally describing how to differentiate talking individuals seems nearly impossible, especially in a computer language. However, we know that given a hearing sense and sufficient cognitive capacities, by listening to a voice several times, we acquire the capacity to recognise it. This led the scientific community to start shifting towards machine learning algorithms, which are introduced in the following section.

1.3.2 Artificial neural networks

Training Artificial Neural Networks (ANNs) is the chosen approach for the automation of PAM throughout this thesis. It is one of the most popular techniques of machine learning, a field of computer sciences that approaches problem solving without programming solutions explicitly. Specifically, in machine learning, the algorithm is designed to approximate (or learn) the optimum solution to a problem, often formulated as a mathematical framework. An analogy could be made that genes encode a brain structure for it to learn but genes do not encode knowledge directly. Similarly, in machine learning, a learning framework is programmed, but the task's solution is to be learnt.

ANNs represent a major branch of today's machine learning, solving tasks in computer vision and speech recognition with performances and robustness highly superior to that of traditional handcrafted algorithms. This has motivated this research to apply ANNs to the field of PAM, making it the central topic of this thesis as described in the following section.

1.4 Thesis objectives

This thesis was co-financed by the GIAS European project, aiming to improve navigation security in the Mediterranean sea (western bassin). This thesis takes part in one axis of this project: the mitigation of whale-ship collision risk. For that purpose, a ‘smart bioacoustic buoy’ was designed, with the intent to acoustically detect the large cetaceans in a target zone (sperm whales and fin whales). Alerts could thus be transmitted close to real-time, for ships to adapt their speed or route accordingly.

Being a thesis in computer science, the goal is to design and implement the acoustic detection algorithms embedded in the buoy (collaborating with third parties on the hardware development). Moreover, motivated by the recent advances in deep learning, the ANN approach was chosen.

The work of training ANNs for the detection of cetacean vocalisations quickly expanded well beyond the initial needs of the GIAS project. Indeed, the team participates in a variety of projects, described in section 3.2.1. In each of them, the role of the team is typically to analyse large amounts of recordings to advance on biological questions. Hence the need for cetacean acoustic detection and/or classification mechanisms. Moreover, the performance demonstrated by ANNs in early experiments motivated to use them extensively on other species. This is how the objective of this work dissociated from the GIAS implementation to become a general study of applying ANNs to cetacean acoustic detection and classification.

1.4.1 Structure of the manuscript

This document is organised as follows. First, chapter 2 introduces the State Of The Art (SOTA) of the deep learning techniques that will be used in this study, along with their use for automated PAM. Then, chapter 3 will go through the species of interest for this work, the signals they emit, and the recordings available.

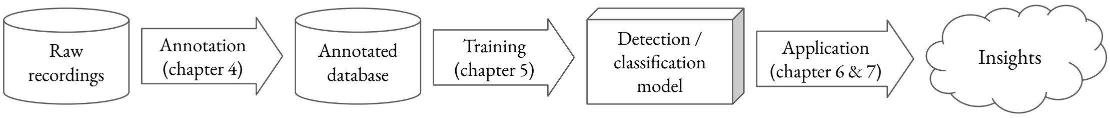


Figure 1.8: Flowchart of the typical process when using ANNs for bioacoustics. The main steps covered by this thesis are shown by arrows with their associated chapters.

The rest of the manuscript then revolves around the three main steps needed to address PAM with ANNs (Fig. 1.8). It starts with the construction of training databases, describing annotation procedures suited for a variety of constraints (depending on the recordings at hand and the target signals). Then, to train ANNs on these databases, architectures and frameworks are described to yield robust detection and classification mechanisms (again depending on constraints of computational power and target signals). Finally, for some of the trained models, applications are illustrated around two main uses: species conservation and communication modelling.

2
State of the art

Contents

The following chapter introduces the main technical aspects relevant to the subsequent work, lying between pedagogic and bibliographic objectives. It starts with the main techniques involved in building and training ANNs, in their most prevalent context in the literature (computer vision). Then, PAM of cetaceans (in general and using ANNs) are reviewed. Finally, the last section of this chapter intends to put past work into perspective with this thesis.

2.1 Neural networks for computer vision

If computer vision techniques can be used to tackle acoustic tasks, it is in part because sound can be represented as time-frequency images (spectrograms for instance). They describe content such that vocalisations appear as patterns with identifiable shapes for instance[1]. We will therefore first go through the state of the art in image pattern recognition before applying similar methods to our acoustic tasks. This is obviously not an exhaustive review of deep neural networks, but rather an overview of the key elements used in this thesis to build detection and classification systems.

2.1.1 Introduction to Artificial Neural Networks

The idea of emulating brain neural systems computationally emerged in the mid 20th century [63]. It is however only recently that ANNs have taken such an important part in applied mathematics and computer sciences, with the increased availability of data and computational power. The underlying approach to ANNs is to reproduce advanced processes emerging from the accumulation of simple operations, alike brains with neurons. Put mathematically, neurons would typically take the form of a simple linear transformation of an input x into an output y ($y = wx + b$). With their combination into large networks emerges the capacity of modelling high level functions such as classifying cat and dog images.

An ANN is defined by a network architecture (interconnection of neurons) and its neurons' weights (the linear transformations' coefficients, namely w and b). Like so, we can formulate the model g as a composition of linear layers $l_{\boldsymbol{\theta}_i}$, and the concatenation of all their weights (Eq. 2.1).

$$g_{\boldsymbol{\theta}}(\mathbf{x}) = l_{\boldsymbol{\theta}_1} \circ l_{\boldsymbol{\theta}_2} \circ l_{\boldsymbol{\theta}_3} \circ ...l_{\boldsymbol{\theta}_n}(\mathbf{x}) \tag{2.1}$$

We first design an architecture g before optimising its weights $\boldsymbol{\theta}$ for our task, typically with supervised learning. This paradigm consists in feeding the model

[1] In a way, our hearing system itself processes sound via a frequency decomposition with the cochlea

examples with their associated labels. For instance with our cats and dogs task, this means giving the model images of each class and asking it to predict the associated label, namely ‘cat’ or ‘dog’. An error $\mathcal{L}$ is then computed between the expected and the predicted labels. Like so, the training objective can be formulated as Eq. 2.2 to find the optimum weights $\hat{\theta}$.

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \; \mathcal{L}(\mathbf{y}, g_{\boldsymbol{\theta}}(\mathbf{x})) \tag{2.2}$$

Under the hood, the network learns a projection of the input images (often called embedding) from the pixel space to a new abstract one. Put simply, the more neurons in a network, the more complex the resulting projection can be[2]. Training thus becomes trying to learn the optimum embedding space to solve a given task.

There are two main limitations here, the first being the necessary computational power. Training a large ANN typically demands thousands of iterations, each of which consists in an update of millions of neurons. This is in part why we had to wait for the development of parallel computation with Graphical Processing Units (GPUs) to see the democratisation of ANNs. The second limitation, this time a human effort cost, is the necessary training data. To learn a robust solution, training typically demands thousands of examples for each class, with their associated label (often manually annotated) for the computation of the performance metric that will be optimised.

This leads us to the major challenge of training ANNs and modelling in general: robustness, or generalisation. Indeed, optimising a performance metric on a limited amount of examples might bring the curse of overfitting: when the model finds a solution that works for its given training data, but not the generalised solution that we desire (see Fig. 2.1). To give an example, coming back to the cats and dogs task, if all the cats we show the ANN are white and all dogs are black, it might just discriminate based on average pixel colours. This will lead to great performances on the training data, but will fail as soon as we try our ANN on a black cat image.

[2] Neurons are put in a stack of layers, thus the appellation ‘deep learning’

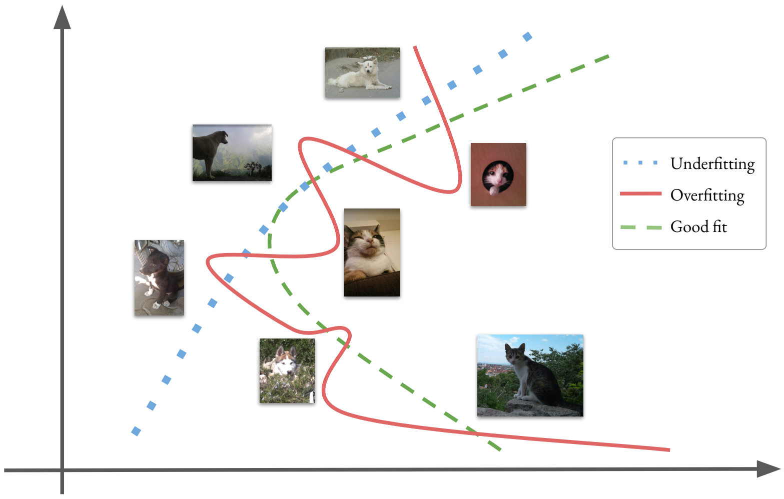


Figure 2.1: Illustration of the concepts of underfitting and overfitting, for the cats and dogs classification task. Lines denote discrimination boundaries, in a two-dimensional abstract embedding space.

As we will see throughout this thesis, most of the struggle in training ANNs comes down to enforcing generic solutions with limited training data.

2.1.2 Performance optimisation

As previously mentioned, training ANN comes down to trying to find the optimum weights for a task. This optimisation takes form as the minimisation of some error function, or loss (Eq. 2.2). This section describes the methods involved in optimising this loss, especially with Stochastic Gradient Descent (SGD). Then, the different loss functions that will be needed in this thesis will be introduced. Finally, we will go through the 'second level' of performance estimation and optimisation, employed to account for architecture and training quality after the weights have converged.

Optimising the loss

Depending on our task and label availability, let's consider a differentiable loss L to be minimised. A straightforward way of finding some function's minima is to follow the slope downwards iteratively ("gradient descent"). Furthermore, having multiple data points to account for in the computation of the loss, a stochastic estimate of the gradient can be used. This is the approached followed by the SGD algorithm [160], as expressed in Eq. 2.3. The amplitude of the update is defined by the learning rate α, which takes values between 0 and 1.

$$\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^{(i)} - \alpha \times \mathbb{E}_{\mathbf{x},\mathbf{y}}[\nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{y}, g_{\boldsymbol{\theta}}(\mathbf{x})]. \tag{2.3}$$

The choice of learning rate is critical to achieve convergence of the model's parameters. Indeed, a too small learning rate might result in getting stuck in a local minima, whereas with a too large one the actual minima might be skipped and the procedure may diverge. No generic learning rate is good for every task, so it will be one of the hyper-parameters to be tuned (see section 2.1.2). Moreover, the data used to compute the loss and update weights at each step (Eq. 2.3) needs to be defined: it is called a batch. Using the whole dataset at each step would be too costly in memory and computation, and using only one sample would hardly converge (the gradient would oscillate in different directions). Mini-batch SGD thus consists in using only a sub-sample of the available data at each $\boldsymbol{\theta}$ update. In a compromise between computation cost and each batch being representative of a global direction to follow, a "batch size" (number of data points) needs to be defined. It is part of the hyper-parameters to be tuned (see section 2.1.2).

To enhance convergence quality and speed, the community is now opting for learning rates that evolve through the course of the optimisation. This evolution (termed learning rate scheduling) can be a simple exponential decay, a decay when the loss plateaus, or more advanced periodic schedules with warm restarts [117]. No definite agreement has yet been made on the right schedule, and the answer might again be task specific.

Methods like SGD to iteratively update the model's parameters depending on the loss gradient are called optimisers. Several variations of SGD have been proposed since its original formulation, especially with gradient smoothing. The nesterov momentum [188] as well as the gradients' moments [101] serve that purpose.

Classification and detection losses

Because it will be needed for SGD, the chosen loss to optimise needs to be convex and differentiable. For our classification tasks, the accuracy is therefore not suitable since it relies on the *argmax* of the output vector. We will rather choose the Cross-Entropy (CE) instead, and will keep the accuracy for model evaluation, selection and validation (section 2.1.2).

The definition of the CE classification loss H is given in Eq. 2.4, with $\mathbf{y}$ the one-hot encoded label[3], $\hat{\mathbf{y}}$ the vector of predicted probabilities for each class, and C the set of possible classes.

$$\mathrm{H}(\mathbf{y}, \hat{\mathbf{y}}) = -\sum_{c \in C} y_c \log(\hat{y}_c). \tag{2.4}$$

To get normalised predictions of the model homogeneous to a probability distribution, we use the SoftMax function described in Eq. 2.5, given the unnormalised model output $\mathbf{z}$ (also called logits).

$$\hat{y}_c = p_{g,\boldsymbol{\theta}}(x|c) = \mathrm{SoftMax}(\mathbf{z})_c = \frac{e^{z_c}}{\sum_k e^{z_k}} \in [0,1],\ \sum_c p_{g,\boldsymbol{\theta}}(x|c) = 1. \tag{2.5}$$

This is appropriate for the multi-class classification tasks, when a higher confidence for an class implies lower probabilities for others. When solving multi-label classification tasks however, a sample can be assigned multiple classes, making the SoftMax assumption not appropriate. The Sigmoid function is then rather used to normalise logits to probability distributions (Eq. 2.6), and the sum of the independent Binary Cross Entropys (BCEs) as a loss.

$$\mathrm{Sigmoid}(z_i) = \frac{1}{1 + e^{-z_i}}. \tag{2.6}$$

The BCE is simply a special case of the CE, with $C = 2$. However, we can use single valued labels y and prediction $\hat{y}$ for its computation (Eq. 2.7).

[3] Vector of zeros except for the true class which is one

$$\mathrm{BCE}(\mathbf{y}, \hat{\mathbf{y}}) = -\frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i). \tag{2.7}$$

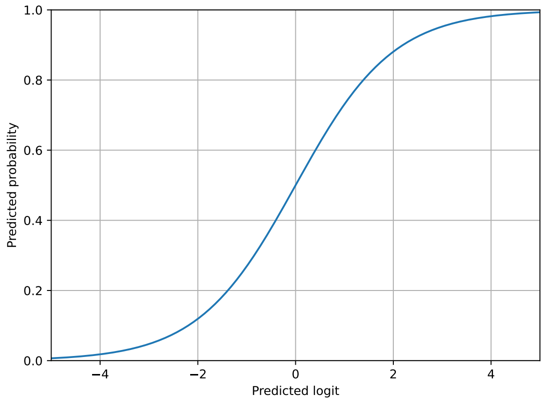


Figure 2.2: Sigmoid function (Eq. 2.6).

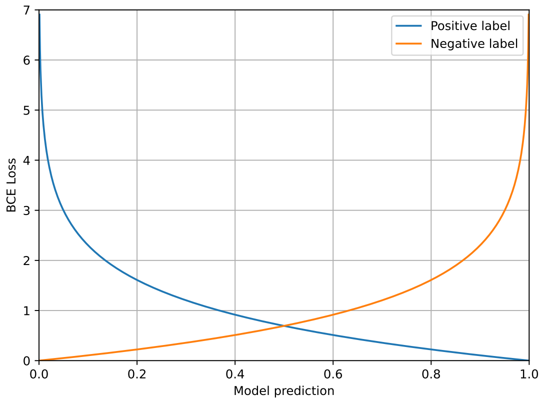


Figure 2.3: BCE loss (Eq. 2.7).

Through this thesis, the binary classification will be used as a proxy to solve detection tasks (one class being the target event to detect, and the other anything else). When running a classifier model post training, the predicted class will be argmax($\mathbf{z}$). For binary classifiers however, the output becomes a single value denoting the confidence in the presence of one class, equivalent to a detection confidence. A threshold is then set to binarise this continuous value (yielding a presence/absence decision).

Representation learning losses

The losses previously mentioned are suited when a sufficient amount of labels are available for supervised learning. When few or no labels are available, the literature proposes frameworks to learn semantically relevant embedding spaces, used subsequently by clustering algorithms or in supervised fine tuning. We call this process deep representation learning. Since this learning paradigm does not rely on labels for optimisation, it is referred to as Self Supervised Learning (SSL).

Triplet loss and contrastive learning Contrastive learning is a branch of SSL algorithms, where we enforce the models' output projection to ignore transformations applied to the input (transformations that do not imply a semantic change to the

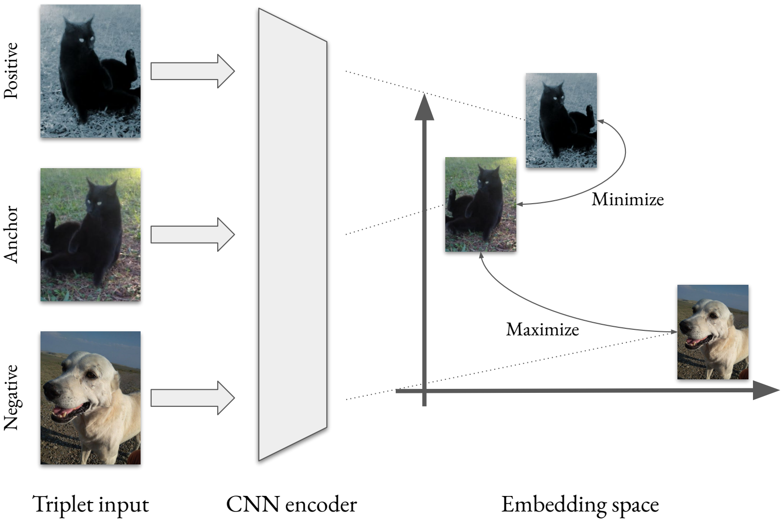


Figure 2.4: Illustration of the contrastive learning approach. The anchor and the negative are randomly sampled from the database, whereas the positive is a hand crafted transformation of the anchor. The distance metric to be minimised/maximised varies among implementations.

data). For that purpose, we will minimise the distance between the projection of a sample and that of its transformation w.r.t. the projection of other samples (see Fig. 2.4). In this way, rather than directly learning an embedding space for discrimination, the model is trained to learn an embedding space that reflects a desired notion of similarity and difference (the contrast). The mathematical formulation of this objective is termed as triplet loss since it uses the projection of three samples: an anchor (the original sample), a positive (the transformation of the anchor), and a negative (another unrelated sample). Several metrics have been used in the literature to measure distances between embeddings :

- The cosine similarity (SimCLR [31])
- The cross-entropy (UDA [209], fixMatch [180])
- The cross-correlation (Barlow [211])
- The mutual information (Invariant Information Clustering (IIC) [94])

These contrastive losses can also be combined with a regular classification loss in a semi-supervised paradigm, as seen in fixMatch [180] and UDA for instance.

They can then be considered as a form of training regularisation (see section 2.1.5).

Triplet loss and Siamese neural networks In a similar fashion than with contrastive learning, the triplet loss can be used in a supervised context. In this case, the positive of the triplet is a sample drawn from the same class as the anchor, and the negative is a sample from another class. We call this approach Siamese networks [23, 103]. Despite its use of labelled samples alike regular supervised classification training, this method focuses on learning an embedding space to measure samples' similarity, rather than an embedding space to discriminate among classes.

Reconstruction loss In other SSL frameworks such as Auto-Encoders (AEs) (see section 2.1.4), we will use a reconstruction loss, that reflects the fidelity of the reconstructed sample w.r.t. the original input. This can simply take the form of a Mean Square Error (MSE) between the input and the reconstructed image (pixel loss). There are also more advanced approaches such as the perceptual loss which uses the MSE in the latent space of an independently trained encoder to have comparison at a higher level than pixel wise [95].

Model validation

Once our model has optimised the loss function until convergence, we usually want to measure its performance with interpretable metrics, and with new data.

Performance validation metrics (detection) For detection tasks, which are the most common in this thesis, these metrics reflect the proportion of target signals that we won't miss (recall) and the proportion of detections that will be the signal we look for (precision). This is typically described via the areas under the Reveiving Operating Characteristics (ROC) and Precision Recall (PR) curves. For varying thresholds, they give average values of recall/fall-out and precision/recall respectively. Note that the area under the ROC and PR curves will be referred to as Area Under the ROC Curve (AUC) and mean Average Precision (mAP) respectively. Equations 2.9 and 2.10 formulate their computation with *rec*, *prec*, and fal denoting recall, precision and fall-out respectively. TP, P, PP, FP, and N denote numbers of true positives, positive ground truths, positive predictions, false positives, and negative ground truths respectively. Some are a function of a threshold noted λ, used to binarise continuous prediction values.

$$rec(\lambda) = \frac{TP(\lambda)}{P}, \quad prec(\lambda) = \frac{TP(\lambda)}{PP(\lambda)}, \quad fal(\lambda) = \frac{FP(\lambda)}{N}, \tag{2.8}$$

$$AUC = \int_0^1 rec(\lambda)\ dfal(\lambda), \tag{2.9}$$

$$mAP = \int_0^1 rec(\lambda)\ dprec(\lambda). \tag{2.10}$$

These two last metrics are similar, but differ on the measurement of false alarm rate: the mAP normalises on the number positive predictions whereas the AUC normalises on the number of negative samples. This difference has a significant impact especially with imbalanced datasets.

Performance validation metrics (classification) For multi-label classification tasks (each sample can be assigned to multiple classes), we will average the independent detection performance of each class. As for multi-class classification (each sample is assigned to a single class), we will rather compute the accuracy as the rate of correct predictions. Averaging methods for the performance metric should be chosen to account for class imbalance or not (i.e. averaging the performance per class before averaging between classes or averaging performances per samples directly).

Performance validation metrics (representation learning) Latent representations learnt by optimising a triplet loss or a reconstruction loss are intended to reflect semantic similarity. Therefore they can serve to measure relevant distances between samples, allowing their clustering.

To measure the relevance of clusters against a set of labels, the Mutual Information (MI) noted $\mathrm{I}(X;Y)$ can be used. It is computed as the Kullback-Leibler (KL) divergence between the joint and the marginal distributions of labels X and clusters Y (Eq. 2.11).

$$\mathrm{I}(X;Y) = \sum_{y \in \mathcal{Y}} \sum_{x \in \mathcal{X}} P_{(X,Y)}(x,y) \log \left(\frac{P_{(X,Y)}(x,y)}{P_X(x) P_Y(y)} \right) \tag{2.11}$$

To compute the Normalized Mutual Information (NMI) (between 0 and 1), one can divide I by the average of the entropy of X and Y (Eq. 2.12).

$$\mathrm{NMI}(X;Y) = \frac{\mathrm{I}(X;Y) * 2}{\mathrm{H}(X) + \mathrm{H}(Y)} \tag{2.12}$$

Validating with new data To reduce human effort, we usually desire models to be applicable to different recording devices, locations and background noise conditions. However, ANNs have the tendency to overfit, showing a decrease in performance on data different from those seen in training. In machine learning, to account for this potential overfitting, models' performance are usually measured on new data (not seen in training). It is called the test set, as opposed to the training set which is used for the iterative loss optimisation.

When designing experiments, one must ensure that the test set is significantly disjoint from the training set to relevantly measure generalisation. For instance, in sound event detection tasks, we might want to test our model on recording devices, environments, and emitters that have not been observed during training. How well the model performs facing such domain shifts is the only reliable measure that should be taken into account, especially if we want the model to be reusable in new conditions. In the contrary, if a model has been trained and tested on similar data, a large performance drop should be expected as soon as the data changes.

Hyper-parameter tuning

We mentioned the iterative optimisation of a loss through the update of the model's weights (Eq. 2.3), but other parameters can be tuned to enhance performance. The model architecture and the optimiser have numerous settings that need to be fixed before training and have a huge impact on the training in both convergence speed and the found loss minima. We call them hyper-parameters.

Often, hyper-parameters are tuned to optimise performance on a separate set of data called ‘validation set’. Doing so, we keep the test set for the final performance evaluation, and avoid finding hyper-parameters that would be specific to the test set. Throughout this thesis, accounting for the efforts put into having a test set disjoint from the training set and their sufficient size (reducing the probability of overfitting hyper-parameters), the test set was directly used to tune hyper-parameters.

Each model training taking at least several minutes on a super computer, the exploration of hyper-parameter combinations to improve model performance is a challenging task. Dedicated Algorithms have been proposed to efficiently explore the hyper-parameter space. They combine several principles among which the early stopping of low performing models [113], as well as muting high performing ones for the next trials [89].

2.1.3 Layers

As previously mentioned, the accumulation of layers of neurons (linear transformations) forms the basis of ANNs functioning. However, several other types of layers exist. Let us dive deeper into the different layers that will be needed for this thesis, and each of their specific utility.

Convolution

Convolution is a mathematical operation that describes the integral of the point-wise product of two functions, with a varying shift on the input variable. It is usually noted with the asterisk symbol (see Eq. 2.13, given a kernel f of size M and a function g).

$$(f * g)[n] = \sum_{m=0}^{M} f[m] \times g[n-m] \tag{2.13}$$

Typically, in image processing, we will use this operator to slide a filter (or kernel) over a larger image. The output of the convolution will be maximal where the filter matches most the image, or in other words where there is the strongest correlation. In 1995, LeCun et al. [111] introduced the concept of using convolution operators in neural networks; Convolutionnal Neural Networks (CNNs) were born.

Before that, pixels where given independently to input neurons. The input image size was thus fixed for a given network architecture, and a displacement of patterns within an image would mean a totally different response of the network. With CNNs, the network's neurons take the form of kernels (or filters), which are convolved onto input images. Like so, patterns are searched all over the image, independently of their placement.

This behaviour is called spatial invariance, and is crucial for pattern recognition in images (looking for a cat within an picture or a vocalisation within a spectrogram for instance, independently of their placement). This characteristic led CNNs to become unavoidable in the field[4].

In terms of mathematical definitions, a traditional ANN layer is described as $\mathbf{y} = \mathbf{W}\mathbf{x} + \mathbf{b}$ with $\mathbf{x} \in \mathbb{R}^{in}$ an input vector, and $\mathbf{y} \in \mathbb{R}^{out}$ an output vector. In deep neural networks, the input of a layer is the output of the preceding one. The weights $\mathbf{W}$ and $\mathbf{b}$ are thus matrices defined in $\mathbb{R}^{out \times in}$ and $\mathbb{R}^{out}$ respectively, with in and out being the number of neurons in the preceding and current layers respectively.

As for CNNs, a layer is no longer composed of a stack of neurons, but rather a stack of kernels. The behaviour of a kernel of width w_k and height h_k is formulated by Eq. 2.14, given an input of width w, height h, and depth d.

$$\mathbf{Y} = \mathbf{W} * \mathbf{X} + b, \quad \mathbf{X} \in \mathbb{R}^{h \times w \times d}, \mathbf{A} \in \mathbb{R}^{h_k \times w_k \times d}, b \in \mathbb{R} \tag{2.14}$$

The convolution integration (sum) is done over the 3 dimensions, but the shift will occur on the width and height dimensions only, making $\mathbf{Y} \in \mathbb{R}^{h \times w}$. The outputs of each kernel of the layer will eventually be stacked to form the depth dimension for the input of the next layer[5] (see Fig. 2.5).

[4] Let aside the recent rise of transformers for computer vision [144]

[5] The colour dimension of input images are also put as depth dimension

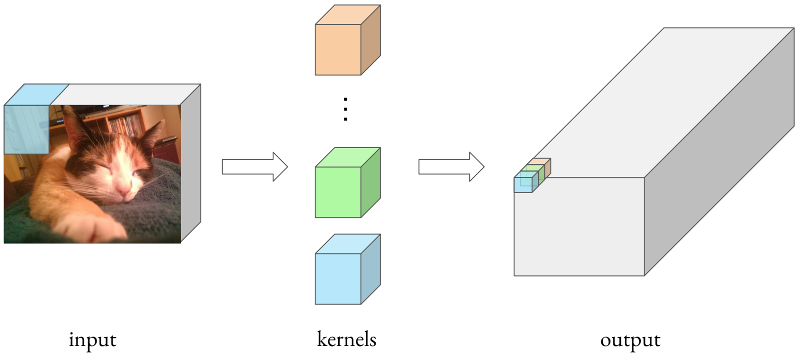


Figure 2.5: Convolution layer. Blue denotes a slice of the image, a kernel, and the resulting point in the output image (the sum of the point-wise product between the two). The number of kernels will define the depth of the output cuboid.

A CNN layer is thus defined by the number of input features it processes, its number of kernels, and their width and height. The number of trainable parameters in a layer is given by Eq. 2.15.

$$\#\boldsymbol{\theta} = d_{in} \times w_k \times h_k \times d_{out} + d_{out}. \tag{2.15}$$

Depth-wise separable convolution

As presented in the previous section, convolution kernels are cuboids, with a depth that fits the depth of the input. The filters are thus designed to find patterns that are interconnected depth-wise. However, often, we might want patterns to be filtered independently through the input image depth, for a subsequent depth-wise combination. This is the idea introduced by depth-wise separable convolutions, first used in the context of a CNN by Chollet [32].

In this new type of convolution layer, we dissociate the spatial filtering and the depth-wise combination in two stages, as opposed to regular convolutions that process it all at once. A kernel remains cubic, but the convolutions are separated depth-wise, thus yielding a cuboid, when a regular convolution kernel yields a flat image. The combination of the features then happens with the point-wise stage,

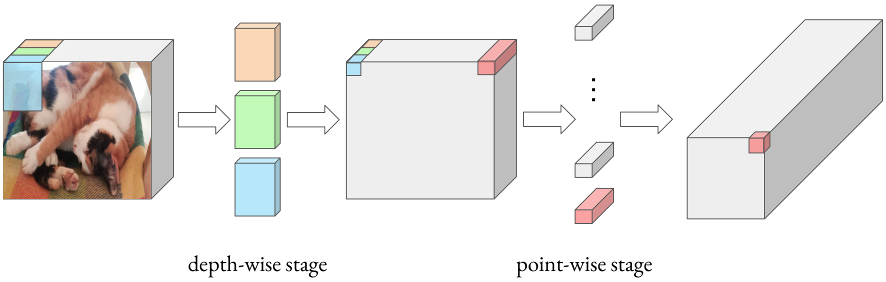


Figure 2.6: Depth-wise separable convolution. In the depth-wise stage, each depth bin is convolved with its own kernel independently. For the point-wise stage, the depth dimension is combined point by point by various vector kernels, each of which will result in a depth bin in the output.

similar to a convolution with a kernel of width and height 1. This stage can be repeated to obtain an output depth (see Fig. 2.6).

For comparison with the regular convolutions, the number of trainable parameters in a depth-wise separable convolution layer is given by Eq. 2.16.

$$\#\boldsymbol{\theta} = d_{in} \times (w_k \times h_k + 1 + 2 \times d_{out}) \tag{2.16}$$

Having less parameters involved in a network means less computational complexity for inference and for weight updates. Moreover, this type of convolution has shown improved generalisation performances for computer vision tasks [32]. Indeed, limiting feature inter-dependence could limit potential overfitting, alike the dropout [184] technique introduced later on.

Pooling

As previously mentioned, convolution enables spatial invariance. However, it doesn't treat the scale problem. Indeed, some patterns might have to be detected independently of their scale in input images. Moreover, detecting large patterns would require large kernels, which are expensive in computation and memory. For this purpose, pooling layers enable a progressive decrease in image resolution (on the width and height dimensions), so that deeper layers can have a larger scale view without requiring larger kernels.

Often, we want to simply denote if features (depth bin) were activated in a given area, with a lower resolution. Max-pooling layers are well suited for this, simply keeping the maximum value in a window with a *stride* > 1 (the stride is the amplitude of windows' steps in pixels). Typically, max-pooling layers are placed after 2 or 3 convolution layers.

Non-linearity layers

Even if they are spatialised, convolution layers remain a simple linear transformation of the input, and accumulating linear transformations successively is equivalent to a single linear transformation (see Eq. 2.17).

$$w_2(w_1x + b_1) + b_2 = (w_2w_1)x + (w_2b_1 + b_2) \tag{2.17}$$

Therefore, building deep networks by accumulating layers of neurons would not add to the complexity the network is able to model. In order to model non-linear functions up to a great complexity, non-linearity layers in-between linear layers are thus needed. Common non-linearity layers are Rectified Linear Unit (ReLU) ($y = \max(0, x)$), leaky ReLU, TanH, among others.

Moreover, functions such as ReLU allow to insert zeros in numerous dimensions of vectors. This serves the stabilisation of gradients during the optimisation and has an effect of sparsity enhancement (latent representations lie in lower-rank manifolds).

2.1.4 Architectures

Projection using CNNs

Before ANNs, non linear Support Vector Machines (SVMs) [2] were used for a similar purpose: learning the optimum projection of data points to make them linearly separable. Only their approach to optimisation differ. In our case study of CNNs, we typically want to project an image from the pixel space to a lower dimensional space that embeds semantic content. We often refer to CNNs as encoders for this projection property.

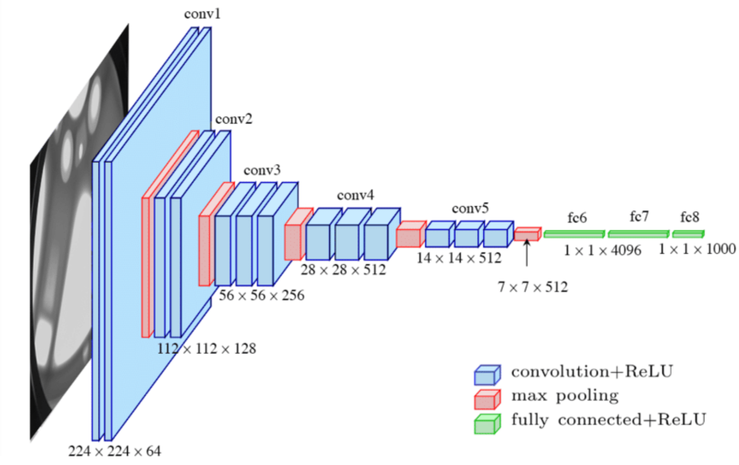


Figure 2.7: Visual Geometry Group (VGG)16 architecture [175]. Dimensions at each layer are given in this order: *height* × *width* × *depth* (image taken from Ferguson et al. [54]). The encoder part of the CNN is in blue and red, and the projection part is in green.

The projection is usually the last operation of a network, and done using fully connected layers after flattening the image (compression of the width, height, and depth into a single dimension, see Fig.2.7). In the case of classifiers, the dimensionality of the output projection is defined by the number of possible classes, each dimension denoting the confidence for one class.

Interpretation of the model's output

For classifiers, the dimensionality of the output is defined by the number of possible classes for our task. Indeed, each output feature will describe the confidence of the model on the presence of one class in the input. We will thus train our model to, given an input and its associated class(es), maximise the confidence value of the 'present' class(es), while minimising those of the 'absent' class(es). For the case of binary classification, networks can have either one or two output feature(s)[6].

Standard architectures

Numerous architectures have become a *de facto* standard and are commonly used by the deep learning community. In most cases, starting the design of a new

[6] Single output models can be seen as detection systems, as we will see through several use cases in this thesis.

architecture from scratch seems unnecessary and counterproductive (as long as the task at hand is relatively similar to the one of the standard architecture). Through this thesis, experiments will make use of three architectures coming from the ImageNet computer vision benchmark [43]: VGG [175] and ResNet18/50 [81].

- The VGG16 architecture is presented in Fig. 2.7, and is a classic convolutional encoder tailed by fully connected layers.
- The ResNet18 and ResNet50 architectures are composed of residual blocks, which introduce 'skip-connections' (the output of a block is the sum of its processed input and the original input). Their associated number denotes the number of layers that composes them.

These two types of architectures were chosen as they are (or have been) the baseline in image classification tasks, therefore considered standard CNN architectures, even for bioacoustic tasks (see section 2.2.4).

Auto-Encoders

As seen in the previous section, encoders can serve classification tasks, but they can also take part in bigger systems such as AEs. AEs may serve tasks of dimensionality reduction, operated with an encoder (see Fig. 2.7). To enforce the conservation of information, the encoder is followed by a decoder, that reconstructs the input image from the low dimensional space (called bottleneck). The encoder and decoder combination (called AE) is trained to compress and reconstruct the input most faithfully, despite the low dimensional bottleneck.

The compression that AEs offers enables a removal of random or unstructured information (denoising), and a lower dimensional space which often facilitates clustering. Indeed, clustering relies on sample distance estimations which are unreliable in the pixel space and suffer the curse of dimensionality.

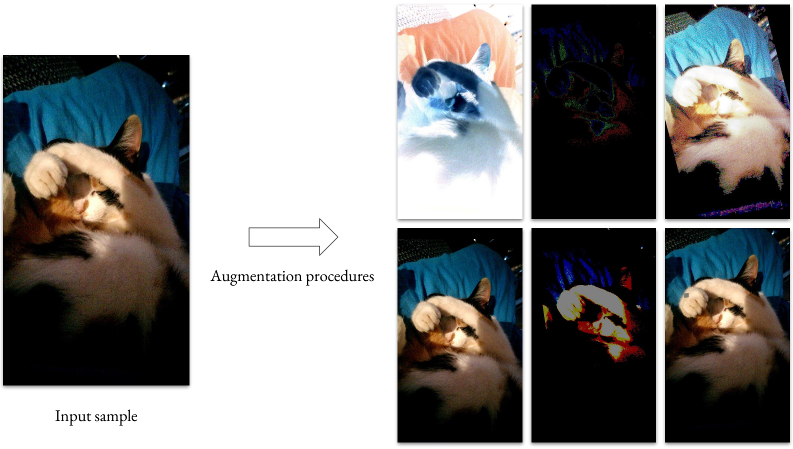


Figure 2.8: Using data augmentation enables new samples to be derived from original ones, while conserving the label. Transformations are randomly sampled among several texture and position alterations.

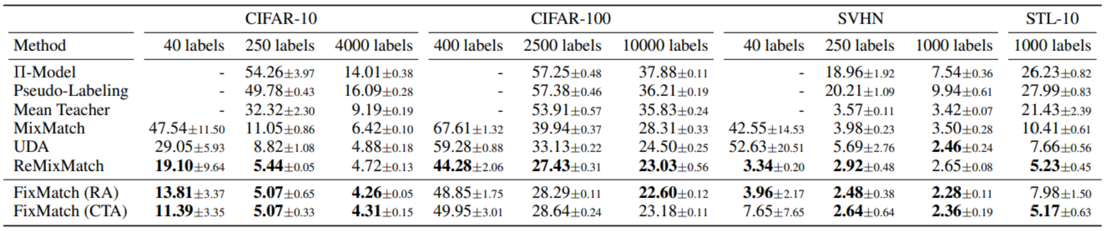
2.1.5 Training regularisation

Methods employed during training to reduce potential overfitting and enhance generalisation are called regularisation. They are especially relevant when a limited amount of training data is available (the case of many bioacoustics tasks). Some of these approaches come down to increasing the variability of the data both in the input and directly in the activations of the network.

Data Augmentation

Introducing variability to the input data is widely used to avoid overfitting, especially with small datasets. The idea is to generate new data samples out of the existing ones, thus increasing the dataset size, without needing more annotation. To do so, we apply randomised transformations, realistic or not, with the only constraint that we must ensure not to change the sample's class. For image classification, RandAugment [37] has now been accepted as the standard augmentation policies, combining texture and shape transformations (see Fig. 2.8). We will go through data augmentation for acoustic tasks in section 2.2.3.

Another branch of data augmentation worth mentioning is MixUp [214], which combines two input samples and their labels, thus creating 'in-between' data points. The combination takes form as a simple weighted sum of inputs and labels, which



	CIFAR-10			CIFAR-100			SVHN			STL-10
Method	40 labels	250 labels	4000 labels	400 labels	2500 labels	10000 labels	40 labels	250 labels	1000 labels	1000 labels
Π-Model	-	54.26±3.97	14.01±0.38	-	57.25±0.48	37.88±0.11	-	18.96±1.92	7.54±0.36	26.23±0.82
Pseudo-Labeling	-	49.78±0.43	16.09±0.28	-	57.38±0.46	36.21±0.19	-	20.21±1.09	9.94±0.61	27.99±0.83
Mean Teacher	-	32.32±2.30	9.19±0.19	-	53.91±0.57	35.83±0.24	-	3.57±0.11	3.42±0.07	21.43±2.39
MixMatch	47.54±11.50	11.05±0.86	6.42±0.10	67.61±1.32	39.94±0.37	28.31±0.33	42.55±14.53	3.98±0.23	3.50±0.28	10.41±0.61
UDA	29.05±5.93	8.82±1.08	4.88±0.18	59.28±0.88	33.13±0.22	24.50±0.25	52.63±20.51	5.69±2.76	**2.46**±0.24	7.66±0.56
ReMixMatch	**19.10**±9.64	**5.44**±0.05	4.72±0.13	**44.28**±2.06	**27.43**±0.31	**23.03**±0.56	**3.34**±0.20	**2.92**±0.48	2.65±0.08	**5.23**±0.45
FixMatch (RA)	**13.81**±3.37	**5.07**±0.65	**4.26**±0.05	48.85±1.75	28.29±0.11	**22.60**±0.12	**3.96**±2.17	**2.48**±0.38	**2.28**±0.11	7.98±1.50
FixMatch (CTA)	**11.39**±3.35	**5.07**±0.33	**4.31**±0.15	49.95±3.01	28.64±0.24	23.18±0.11	7.65±7.65	**2.64**±0.64	**2.36**±0.19	**5.17**±0.63

Figure 2.9: Error rates on CIFAR-10, CIFAR-100, SVHN and STL-10 on 5 different folds (taken from Sohn et al. [180]).

we will feed our model with like a regular sample. This simple concept of giving mixtures of 2 instances as training samples has shown to improve generalisation in most computer vision tasks with standard architectures [214].

Within-network regularisation

By introducing perturbations and variability within the network, we can mitigate its dependency to highly specific events, presumably increasing its robustness. Dropout [184] follows that incentive by randomly deactivating neurons or kernels (putting their activation to 0). The probability of discarding is defined by the dropout hyper-parameter p, commonly set to 0.25.

A second common way to regularise the network while training is to enforce the model to rely on as few weights as possible [106]. To do so, we introduce a new term in the loss: the L_2 norm of all parameters, weighted to control its impact. We call this method weight decay, and its weight introduces another hyper-parameter to the learning framework.

Leveraging unlabelled samples

As seen in section 2.1.2, contrastive losses can be used to train encoders for resilience to data augmentation. Several algorithms have been published to incorporate this, often termed as consistency training. Tab. 2.9 summarises their performances on semi-supervised learning datasets with varying proportions of labelled samples. The fixMatch algorithm [180] combines a supervised loss, pseudo labelling, and consistency training in one framework to achieve SOTA performances for the tasks with the fewest labels.

2.2 Cetacean acoustic detection and neural networks

After presenting CNNs in their original context of computer vision, let us discuss their application to cetacean monitoring. This section starts with techniques used before the apparition of ANNs in the field. It will then review acoustic pattern recognition via spectrogram images, followed by specific techniques and past use cases of PAM using ANNs.

2.2.1 Automated PAM

Template Matching

A straightforward way of implementing cetacean vocalisation detection mechanisms is to search for localised energy in a target frequency band, and yielding a detection when it surpasses a given threshold. For instance, it is known that some fin whale vocalisations are 20 Hz centered pulses that last approximately 1sec. The signal can thus be analysed in search for localised energy peaks in that time / frequency range.

Further extending this concept, strong correlations between recordings and a prototype of target signal can be looked for directly. This can be achieved either in the time domain (waveforms), the frequency domain (spectrums), or in the spectro-temporal domain (spectrograms). We call these techniques template matching, or matched filter.

Such approaches have been used extensively [22, 125, 204, 8, 125], but still suffer from the fact that they only work when target signals show enough consistency to be described by one or several templates. This is not the case for orca vocalisations for example, that show great spectro-temporal variability. Techniques such as dynamic warping can, to some extent, help coping with this challenge, as demonstrated by Somervuo [181] for bird classification.

Pitch tracking

Other detection and classification algorithms rely on the fundamental frequency (or pitch) contour of vocalisations. It can be estimated via the instantaneous peak

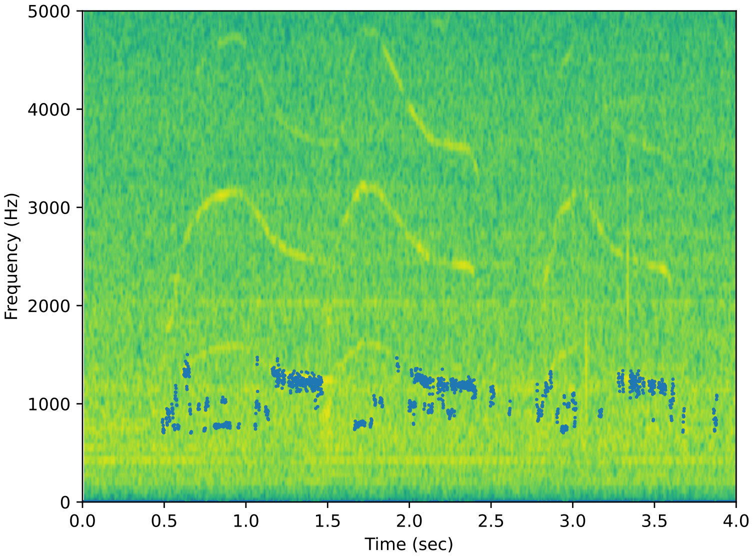


Figure 2.10: Difficulty in estimating the pitch on orca calls (OrcaLab recording). This estimate was done via the auto-correlation algorithm using the parselmouth python package [90].

frequency, spectrogram thresholding, or spectrum auto-correlation for instance. Once the pitch contour is extracted, one can infer features such as the duration, frequency range, or frequency variation. These can later serve filtering and/or clustering of vocalisations, potentially enabling the identification of species or vocalisation units [12]. Contours can also be compared directly as pitch sequences to measure similarity between vocalisations. In this context, dynamic time warping can be used to cope with temporal distortions, as shown by Brown et al. [26] for orca call classification, and by Deecke and Janik [40] for automated unit categorisation.

However, as Figure 2.10 demonstrates, these pitch based methods still suffer from the difficulty to robustly estimate frequency contours, especially in low Signal to Noise Ratio (SNR) conditions and in the presence of transitory impulses (odontocete clicks for instance). Nonetheless, more robust frequency contour estimation methods are being developed [114], (as compared to spectrum auto-correlation presented in Figure 2.10 and used by Deecke and Janik [40]).

Machine learning

Once vocalisation features were extracted (via the pitch or Mel Frequency Cepstral Coefficientss (MFCCs) for instance), machine learning algorithms have been used to classify them in supervised and unsupervised settings. Roch et al. [161] for instance compared SVMs and Gaussian Mixture Models (GMMs) for the classification of odontocete clicks based on MFCCs. Brown and Smaragdis [25] on the other hand used a GMM and Hidden Markov Model (HMM) based approach to classify orca calls. Esfahanian et al. [50] on the other hand explored the classification of dolphin whistles using time-frequency contour features and an SVM.

These methods heavily depend on their input features, often either too specific and not estimated accurately (pitch) or too generic and giving only a gross description of the signals (MFCC).

Overall limitations

All in all, despite efforts to build robust algorithms [82], they hardly cope with the wide variety of perturbations found in underwater recordings. Indeed, these induce acoustic masking and heavily alter signals, hindering template correlations and/or pitch estimates. Furthermore, noise from boats, waves, currents, sonars, or even earthquakes take a variety of acoustic forms, that potentially strongly correlate with whale vocalisation templates [204].

In a general sense, for studies to base their results on automatic detections, underlying algorithms need to be robust to low SNR conditions and heavy disturbances, or important biases will be introduced. Take for instance studies on the impact of marine traffic on the wildlife: if boats trigger or impeach detections, further interpretations will be dramatically falsified.

Tuning templates and/or thresholds to cope with all possible perturbations can be very demanding, and sometimes the global compromise simply does not exist. In that sense, ANNs might be able to push forward automated PAM systems, by seamlessly learning robust feature representations for the detection and classification of cetacean vocalisations.

2.2.2 Preparing the network's input (frontend)

A widespread preliminary feature extraction of acoustic signals is its frequency decomposition, this section describes how waveforms can be compiled into images (spectrograms).

Let us start with an acoustic recording. It is described digitally by a sequence of samples $\mathbf{x} = \{x[i]\}_{1..n}$ that denotes the evolution of pressure through time. The number of samples recorded per second is given by the sampling frequency, noted f_s.

Fourier

The Fourier transform is a major tool in signal processing. It allows to describe any signal as a sum of sinuses, each characterised by an amplitude and a phase. This representation is called the spectrum. Given our acoustic signal $\mathbf{x}$, the Discrete Fourier Transform (DFT) will yield a spectrum $\mathcal{X}_f$ that gives complex numbers as a function of frequencies. These complex numbers describe each frequency component of the signal, with the amplitude as the modulus and phase as the angle. The behaviour of the DFT $\mathcal{F}$ of a signal of size N is described by Eq. 2.18.

$$\mathcal{X}_f = \mathcal{F}(\mathbf{x})_f = \sum_{n=0}^{N} x[n] e^{-i\frac{2\pi}{N}fn}. \tag{2.18}$$

Fast Fourier Transform (FFT) implementations of the DFT are available, enabling a reduction of the complexity from $O(n^2)$ to $O(nlog(n))$.

Short Term Fourier Transform

Numerous signal processing techniques, especially those presented in this thesis, rely on spectrograms. A spectrogram is a matrix representation of a signal, with values denoting magnitudes (square modulus of Fourier values) for each frequency and time bin (rows and columns respectively). It results from the juxtaposition of successive DFTs, computed by sliding a window over the signal.

$$\mathbf{S}_{f,t} = \left| \sum_{n=0}^{NFFT} \mathbf{x}[t \times hop + n] e^{-i\frac{2\pi}{NFFT}fn} \right|^2. \tag{2.19}$$

Throughout this thesis, spectrogram rows and columns will be referred to as frequency bins and time bins respectively.

Several parameters are to be set prior to the Short Term Fourier Transform (STFT) computation, that define the sliding window's behaviour: the window size $NFFT$ and the hop size *hop*. Along with the sampling frequency f_s, these will define the range and resolution of our resulting spectrogram :

- The sampling frequency will affect the maximum frequency represented by our spectrogram: $f \leq \frac{1}{2} f_s$ (Nyquist theorem).
  We sometimes downsample the signal during preprocessing to withdraw high frequency contents when non relevant. Downsampling also drastically reduces the downstream computation complexity.

- The window size defines the length of the signal to be decomposed.
  A bigger window will yield a more detailed representation frequency wise. However, it will also blur short transitory events (the yielded spectrum is an average of the frequency contents in the window).

- $NFFT$ is the number of points used in each DFT. It will define the number of frequency bins of the resulting spectrogram: $\Delta f = \frac{f_s}{NFFT}$. The DFT size can be larger than the window size, in which case borders are filled with zeros (zero padding). Like so, short transitory events are preserved as compared to using a larger window. Note that a larger $NFFT$ also implies more computation per DFT.

- The hop size defines the temporal sampling rate of the spectrogram:
  $\Delta t = \frac{hop}{f_s}$.
  A smaller hop size will yield a more detailed spectrogram, but also implies more computation (each step demands a DFT).

These parameters have a crucial impact on the spectrogram, and thus on how well our target signal(s) will be represented (see Fig.2.11). Finding the appropriate spectrogram settings is thus the first step in building any spectral based detection algorithm, ANNs included.

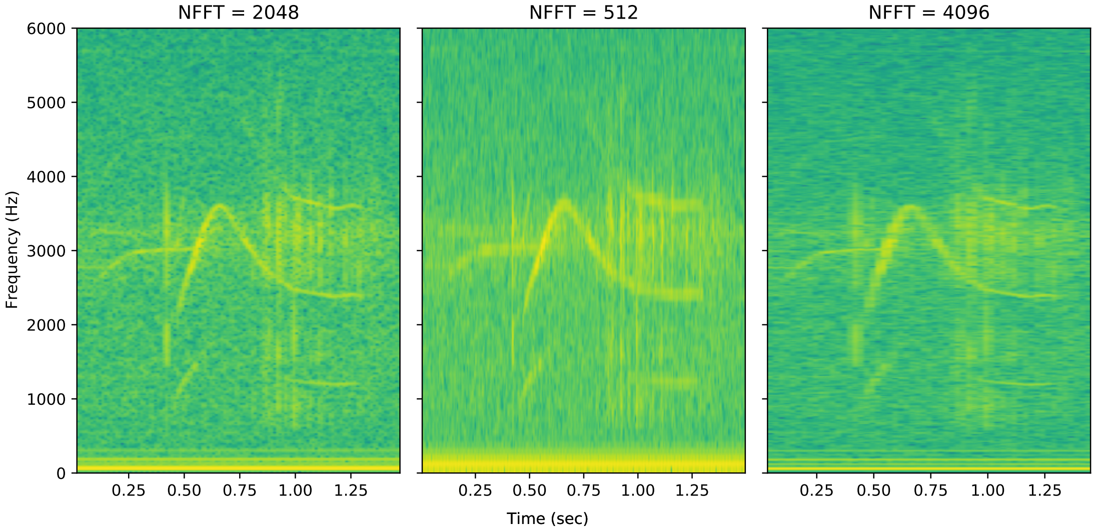


Figure 2.11: Spectrograms of an orca call (OrcaLab recording) with varying $NFFT$.

Alternative to the Fourier transform Sometimes, both temporal and frequency high resolutions are needed, and a satisfying Fourier window size does not exist. While Fourier uses the same window size for all frequencies, algorithms such as the wavelet transform propose a non uniform sampling of the time-frequency space, allowing for a better compromise in terms of temporal and frequency resolution.

Only a wavelet transform allows for a satisfactory representation of both low frequency and high frequency events, which can be useful in PAM applications.

Researchers have thus studied the use of wavelet transforms as frontends, for instance with cetacean click detection [116]. Further studies have also experimented on combining Fourier and wavelet transforms into multi-channel spectrograms, as a frontend for speech recognition [6] or bird classification [213]. Also, Stowell and Plumbley [187] have used chirplet transforms to analyse bird songs.

Nonetheless, the Fourier transform remains the choice in a wide majority of applications, because of its convenience of use and its efficient FFT implementation. Throughout this thesis, the large majority of experiments are based on the Fourier transform, to focus more on the effects of other components of the analysis such as downstream processing.

Beyond the empirical choice of STFT parameters In machine learning frameworks, finding the best spectrogram parameters can be part of the optimisation process. For instance, it can be done via STFT differentiation [215], or trainable Gabor filters [212] that recently reached SOTA performances on several acoustic recognition tasks.

Indeed, when there is a wide variety of target signals, the empirical choice of the right STFT parameters can be challenging. Optimising them through learning will lead to a compromise between several parameters, but may not be optimally suited for each type of target signals. Multi-channel spectrograms offer a solution to this issue, by giving a stack of spectrograms with different parameters to the model (they must be interpolated to match in time / frequency resolutions). Studies that have experimented on this technique have not seen a significant improvement so far [132, 193], let aside the computational cost implied by such approach.

Mel-spectrograms

Humans have a logarithmic sensibility to frequencies: we perceive a constant tonal shift when frequencies are multiplied by a constant. Besides, harmonic structures of acoustic signals also often show logarithmic behaviours. To have a spectrogram representation that reflects this phenomenon, the Mel transform changes the linear layout of a frequency domain into a logarithmic one: the Mel scale. The Mel scale describes a frequency layout that follows human perception of tones in terms of hearing range, but also such that a constant shift in Mel bin will be perceived as a constant shift in tone. However, we can extend this scale to a wider range of frequencies, thus extrapolating the human perception into frequencies suited for the hearing range of cetaceans for example.

To build a Mel-spectrogram, a dot product is computed between a matrix of logarithmically spaced triangular filters, and the STFT magnitudes. The relationship in Eq. 2.20 is used to convert frequencies to mel bins, and an example of a resulting Mel-spectrogram is given in Fig. 2.13.

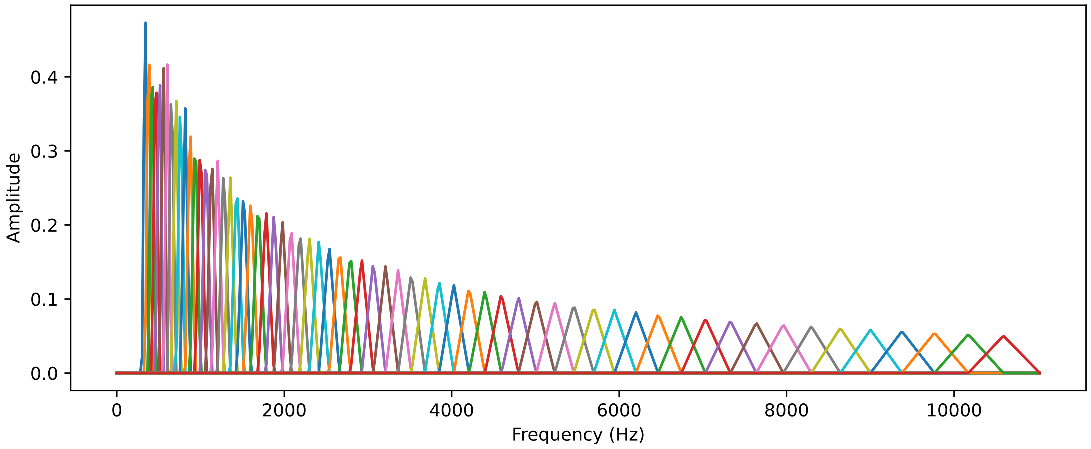


Figure 2.12: Example of Mel filterbank (set of triangular filters). Each colour represents a filter, which will ponderate the input spectrum to yield a Mel frequency bin.

$$f_{mel} = 1127 \times \log\left(\frac{f_{Hz}}{700} + 1\right). \tag{2.20}$$

Range compression

In acoustics, the energy is usually measured in decibels (dB), a logarithmic transformation of measured magnitudes such that $\mathbf{E} = 10\log_{10}(\mathbf{S})$. Typically, for spectrograms, this will shift the values from a logarithmic distribution to a Gaussian distribution. The strength of the shift can be modulated by applying a factor 10^a to the magnitudes before computing the logarithm ($log(1 + x \times 10^a)$, see Fig. 2.13). The optimisation of the exponent a can also be part of the learning framework, as proposed by Schlüter [169].

On the other hand, presumably more robust methods have emerged, especially with the Per-Channel Energy Normalisation (PCEN) [199]. This method introduces a dynamic gain control to adapt the compression range depending on local loudness and reduce stationary noise (estimated via an infinite impulse response (IIR) filter for each frequency bin, see Fig. 2.13). The formula for PCEN is given by Eq. 2.22, given an input spectrogram $\mathbf{E}$, and parameters ϵ, α, δ and r. $\mathbf{M}$ denotes the IIR filtered version of the spectrogram, as given by Eq. 2.21, depending on the smoothing coefficient s that impacts the filter's latency. This method requires

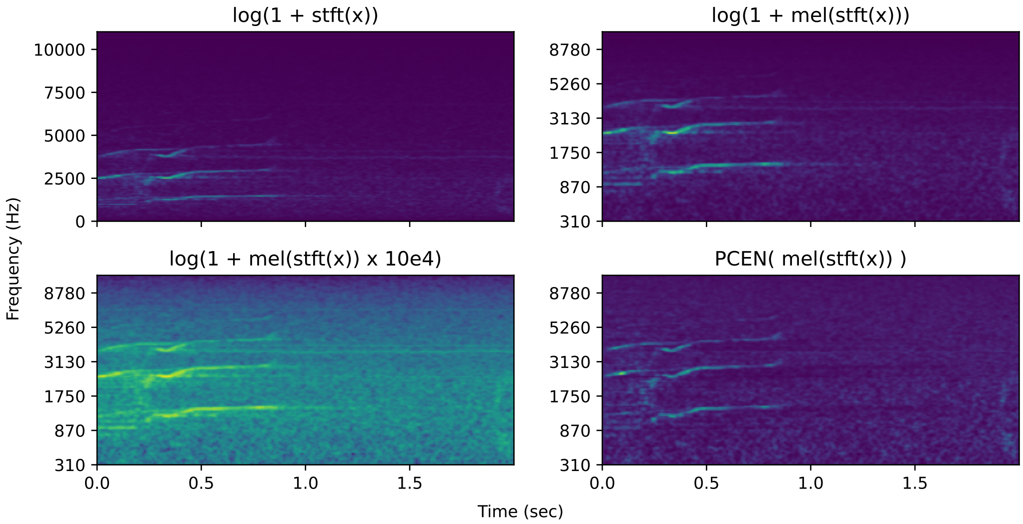


Figure 2.13: Comparison of frequency layouts (regular STFT versus Mel transformed) and compressions for an orca call spectrogram. Notice how the 4 kHz stationary noise gets removed with PCEN.

5 hyper-parameters to be set for initialisation and potentially to be optimised end-to-end with the downstream model.

$$\mathrm{M}_{f,t} = (1-s)\mathrm{M}_{f,t-1} + s\mathrm{S}_{f,t}, \tag{2.21}$$

$$\mathrm{PCEN}_{f,t} = \left(\frac{\mathrm{S}_{f,t}}{(\epsilon + \mathrm{M}_{f,t})^{\alpha}} + \delta\right)^{r} - \delta^{r}. \tag{2.22}$$

Learnable frontends

Probably more than in any field, in the machine learning community, researchers flee ad-hoc and hand-crafted approaches to rather choose fully learnable adaptive frameworks. This is applicable to the spectrogram computation, a major step in acoustic recognition. Several approaches have been proposed to learn custom spectrograms and break free from the STFT.

Some directly learn convolution kernels from scratch to be applied in the time domain [56, 141]. Others optimise known filters parameters, such as cardinal sinus [157], spline [9], gammatone [167], or gabor [212].

The latter, called Leaf, has outperformed SOTA in 8 different acoustic recognition tasks, but is still quite recent and remains very costly in computation (two orders of magnitude higher than a regular STFT). Thus, to this day the STFT and optionally its Mel transform remain the standard approach to feature extraction for acoustic recognition, despite their debatable anthropocentric nature and all the efforts invested in replacing them.

2.2.3 Data augmentations for acoustics

As previously mentioned for image classification tasks, data augmentation is a crucial regularisation method, especially when dealing with bioacoustics tasks with very few labels available [186]. This section presents known acoustic data augmentation methods, for the time domain and the spectro-temporal domain.

Addition of noise

Acoustic signals can simply be summed to be combined. A first augmentation technique thus comes down to adding randomly generated noise to the input sample in the time domain [133]. One can add white noise (flat spectrum), pink noise (spectrum following $1/f$) or brown noise (spectrum following $-20dB/decade$). The latter being the closest to underwater ambient noise, it is the most relevant to PAM of cetaceans.

Instead of synthesising random noise, one can also add soundscape recordings [109]. To some extent, this is equivalent to the MixUp approach aforementioned.

Whether it is synthesised or recorded in situ, a weight needs to be set when adding noise, defining its strength relatively to the input signal (the SNR). This value can be fixed for the whole training, or sampled randomly for each generation.

SpecAugment

Alike RandAugment for images [37], a suite of generic audio augmentation policies has been proposed for spectrograms: SpecAugment [143]. It includes time wise dilation or compression (via the interpolation of pixels values), as well as the masking of random time and frequency bands (see Fig.2.14).

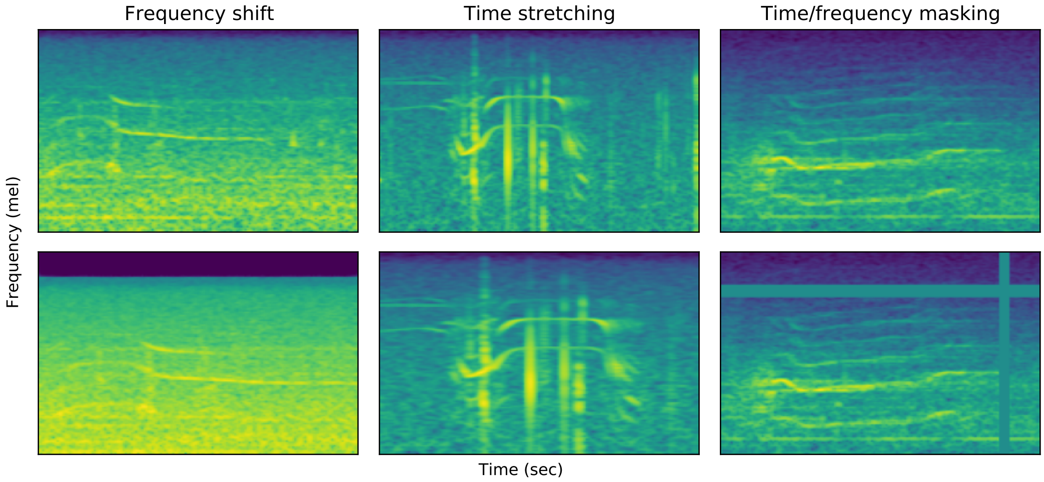


Figure 2.14: Demonstration of three common augmentation policies on orca calls recorded at OrcaLab (top: original sample, bottom: augmented version).

The authors did not include frequency stretching in their SpecAugment suite, perhaps because it was not appropriate for their task at hand, or since it is more common to operate pitch shifts on waveforms [118] rather than on spectrograms [88].

SpecAugment has shown SOTA results in several acoustic recognition tasks [143], with the drawback that it potentially converts the overfitting problem to an underfitting problem. To cope with this, the authors propose larger networks and longer training schedules.

Temporal / frequency shifts

Temporal and frequency shifts of spectrograms seem like a straightforward efficient way of augmenting the data, by simply displacing patterns to be recognised (in realistic ranges). As for the time shifts, as previously mentioned, CNNs offer spatial invariance, making such data augmentation non significant. However, pitch modulation potentially implies more than just a vertical shift for the resulting spectrogram (see Fig.2.14). By simply speeding up or slowing down the input sample (via resampling the waveform), the spectrogram is shifted frequency wise but also stretched time wise. Pitch shifting has thus proven to be a relevant data augmentation approach [118].

2.2.4 Applications to bioacoustics

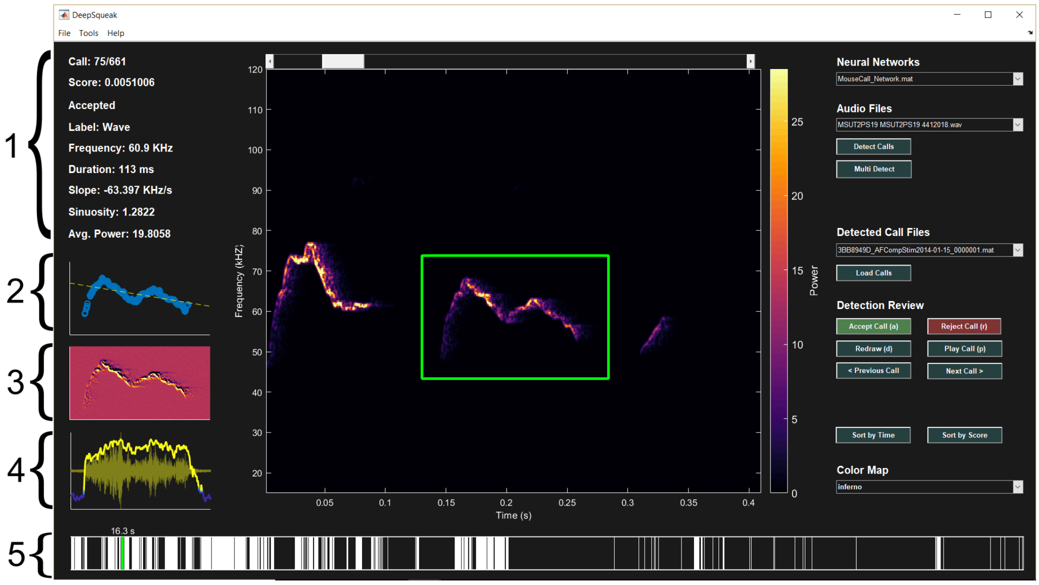
Now that CNNs have been introduced in their original field of computer vision, along with their use in acoustics, I will finally present how the bioacoustics community has made it its own, trying to overcome the domain specific challenges that come along.

Annotation optimisation

The amount of training data is crucial to a robust deep learning model. Besides, despite efforts to develop reliable unsupervised algorithms, the performance they offer is still not sufficient for relevant use in PAM contexts without human intervention. Therefore, large amounts of labelled examples are still needed prior to developing automated detection systems.

The usual annotation scenario starts with the access to a bank of audio signals. When recorded by autonomous antennas, this typically means weeks or months of recordings, with no other prior information than the presumed presence of some species' vocalisations. Listening to the whole recordings would be too tedious and is therefore not viable. To efficiently browse through recordings and potentially annotate certain sections, several approaches are found in the literature :

- Long Term Spectral Average (LTSA) enables a quick glimpse at frequency distributions of several hours of data at a time [177],
- Running high recall hand crafted detection mechanisms allows for a first extraction of potential signals of interests (pre-detections) [67, 49],
- Hand crafted filtering rules can sort out known false positives among pre-detections [204],
- Clustering pre-detections via hand crafted features can group similar acoustic events together [67],
- Dedicated interfaces can improve the efficiency of visualisation and annotation of pre-detections and clusters [185, 34] (see Fig. 2.15).



Call: 75/661
Score: 0.0051006
Accepted
Label: Wave
Frequency: 60.9 KHz
Duration: 113 ms
Slope: -63.397 KHz/s
Sinuosity: 1.2822
Avg. Power: 19.8058

Neural Networks: MouseCall_Network.mat
Audio Files: MSUT2PS19 MSUT2PS19 4412018.wav
Detected Call Files: 3BB8949D_AFCompStim2014-01-15_0000001.mat
Color Map: inferno

Figure 2.15: Example of an advanced annotation interface for mice vocalisations: DeepSqueak [34]. (1) call statistics, (2) extracted contour, (3) spectral gradient of spectrogram, (4) tonality and sound wave, (5) position in file.

Using one or several techniques mentioned above should suffice in yielding several dozen positive and negative annotations, enough to start learning small CNN models. To further increase the system's performance, which is often not yet robust enough when trained with few samples, active learning is the usual adopted solution. Active learning consists in an iteration of three steps: training the model, running it on unlabelled data, and validating or invalidating the model's predictions via human intervention. Looking for false positives with strong confidence (hard negative mining) and vice versa will enhance the process by focusing on samples that confuse the model [173].

The yielded annotations will found the basis of knowledge for our model, and will further serve performance measurements. It is therefore crucial that no labeling errors slip into our database, or it will negatively impact all following procedures. A first pitfall is in the gathering of the initial database for active learning. Depending on their frequency and source level, certain vocalisation might be missed by the chosen algorithm or annotation procedure. If so, the model will never learn to detect them. A second source of bias can be human annotators, for which Duc et al. [46] has demonstrated the potential subjectivity. To mitigate this effect, we can cross-validate labels with several experts, and propose the 'unsure' label during annotation.

Deep representation learning for bioacoustic signals Clustering similar signals into groups can drastically reduce the annotation effort. For this purpose, similarity can be measured using hand-crafted features such as MFCC [38], or features learnt via SSL for instance. Several papers explored this approach such as Tolkova et al. [195] for birdsong annotation using an AE framework, Goffinet et al. [76] similarly with a variational AE [102], or Jahangirnezhad and Mashhadi [91] combining an AE reconstruction loss with the Deep Embedded Clustering (DEC) loss [208].

The embedding space learnt via deep representation learning can not only enable clustering for efficient annotation, but also serve classification models directly. Indeed, either via a semi-supervised loss, or via network pre-training, the performance of classifiers can be enhanced despite a reduced quantity labels when fine tuning from relevant embedding spaces. In the case of using embeddings for annotation via clustering, attention should be paid to the potential biases induced (some classes might be favored by the similarity metric employed).

Transfer learning The method of using weights optimised on a third party task to initialise a model (pre-training or transfer learning) has indeed proven its effectiveness, especially when dealing with small datasets. The third party task can be for instance a SSL paradigm such as training an AE on data similar to that of the target task [13, 190]. On the other hand, it can also be a a totally unrelated task. Indeed, fine-tuning from models trained on AudioSet [85] or even ImageNet [43] was shown to be relevant for bioacoustics event detection [10, 216, 190]. The assumption here is that early feature extraction are quite generic, and that knowledge gained from very large datasets are useful for other tasks.

Available databases

A common practice in the computer science community is to publish databases for researchers to try their automatic systems on. They enable shared performance metrics, essential to the objective comparison of models. An important element is to be taken into account regarding the types of available annotations: some denote the presence of events in a large window of recordings (up to several minutes), they are called weak labels. On the other hand, strong labels give exact time positions associated with target events. I will hereby present some of the available databases for the detection and classification of cetacean vocalisations.

- The Watkins marine mammals sound database [168] proposes excerpts for numerous marine mammal species from different recorders. It was used by Lu et al. [119] with an AlexNet architecture (pretrained on ImageNet), and by Murphy [132] with a ResNet architecture and multi-channel spectrograms. One limitation of this database is that most of the recording devices and locations are species specific, which hinders good generalisation measures.

- The Orchive database [134] presents annotation of Northern Resident killer whales (NRKW) calls. It includes calls with their class label (call type, see section 3.1.3) or just as positives, along with negative samples (boats and other noises); all recorded at the OrcaLab laboratory. It was used by Bergler et al. [14] with a ResNet architecture for call detection and unit classification, and by Vargas [197] for classification using SVMs.

- The Detection Classification Localisation and Density Estimation of marine mammals (DCLDE) workshops have published numerous datasets with different target species and labeling (some of them offer only weak labels). It was used by Shiu et al. [173] for Northern Right Whale (NRW) upcall detection using LeNet and BirdNet architectures.

- The DOCC10 database [56] is an extension of the DCLDE 2018 dataset that used an automated algorithm to extract strong labels from the available weak labeling. Samples include clicks from 10 odontocete species. It was used in the same study to train an end to end deep classifier of a custom architecture.

- The acoustic trends blue fin library [128] offers almost 2,000 hours of recordings from the Southern Ocean, annotated by a consortium of experts. Several thousands of samples are available for each of the 7 call types from 2 mysticete species: the blue whale and the fin whale. By covering several recorders, locations, environmental conditions and years, this database offers an opportunity to robustly measure models' generalisation performances.

Deep classifiers for bioacoustics

Since the introduction of CNNs in bioacoustics a few years back, numerous experiments were published on the topic, either with public or private databases. Most of them report their experiment with a standard CNN architecture on some database, like a ResNet for orca vocalisation detection for instance [14]. Some also report empirical studies of varying parameters such as data augmentation, frontend or architecture [173, 4].

Other architectures than regular CNNs are encountered, such as a Recurent Neural Networks (RNN)+CNN that integrates the prior of call rates into the detection process [120], an LSTM on spectrogram for click detection [45], an ANN that classifies odontocetes' clicks without convolution [162], Siamese networks for classifying blue whale calls [217], transformers for bird recognition [156], or a context adaptive CNN that makes use of soundscape features to gain robustness [118].

Stowell [186] proposes a review gathering 159 articles on bioacoustics using deep learning, 30 of which concern marine mammals. One important insight of this review is a report on chosen CNN architectures. The most popular mentioned are Resnet (23 papers), and VGG or VGGish (17 papers). Other tendencies are described, but besides perhaps the use of spectrograms as inputs for CNNs, no clear advantage emerges for a specific architecture or set of hyper-parameters.

Such reviews demonstrate how the automatic analysis of bioacoustic recordings is still an open research subject. In this context, Brown et al. [24] explored a wide range of settings evaluated on multiple bird recognition tasks. They show how there is not one generic workflow that is well adapted to every task. In a similar paradigm, this thesis explores several methods trying to extract reusable knowledge on their potential efficiency.

2.3 Conclusion

Before diving into this thesis' contributions, let us take a step back and get an overview of the challenges and opportunities that come along the research problematic.

Challenges

ANNs, despite having already some implementation in industrial systems, is still an open research topic. This is even more true when it comes to its application to PAM. Indeed, PAM brings specific problems uncommon to other domains of application of ANNs, the main ones probably being the lack of annotations and the scarcity of events to detect. As mentioned previously, training ANNs demands large quantities of labels, which are costly to produce in terms of human effort. When implementing image classification or speech recognition systems, one can make use of large databases already available for these quite popular tasks. High quality databases of cetacean vocalizations (large amount of annotations with precise timestamps) are more rare. To cope with this, I will show in this thesis how annotation processes can be optimized to reduce human effort.

Another challenge comes from the underwater conditions that highly impact acoustic properties of signals. Since few researchers apply ANNs underwater, they have to find their own way to cope with these conditions yet relatively unexplored in the literature. Moreover, detection systems are most useful when reusable across acoustic stations. This demands highly robust models, taking into account the variability in potential noise exposition (e.g. depending on depth, boat traffic, bathymetry).

Also, we will later discuss the need for PAM systems to be embedded into field stations (section 6.2). This demands efforts in reducing the computational needs, as well as building trustworthy algorithms, which can be challenging when having relatively low control on ANNs' behavior (ANNs are often described as 'black boxes' since their functioning is hardly interpretable).

Another important challenge faced during this thesis was to sort out relevant methods to explore among the wide variety of propositions. Indeed, ANNs being a highly popular topic, dozens of different approaches are still being explored, with few consensuses on universally reliable techniques.

Eventually, despite numerous occurrences of trained ANNs for marine bioacoustics in the literature, very few are found to be put in production yet [4] (using the prediction to yield biological analysis). It is thus an ambitious objective to finally bridge that gap between training experiments and production use of deep learning models.

Opportunities

Even with few annotations available, large amounts of data can still be useful when training ANNs, as section 5.5 discusses with unsupervised approaches. The democratisation in autonomous recording units (ARUs) has already yielded Terabytes of data which, even when containing only few signals of interest, can come handy to train ANNs because of the data diversity they provide. Indeed, these long recordings often demonstrate a wide variety of noises (e.g. from boat engines, sonars, reef activity, waves, currents, or earthquakes). Moreover, data variability can also arise from differences in recorders' frequency responses, and/or placements regarding the bathymetry.

This is a major challenge when building handcrafted algorithms, having to compensate for each potential acoustic disturbance independently. However, ANNs represent a great opportunity in that sense since they have the potential of learning robust representations that can be resilient to the most diverse perturbations. Section 4.6 discusses how one can make use of the variations in the available data to rigorously measure a model's generalisation performances, and/or use it to train projections of sound that are most stable against such noise diversity.

Eventually, PAM strongly benefits from such robust systems, since they help reduce the minimum SNR for detection as well as the amount of false alarms. A first axis of benefits offered by this characteristic is that it facilitates the use of such systems in real-time applications, with implication for species conservation via ship strike mitigation for instance.

A second axis of benefits is the yielded reliable statistics for large scale analysis that would not have been feasible otherwise. This enables learning on presence patterns, song structure evolution and to characterise communication systems for instance (as demonstrated in this thesis' chapter 7). Indeed, long term surveys represent a good opportunity to yield biological insights, especially in blind datasets (audio recordings with no complementary data such as behaviour) and in uncontrolled settings. Finally, these uncontrolled settings ensure that no behavioural bias is induced, conversely to many laboratory experiments.

3
Material

Contents

In this chapter will be introduced the material used for the experiments conducted throughout this thesis. It takes form as underwater acoustic data, containing several types of signals, and recorded at different places and times. This chapter will thus revolve around two axis: the studied signals, and the recording setups.

3.1 Target species and signals

The diverse set of target signals described here are those for which detection and classification systems were built. Their characteristics are summarised in Table 3.1, and each subsection then underpins the current knowledge about them, especially regarding their context of emission.

Note that for the signal types of Tab. 3.1, and throughout this thesis, stationary refers to "signals locally stable in frequency" (calls, whistles) as opposed to transitory signals (clicks).

Species	**Sperm whale**	**Fin whale**	**Orca**	**Dolphin**	**Humpback whale**
Sub-order	Odontoceti	Mysticeti	Odontoceti	Odontoceti	Mysticeti
Signal	clicks	20 Hz pulses	pulsed calls	whistles	calls
Signal type	Transitory	Transitory	Stationary	Stationary	Stationary
Frequency (Hz)	12,500	20	[500; 5,000]	[5,000. 20,000]	[300; 3,000]
Bandwidth	20 kHz	2 Hz	100 Hz	20 Hz	50 Hz
Duration (sec)	0.001	1	[0.5; 2]	[1; 2]	[0.5; 1]

Table 3.1: Summary of the target signals for the detection systems built throughout this thesis. For transitory waves, the frequency denotes the approximate centroid frequency, for stationary signals it denotes its range

3.1.1 Fin whale (*Balaenoptera Physalus*) 20Hz pulses

As the second largest animal on earth, the fin whale produces very low-pitched vocalisations, barely noticeable to the human ear. So far, bioacousticians have documented 3 main types of signals emitted by fin whales: 100-30 Hz down-sweeps, 30 Hz rumbles, and 20 Hz pulses. They supposedly serve group cohesion [149, 200], food signaling [166], and mate attraction [201, 36].

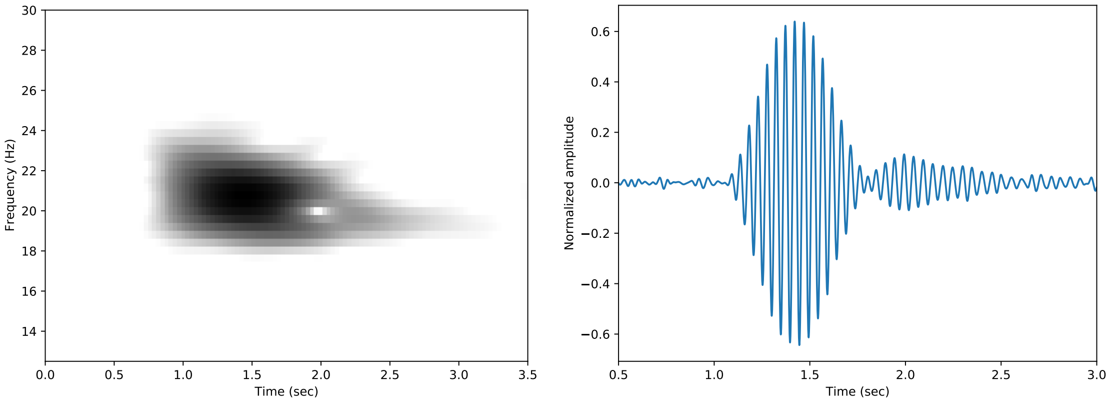


Figure 3.1: Spectrogram (left) and waveform (right) of a fin whale pulse recorded by Bombyx (the two figures share the same abscissa). STFT parameters are: $fs = 100Hz$, $NFFT = 128$, $padding = 50\%$, $hopsize = 3$.

In this thesis, I will focus on the most common signal: the 20 Hz pulse. It is often further classified into two sub categories, named A and B, or classic pulse and

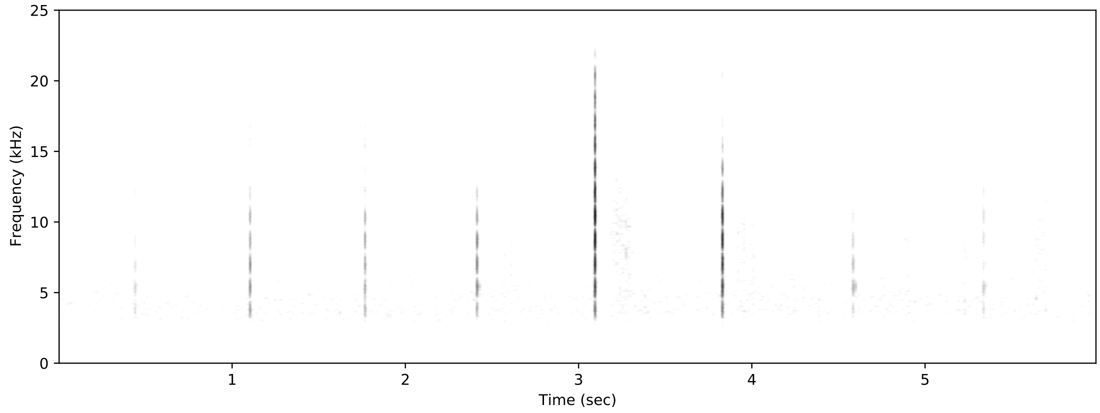


Figure 3.2: Sequence of sperm whale echolocation clicks recorded by Bombyx in july 2018. STFT parameters are: $fs = 50kHz$, $NFFT = 1,024$, $hopsize = 896$, $padding = 0\%$.

back-beat [171]. They highly resembles a Gabor wavelet: a sine wave enveloped by a Gaussian (see Fig.3.1), and can be emitted either as single pulses, or in patterned sequences, termed as songs [174]. The pulses and the sequences they take part in are highly stereotyped: pulses show very low variability both in frequency and duration, and when in sequences, the Inter Note Interval (INI) remains highly stable.

Fin whale song characteristics, especially the INI, are population specific [42, 29]. They also are subject to seasonal cyclic variations [138, 130], as well as long-term trends [204, 83] (e.g. linear increase of the INI through years).

3.1.2 Sperm whale (*Physeter Macrocephalus*) clicks

Sperm whales produce echolocation clicks to navigate and locate preys during hunts. Their large head contains a series of oil sacks surrounded by sound-reflecting air sacs that allows for the amplification of the impulses [135], making it the most powerful sonar in the animal kingdom [129] (the loudest recorded click was at 230 dB re: 1μPa rms).

Echolocation clicks usually come in sequences (see Fig.3.2), with the Inter CLick Interval (ICI) ranging between 0.01 and 1 sec, usually decreasing when approaching a prey [53]. The clicks lie around relatively low frequencies compared to other smaller odontocetes (between 3 kHz and 30 kHz).

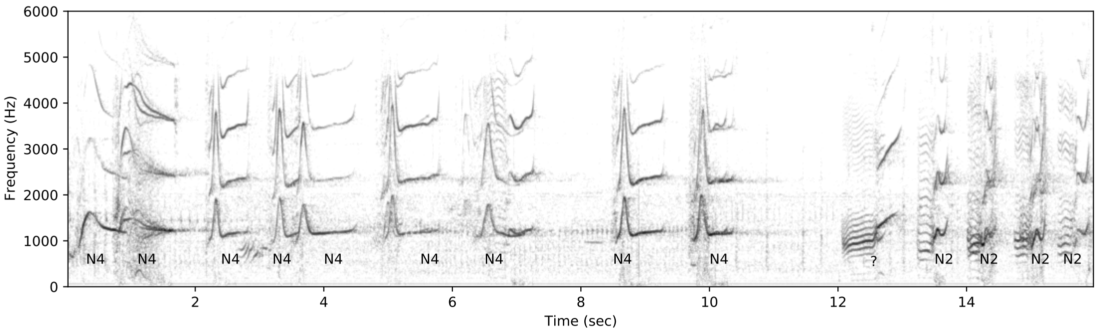


Figure 3.3: Sequence of orca tonal calls recorded at OrcaLab in September 2016. STFT parameters are $fs = 22050Hz$, $NFFT = 1024$, $hopsize = 50$, $padding = 0\%$. The N.. labels denote each call type, with a '?' showing an ambiguous one.

3.1.3 Orca (*Orcinus Orca*) calls

Orcas produce three types of signals: clicks, pulsed calls, and whistles [66]. As for most dolphin species, clicks presumably serve echolocation, while the two other more stationary signals would rather be used for communication. Pulsed calls are highly harmonic, typically lying between 500 Hz and 5 kHz, and lasting up to 1.5 seconds (Fig. 3.3). On the other hand, whistles show little or no harmonic structure, lay between 6 kHz and 12 kHz, and can last up to 12 seconds. In this thesis, I will focus on the pulsed calls, referring to them as calls or vocalisations.

As shown in Fig. 3.3, some orca calls have stereotyped frequency contours that have been classified into discrete types. These were proven to be community specific (dialectic) [65], and subject to cultural evolution [41, 59]. The identification of call types strongly contributed to the study of the orca's social structures, and its categorisation is widely accepted by the scientific community. Difficulties remain however, for some calls to be attributed to one class or another, especially for non experts. Indeed, despite calls being stereotyped, they still are prone to variability which might lead to overlap between classes' characteristics [66].

3.1.4 Humpback whale (*Megaptera Novaeangliae*) calls

The humpback whale song is among the most widely studied cetacean acoustic signals. These sequences are mostly emitted by males during the reproductive

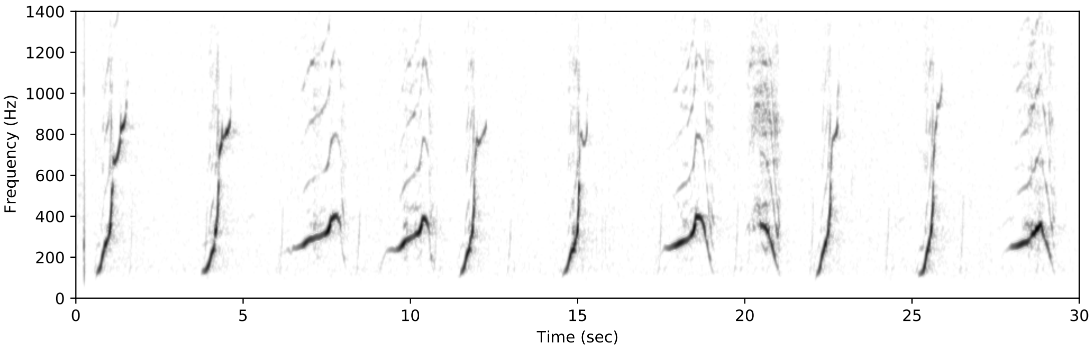


Figure 3.4: Extract of a humpback whale song from the Carimam dataset. STFT parameters are $fs = 22050Hz$, $NFFT = 4096$, $hopsize = 48$, $padding = 50\%$.

season, presumably playing a role in courtship [84] (male-female and/or male-male interaction). They follow strict hierarchical structures: series of units form phrases that are arranged into themes, themselves combined in songs that can last several hours [148].

Each component of the hierarchical structure of the humpback whale songs are stereotyped, as seen in Fig. 3.4 with a sequence of stereotyped units. Moreover, song structures are shared by individuals at a given place and time, with cultural implications for their spatio-temporal evolution [205].

3.1.5 Dolphin (*Delphinidae*) whistles

Exceptionally for this type of signal, we do not target a single species, but rather a family of species, the *Delphinidae* which includes sub-families such as *Globicephalinae*, *Delphininae*, and *Orcininae*. They all produce whistles, which are typically high pitched, tonal, and narrow-band. Their frequency contour can be stereotyped [198], individual specific [27], and serve group cohesion [93].

3.2 Data at hand

In order to experiment on detection and classification mechanisms for the target species and signals aforementioned, datasets are needed. Through this thesis, work has been conducted on both recordings from local projects and publicly available ones. They involve a variety of recorders, locations and time spans which are

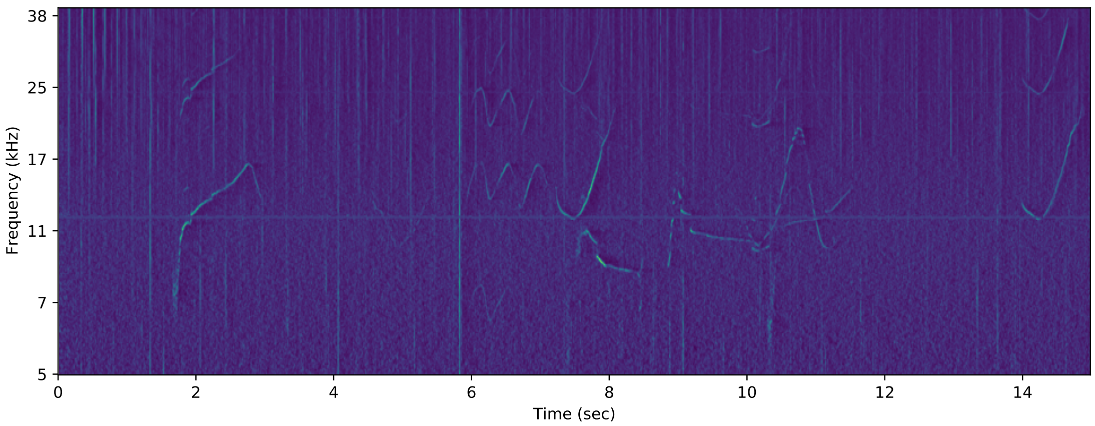


Figure 3.5: Sequence of dolphin whistles from the Carimam dataset. STFT parameters are $fs = 256kHz$, $NFFT = 8192$, $hopsize = 512$, $padding = 0\%$, Mel transformed from 5 to 40 kHz, and PCEN normalised. The stationary signal around 12 kHz is the remaining self noise of the sound card used [11] (despite heavy mitigation via the PCEN).

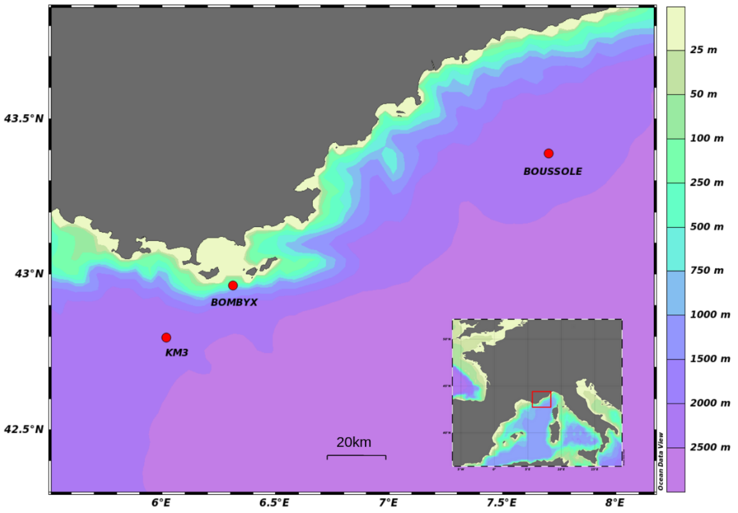


Figure 3.6: Map of the 3 Mediterranean antennas used throughout this thesis.

described in this section, starting with the local projects of the DYNI team (Toulon University), and followed by the public Blue and Fin whale acoustic trends dataset.

3.2.1 Data from DYNI

Table 3.2 summarises some of the data yielded by the partnerships and projects that H. Glotin co-set up at Toulon University. They are stored locally in a Network-Attached Storage (NAS) system, funded by the DYNI team projects and maintained by the LIS laboratory. Each are briefly introduced in the following sections.

Data source	**Boussole** [107]	**Bombyx** [74]	**OrcaLab** [183]	**Carimam** [75]	**KM3Net** [1]
Location	Côte d'Azur	Côte d'Azur	British Columbia	Caribbean	Côte d'Azur
Depth (m)	10 - 25	25	0 - 20	5 - 20	2,440
Recording year	2008-2009	2015-2018	2015-2021	2021-today	2020-2021
Sampling rate (Hz)	32,000	50,000	22,050 - 44,100	384,000	195,312
ON/OFF protocol (min)	5/10	1/5 - 5/15	Continuous	1/5	Continuous
Channels	1	2	5	15	3
Recorded time (hours)	1,752	3,533	$\approx$ 40,000	5,677	514

Table 3.2: Summary of the recording characteristics for each data source available at Toulon University.

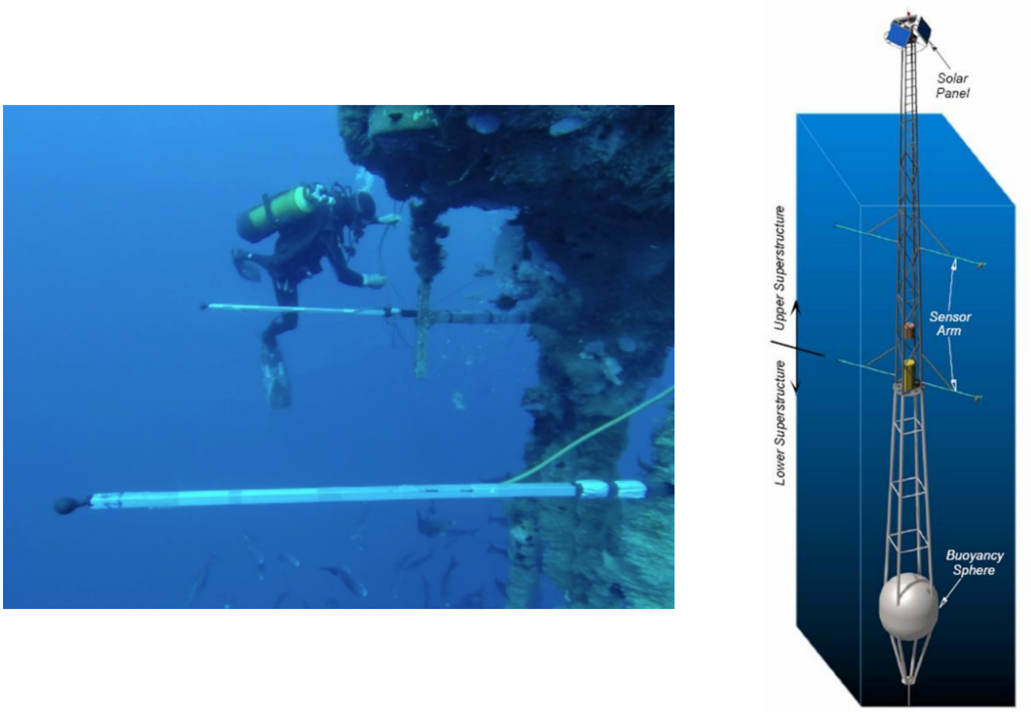


Figure 3.7: (left) Installation of the Bombyx stereophonic antenna [74]. (right) Structure of the Boussole antenna [107]

Boussole

This project consisted in a partnership between GIS3M, Pelagos marine mammal sanctuary, and Port-Cros National Park. In order to study marine mammals acoustic activity, a monophonic recording system was placed on the Boussole buoy. Originally dedicated to marine optics, this buoy designed to be transparent to swell was moored on the 2,440 meters deep sea floor, off the coast of Nice (France). During 4 phases between October 2008 and September 2009, the system recorded at 32 kHz, enabling the detection of vocalisations from sperm whales, fin whales, and delphinids of the area (*Stenella coeruleoalba, Globicephala melas, Grampus griseus, Tursiops truncatus and Delphinus delphis*).

A study prior to this thesis intended to monitor the acoustic presence of sperm whales and fin whales in the yielded recordings. Sperm whale clicks were successfully detected automatically but the processing of fin whale 20 Hz pulses was hindered by the self noise of the system [107] (Fig. 3.8).

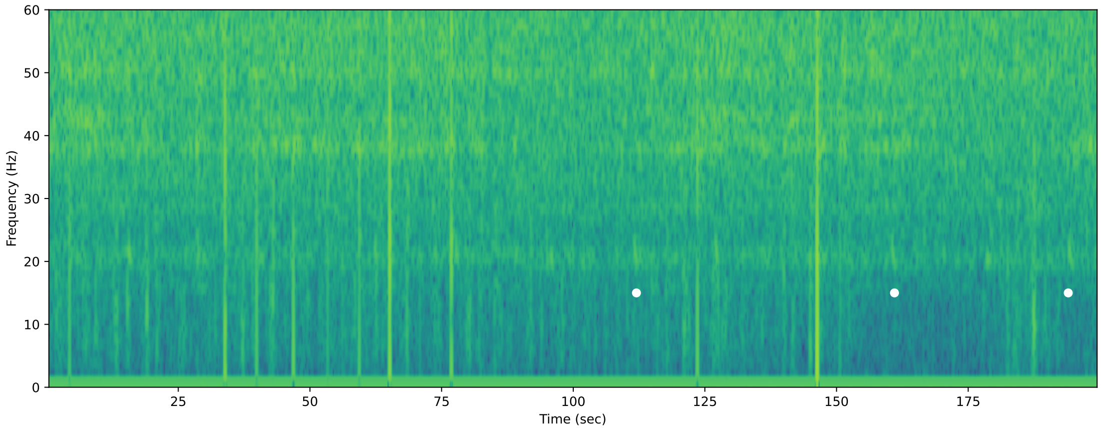


Figure 3.8: Spectrogram of a noisy recording from the Boussole antenna ($f_s = 32kHz$, $NFFT = 32768$, $hop = 5568$). White dots denote the temporal position of some confirmed fin whale 20 Hz pulses.

Bombyx

The Bombyx antenna was set up by a partnership between Toulon University, Port-Cros National Park, TVT Innovation, and the Pelagos marine mammal sanctuary. Being placed right on the rift of a 2000 meters deep canyon, it intends to allow the monitoring of sperm whales of the area [74]. It did so during several phases spread across 4 years (2015 to 2018). The area is of interest because of the nearby canyons prone to sperm whale hunts [60], but also because of the ferries that travel across on a daily basis. In addition to the noise that the latter generate, Bombyx recordings are also subject to self noise (Fig. 3.9).

OrcaLab

Paul Spong founded OrcaLab in the 1970s [183], an in-situ observatory in the Johnstone Strait (British Columbia, Fig. 3.10). It serves the visual and acoustic monitoring of orcas, especially the population that feeds on the local salmon every summer, the NRKW. From 2015 to 2020, the 5 hydrophones' signals have been recorded continuously (at 22,050 Hz until march 2018, then at 44,100 Hz).

The fact that the orcas regularly come to this relatively confined space represents an unique opportunity to observe and listen to them 24/7 from the shore. Most

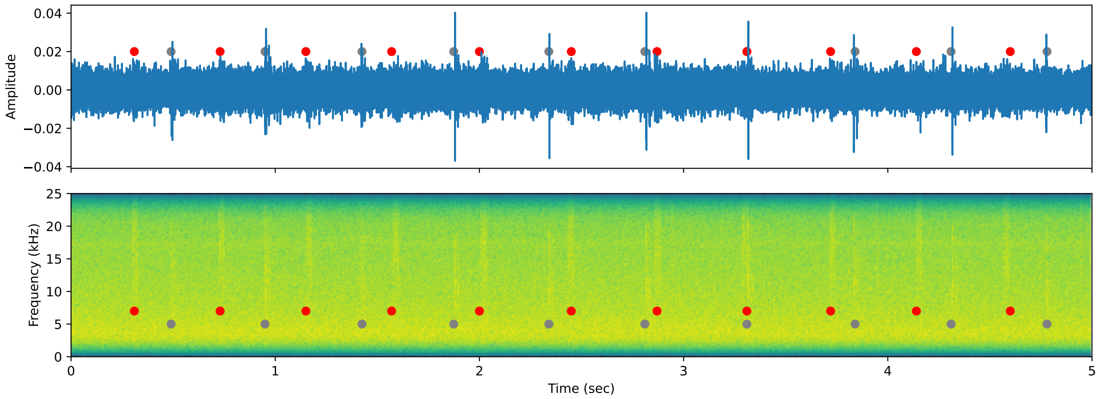


Figure 3.9: Example of signal from the Bombyx antenna (high pass filtered, order 3 butterworth at 3 kHz). Grey dots denote sperm whale clicks, and red ones self noise from the recording device. (top) Waveform. (bottom) Spectrogram ($f_s = 5kHz$, $NFFT = 512$, $hop = 256$).

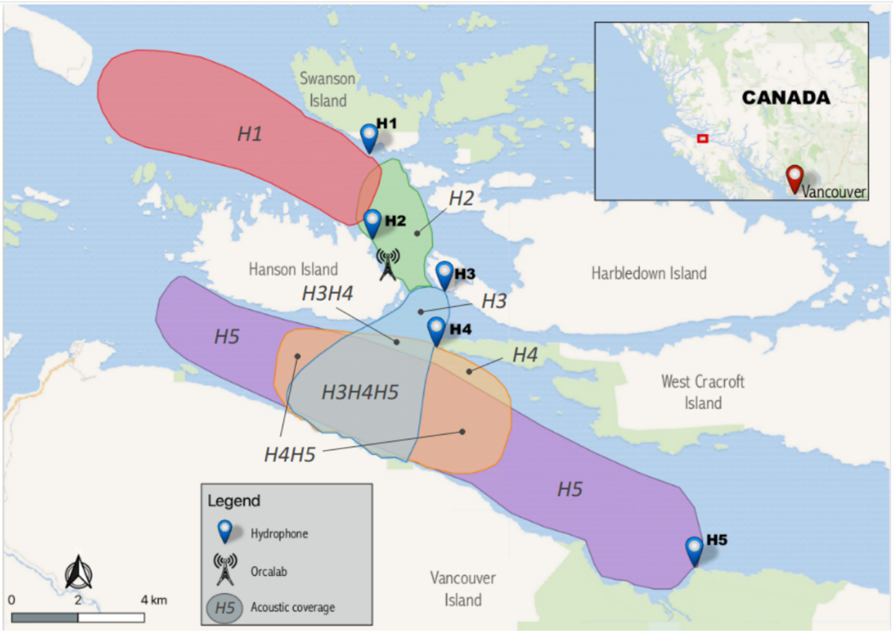


Figure 3.10: Map of the OrcaLab observatory, with its 5 hydrophones and associated acoustic range.

importantly, it guarantees no behavioural disturbance and continuous power and data storage supply, the main constraints of most PAM approaches.

KM3Net

The ORCA detector of the KM3Net observatory is an array of detection units allowing the measurement of neutrino particles [1]. It was installed on the seabed

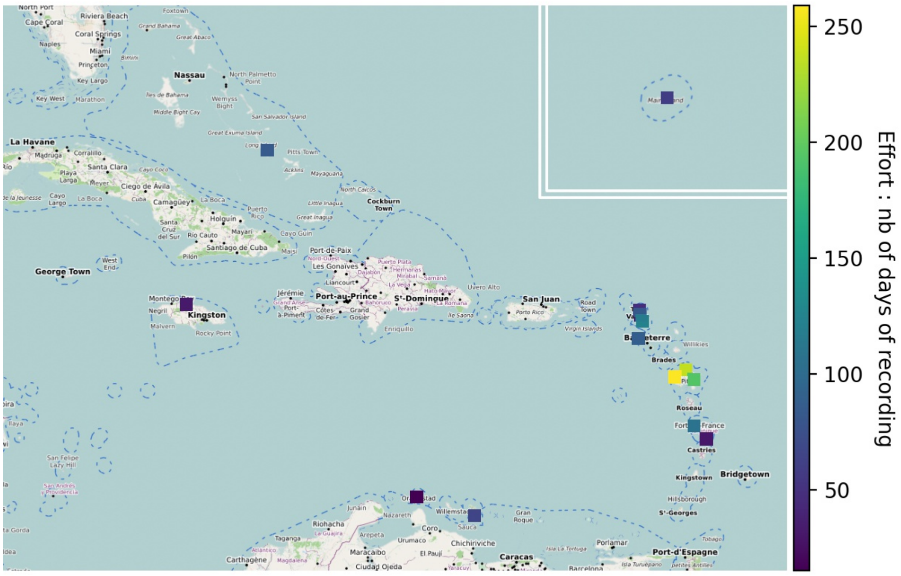


Figure 3.11: Map of recording stations with their recording effort for the Carimam project, in the Caribbean archipelago.

2,440 meters deep, connected to the shore of Toulon (France) via fiber cable. Hydrophones are used as part of a positioning system, but as a by-product, also serve the PAM of local cetaceans.

Carimam

The Carimam project, led by a consortium composed of AGOA, the OFB and Toulon University, is a network of 16 monophonic acoustic stations spread through the Caribbean archipelagos. It aims at monitoring the rich marine mammal activity of the area. To manage such a wide number of stations, low-cost and easy to install recording devices [11] were sent to local environmental managers, who set them up on existing mooring lines close to the shore.

Spatialisation

In Table 3.2, number of channels of each recording system are given. When synchronised and with overlapping acoustic coverage, multi-channel data can serve the spatialisation of acoustic sources. This is done via the computation of TDOAs for signals to be triangulated. For Bombyx, since two hydrophones record 1.8 meters apart (on the same horizontal plane, see Fig. 3.7), the two possible azymuths of acoustic sources can be computed. For Carimam, the stations' acoustic coverage do not overlap: the spatial precision is the acoustic range of the antennas. For KM3Net, the 3 hydrophones are approximately 30 meters apart. With prior knowledge on the depth of a source, its coordinates could be estimated (see section 8.2). Finally, for the OrcaLab network, hydrophones are several kilometers apart, but sent to a centralised Digital Analog Converter (DAC) via radio waves, which makes them temporally synchronised. Therefore, spatialisation could be performed in the zones of acoustic overlap.

3.2.2 Blue and Fin whale acoustic trends dataset

In early 2021, a large acoustic dataset of antarctic mysticetes was made publicly available [128]. It was built by a working group from the Southern Ocean Observing System (SOOS) titled Acoustic Trends of Antarctic blue and fin whales (Acoustic Trends Working Group; ATWG). The following is an extract from their terms of reference [182]:

SOOS Capability Working Group Key Objective(s): *Continue to develop and mature a long term acoustic research program to understand trends in Southern Ocean blue and fin whale distribution, seasonal presence, and population growth through the use of passive acoustic monitoring techniques. Implementation of these objectives will occur via:*

1. *analysis and interpretation of existing ad-hoc acoustic datasets in from the Southern Ocean,*
2. *the development and implementation of an ongoing network of long-term circumpolar underwater listening stations, and*
3. *development of novel and efficient methods for standardised analysis of acoustic data collected in the Antarctic and sub-Antarctic*

It is regarding this third axis of work that the Acoustic Trends dataset was built and published, especially to share performance metrics for detection systems. It

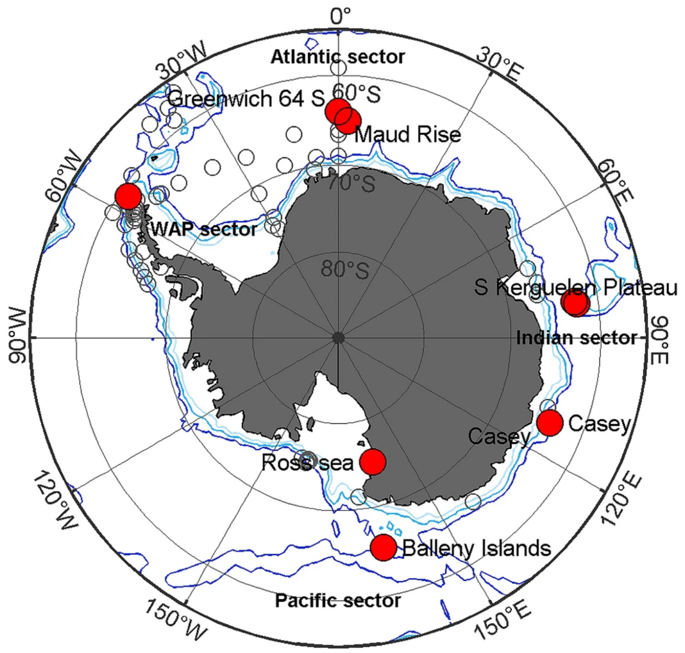


Figure 3.12: Map of the recording stations used in the Acoustic Trends dataset. The map was published by Miller et al. [128].

gathers annotations from a group of experts, on data yielded by several recorders at different locations from 2005 to 2017 (see Fig. 3.12 and Tab. 3.3).

Target signals are of 7 classes, 4 vocalisation types from blue whales (*Balaenoptera Musculus (Bm)*) and 3 vocalisation types from fin whales (*Balaenoptera Physalus (Bp)*). They all lie in low frequencies (between 20 Hz and 100 Hz) lasting from 1 to 15 seconds (Fig. 3.13). Since this data has not been subject to custom annotations, I won't expand on the target signals which are described in the dataset publication [128].

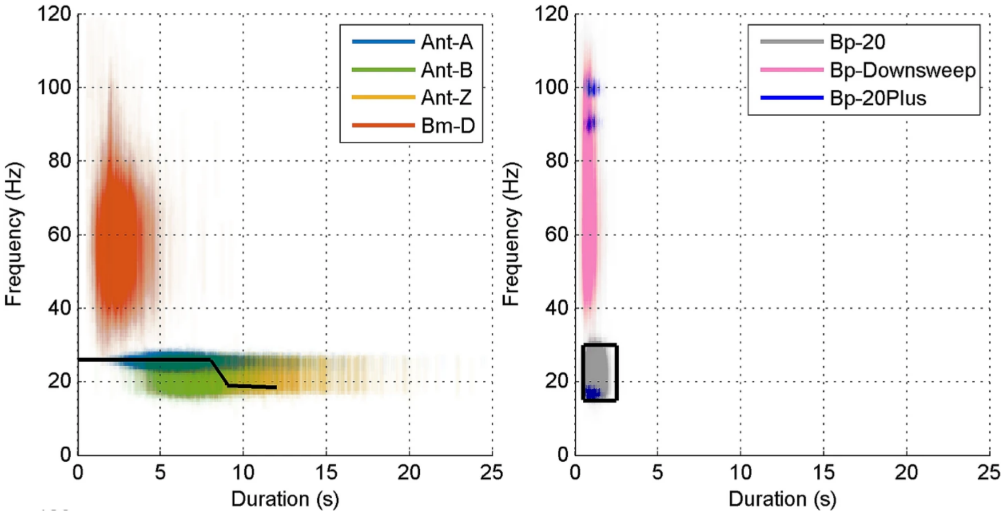


Figure 3.13: Distributions of lengths and frequencies for each of the 7 call types of the Acoustic Trends dataset. (left) *Balaenoptera Musculus*, (right) *Balaenoptera Physalus*. The figure was taken from [128].

Location	Year	Instrument	Recordings (hours)
Balleny Islands	2015	PMEL-AUH	204
Elephant Island	2013	AURAL	707
Elephant Island	2014	AURAL	216
Greenwich 64S	2015	Sono.Vault	32
MaudRise	2014	AURAL	80
Ross Sea	2014	PMEL-AUH	184
Casey	2014	AAD-MAR	194
Casey	2017	AAD-MAR	187
Kerguelen 1	2005	ARP	200
Kerguelen 2	2014	AAD-MAR	200
Kerguelen 2	2015	AAD-MAR	200

Table 3.3: Summary of recorders' characteristics and amounts of data available in the Acoustic Trends dataset.

4
Optimising annotation processes

Contents

4.1 Context and objective

Given the large amount of available recordings presented in the previous section, the objective of this thesis is to build robust detection and classification mechanisms for the vocalisations of species of interest. For this purpose and with the chosen approach

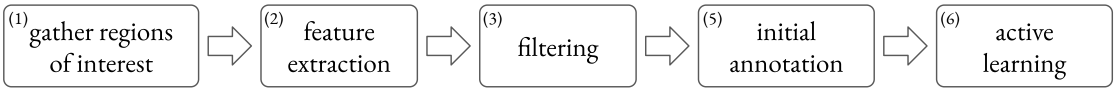


Figure 4.1: Flow chart of procedures employed in the annotation processes.

of ANNs, annotated databases are needed. In the following chapter, procedures and User Interfaces (UIs) suited for bioacoustic use cases are proposed, with an objective of optimising annotation quantity while minimising human effort. For all tasks, the annotation procedure can be summarised in 5 steps that are introduced Fig. 4.1.

This chapter starts by introducing a versatile and efficient approach to annotation (thumbnail picking), which will be needed in the subsequent experiments. Then, algorithms and UIs are proposed for several use cases, each being adapted to specific constraints:

- To detect stationary signals (orca calls) and given some samples to tune a handcrafted algorithm, a spectrogram binarisation approach is described (section 4.3.1).
- Looking for transitory signals (sperm whale clicks) in stereophonic recordings, I propose an interface to visualise and annotate TDOAs tracks (section 4.3.2).
- For a case when no target signals are available a priori, a generic extraction of spectral distributions is used to cluster similar acoustic events (section 4.4.1).
- In contrast, when a large quantity of signals of interest are available, an AE demonstrated the ability to learn relevant features to measure similarity and enhance annotation efficiency (section 4.4.3).

Finally, after these methods were employed to gather an initial set of annotations, active learning was conducted until a satisfying amount of labels are available (section 4.5). They are presented with their chosen train / test split in the last section of this chapter.

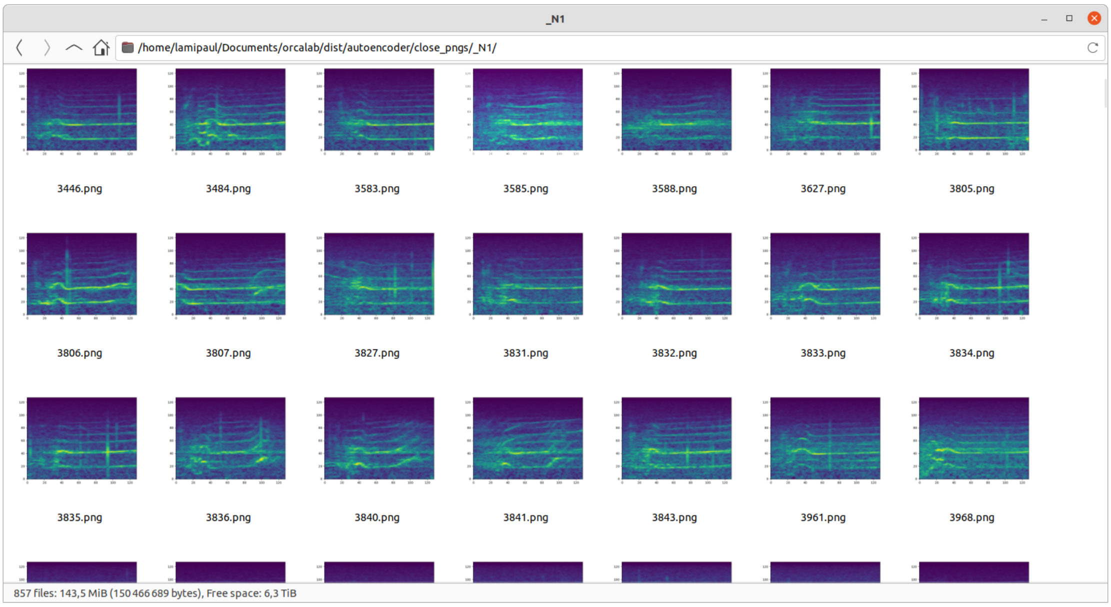


Figure 4.2: Example of thumbnails ready to be annotated (picked), using the Thunar file explorer [207]. Here, files are clustered spectrograms of orca calls (see section 4.4.3).

4.2 Thumbnail picking

Often during annotation procedures, we want to manually sort out true and false positives from a set of detections. It occurred numerous times during this thesis, after the aforementioned preliminary detection algorithms or during the active learning process (section 4.5). Picking spectrogram images from their thumbnails in file explorers appeared to be the most efficient way to do it (see Fig. 4.2).

The typical scenario in which this procedure was used is to pick false positives from a set of detections. In a few minutes, an annotator can browse hundreds of samples (exhaustively or not), and select dozens of files to move them to a new folder. Using table identifiers as filenames then allows to retrieve the annotator's decision and save it for later use.

Annotating by organising of thumbnails in folders is not only efficient in time, but also very generic (it requires no specific software installation). This comes practical especially when needing annotation efforts from different people with different operating systems for example.

4.3 Gathering regions of interest

When choosing machine learning to build detection systems, we must first gather annotations. For this purpose, we can start by running an algorithm that filters the data using our prior knowledge of the target signal(s). These handcrafted algorithms present limitations (as argued in section 2.2.1), but avoid having to go through the whole set of available recordings to find our first training examples.

In detection algorithms, the user usually sets a threshold to binarise continuous prediction values. For instance with cetacean vocalisation detection tasks, the threshold is typically on the energy level at a specific frequency, or on the cross-correlation coefficient (template matching approaches). The lower we set this threshold, the lower the specificity (higher risk of false detections) but also the higher the sensitivity (lower risk of missed detections). Conversely, by increasing this threshold, we increase the specificity but decrease the sensitivity.

This trade-off is to be kept in mind when tuning handcrafted algorithms to build a first database: we want just enough sensitivity to yield some true positives (perhaps the ones with the highest SNR), while keeping the number of detections low enough so that we can go through them in a reasonable amount of time.

The following paragraphs introduce two case studies of such approaches, one with stationary signals (orca calls) and one with transitory ones (sperm whale clicks).

4.3.1 Spectrogram energy thresholding (orca calls)

This work was conducted in collaboration with Jan Schlüter and Marion Poupard, on the OrcaLab data (see section 3.2.1).

The chosen approach to the preliminary detection of orca calls was inspired by Lasseck [108] on spectrogram segmentation for bird call detection. We first binarise spectrograms (see Fig. 4.3) with adaptive thresholds using rows and columns moments. The original formulation proposed by Lasseck [108] for the threshold $T_{f,t}$ given a log compressed spectrogram $\mathbf{E}$ is given by Eq. 4.1. The goal being to detect pixels with energy values above the distribution of their row and column, we propose to rather use Eq. 4.2.

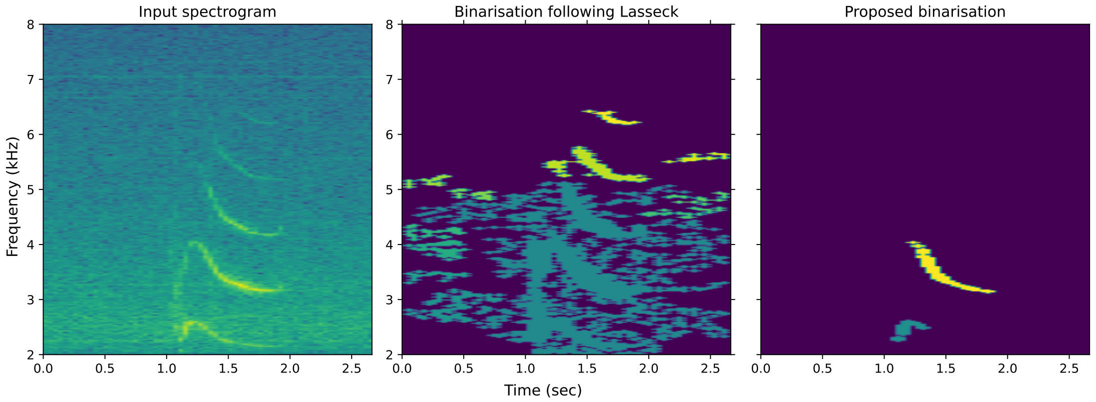


Figure 4.3: Comparison of the spectrogram binarisation procedure following Eq. 4.1 (middle) and Eq. 4.2 (right).

$$\mathrm{T}_{f,t} = \max(3 \times \underset{j}{\mathrm{median}}(\mathrm{E}_{f,j}),\ 3 \times \underset{i}{\mathrm{median}}(\mathrm{E}_{f,i})). \tag{4.1}$$

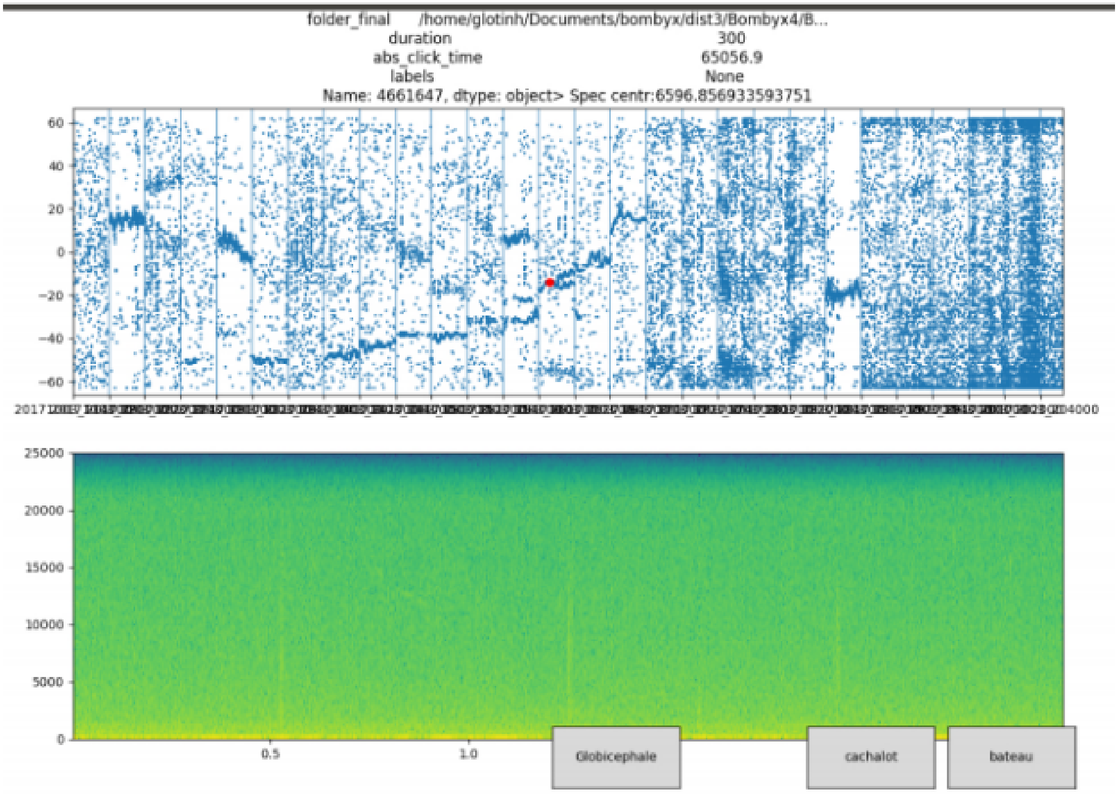
$$\mathrm{T}_{f,t} = \max(\underset{j}{\mathrm{median}}(\mathrm{E}_{f,j}) + 2 \times \underset{j}{\mathrm{std}}(\mathbf{E}_{f,j}),\ \underset{i}{\mathrm{median}}(\mathbf{E}_{i,t}) + \underset{i}{\mathrm{std}}(\mathbf{E}_{i,t})). \tag{4.2}$$

Connected positive pixels are later grouped by regions, from which we will extract features such as minimum and maximum frequencies, length, and mean and maximum decibels. We finally use our prior knowledge of orca calls to filter out impossible regions (out of range features), and plot them for annotation via thumbnail picking (see section 4.2).

4.3.2 TDOA tracking (sperm whale clicks)

This work was conducted in collaboration with Maxence Ferrari and Marion Poupard, on the Bombyx data (see section 3.2.1).

For sperm whale clicks, time domain signal processing is more appropriate than the spectral based energy detection presented above. In a first pre-processing step, sperm-whale clicks are emphasised by correlating the signal with a sinus of their centroid frequency (12.5 kHz). Then, the permissive detection mechanism is based on the Teager-Kaiser (TK) energy operator (inspired by Kandia and Stylianou [97]). The TDOA of the detections were then computed between the two hydrophones of the antenna, as we will see that spatial information is quite useful for the identification of sperm whales.



```
folder_final      /home/glotinh/Documents/bombyx/dist3/Bombyx4/B...
duration                                                        300
abs_click_time                                              65056.9
labels                                                         None
Name: 4661647, dtype: object> Spec centr:6596.856933593751
```

Figure 4.4: Custom UI built in matplotlib [87] for the annotation of sperm whale clicks. (top) TDOAs versus time of detected clicks, with vertical bars denoting gaps between recorded files. (bottom) Spectrogram of the signal surrounding the selected click, shown with a red dot on the top panel.

In our data the three main signals that trigger such a detector are those produced by sperm whales, boats, and other odontoceti such as long-finned pilot whales (*Globicephala melas*). To discriminate between these three for annotation, while browsing the large amount of recordings, the custom UI shown in Figure 4.4 was built.

This UI shows a scatter plot of TDOA of preliminary detections versus time. This allows for the identification of tracks revealing moving acoustic sources, with the slope reflecting angular speed relative to the antenna. With such a plot, we display 10 hours of signal at once, enabling a quick browse through large amounts of data. When clicking on a point of the scatter plot, it is signaled with a red dot, and the surrounding signal's spectrogram is displayed on the bottom pane while the sound is played. This allows for the identification of the source responsible for the selected track. The user can eventually click on buttons to save an annotation along with its timestamp (noise, pilot whale or sperm whale).

4.4 Feature extraction and filtering

Clustering allows for a strong optimisation of the annotation process. Indeed, once signals are grouped by similarity, browsing and sorting becomes much more efficient, especially by avoiding to go through large amounts of void.

The key to clustering quality is the extraction of relevant features for similarity measurement. Hereby are presented three feature extraction approaches on different kinds of signals : humpback whale vocalisations, toothed whale clicks, and orca calls.

Once features were extracted, they were usually projected using Uniform Manifold Approximation and Projection (UMAP) [124] before a Density Based Spatial Clustering of Applications with Noise (DBSCAN) clustering [51]. Allaoui et al. [3] have shown that dimensionality reduction using UMAP would improve the performance of clustering such as density based ones. The distribution of projection in turn motivated a density based approach to clustering such as DBSCAN (Fig. 4.6).

4.4.1 Spectro-temporal features (humpback whale calls)

This work was conducted on the Carimam dataset (see section 3.2.1).

The objective of the following procedure is to explore a large dataset with no samples of target signals given a priori. For this purpose, a relatively generic feature extraction was conducted before plotting and clustering their projection. Like so, we intend to isolate groups of similar events, and allow for a more efficient exploration of the data. Especially, the events we hope to find are click trains and cetacean vocalisations, but we also expect to retrieve events from other noise sources.

The extracted features process spectrogram chunks in two main steps in order to emphasis potential signals of interests (Fig. 4.5). First, to get a representation that preserves short events (such as clicks) but with a reduced size, we max-pool spectrograms time wise. Second, to make abstraction of the temporal information (whether an event is at the beginning of a chunk or at its end) we sort each frequency bin (time wise) in descending order.

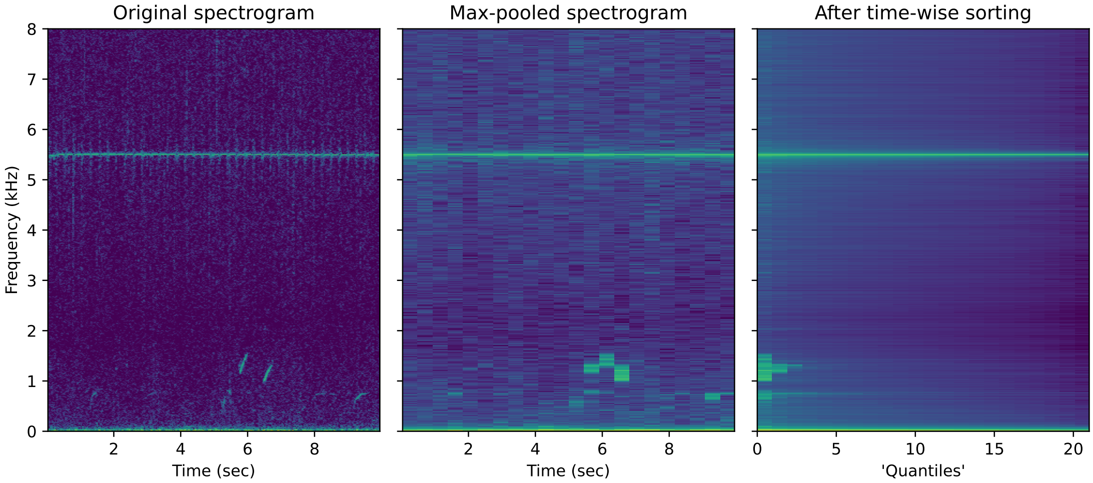


Figure 4.5: Main steps of the spectral feature extraction procedure. The spectrogram is first max-pooled time-wise by a given factor. Then, each frequency bin is sorted (time-wise) to make abstraction of the temporal position of events.

Doing so, the resulting matrix is no longer a spectrogram, but rather a representation of the energy distribution for each frequency bin. This allows to select specific columns as discriminating features, the first denoting the highest energy in the chunk for each frequency bin, and the last their lowest. For instance, looking for short events, we can select the first and second columns. Chunks with a large gap between the two should contain a short but strong acoustic event. It could thus be differentiated from chunks with stationary strong energy and chunks with a low overall energy, and this for specific frequency bins. For simplicity, this set of columns to be kept will be referred to as 'quantiles'. The full procedure for this analysis is described in Listing 4.1.

Listing 4.1: Feature extraction and clustering for humpback whale vocalisations. Steps are operated over a batch of signals on GPU for computation speed.

```
from torchaudio.functional import resample
import torch
from sklearn.cluster import DBSCAN
from umap import UMAP
gpu = torch.device('cuda')

# load a batch of signals using pyTorch DataLoader
sigs = ...
sigs = sigs.to(gpu)
sigs = resample(sigs, source_fs, fmax * 2)
```

```
# compute the magnitude spectrogram using the STFT
specs = torch.stft(sigs, n_fft=1024, hop_length=512)
specs = 20 * torch.log10(specs.norm(p=2, dim=-1))
# substract a background noise estimate
specs = specs - specs.median(dim=1, keepdim=True)[0]
# apply the mel-transform
spec = torch.matmul(melbank, specs)
# undersample the spectrogram over the time dimension
specs = torch.nn.MaxPool1d((uds,))(specs)
# rearange the tensor into a list of time chunks
specs = specs.permute(1, 0, 2)
specs = specs.reshape(specs.shape[0], -1, chunksize)
specs = specs.permute(1, 0, 2)
# sort frequency bins and select quantiles
features = torch.sort(specs, dim=2, descending=True)[0]
features = features[:,:,quantiles].numpy()
# project and cluster each time chunk
features = features.reshape((specs.shape[0], -1))
embeddings = UMAP().fit_transform(features)
clusters = DBSCAN().fit_predict(embeddings)
```

The variables `fmax`, `uds`, and `chunksize` need to be tuned to the type of signals we desire to isolate, especially in terms of spectrum range and spectrogram temporal resolution. `fmax` determines f_s at which the signal is resampled, `uds` determines the downsampling factor used for max-pooling, and `chunksize` determines the sampling rate of the feature extraction process. As for the humpback whales, they were set to 8,000 Hz, 14 and 20 respectively (chunks of 10 sec with 20 time bins). Then, the first seconds and fifth quantiles were chosen.

These choices were made via intuition and empiric testing. Once annotations were gathered, experiments were carried out to measure which configuration would have been the most efficient.

Trials with varying values for the size of chunks and the choice of quantiles were conducted, using the NMI between clusters and annotations as a metric of configuration quality. The choice of configuration appeared to have a relatively small impact on the resulting NMI, with values ranging between 0.15 and 0.18 (the random baseline being under 0.01). The highest scoring configuration was to cut chunks of size 10 with only the first quantile.

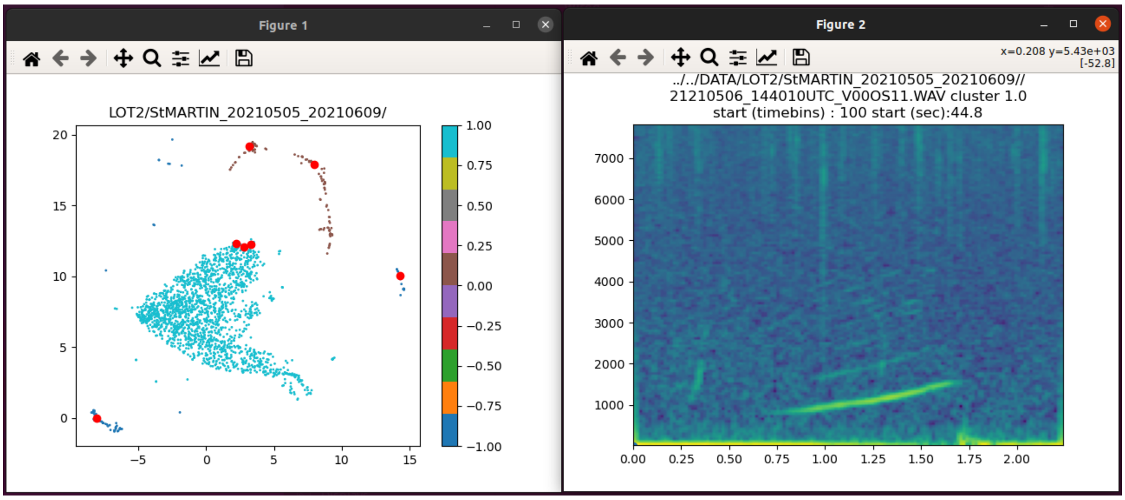


Figure 4.6: Interface for browsing clusters. The left panel displays clustered UMAP projections of audio chunks spectral features, with red dots signaling points that have been clicked on. The right panel displays the spectrogram of the last selected audio as well as its metadata.

Once features have been extracted for a large amount of samples, we reduce their dimensionality (using UMAP), and cluster them (using DBSCAN). A custom made interface then enables a seamless browsing of this clustered projection (see Fig. 4.6).

Users can select an audio chunk by clicking on its projection on the scatter plot. The interface will then play the corresponding sound extract and display its spectrogram on a secondary window. This allows for the identification of discriminant clusters to be retained (containing only vocalisations, or only noise for instance). Eventually, we can plot samples belonging to selected clusters as .png files and use thumbnail picking (see section 4.2) to sort out misclassified samples.

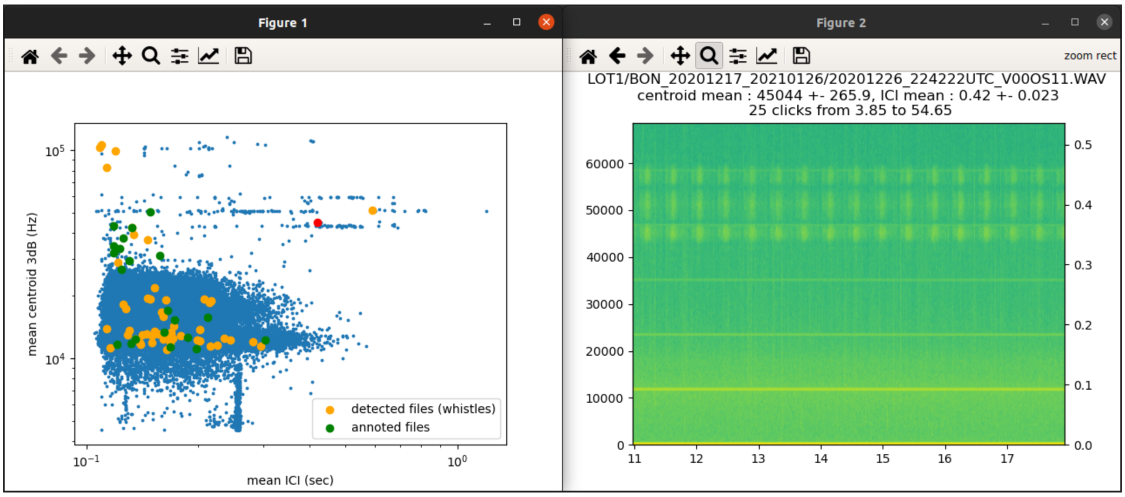
4.4.2 Impulses' features (toothed whale clicks)

This work was strongly inspired by Frasier [67], conducted in collaboration with Maxence Ferrari and Marion Poupard, on the Carimam data (see section 3.2.1). In a similar approach, we might want to cluster clicks for their spectral features to infer ICI characteristics, which helps discriminate toothed whales click trains from reef noise. To do so, using the STFT as seen in the previous section is not appropriate. We would rather use a generic impulse detection mechanisms on the waveform, and extract their features. I thus propose the following steps :

- Generic impulse detection:
  - high pass the signal $x(t)$ at 5 kHz,
  - compute the Hilbert transform $H(t)$ of $x(t)$,
  - compute a running average $a(t)$ to smooth $H(t)$,
  - convert $a(t)$ into decibels with $20 \times \log_{10}(a(t))$,
  - compute the median and std of $a(t)$,
  - find peaks of $20 \times \log_{10}(x(t))$ that are 3dB above the noise level expressed as median $+3 \times$ std,
  - retain peaks with widths between 0.008 and 1.2ms, and retain its highest sample.
- Feature extraction:
  - compute the FFT of a 1ms window surrounding the detected impulse,
  - compute the 3 dB centroid frequency,
  - cluster impulses by their centroid frequency,
  - compute ICIs as the time difference between impulses of the cluster,
  - fit a gaussian Kernel Density Estimate (KDE) on the ICI distribution of the cluster,
  - estimate the peak of the KDE,
  - for each cluster, save the peak of the KDE (most frequent ICI), its width (ICI variability), and the mean 3dB centroid frequency.

The user can eventually filter data on the KDE peaks height and width depending on the desired specificity. An interface then displays a scatter plot of ICIs vs centroid frequencies (Fig. 4.7). Again, a click on a point triggers a spectrogram display of the corresponding signal, which can be further analysed and eventually saved for annotation.

This method was used to explore the data in view



```
LOT1/BON_20201217_20210126/20201226_224222UTC_V00OS11.WAV
centroid mean : 45044 +- 265.9, ICI mean : 0.42 +- 0.023
25 clicks from 3.85 to 54.65
```

Figure 4.7: Interface for browsing cluster of clicks. The left panel displays the mean ICI vs 3dB centroid frequency for each cluster of impulses. Red dots signal the selected cluster of clicks. The right panel displays the spectrogram of audio surrounding the selected cluster as well as its meta data.

4.4.3 AE embeddings (orca calls)

This work was conducted on the OrcaLab data (see section 3.2.1), and has been subject to a workshop intervention [19].

For this section, we are interested in the classification of pre-detected orca calls (dataset of 114k orca calls detected by a CNN presented in section 5.3.1). Call types, as defined by Ford [65] for NRKWs, are determined by their temporal pitch patterns. First experiments were thus conducted using a pitch based feature extraction to cluster calls [152]. However, the estimation of the pitch appeared to be quite unreliable in low and medium SNR conditions (see section 2.2.1). This led to a switch towards a larger scale extraction of shape (as opposed to local pitch estimates).

Auto-encoders (introduced in section 2.1.4) are trained to compress data in a lower dimensional embedding space while being able to reconstruct it. Moreover, since the reconstruction relies on learning structure in the data, the noise in the input (random and unstructured) is omitted in the output. This motivates the use of AEs for the feature extraction of orca calls, expecting the bottleneck to contain the shape of the call in a low dimensional space.

The training framework of the AE was designed as follows (see Fig. 4.9):

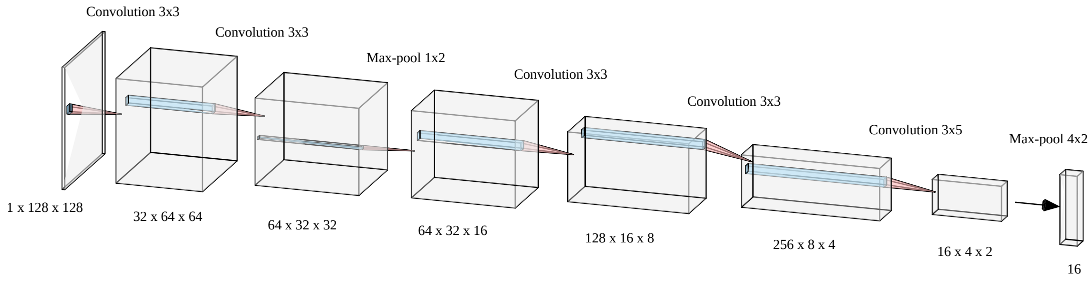


Figure 4.8: Architecture of the encoder part of the AE. (Bottom) shapes of volumes as ($depth \times height \times width$). (Top) Operations and kernel shapes as ($heightxwidth$).

- Compute Mel-spectrogram on windows of 2sec around detections ($f_s = 22050$, $NFFT = 1024$, $hop = 330$, $\#Melbands = 128$, $f_{min} = 300$, $f_{max} = 11,025$),
- Run the encoder to compress the 128x128 image to 16 dimensions (Fig. 4.8). Each convolution is followed by batch normalisation and leaky rectifier linear units. The resolution is lowered via strides of 2 for each convolution, and a max-pooling layer,
- Run the decoder as the mirror of the encoder. The first 16x4x2 volume is created via a linear layer, and each resolution increase consists in upsampling by nearest value followed by two convolution layers of kernels 3x3,
- Compute the VGG embedding of the input and the reconstructed images as the activations after the 6th convolutionnal layer,
- Use the MSE between the two VGG embeddings as the loss ($\mathcal{L} = \Sigma \|\mathrm{VGG}(\mathbf{E}) - \mathrm{VGG}(\hat{\mathbf{E}})\|^2$).

The VGG mentioned, used for the perceptual loss [95], is a VGG16 pretrained on the ImageNet dataset [43].

The size of the bottleneck was empirically chosen as the minimum that still enables satisfactory reconstructions. Fig. 4.10 demonstrates how, in reconstructions, details of some calls are omitted, and background noise becomes patterned. Indeed, due to the limited amount of information that the bottleneck can fit, the decoder is forced to learn common data structures to reconstruct the data. This is actually

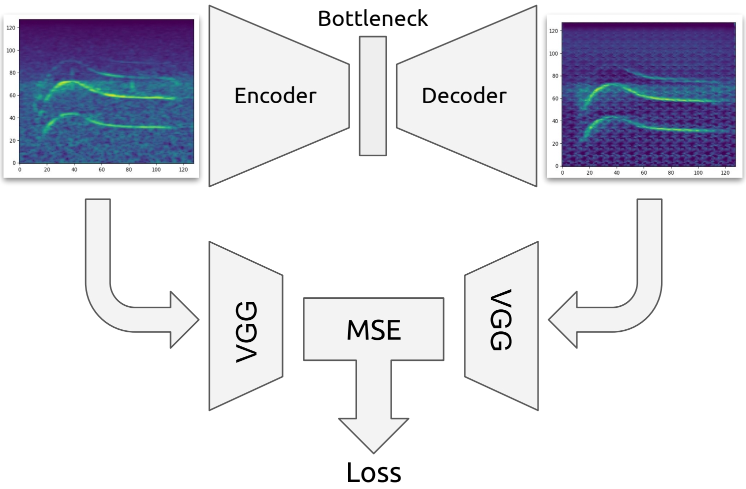


Figure 4.9: Architecture of the training framework for the AE of orca calls using a perceptual loss [95].

beneficiary for our end goal of grouping roughly similar shapes together, and it explains why random background noise, transient clicks, and small variations in call shapes are not found in output spectrograms.

The bottleneck embeddings were later used as features for DBSCAN clustering (after UMAP dimensionality reduction [124]). This enabled a drastic reduction of the annotation effort by grouping similar calls together. Thumbnail picking (see Fig. 4.2) was then conducted to verify clusters and associate them with the orca call types defined by Ford [65].

4.5 Active learning

Active learning is the process of iteratively training and annotating to improve a database (qualitatively and/or quantitatively). It is relevant when one has a training database that is not large enough to ensure satisfactory performances. By correcting the model's predictions at each iteration, we emphasis on difficult examples and guide it towards robustness.

This active learning process was conducted via thumbnail picking to gather annotations for fin whale pulses, dolphin whistles, humpback whale vocalisations, and orca calls.

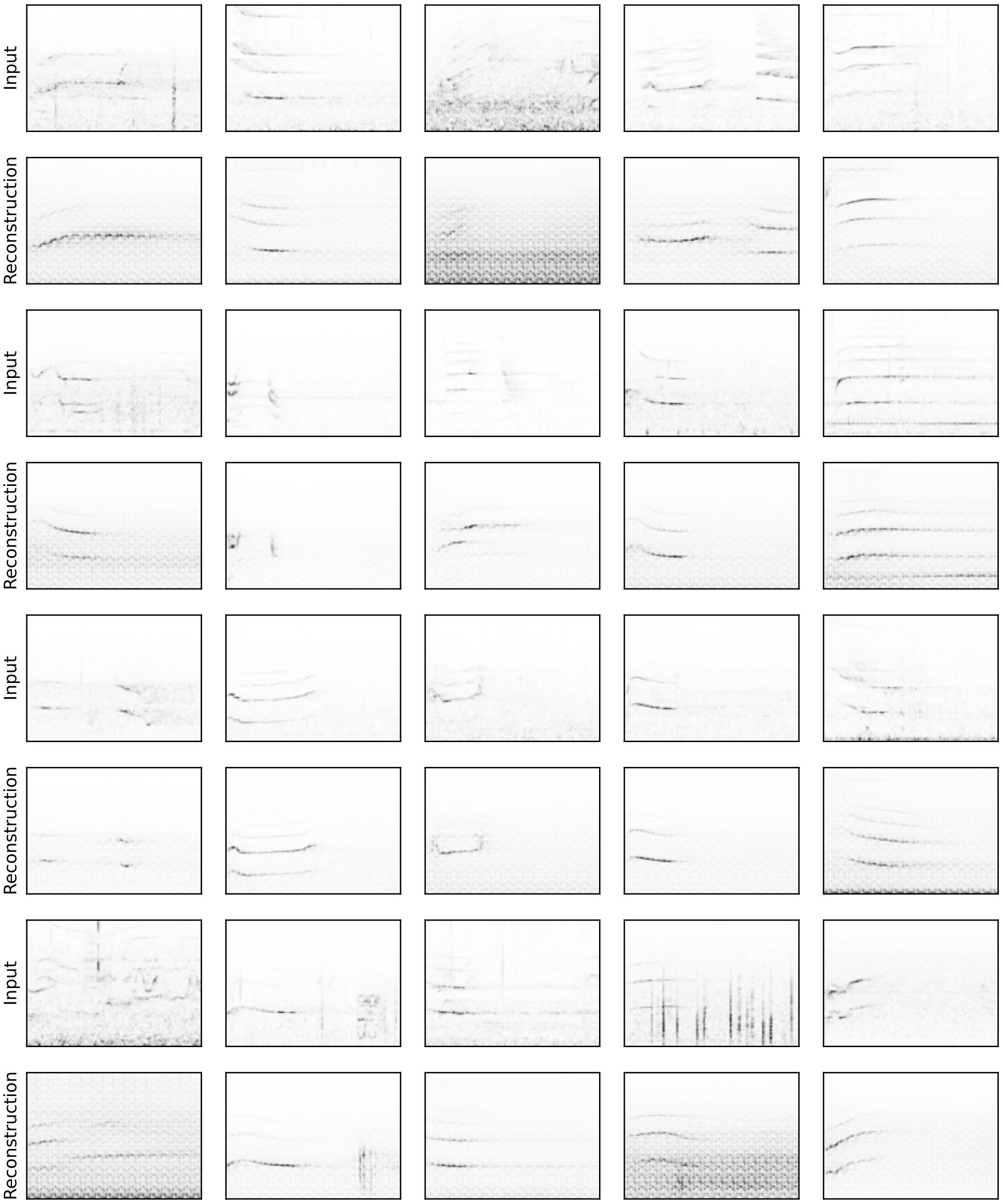


Figure 4.10: Comparison of input and AE reconstructed spectrograms.

4.5.1 Transfer learning (fin whale pulses, dolphin whistles)

Pre-training a model on a database before fine tuning on a different one is called transfer learning. Similar approaches were used to kick-start the active learning process on two detection tasks, as described in the following paragraphs.

Fin whale pulses

To gather a database of fin whale 20 Hz pulses from the recordings of Bombyx and Boussole (see section 3.2.1), several handcrafted algorithms were first tested (looking for strong energy peaks in realistic time and frequency ranges). Without any exemplary signal to tune them, and with the wide variety of noises present on both banks of recordings, this approach failed to yield any fin whale signals.

We were lucky to eventually get some help from M. Giani Pavan, who shared some of his recordings of Mediterranean fin whale songs containing 100 pulses [146]. Despite the limited amount of samples, training a small neural network (see section 5.2.1) on this data allowed to find similar signals on the Bombyx and Boussole datasets (see section 3.2.1). This demonstrates the capacity of small neural networks to generalise to different recorders even with very few training samples.

Active learning with thumbnail picking further helped increase the database to a satisfactory size (see Tab. 4.1).

Dolphin whistles

This work has been conducted in collaboration with Marion Poupard.

For this task, as for the fin whale pulses, we used other sources of data available at the lab as a starting point to the active learning process. This time though, the variability of the signals to be detected prevented the use of a low complexity architecture.

Thus, to enforce the generalisation of the model to other recording systems, the available data was augmented with negative samples from the target recording system (Carimam). By mixing annotated foreign inputs with negative samples from Carimam, we teach the model to be robust to common Carimam perturbations (self noise from the sound card, reef noise), while training on positive samples of the target signal. This 'mixing' takes form as a simple summation of the waveforms after their standardisation.

Active learning with thumbnail picking further helped increase the database to a satisfactory size (see Tab. 4.1).

4.6 Resulting annotations and train / test splits

The methods proposed in this section yielded enough annotations to train ANN models on each detection / classification tasks (Tab. 4.1).

Target signal	Positives	Negatives	Total
Sperm whale clicks	42%	58%	5,554
Fin whale 20 Hz pulses	14%	86%	5,790
Orca calls	78%	22%	6,004
Humpback whale calls	42%	58%	1,377
Dolphin whistles	12%	88%	1,595

Table 4.1: Summary of the annotations gathered on the data at hand for detection task.

The performance measurement methods employed need to reflect our end goal, namely training robust detections models. Robustness, can be defined as the capacity to ignore perturbations, some kind of resilience. In our case, perturbations are sound events and background noises, especially those not seen in training. To measure robustness, our test data must thus contain new acoustic content, somewhat different from training.

The randomly sampled train / test splits often seen in the machine learning community is insufficient in that sense. Indeed, train and test samples will be extracted from the same vocalisation / noise sequences, thus sharing most of their characteristics. On the other hand, choosing a specific source of data, or if not available a distinct time period, should yield novelty in the test set, and give relevant robustness measures for our models.

Fin whale pulses

The gathered annotated database of fin whale 20 Hz pulses offers three different data sources (Table 4.2). Thus, in the experiments, three folds were used : each using two sources for training and the remaining one for testing. The Magnaghi data corresponds to the extracts provided by G. Pavan (see 4.5.1).

Data Source	Positives	Negatives	Total
Magnaghi	15%	85%	688
Boussole	9%	91%	4,528
Bombyx	49%	51%	574
Total	14%	86%	5,790

Table 4.2: Distribution of annotations of 20 Hz fin whale pulses. Each source of data was used as test set in a 3 fold manner.

Sperm whale clicks

The annotated database of sperm whale clicks coming from only one source of data (Bombyx), The year 2017 was chosen for testing and the remaining for training (Table 4.3). This choice is motivated by the fact that 2015 has too few samples for the test to be relevant, 2016 has a positive / negative distribution too different than the global dataset, and 2018 has the largest amount of samples which is desirable for training. To improve the annotation comes from separate files.

Experiments showed that the model would tend to lack sensitivity, with the exception of pilot whale samples which would trigger a low specificity. To tackle this issue, and accounting for the imbalance in the data (Tab. 4.3), sperm whale and pilot whale samples were over-sampled during training, by a factor 3 and 10 respectively.

Recording year	Sperm whale	Boat / Noise	Pilot whale	Total
2015	48%	52%		256
2016	75%	23%	2%	1,383
2017	32%	67%	1%	1,363
2018	28%	68%	4%	2,552
Total	42%	55%	3%	5,554

Table 4.3: Distribution of annotations for the sperm whale click detection task. The year 2017 was used as test set.

Humpback whale calls

For the detection of humpback whale calls, the data recorded from Sint Eustatius island was selected as the test set (Table 4.4). The Sint Eustatius antenna was chosen for the test set as it has a representative distribution of classes and is neither too small nor too big ($\sim$ 10%).

Station	**Positives**	**Negatives**	**Total**
Anguilla	100	0	100
Bahamas	0	45	45
Bermude	276	27	303
Guadeloupe	666	26	692
Jamaica	0	11	11
Martinique	354	37	391
Saint Barthélémy	173	0	173
Sint Eustatius	204	103	307
Saint Martin	163	242	405
Total	67%	33%	2,427

Table 4.4: Distribution of humpback whale calls annotations through the Carimam recording stations. The Sint Eustatius data source was used as a test set.

Dolphin whistles

For the detection of dolphin whistles, the data recorded from Guadeloupe Breach was selected as test set (Table 4.5).

Station	**Positives**	**Negatives**	**Total**
Guadeloupe Breach	36	354	390
Gualdeloupe Anse Bertrand	0	49	49
Saint Barthélémy	0	16	16
Sint Eustatius	37	111	148
Saint Martin	0	34	34
Jamaica	24	10	34
Bonaire	74	25	99
Bermude	25	439	464
Bahamas	0	16	16
Anguilla	0	345	345
Total	12%	88%	1,595

Table 4.5: Distribution of dolphin whistles annotations through the Carimam recording stations. The Guadeloupe Breach data source was used as test set.

Orca call detection

A special recording session was run at OrcaLab in 2019 by Poupard et al. [154], for the study of group dynamics via triangulation. The manual annotations gathered for this experiment were used in this thesis, bringing an opportunity to measure the impact of a change in recording hardware on detection mechanisms with no additional annotation effort. Two test sets were thus used for the orca call detection task, one from the same antenna than in training but in a different year and one from a different antenna (see Tab. 4.6). A preliminary study using this dataset was published in a conference paper [17].

Recorder	Year	Positives	Negatives	Total
OrcaLab network	2015 - 2017	846	3,777	4,623
OrcaLab network	2019	111	177	288
Poupard et al. [154]	2019	368	725	1,093

Table 4.6: Distribution of orca calls binary annotations. The data from 2019 (two different antennas) was used as test set.

Orca call classification

Given the diversity of classes and the singular recording source, for the orca call classification task, the train / test split was simply done by sorting by date and choosing a proportion for test and the rest for train. For instance, the first 10% of each class were taken for test, and the remaining 90% were used to train the model.

call type	instances
N1	854
N2	191
N3	192
N4	1213
N5	209
N9	609
N23	469
other	109
Noise	814

Table 4.7: Distribution of annotations of orca call types [65]. The ‘other’ class corresponds to infrequent calls that did not have enough occurrences to form an independent class.

Antarctic blue and fin whale calls

Table 4.8 summarises the distribution of labels for each data source available in the Acoustic Trends dataset [128]. The Kerguelen 2005 data source was chosen as a test

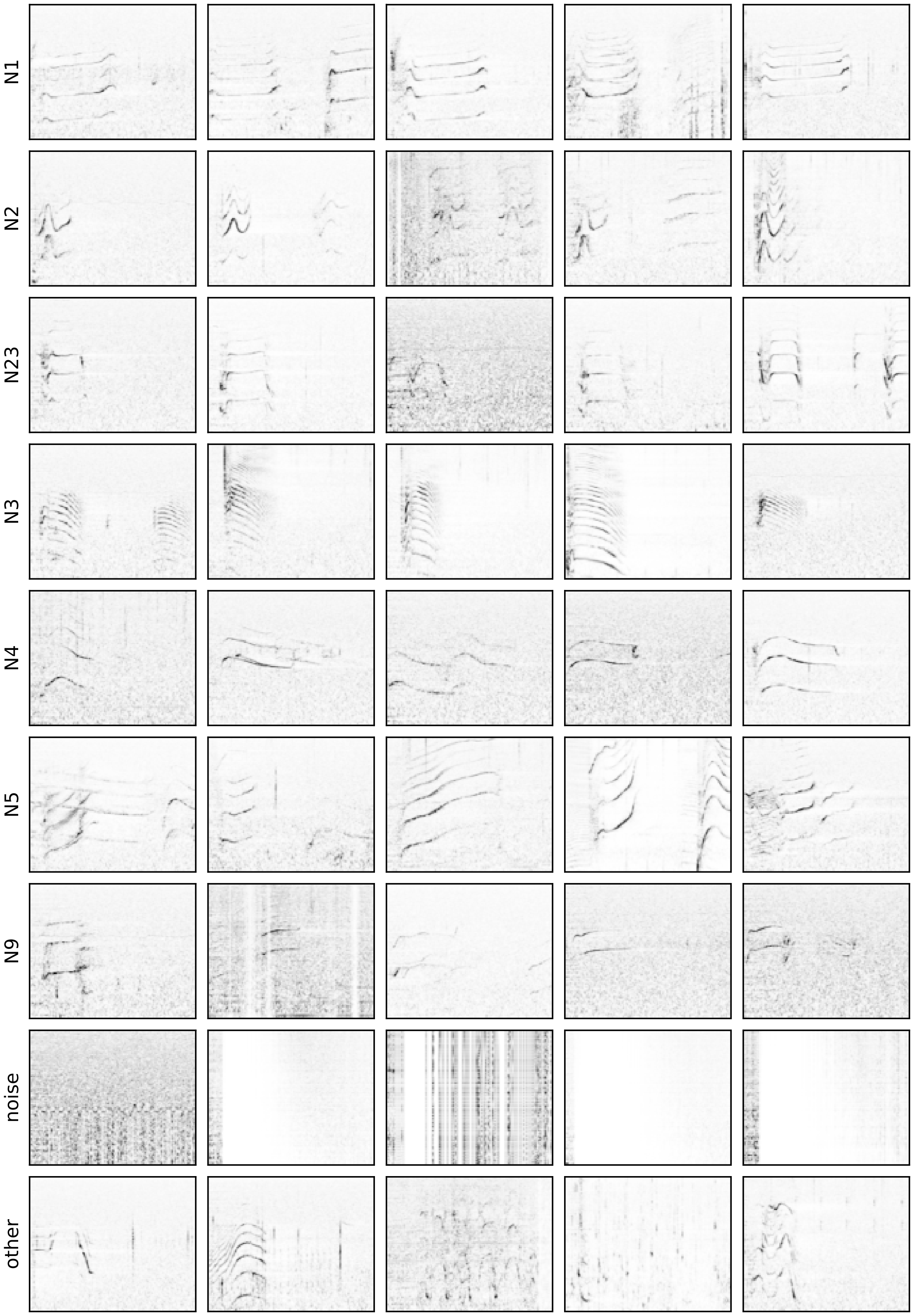


Figure 4.11: Examples of each class of orca call types annotated using clusters of AE embeddings. The terminology as defined by Ford [65] has been used by associating calls with their closest class in the catalogue.

set. Its specific recording system and location, as well as its sufficient support of all classes motivated this choice. The remaining recordings were used for training.

4.7 Discussion

As seen throughout this chapter, techniques employed in pre-detection, feature extraction and filtering need to be adapted to the type of target signal and the available recordings. For that matter, Tab. 4.9 recapitulates the choices made for each of the 6 annotation procedures conducted.

Let us get an idea of the time it would have taken to annotate the sperm whale click database via random sampling for instance. Sperm whales were confirmed on 6% of the files from Bombyx (see section 6.3.1). If we consider 30 seconds to manually check a file (between 1 and 5 minutes long), to yield the 2,300 positive labels collected here (they are each on separate files), one would need 320 hours. It took approximately 40 hours in total collect this database with the annotation approach described in 4.3.2.

The following paragraphs summarise the advantages of some methods employed through these experiments, along with potential pitfalls to be kept in mind.

4.7.1 Active learning

Active learning has proven to be very efficient in iteratively increasing database sizes. It can be started as soon as few dozen annotations are at hand. Indeed, in that case, rather than spending time in tuning pre-detection mechanisms to collect more samples, deep learning models help to collect occurrences of the target signal as well as disruptors (e.g. boats, signals from other species). Moreover, it is worth the efforts of developing the training procedures since they will be used subsequently, as opposed to the pre-detection algorithms which are rather a one time usage.

Nonetheless, there is a danger that comes along with active learning: the progressive specialisation of the model to detect only one type of signal. Indeed, if the initial annotations omit some type of signals from a target species, it is likely that the model will never learn to detect them. This especially comes from the

Location	Year	Instrument	Bm A	Bm B	Bm Z	Bm D	Bp 20 Hz	Bp 20+	Bp DS	Negatives
Balleny Islands	2015	PMEL-AUH	4%	1%	1%		7%	2%	1%	10%
Elephant Island	2013	AURAL	10%	26%	6%	71%	28%	24%	16%	30%
Elephant Island	2014	AURAL	28%	14%	4%	7%	38%	38%	64%	6%
Greenwich 64S	2015	Sono.Vault	3%	2%	1%				1%	1%
MaudRise	2014	AURAL	9%	1%	1%					3%
Ross Sea	2014	PMEL-AUH								9%
Casey	2014	AAD-MAR	15%	20%	43%	4%				8%
Casey	2017	AAD-MAR	7%	8%	5%	4%	1%	3%		8%
Kerguelen 1	2005	ARP	6%	3%	7%	3%	6%	1%	7%	9%
Kerguelen 2	2014	AAD-MAR	10%	17%	22%	3%	15%	24%	5%	8%
Kerguelen 2	2015	AAD-MAR	8%	8%	9%	8%	4%	9%	5%	8%
Total			25,177	6,903	2,515	15,339	12,933	7,761	6,381	357,765

Table 4.8: Distribution of annotations published by Miller et al. [128]. The Kerguelen 2005 was chosen as test set.

Target signal	Pre-detection	Feature extraction	Filtering
sperm whale clicks	TK filter	TDOA	custom UI
humpback whale calls	NA	spectral features	custom UI and clustering
orca call detection	spectrogram thresholding	region statistics	hand-crafted rules
orca call classification	CNN	auto-encoder	clustering
20 Hz fin whale pulses		transfer learning	
dolphin whistles		transfer learning	

Table 4.9: Summary of steps employed in the initial annotation process of each target signal.

fact that we often only correct the positive predictions of the model, sorting out false positives (negative predictions usually come in much larger numbers, making it fastidious to find false negatives). To mitigate this effect, one can manually browse recordings around detections and annotate full sequences, or look for false negatives in low confidence negative predictions.

4.7.2 Thumbnail picking

Thumbnail picking allows to quickly validate detections or clusters to collect annotations. The only condition is to find a visualisation that fits a small size and still allows to make a decision on sample's classes (small spectrograms work well for most stationary signals, see Fig. 4.2). It is versatile and easily shareable to experts (the only prerequisite is to have a graphical file manager). It is fast and user friendly: you just need to click on files to select them and move them to a separate folder (cut and paste). Also, seeing multiple samples at once strongly helps the eye in discriminating singularities.

This last advantage can also be dangerous in the annotation process. Indeed, when sorting large folders to try and keep only a class of call, one might always see similar calls at a time on the screen, but through scrolling, pitch contour shapes might shift progressively. When this occurs, the annotator might feel like all the calls in the folder are similar, when in fact, the ones at the beginning are very different from the ones at the end. To mitigate this, indexes should be randomly permuted, since this progressive shift in call contour is likely to occur if files are sorted time wise, but very unlikely otherwise.

4.7.3 'Generic' spectral features extraction

Section 4.4.1 proposes a procedure suited to explore large banks of recordings by grouping events with similar content in terms of frequency energy distribution. It allowed to collect a first database of humpback whale calls. The success of such approach relies on several assumptions:

- A minimal knowledge of the target signals is needed to configure the algorithm (e.g. frequency range, length).
- Events are grouped independently of the temporal distribution of the energy in the spectrogram (e.g. upwards and downwards chirps will yield the same features). This is suited to discriminate between events of different temporal support, but would not work to discriminate some pitch patterns.
- For the projection and the clustering to reveal a group of events, a sufficient number of instances are needed. This is the most probable explanation of why we failed to retrieve dolphin whistles and click trains using this method.

5 Training detection and classification mechanisms

Contents

5.1 Context and objective

The gathered annotations previously mentioned represent an important step towards the objective of this thesis: building robust detection and classification mechanisms for several target signals. For that purpose this chapter discusses ANN training in a supervised learning context. The detection of sperm whale clicks and fin whale 20 Hz pulses is first experimented with a constraint on computational cost (in order to be embedded in a sono-buoy, see section 6.2) . For that matter, the effect of several complexity reduction approaches is studied. Then, heavier models are used to detect Antarctic mysticetes and orca calls. Experiments focus on the effect of network frontends, architectures and hyper-parameters on performances. Furthermore, given orca call detections, trials with deep metric learning and semi-supervised learning are reported for the call type classification task.

5.2 Light weight detectors

The initial funding of this PhD was oriented towards the implementation of a real time alert system for the presence of large cetaceans in the Ligurian (Mediterranean) sea (GIAS Project). This system takes form as a battery powered sono-buoy with acoustic and processing capacities.

Motivated by the objective of deploying detection mechanisms into this embedded systems with low computing capacity, several complexity reduction approaches have been experimented with. Some measures will be given according to the specific embedded Microcontroller Unit (MCU) of the buoy: the PIC32 by MicroChip.

Two large cetacean species evolve in the Ligurian sea, and therefore are to be detected by the system: sperm whales and fin whales. Two target signals are thus concerned by the following experiments on low computational detection: sperm whale clicks and fin whale 20 Hz pulses.

This section first reports on experiments with three complexity reduction approaches (depth-wise convolution, weight pruning and weight quantisation), comparing their computational needs and performance. Then, with the chosen approach of depth-wise convolution, we investigate on optimal number of features per layer and kernel sizes via a grid search. Finally, the two selected detection mechanisms are compared with baseline algorithms of the literature.

5.2.1 Complexity reduction

The base architecture for the following experiments is a 3 layer network of 1D convolutions. It takes 64 bins Mel-spectrograms as an input :

- Sperm whale clicks: $f_s = 50\,\mathrm{kHz}$, $NFFT = 512$, $hop = 256$, $f_{min} = 2\,\mathrm{kHz}$, $f_{max} = 25\,\mathrm{kHz}$
- Fin whale 20 Hz pulses: $f_s = 200\,\mathrm{Hz}$, $NFFT = 256$, $hop = 32$, $f_{min} = 0\,\mathrm{Hz}$, $f_{max} = 100\,\mathrm{Hz}$

Following Schlüter [169], the spectrograms are compressed with $\log(1 + \mathbf{S} \times 10^a)$ with a a trainable parameter.

The frequency bins (spectrogram rows) are considered as input channels for the first 1D convolution. This choice was motivated by the fact that large spectral shifts are not expected for these target signals. Convolving frequency-wise is thus inappropriate. Using 1D convolution also significantly reduces training and inference time.

The following experiments make use of the annotated databases described in section 4.6.

Depth-wise layers

As demonstrated in section 2.1.3, using depth-wise separable convolutions is an efficient way of reducing the amount of multiplications needed in neural network systems. Fig. 5.1 compares the number of multiplications needed for an inference with regular convolution networks and depth-wise separable networks. The lower bound complexities are of $O(n^2)$ and $O(n)$ respectively (with n the number of features per layer).

Weight pruning

In ANNs, weight pruning consists in putting to 0 a proportion of weights after training [110] (e.g. the ones with the smallest L1 norm). The idea is to avoid computing multiplications for weights that are of low impact for the end prediction. Experiments were conducted to measure the effect of pruning as compared to reducing the number of features per layer before training (see Fig. 5.2).

For the model with 32 features per layer, pruning until 20% included had a non-significant effect. As for the larger models, performances were impacted starting

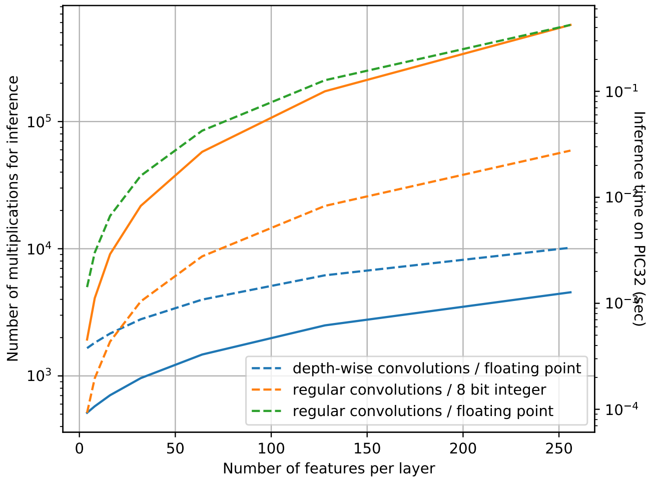


Figure 5.1: Number of multiplications needed per forward pass against the number of features per layer, for two types of architecture (solid lines). The number of multiplications were estimated for a 3 1D convolutions architecture (64 channeled input and single channeled output), stride of 1, and a kernel of size 4. Estimated inference time on the PIC32 MCU are also given (dashed lines).

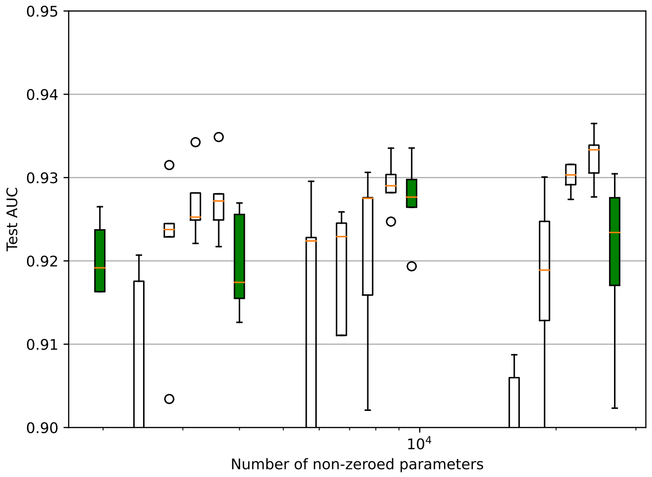


Figure 5.2: AUC performance on the sperm whale click detection task before and after pruning. Models consisted in 3 depth-wise layers with varying numbers of features (each randomly initialised 5 times). Green boxes denote the performance of models before pruning, with 16, 32, 64, and 128 features per layer. For each of them, pruning was applied over 10%, 20%, 30%, and 40% of the weights, whose performances are shown in white boxes.

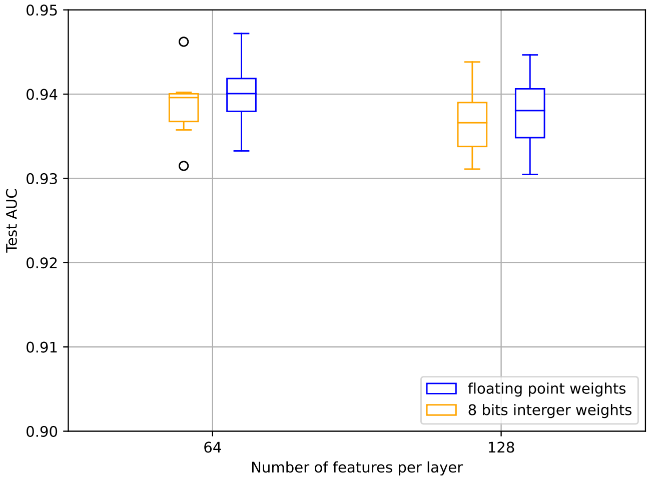


Figure 5.3: Performance for sperm whale clicks detection, before and after quantisation to 8 bits integers. 3 layer regular convolution architecture were trained 5 times for each configurations. AUC are given for the test set (see section 4.6)

from 20% of pruning. Pruning can therefore be considered a relevant option to reduce the complexity of CNN detection systems, but can only offer a marginal gain (between 10 and 20% of multiplications can be avoided).

Weight quantisation

The type of variable in a multiplication has an important impact on the cost of the operation. For instance, on the target MCU of the GIAS project (see section 6.2), the PIC32 from Microchip, a multiplication of two floating point variables takes 736 ns while multiplying two 8 bit integers takes 48 ns [30] (a factor 16 of difference). Fig. 5.1 compares inference times on the PIC32 for a depth-wise architecture of floating points against a regular convolutional architecture of 8 bit integers.

Weights were thus quantised to 8 bit integer variables in an attempt to reduce computation time. To do so, using the Pytorch [145] quantisation module, inputs weights and activations are quantised after training regularly with floating point numbers (post-training quantisation). Nonetheless, inference on a subset of the dataset is conducted to calibrate the quantisation parameters for the activations and mitigate information degradation. This quantisation approach was experimented on 3 layer architectures with regular convolution and varying number of features (Fig. 5.3).

The quantisation procedure appeared to have a non-significant impact on performance (the Kruskal-Wallis H test between the two distributions gave p-values > 0.1). Quantisation can thus be a relevant approach to the complexity reduction of models.

Conclusion

The depth-wise approach shows a significant complexity reduction, even with floating point weights numbers, and this until 16 features per layer (Fig. 5.1). At 128 features per layer (the chosen configuration for fin whale 20 Hz pulse detection), such architecture yields an inference 50 times faster than a regular convolutional one, and 5 times faster than its quantised version. Depth-wise convolutions has thus been retained for the detection systems of sperm whale clicks and fin whale pulses, the two target species of the GIAS project (section 6.2).

Implementing quantised and pruned depth-wise architectures would have been possible, but appeared to be demanding in development efforts. Moreover, as section 6.2 shows, the main cost of the buoy embedded analysis lies in the spectrogram computation rather than in the model inference (given the already reduced complexity of the CNN). Accounting for this, no further efforts were put into researching complexity reduction for these detection systems.

5.2.2 Hyper-parameter search

With the chosen 3 layer depth-wise architecture, experiments were conducted to select the optimal kernel sizes and number of features per layer. These small neural networks being quite fast to train (less than 5 seconds per epoch using the GPU), a simple exhaustive search is possible. They are summarised in Fig. 5.4 and Fig. 5.5. Networks were trained with batch normalisation, dropout ($p = 0.25$) and leaky rectifier units after the two first convolution layers. Learning rate and weight decay were manually tuned before training with varying numbers of features and kernel sizes. Kernel size and number of feature per layer were chosen to study as they were found to have the largest impact on computation cost and performances.

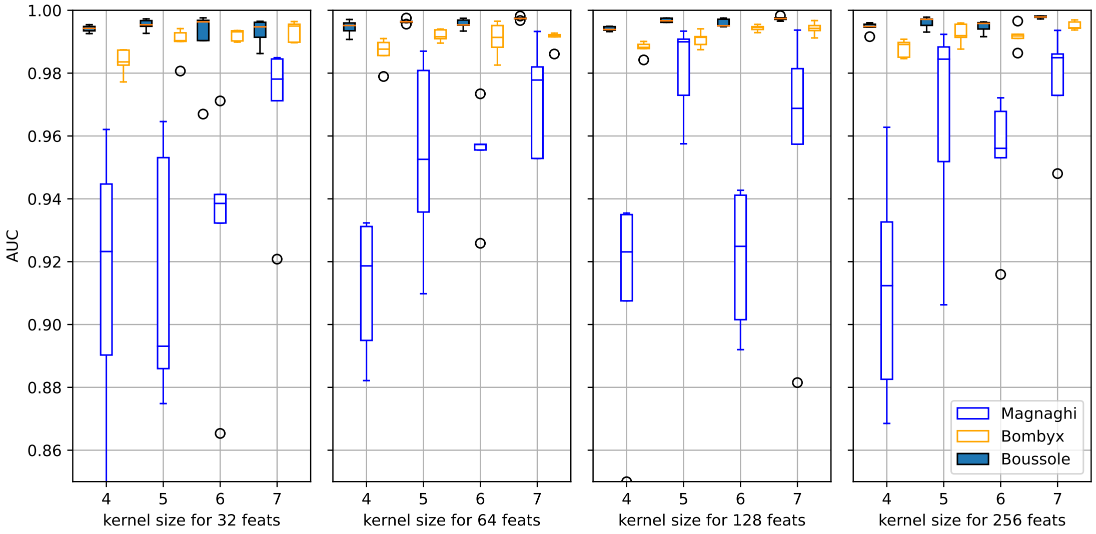


Figure 5.4: AUC performance for the 20 Hz fin whale pulse detection task. Depth-wise architectures have been experimented with several combinations of hyper-parameters (number of features per layer and kernel size). For each configuration and train/test fold, 5 runs were conducted. Folds are labelled with their test set (Bombyx scores report the performance of models trained on Magnaghi and Boussole data.

On the fin whale 20 Hz pulse detection task, the Magnaghi test set showed a great variability to multiple network initialisation, even with the same hyper-parameters. This is perhaps a consequence of specific recording setup properties, or a large gap between convergence points accounting to the two different training sources. On the two remaining folds however, performance is relatively resilient to hyper-parameter choice and initialisation. Performances of 0.99 AUC seem satisfactory for the test set.

As for the sperm whale click detection, larger kernels and deeper layers (number of features) appeared to induce some overfitting. For some configurations however, the depth-wise architectures, despite a lower amount of parameters, yield performances similar to those of regular CNNs (Fig. 5.3).

For the following experiments, the architecture with kernels of size 5 and 32 features per layer was retained for the sperm whale click detection, and kernels of size 5 with 128 features per layer was retained for 20 Hz fin whale pulse detection.

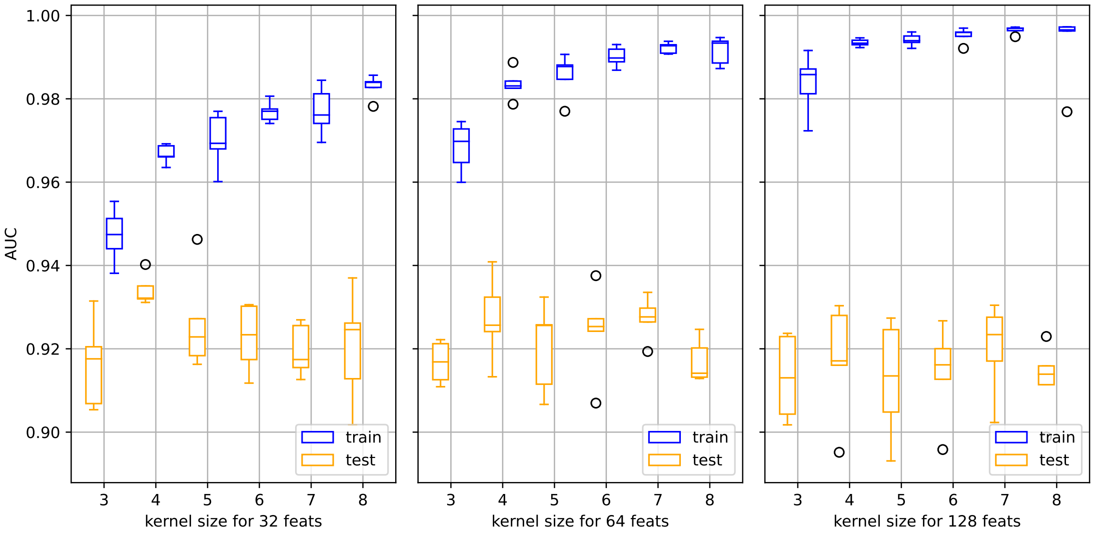


Figure 5.5: AUC performances for the sperm whale clicks detection task. Depth-wise architectures were experimented with several combinations of hyper-parameters (number of features per layer and kernel size). For each configuration and train/test fold, 5 runs were conducted.

5.2.3 Baseline comparison

The performances reported in the last section only have value relatively to that of previous systems (baselines). This section first reports on a common technique used in sperm whale click detection: the TK filter. Then, two experiments were conducted to validate the 20 Hz fin whale pulse detection procedure: comparison to a commonly used template matching method and comparison to a state-of-the-art ANN based system on an unseen dataset.

TK filter (sperm whale clicks)

The chosen baseline for the sperm whale click detection is inspired from the work of Ferrari [55]. It makes use of the TK energy operator to find impulses, before filtering them by an estimation of the background noise with a rolling median.

This algorithm was used on the whole dataset of sperm whale clicks for comparison with ANN performances. Using the maximum energy value of samples as prediction, the AUC score was of 0.86, around 0.07 points below most of the trained depth-wise models (Fig. 5.6). This translates to, for instance if we fix

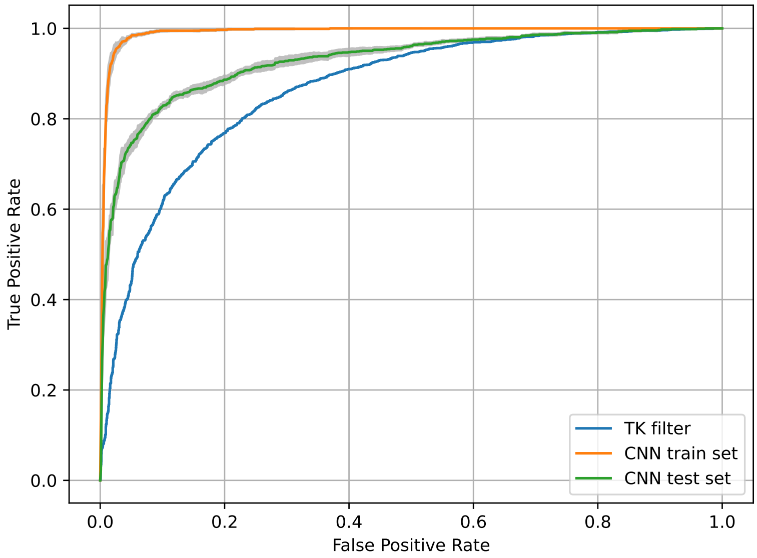


Figure 5.6: ROC curves for the sperm whale click detection task. Performances are given for the TK filter (baseline) and for the 5 initialisations of the 3 layer depth-wise architecture (median ± standard deviation).

a 10% fall-out (false positive rate), a precision of 62% for the TK filter, against 82% in average for the depth-wise models.

Different base for spectrogram computation

Through numerous research, the scientific community has looked for alternatives to the Fourier transform as feature extraction before the main neural network. The sinus base the FFT offers seems too generic, not suited for particular signals such as the transient sperm whale clicks. Experiments were thus conducted using the sincnet frontend proposed by Ravanelli and Bengio [157] which is based on cardinal sinuses with trainable cut frequency. Performances never exceeded 0.86 of AUC on the sperm whale click database (6 points below average performances of FFT based models).

Template matching (20 Hz fin whale pulses)

As mentioned in section 2.2.1, spectrogram correlation is a common approach for cetacean signals detection, especially for mysticetes. To compare our ANN system with this baseline, we built a template of fin whale 20 Hz pulse by averaging the Mel-spectrogram of all annotated pulses in the training set. We then threshold on the cross-correlation product of samples with the template. The resulting detection performances are presented in Figure 5.7. The AUC of the template matching method is 0.898 (5 to 10 points less than the CNN model, depending on the fold).

Larger ANN architecture (20 Hz fin whale pulses)

The dataset published by Madhusudhana et al. [120] which also studies a CNN based fin whale 20 Hz pulse detection seems relevant to test this thesis' proposed system on foreign data. The resulting mAP and peak F1-score are 0.96 and 0.88, when the best overall performances of the study are 0.95 and 0.91 respectively (note that the dataset published is only a subsample of the dataset used in the study, and thus scores are not reliably comparable). This demonstrates that the proposed model generalises well to new data, with scores comparable to a larger architecture that exploits the sequentiality of the pulses.

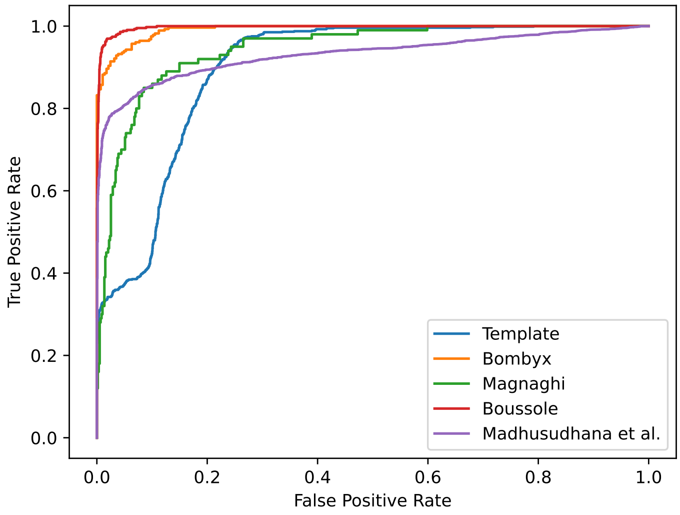


Figure 5.7: ROC curves for fin whale 20 Hz pulse detection over each test set (the two remaining sources serving as training set, see section 4.6 for details). Performances of the template matching method and over the dataset published by Madhusudhana et al. [120] are also displayed.

Conclusion

To challenge this thesis' choice of architecture, handcrafted algorithms, a different frontend, and tests on foreign data were implemented. All results comfort the fact that the FFT based depth-wise architectures are successful at the task, and that with a relatively low computational cost, they show better performances than handcrafted algorithms.

5.3 Deeper and wider models

The remaining target signals treated in this thesis present more variability than sperm whale clicks and fin whale 20 Hz pulses. Larger architectures than simple 3 depth-wise convolutions were thus experimented. We followed the community by opting for the ResNet architecture, widely used in image and sound classification tasks, and the most used for bioacoustics applications [186].

Note that when using ResNet architectures, the last layers consist of an average pooling of the spatial dimensions, followed by a fully connected layer (with the number of output channels set to the number of target classes). In bioacoustic applications, it is often more convenient to yield a sequence of predictions through time rather than one prediction regardless of the size of the input spectrogram. To retrieve this behaviour while conserving the main ResNet architecture, one can discard the average pooling and replace the fully connected layer by a 1x1 convolutional layer (kernel of size 1).

During training, the sequence of predictions can be max-pooled before the loss computation. Max pooling is more suited than average pooling for detection (or multi-label classification) tasks since we want the prediction to be invariant to the amount of void surrounding a target signal. In other words, whether there is one or 10 calls in the input, the detection should be the same: it denotes the presence of at least one event in the window. Note that when using a max-pooling layer, during back-propagation, only the temporal frame with the maximal prediction serve the gradient computation.

In this section, experiments study the effect of the choice of frontend (especially spectrogram range compression), architecture (among ResNet-18, ResNet-50 and sparrow [77]), training hyper-parameters and evaluation metric. In these regards, it intends to assist decision making, by discussing on their impact to solve two detection tasks (orca calls and Antarctic mysticetes calls).

5.3.1 Hyper-parameter search for orca call detection

Contrary to the smaller architectures aforementioned, heavier models need around 1min per epoch on the orca call detection dataset (see section 4.6). An automatic hyper-parameter search was thus employed using Async Successive Halving Algorithm (ASHA) [113], implemented by the Ray python package [131]. It uses the hyperband algorithm with successive halving to explore the hyper-parameter search space, with aggressive early stopping of low performing models. Moreover, to optimise computations, models with plateauing performance are also stopped rather than trained until the maximum number of epochs is reached.

Hyper-parameter combinations are drawn from the following search space:

- Learning rate (log uniform distribution between 0.00001 and 0.1)
- Weight decay L2 loss (log uniform distribution between 0.00001 and 0.1)
- Batch size (sampled uniformly from [8, 16, 32, 64, 128])
- Weighting of positive samples in the loss computation (uniform distribution of integers between 1 and 5)
- Brown noise data augmentation (boolean)
- MixUp data augmentation (boolean)
- SpecAugm [143] spectral data augmentation (boolean)
    - maximum frequency dilation for SpecAugm (uniform distribution between 1% and 30%)
    - maximum temporal dilation for SpecAugm (uniform distribution between 1% and 30%)
    - maximum mask height (number of frequency bins) for SpecAugm (uniform distribution between 10 and 50)

  - maximum mask width (number of time bins) for SpecAugm (uniform distribution between 10 and 50)

Several architectures are studied: sparrow [77] (simple VGG-like model) and ResNet-18 models (one randomly initialised and one pretrained on ImageNet noted 'resnetPT'). For each of the 3 possible architectures, logarithmic and PCEN spectrogram range compression were tested, yielding 6 independent hyper-parameter searches. The searches were ran independently in order to have a fair comparison of the 6 types of models: each have their hyper-parameters optimised via a systematic procedure with a fixed computational budget.

The main objective of this study is to compare architectures on their best possible performance on the test set (both same antenna and different antenna). This is why no validation set was kept apart, and the whole test set mAP was used for early stopping (both low performing and plateauing trainings), and making halving decisions.

Nonetheless, in the following, scores of the two sets are reported separately. Indeed, we will see that a change in recording system (with a different frequency response) can introduce a performance drop. To emphasis on this generalisation problem, we report performance separately on a close test set (same antenna than seen in training) and a foreign test set (different antenna).

The search algorithm was run with 100 trials, for all architectures and range compression combinations independently. This allows for a fair comparison of the architectures, each having their hyper-parameters optimised in a systematic way.

Figure 5.8 summarises the resulting performance of the 100 trials for the two test sets. The sparrow architecture appears more resilient to the choice of hyper-parameters, especially with the PCEN range compression. The pcen-sparrow models reach the best scores, with an especially strong performance gain on the foreign test set (different antenna), demonstrating generalisation capabilities. These findings will be further studied in section 5.3.1, with repeated initialisation with the best set of hyper-parameter for each of the architectures.

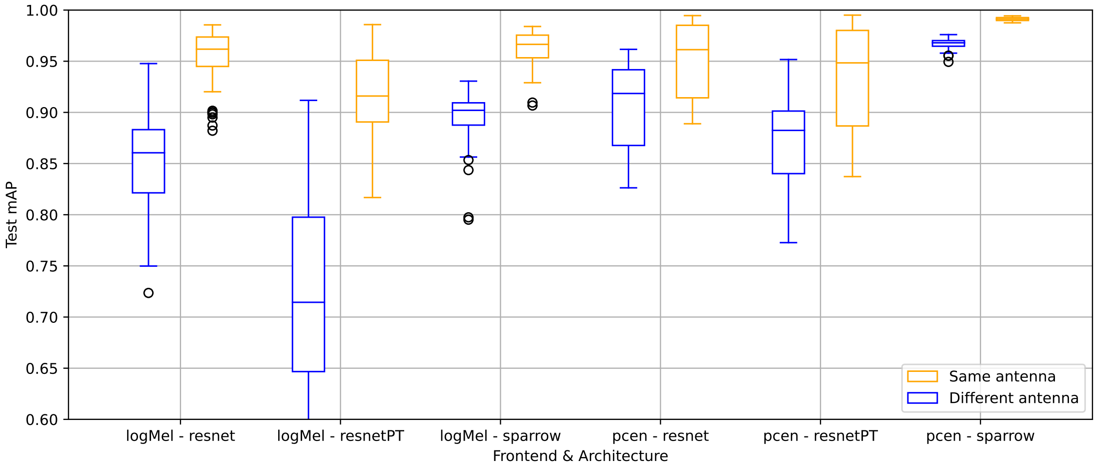


Figure 5.8: Test mAP for the two test sets of orca call detection. Scores of the 50 best trials os the ASHA search are given for each combination of architecture and spectrogram range compression.

Impact of hyper-parameters on model performances

To learn insights from this systematic search, correlations were measured between hyper-parameters and the resulting model performances.

archi	posweight	batchsize	lr	augm	mixup	brownnoise
logMel - resnet		-0.240		False 0.37		
logMel - resnetPT					False 0.06	
logMel - sparrow		-0.216	0.371	False 0.25		True 0.16
pcen - resnet				False 0.06	False 0.08	
pcen - resnetPT				False 0.10		
pcen - sparrow			0.312			

Table 5.1: Statistical analysis of the impact of hyper-parameters on model performances (test mAP). For numeric variables (posweight, batchsize, and lr), the Pearson correlation was computed, and its coefficient is reported for p-values < 0.05. For boolean variables (augm, mixup, brownnoise), the Kruskal-Wallis H-test was computed, and the beneficial value along with medians difference are reported for p-values < 0.05. Empty slots denote p-values below 0.05.

Table 5.1 reports the statistically significant hyper-parameters on the end model performances (p-value < 0.05). Hyper-parameters appeared to have identical impacts on the same antenna and different antenna test sets, and thus the analysis was conducted on the combination of the two. This representation yields several insights:

- Smaller batch sizes can improve generalisation. This is consistent with the study by Kandel and Castelli [96]. It is especially relevant for small datasets, where large batch sizes imply a reduced variability of batch compositions which can yield overfitting models.

- As for the learning rates, several biases have to be taken into account. A small learning rate implies slower training, and thus could be early stopped by the search algorithm before they would plateau to their top performance. Moreover, if selecting the learning rates above 0.001, the Pearson correlation coefficient changes sign with a higher p-value ($r = -0.1$, $p_{value} = 0.06$).
- SpecAugment surprisingly not only does not improve generalisation but reduces it, despite the joint optimisation of augmentation strength. This is presumably related to the underfitting problem reported by the SpecAugment authors [143]. Indeed, data augmentation can make learning 'harder', and thus demand longer trainings and / or heavier models. Note that longer trainings are especially disadvantageous in this paradigm of hyper-parameter search with early stopping.
- Other hyper-parameters do not have a clear significant impact on end performances.

Search findings validation

Frontend **Architecture**	logMel resnet	logMel resnetPT	logMel sparrow	PCEN resnet	PCEN resnetPT	PCEN sparrow
Batchsize	8	8	128	64	128	32
Learning rate	8e-3	7e-4	2e-3	2e-2	1e-2	4e-2
Weight decay	4e-4	9e-3	8e-5	1e-2	1e-3	2e-2
Posweight	4	3	1	5	3	1
Brown noise	False	False	True	True	False	True
SpecAugment	False	False	False	False	False	True
MixUp	False	False	True	False	True	True
# epochs	6	13	9	5	6	5
Same antenna	0.98	0.97	0.98	0.99	0.99	0.99
Different antenna	0.95	0.90	0.91	0.96	0.95	0.98

Table 5.2: Best scoring hyper-parameters resulting from the ASHA search of 100 trials for each frontend / architecture combination. Corresponding mAP scores are given for the two test sets.

To follow up on this hyper-parameter exploration and validate its findings, using each architecture's best scoring hyper-parameters (see Tab. 5.2), 5 training procedures were run with random initialisation. Performances of the latter are displayed on Fig. 5.9. These results reveal several insights:

- The pretrained ResNet ('resnetPT') shows a lower performance than its random initialised relative. For that matter, it is worth mentioning that the first convolutional layer had to be replaced prior to training (switching from a 3 channel input to a single channel input). As a result, the pre-learnt projection at initialisation might be dysfunctional, and even counterproductive for final convergence.
- For the remaining architectures (ResNet and sparrow), PCEN yields a performance more resilient to random initialisation (smaller variance), and show significantly improved performance. This will be studied in greater details in the next section.
- Comparing sparrow and ResNet given PCEN normalised spectrograms, sparrows gives a more stable higher performance. One possible explanation for this is the total number of weights of the architectures. Sparrow has around 300k trainable parameters, and the ResNet-18 has 11M. With a relatively small datasets like this one, smaller models might decrease the risk of overfitting.

PCEN beneficial behaviour

The PCEN range compression procedure appeared to be beneficial with some but not all datasets. For the orca call detection task, it appeared to be beneficial (Fig. 5.10). To verify the significance of the impact of PCEN, a statistical analysis was run to compare the two distribution of scores. To discard low performing models that were early stopped by the search algorithm, only the top 50% of the scores were kept for each distribution.

The two distribution were significantly different (Kruskal-Wallis H test, p-value < 0.001) and the gain in performance was higher for the test set from the different

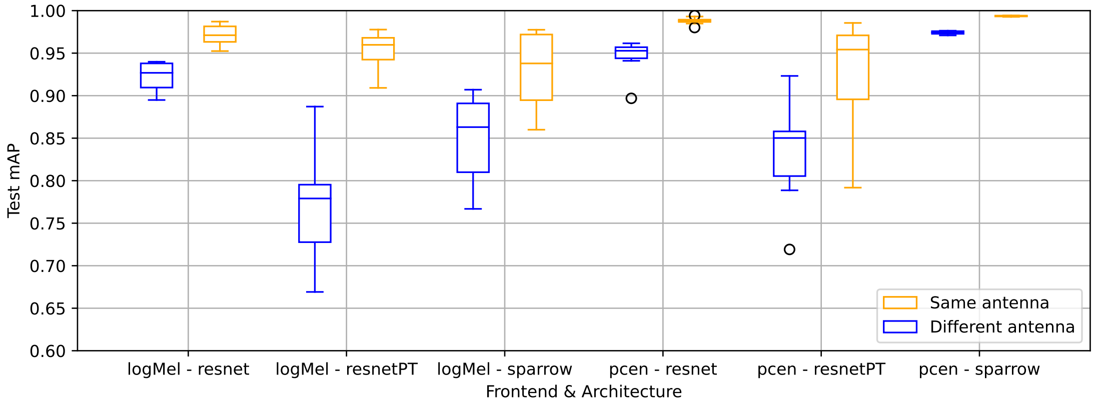


Figure 5.9: Distribution of performances after 5 runs on the best scoring hyper-parameters of each architecture. Best scoring hyper-parameters were tuned systematically using the ASHA algorithm for 100 trials on each architecture independently.

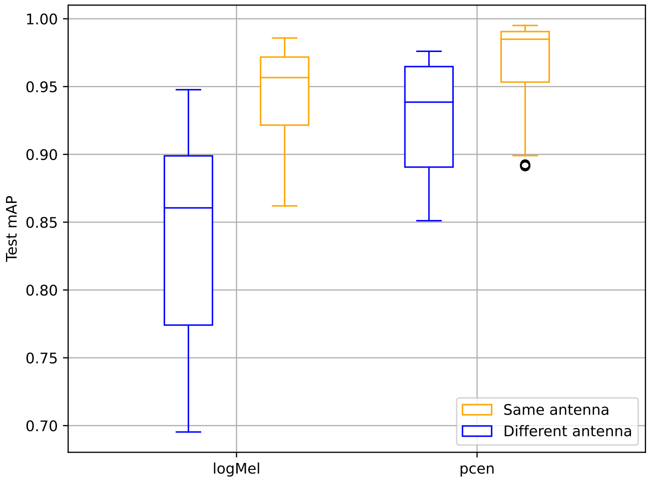


Figure 5.10: Distribution of performances on the orca call detection task depending on spectrogram range compression. Scores are taken from trials of the systematic ASHA hyper-parameter search (all architectures are grouped together). For reach frontend, only the top 50% scores are reported.

antenna (median gain of 0.03 and 0.08 of mAP for the same antenna and the different antenna test sets respectively).

The trainable parameters (s, δ, α and r) remained stable around their initialisation value for a large majority of the training runs. This was not the case during experiments with other datasets such as the Antarctic blue and fin whale vocalisations, where the PCEN parameters appeared to diverge towards irrelevant values (see section 5.3.2). On this orca call detection dataset however, PCEN significantly improves generalisation, especially facing domain shift (foreign test set). This result is consistent with the study by [4] on humpback whale vocalisation detection.

5.3.2 Experiments on a large public dataset (Antarctic mysticetes)

This work has been subject to a workshop intervention [21].

The Antarctic mysticete dataset (introduced in section 3.2.2) offers two main opportunities: its public aspect allows a common mean of evaluation for detection systems among researchers, and its large size enables this evaluation to be the most relevant. Indeed, as Table 4.8 summarises, annotations come in large numbers (close to 80k in total, 2.5k for the least represented class) and are spread across multiple recording locations, devices and years. As discussed earlier in section 2.1.2, this gives us a chance to learn robust models and measure their generalisation capabilities.

Spectrogram	**Architecture**	**SpecAugm**	**Train mAP**	**test mAP**
logarithm	sparrow	no	0.47	0.37
logarithm	ResNet-18	no	0.86	0.54
logarithm	ResNet-50	no	0.84	**0.66**
PCEN	ResNet-50	no	0.82	0.57
fixed PCEN	ResNet-50	no	0.80	0.58
logarithm	ResNet-50	yes	0.70	0.60

Table 5.3: Experiments on spectrogram range compression, architecture, and data augmentation for the detection of Antarctic mysticetes calls. mAP scores are computed over each class independently before averaging to ignore class imbalance.

With this dataset at hand, several architectures were first experimented, with trials on different spectrogram range compression and data augmentation. They are summarised in Tab. 5.3, and demonstrate several insights:

- Non residual architectures such as sparrow don't have the capacity to learn even the training set,
- The larger architecture (ResNet-50) generalises better to the test set,
- Spectral data augmentation produces underfitting,

- PCEN normalisation, whether with trainable or fixed parameters, decreases generalisation.

The next section will try and get a sense of the latter insight which goes against the observations on the orca call detection dataset (section 5.3.1).

PCEN unfavorable behaviour

A reasonable hypothesis of why PCEN appears counter productive is that it filters the long stationary signals of the blue whale (10 to 15 seconds long). In PCEN, the s parameter describes the coefficient of the IIR filter, which yields the smoothed version of the spectrogram $\mathbf{M}$. $\mathbf{M}$ is then used to withdraw background noise from the input $\mathbf{S}$ (Eq. 5.1).

Accounting for this, we want the IIR to have a high enough time constant $\tau = \frac{-1}{\log(1-s)}$. Indeed, the time constant of a filter is the time it needs to reach $1-\frac{1}{e} \approx 0.63$ given an logical gate input [115] (we could make the analogy with the blue whale calls being logical gates on their frequency bin). Using this relationship, with $s = 0.01$, it takes 13 seconds for $\mathbf{M}$ to integrate 63% of the energy of $\mathbf{S}$. Figure 5.11 illustrates this effect of s on PCEN normalisation and compares it to the log compression.

This value of $s = 0.01$ seemed sufficient to avoid withdrawing too much of the blue whale calls, and was used to train a model with a non-trainable ('fixed') PCEN. On the other hand, the intuition is that if a better value exists, the trainable s would converge to it during optimisation.

Unexpectedly, the trainable PCEN s parameter converged towards 0.9, an almost instantaneous smoothing coefficient, high enough to integrate blue whale calls in the smoothed spectrogram $\mathbf{M}$ and subtract them from $\mathbf{S}$. The other trainable parameters α, δ, and r converged around 0.94, 1, and 0.94 respectively. Considering these parameters (the smoothed spectrogram $\mathbf{M}$ being approximately equal to $\mathbf{S}$ with $s \approx 1$), the PCEN equation can be rewritten as Eq. 5.1.

$$\mathrm{PCEN}_{t,f} = \left(\frac{\mathrm{S}_{t,f}}{(\epsilon + \mathrm{M}_{t,f})^{\alpha}} + \delta\right)^{r} - \delta^{r} \approx \mathrm{S}_{t,f}^{0.06} \tag{5.1}$$

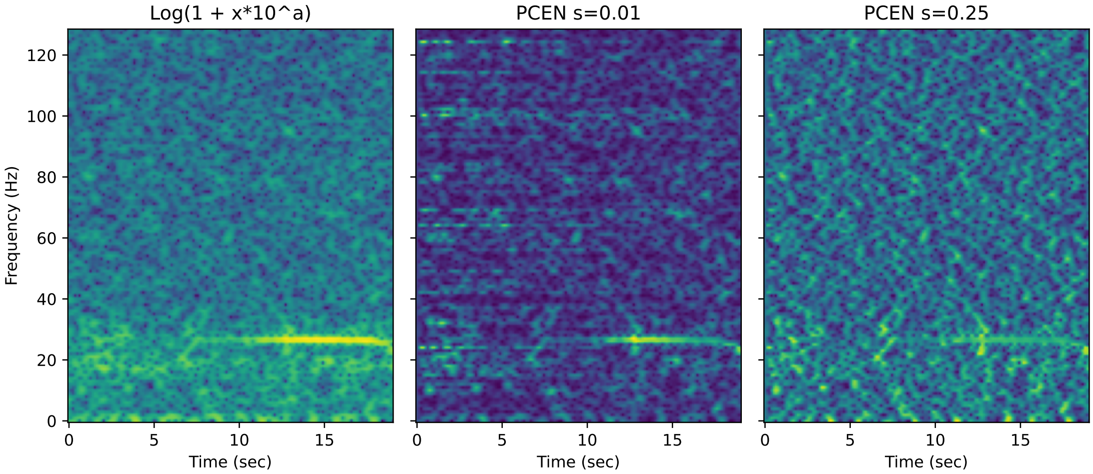


Figure 5.11: Comparison of the different range compression approaches. All spectrograms come from the same sample containing a Bm-A call. For log compression, a converged to 0.3 during training. For PCEN, we show how a too high value for s can lead to the reduction of some target signals. The remaining PCEN parameters were left to the default values proposed by Wang et al. [199].

As for the fixed version, the smoothing parameter was set to $s = 0.01$, corresponding to a 13 sec time constant. It yielded a significant decrease of performance on the test set (10 points of mAP). Trials were conducted with several other values ($[0.001, 0.0025, 0.005, 0.01, 0.025, 0.05, 0.1]$) and the maximum performance was reached with $s = 0.025$ (reported in Tab. 5.3).

These experiments demonstrate that PCEN does not always yield performance gains: it depends on the signals to detect and the noises surrounding them. Also, even if choosing a reasonable parameter s tuned for the target signals, performances might be lowered. This is perhaps explained by the difference in compression compared to the trainable log approach [169]. Experiments should thus be conducted on each task before choosing this spectrogram range compression method.

Another insight on PCEN behavior was yielded by late experiments with the classification of humpback whale sounds (they are preliminary results not reported in this thesis). PCEN was beneficial to detect humpback whale calls, but appeared detrimental to classify then by call type. Put in perspective with the beneficial impact on orca call detection and the opposite effect on the Antarctic mysticete dataset, an hypothesis could be that PCEN hinders performance in mutli-class and multi-label datasets.

Study of performance metrics

After the selection of the best performing model (ResNet-50 with logarithmic range compression), the mAP remains quite low as compared to the AUC (0.11 against 0.99 for Bm B calls for instance, see Tab. 5.4). This is due to the high imbalance of the dataset (ratio close to 50 between amounts of positive and negative samples).

	Bm A	**Bm B**	**Bm Z**	**Bm D**	**Bp 20 Hz**	**Bp 20+**	**Bp DS**
Train AUC	0.99	0.99	0.99	1.00	1.00	1.00	1.00
Train mAP	0.92	0.74	0.75	0.98	0.95	0.96	0.93
Test AUC	0.97	0.91	0.96	0.97	1.00	1.00	0.99
Test mAP	0.73	0.11	0.55	0.83	0.94	0.61	0.86

Table 5.4: Detection performance of the top performing model on the Acoustic Trends dataset (calls from *Balaenoptera Musculus* and *Balaenoptera Physalus*). The model is a Resnet-50 with logarithmic spectrogram range compression trained without SpecAugment.

Indeed, the mAP uses the precision, which normalises true positives by positive predictions, whereas the AUC uses the specificity, which normalises true negatives by negative samples. For a dataset with mostly silent sections like the Acoustic Trends dataset, the AUC will thus be over-optimistic, and the mAP will be over-pessimistic. This motivated to experiment on a different, more informative metric: the number of false positives per hour, previously used by Shiu et al. [173] on automatic cetacean PAM systems.

Figure 5.12 summarises the number of false positives per hour against the recall for each class and data source. It shows how for some calls, the performance is significantly impacted by the data source. This can be explained by a difference in background noise, average SNR of the annoted calls, or both. Moreover, the curve for Bm B calls in the Kerguelen 2005 data confirms the low score reported in Table 5.4, probably due to the presence of hard samples in the dataset (events that trigger false positive even at high thresholds).

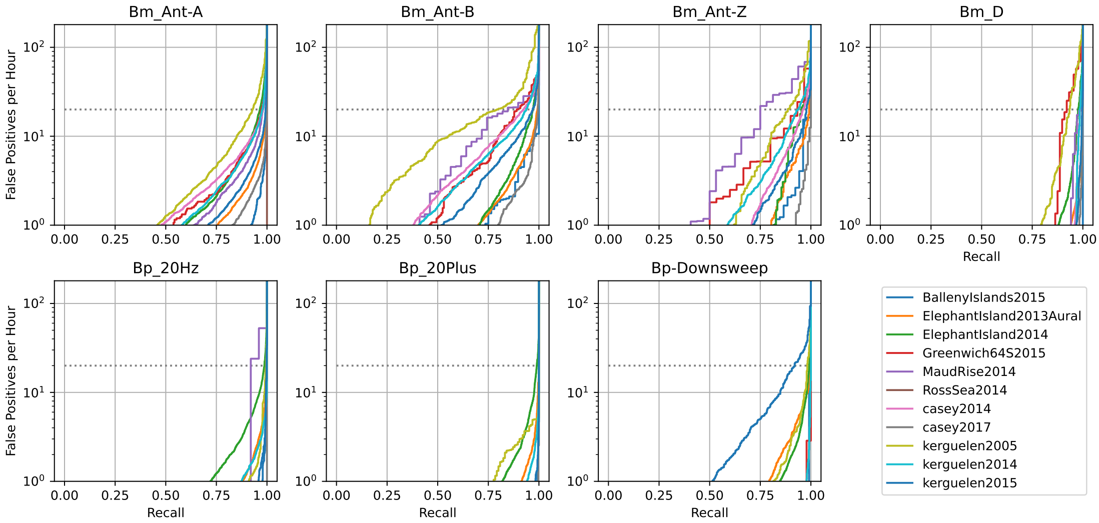


Figure 5.12: Number of false positives per hour as a function of recall. Curves are given for each class and each data source. The dotted horizontal line denotes the 20 false positives per hour threshold.

Data Source	Bm A	Bm B	Bm Z	Bm D	Bp 20 Hz	Bp 20+	Bp DS
Balleny Islands 2015	1.00	1.00	0.98	1.00	1.00	1.00	1.00
Elephant Island 2013	0.99	0.99	0.99	1.00	1.00	1.00	1.00
Elephant Island 2014	0.96	0.97	0.95	0.98	0.99	0.99	0.99
Greenwich 64S 2015	0.97	0.89	0.90	0.91			0.98
MaudRise 2014	0.98	0.82	0.75	0.98	0.92		
Ross Sea 2014	1.00						
Casey 2014	0.98	0.92	0.96	0.99	0.95		
Casey 2017	1.00	1.00	1.00	1.00	1.00	1.00	
Kerguelen 1 2005	0.93	0.79	0.89	0.93	1.00	1.00	0.98
Kerguelen 2 2014	0.98	0.94	0.94	0.98	1.00	1.00	1.00
Kerguelen 2 2015	0.99	0.97	0.98	1.00	1.00	1.00	0.92
All	0.98	0.97	0.98	0.99	1.00	1.00	1.00

Table 5.5: Recall at 20 FP/hr for each class and data source. Cells with less than 20 samples are not reported.

Table 5.5 summarises these curves once more by reporting the recall at which there are 20 false positives per hour. Indeed, Shiu et al. [173] argues that this threshold is the maximum acceptable for quality control processes.

These results emphasis the importance of the choice of performance metric. It needs to account for the datasets' class imbalance, and for the subsequent application needs. In the absence of the latter, the recall at 20 false positives per hour seems a good generic, for its stability facing class imbalance and its high interpretability for production use (other thresholds than 20 can be chosen, depending on project needs).

5.4 Resulting detectors performance

After exploring several ANN architectures on datasets of different characteristics (target signals, amount of annotation, diversity of recording systems), this section intends to get an overview of the resulting detection systems.

Best configurations were kept for each task to report performances. When multiple runs were operated (20 Hz fin whale pulses and sperm whale clicks) the median values are reported. As for the fin whale 20 Hz pulses, since 3 test folds were studied, the median gathers the 5 runs of the 3 folds.

Target signal	Archi	AUC	mAP	Rec(20FP/hr)
Fin whale 20 Hz pulses	3 depth-wise	0.99	0.84	0.94
Sperm whale clicks	3 depth-wise	0.93	0.85	0.65
Dolphin whistles	sparrow	0.98	0.86	0.61
Humpback whale calls	sparrow	0.99	0.99	0.97
Orca calls	sparrow	0.99	0.98	0.87
Antarctic mysticetes	ResNet-50	0.97	0.66	0.93

Table 5.6: Summary of performances for all trained detection systems on their test set (see section 4.6). Reported metrics are, from left to right, area under the receiving operating characteristics curve, area under the precision recall curve, and recall at 20 false positives per hour.

Low complexity architectures such as 3 depth-wise convolution layers suffice in learning to detect low variability signals such as fin 20 Hz pulses and sperm whale clicks. To increase the precision, several detections can be integrated in larger temporal windows, either with handcrafted rules (discussed with the detection of fin whale songs in section 7.2) or with learnt sequential models as proposed by Madhusudhana et al. [120].

The sparrow architecture allows to learn more variable signals, as it was originally designed for bird classification [77]. It is able to yield satisfactory performances despite a reduced amount of labels.

Then, when larger amounts of annotations are available, the ResNet-50 architecture originally designed for image classification can be used to detect multiple calls (Antarctic mysticetes), sharing the same model embedding before discrimination.

Neither sparrow nor ResNet-18 architectures had the capacity to solve this task as the ResNet-50 did.

This thesis' work in annotation and training binary classifiers thus resulted in successful detection systems for 13 different target signals (the Antarctic mysticetes model gathering 7 different signals). The satisfactory performances, especially on test sets that were designed to reflect generalisation capabilities, allow to consider using these trained models in production. Indeed, as we will see in the next chapters, the models served to analyse databases of several thousands recorded hours

5.5 Contrastive learning for orca call classification

A second axis of work conducted on training procedures was applied to a classification task for orca call types. Indeed, call types have been attributed discrete classes, and have served in behavioural and social structure studies [65, 66]. These studies were done by manually annotating calls, a time consuming task that we try to automate here.

This task implied to use other losses than the BCE (the only loss function used so far). Motivated by the lack of annotations, experiments with unsupervised algorithms were first conducted, and are reported in the first part of this section. Then, as annotations were progressively gathered, performances of a semi-supervised learning algorithm are compared to a traditional supervised learning procedure.

5.5.1 Trials with deep representation learning algorithms

Given the large amount of detections of orca calls (section 5.3.1) and the lack of call type annotation, unsupervised learning approaches have been experimented with. This call type classification task comes down to classifying similar pitch patterns together, which fits with the contrastive learning paradigm. Indeed, learning a representation that ignores small distortions of shapes (time and frequency shifts and dilations) seems appropriate: these distortions are found among instances

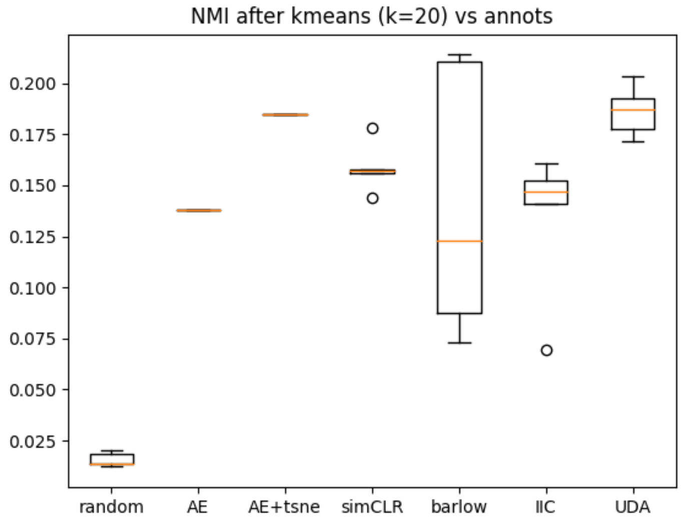


Figure 5.13: Distribution of NMIs between clusters found using k-means on learnt embeddings and labels (5 training initialisation for each algorithm). It needs to be taken into account that annotations were made by filtering some clusters proposed by the AE and t-SNE. Here, for a comparison of deep metric learning, Unsupervised Data Augmentation (UDA) was trained only with its unsupervised loss.

of a call type. Once such a representation has been learnt, supervised learning can be operated using a small amount of labels to optimise discrimination boundaries (fine tuning).

As mentioned in section 2.1.2, numerous algorithms have been proposed in the literature to learn from sparsely annotated datasets using contrastive learning. They mainly differ by the distance metric they use in their embedding space. In search for the right one, papers were in part selected for their top position on the CIFAR-10 with 1000 labels benchmark [105], as it contains a number of classes and labels that is similar to the dataset at hand. Experiments were thus conducted with SimCLR [31], UDA [209], Barlow twins [211]), and IIC, [180].

In a way reflecting the caveat of modern days deep learning research, a plethora of algorithms were implemented, with limited understanding of their underlying behaviours. Moreover, in addition to their proposed main algorithm, each paper comes with a handful of training tricks which are also responsible for the reported performances. This makes a fair comparison between techniques difficult.

Figure 5.13 reports on learnt representation quality for each of the algorithms implemented. The metric used is the NMI between clustering of the embeddings and their associated label. Barlow twins and UDA, for some initialisation, show a slightly higher NMI than the representation used to annotate (t-SNE projected AE embeddings).

Despite the invested efforts, none of the implemented algorithms (SimCLR, UDA, Barlow twins, and IIC) showed a relevant gain in performance after fine tuning for the classification task (as compared to a random initialisation of weights).

5.5.2 FixMatch versus supervised learning

This work has been subject to a workshop intervention [19].

With the progressive collection of labels, semi-supervised learning approaches became more and more relevant. Again, several algorithms were experimented with: Meta Pseudo Labels [151], UDA [209], mixMatch [15] and fixMatch [180]. However, selected for its good loss convergence, reasonable performances, and very few training tricks needed, fixMatch was retained for further comparison with the regular supervised approach.

The fixMatch algorithm combines a supervised loss, pseudo labelling, and consistency training in one framework. Pseudo labelling consists in applying a supervised loss on samples without annotation by using the highest prediction of a model (the ‘pseudo label’). This allows to make use of unlabelled data, especially on easy samples (pseudo labels can be retained only if the confidence is above a predefined threshold, see Fig. 5.14). This can be beneficial because it broadens the diversity of data seen by the model without demanding more annotation.

On the other hand, consistency training is the concept of learning a projection that ignores (is ‘consistent’ against) variations in the data. It is very close to the concept of contrastive learning aforementioned. FixMatch makes use of it by applying different levels of data augmentation to its inputs.

Fig. 5.14 shows how data augmentation and pseudo-labelling were combined for the orca call classification task, following the fixMatch approach. $H(p, q)$ denotes the

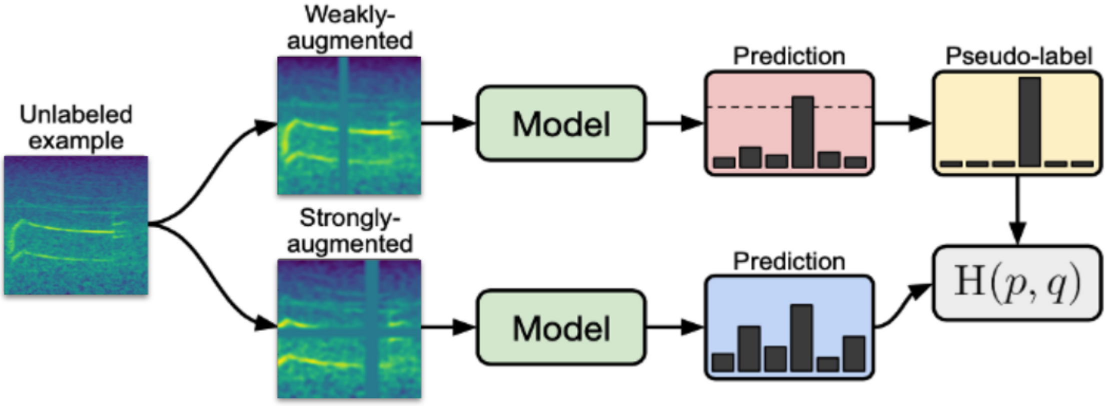


Figure 5.14: The fixMatch algorithm [180], a combination of pseudo-labelling and consistency training. The figure was taken from the original paper, and adapted for the orca call classification task.

cross-entropy between the pseudo label and the prediction after strong augmentation. It represents the unsupervised loss that will be added to the regular supervised cross-entropy loss before the backward propagation.

The main difference with this thesis' implementation is the chosen data augmentation policy. Here, SpecAugment [143] was used (instead of RandAugment [37]). It was applied on PCEN normalised Mel-spectrograms of 2 seconds excerpts, with 128 frequency bins and 128 frequency bins and 262 temporal bins ($f_s = 22,050$, $NFFT = 1024$, $hop = 165$). Strong augmentations allowed until 20% of dilation (time and frequency wise), dropping bands of maximum 20 frequency and temporal bins, and gaussian blurring, whereas weak augmentations capped dilations to 5%, and dropped bands up to 5 bins, without gaussian blurring.

As for the remaining hyper-parameters, (learning rate, cosine scheduling, batch sizes, pseudo-labelling threshold, and loss weighting) they were left as proposed by the paper [180].

	90/10 train/test split		**50/50 train/test split**	
	F1 score	**Accuracy**	**F1 score**	**Accuracy**
Supervised	0.95	0.95	0.94	0.94
FixMatch	0.92	0.94	0.84	0.89

Table 5.7: Comparison of performances for regular supervised learning and semi-supervised learning approaches. Results are given for a regular train/test split, and with a reduced training set (200 samples per class in average).

The resulting performances of semi-supervised and supervised training are compared in Tab. 5.7, with the accuracy computed across all samples and the F1-score being computed for each class independently before averaging. Both were trained with a ResNet-50 and cosine learning rate scheduling.

The results demonstrate a counter productive effect of the unsupervised loss, even when reducing the number of annotations by half (approximately 200 samples per class in average). This might be explained by a too strong augmentation policy which might mask out complete calls in some cases (some calls lie in less than 20 frequency bins for instance). Further work should focus on researching augmentations that are more adapted to the variations found among calls of the same types but without risking to change

6 Application to species conservation

Contents

6.1 Context and objective

Given previously trained detection and classification systems, this section describes how they can be put to production and serve species conservation purposes. Focusing on the sperm whales and fin whales of the Mediterranean sea, a first axis of conservation is the reduction of ship strikes, a significant cause of death for these species evolving in the Pelagos marine mammal sanctuary [142]. Then, the detection mechanisms is run upon the Bombyx long term survey. This yields insights on sperm whale behaviour in relation to anthropic pressure, helping to implement relevant conservation measures in the long term.

6.2 Alert system for collision risk mitigation

6.2.1 Context and objective

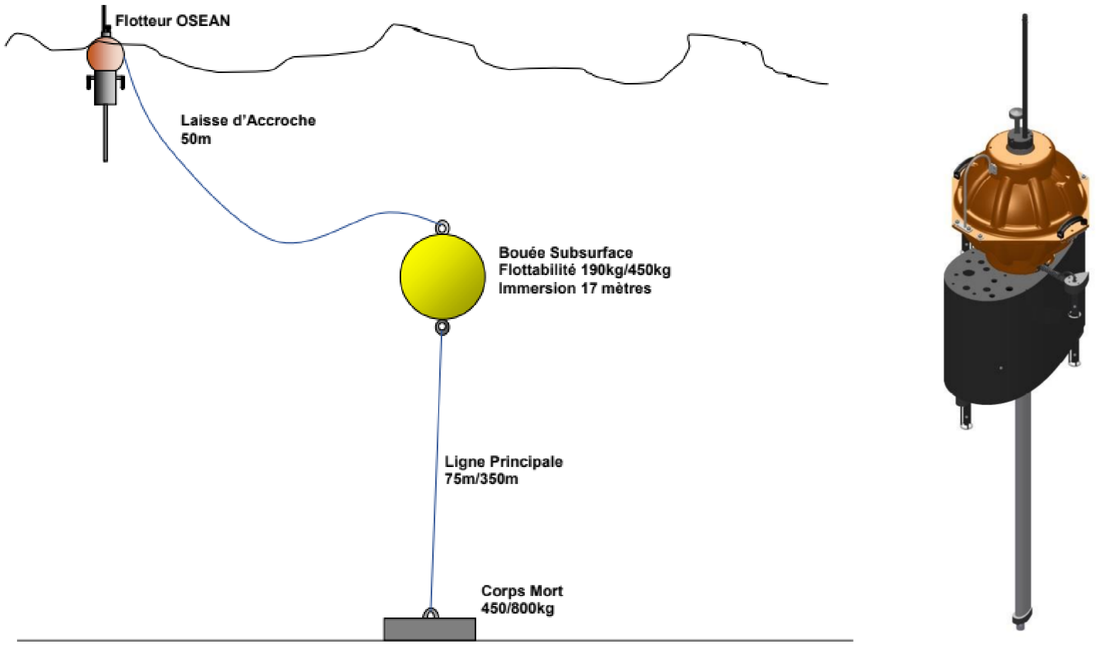


Figure 6.1: Technical plans of the Bombyx 2 system, taken from OSEAN SAS manufacturing report. (left) Mooring system. (right) Pentaphonic acoustic recorder and floatability variation system (total height of 3 meters).

As part of the GIAS project aiming at reducing navigation risks in the Mediterranean sea, the Bombyx 2 buoy was designed, in a collaboration between DYNI and OSEAN SAS. Preliminary work on this project was subject to a conference publication [18]

This buoy is equipped with 5 hydrophones, a floatability variation system, and embedded algorithms for the detection of sperm whale clicks and 20 Hz fin whale pulses. To mitigate surface noise and exposure to strong weather conditions, the buoy parks at 25 meters depth to record and acoustically detect its target species (sperm whales and fin whales). In the event of a detection, the buoy reaches the surface to transmit the alert with supporting data via the mobile network. The alerts then allow ferries of the zone to make decisions to mitigate their risk of collision with nearby whales (reducing speed or changing route for instance).

6.2.2 CNN deployment to an embedded MCU

Section 5.2 introduced low complexity CNNs, especially designed to answer the needs of this alert system. These models, after being trained on GPUs using the Pytorch package [145], were implemented on the embedded system, namely the Microchip PIC32 MCU (integrated on the High-Blue sound card [11]).

This demanded to build a custom interface to export and load architectures and weights via text files. The exports are done in Python, and imported in C (required programming language for the MCU). Design choices were made for the C implementation, for a compromise between flexibility and reduced development effort:

- The model input consists in a Mel-spectrogram,
- Signal length, sampling frequency, window length, hop size, number of Mel-bands, and Mel-frequency boundaries are parametric,
- The architecture consists of successive depth-wise separable convolution layers intertwined with batch normalisation and leaky ReLU,
- The number of layers, and the number of features, kernel sizes and strides for each layer are parametric,
- The last layer is pooled by maximum to yield a global prediction of the signal.

6.2.3 Computation times

Specifications of the input parameters and processing time for the two target signals are given in Tab. 6.1. The longest step is by far the spectrogram computation compared to CNN inference. This comforts the choice of the Fourier transform which offers a fast FFT implementation, rather than others such as the wavelet transforms.

Target signal	Sperm whale clicks	20 Hz fin whale pulse
Signal length (sec)	10	60
Sampling frequency (Hz)	64,000	4,000
FFT window length	512	4096
FFT hop size	256	256
Mel bands	64	64
Mel start (Hz)	2,000	0
Mel end (Hz)	25,000	100
Signal loading (sec)	1	5
Spectrogram computation (sec)	12	26
CNN inference (sec)	4	4

Table 6.1: Specifications and corresponding processing times on the PIC32 MCU, for the detection mechanisms of sperm whale clicks and 20 Hz fin whale pulses.

6.2.4 Detection report

In the event of detections triggered by the CNNs, the buoy is ordered to lift towards the surface to transmit a report supporting the alert. It includes multi-channel chunks of signals (cut surrounding detection peaks), prediction sequences for the two species, and buoy orientation (compass, and magnetometer). These extracts of signals allow experts to confirm the veracity of the alert and to take decisions accordingly. Moreover, the reported extracts being multi-channel (5 hydrophones), triangulation via cross-correlation is possible, increasing the spatial precision of the alert.

The prediction sequences can serve a quick discrimination between false positives, by examining distribution among successive files (Fig. 6.2).

6.3 Long term presence monitoring

In addition to its production use in the context of ship collision mitigation, the sperm whale click detection CNN has been forwarded on the whole Bombyx dataset (3,532 recorded hours from May 2015 to December 2018) for a long term study of sperm whale presence. This work resulted in a journal publication [155], from which some of the results are reported here.

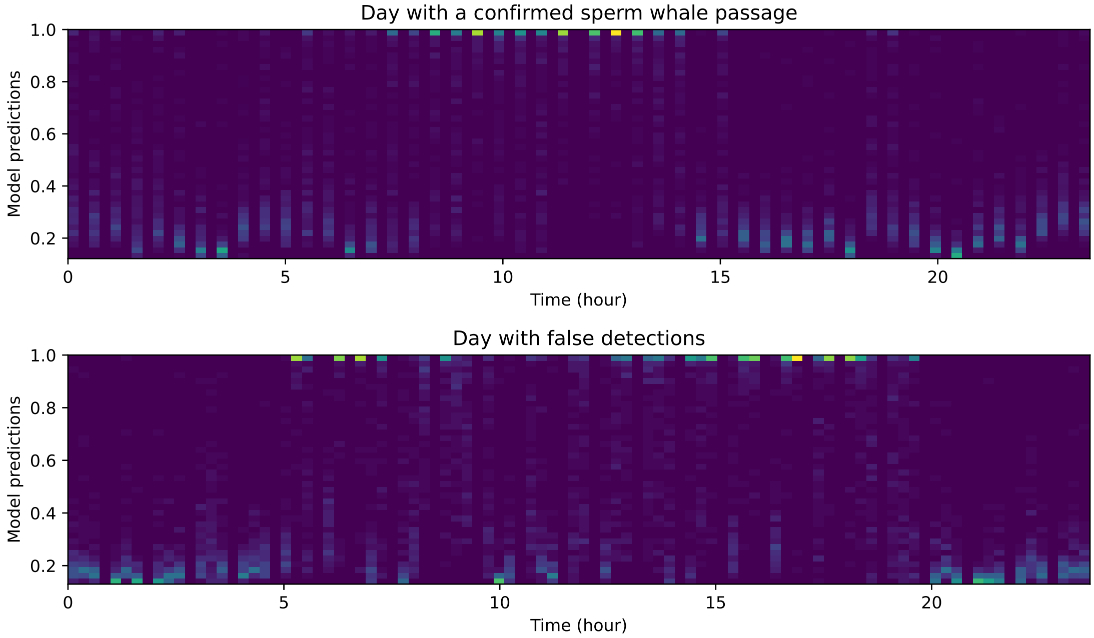


Figure 6.2: Comparison of the distribution of model predictions for a day with a sperm whale (July 7th 2015) and a day with false detections (September 8th 2017).

6.3.1 Sperm whale acoustic presence

A first analysis focused solely on reporting the presence of sperm whales through the recorder years. Files (1min long) with more than 40 CNN predictions above 0.95 were manually verified using the interface described in section 4.3.2. Like so, automatic detections were validated and number of individuals were estimated (inferred from simultaneous click trains and TDOA tracks). This process yielded 57 new sperm whales passages (missed during the annotation procedure described in section 4.3.2), and 25 false positives (including 15 triggered by sound card malfunctions). The notion of passage was used to account for sperm whale presence, considering that clicks belong to the same passage if separated by less than 1h.

In total, 226 sperm whale passages have been recovered, with a total of 347 individuals. Fig. 6.3 presents the number of detected individuals each day during the 4 years of recording. Sperm whales were found all year round, with no statistically significant seasonal pattern. The number of animals per passage varied from 1 to 9 individuals, with a mean duration of 4 hours.

To evaluate dial patterns, the probability of presence was computed for each hour of the day. Grouping probabilities into four periods (Night, Morning, Afternoon, and Evening) demonstrates a statistically significant differences among periods of the day : sperm whales are more present during morning or afternoons than in the evening (Fig. 6.3, Kruskal-Wallis test : p-value < 0.01).

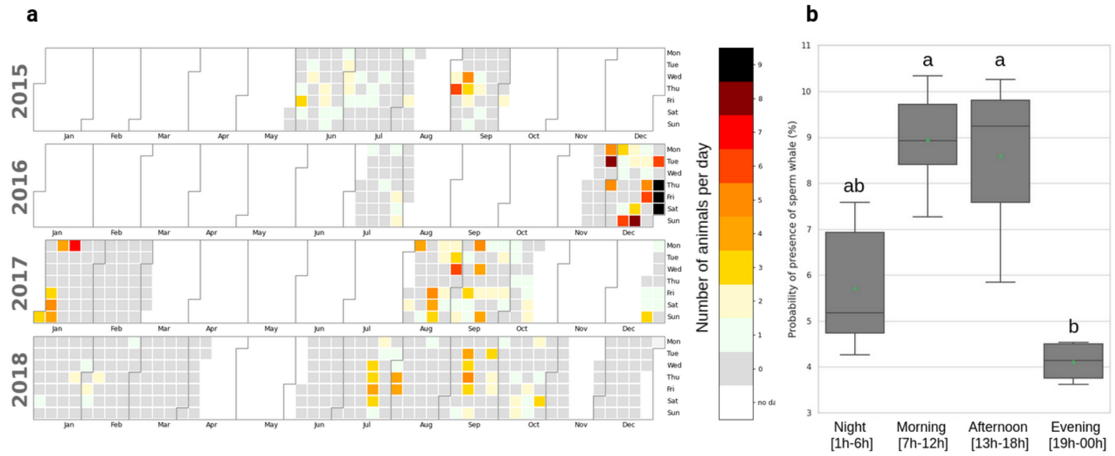


Figure 6.3: Left (a): The Number of detected sperm whales per day during the 4 years of recordings (white region: *no d* = *no data*). Right (b): Distribution of hourly probabilities of presence for each period of the day.

6.3.2 Presence in relation to anthropogenic noise pressure

To assess the performance of the detection system as well as to measure the impact of noise on the presence of sperm whales, the amplitudes of different octave bands were computed and analysed. The distribution of the background noise (octave 800 Hz) through the day is shown in Fig. 6.4. All octaves' dial distributions have the same shape as the 12,800 Hz octave, with the energy peaking around 4am and 9pm.

Ferries cross the study area daily, connecting Toulon or Marseille to Corsica, with scheduled times between 3am - 6am and from 8pm - 9pm. The closest ferry route is approximately 3km away from the antenna. For all octaves, dB amplitudes are significantly higher during ferry schedules (Mann–Whitney test, *p-value* < 0.05), with an average gain of approximately 3 dB.

Moreover, as Fig. 6.4 illustrates, the data shows a significantly lower noise during the sperm whales' presence (Mann-Whitney U=14.44, sample size=300,

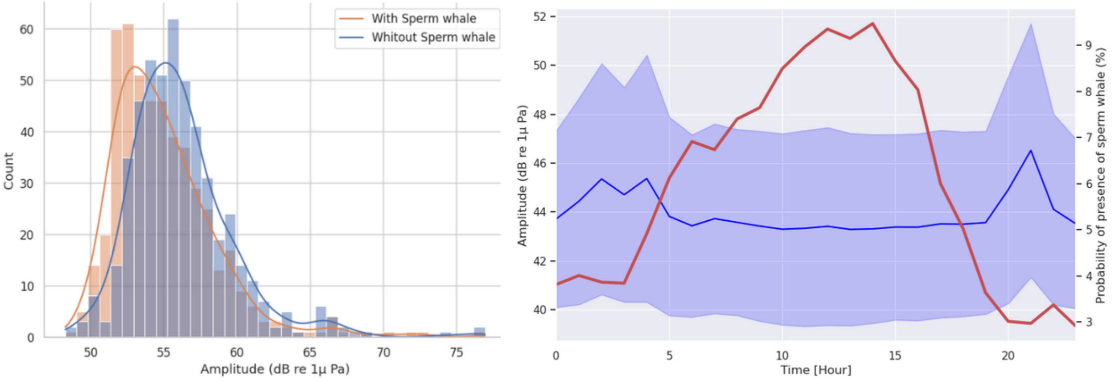


Figure 6.4: (left) Distributions of 12,800 Hz amplitudes during and outside sperm whale passages. (right) Superposition of dial pattern of amplitudes for the octave 12,800 Hz and probability of presence of sperm whales.

p-value < 0.01) for all octaves except 6,400 Hz and 12,800 Hz. This is further demonstrated in Fig. 6.4, where, during 4 AM and 9 PM (noise peaks), the presence of sperm whales is lowest. This last figure also shows that the reduced sperm whale presence is not due to an increased background noise, since sperm whale probability drops before the background noise rises.

6.4 Conclusion

These studies are a first demonstration of the versatility of the detection systems designed through this thesis. Indeed, they can be applied to a real-time alert system to mitigate collision risks, but also in long-term surveys, revealing presence patterns that are crucial in the implementation of relevant conservation measures.

7
Application to communication modelling

Contents

7.1 Context and objective

The previous chapter demonstrated how robust detection systems can be used for species conservation purposes. A second axis of use can also be the study of animal communication systems. In the past, PAM has put forward several examples of song and social communication systems in cetaceans. This allows comparative studies to reflect on the natural evolution of music and language [64]. For that same purpose, robust detection and classification mechanisms are able to analyse large datasets and yield new insights. This is demonstrated in the following chapter with the long term evolution of the Mediterranean fin whale song and communication patterns of the NRKW.

7.2 Fin whale song structure and temporal trends

7.2.1 Context and objective

In parallel to the sperm whale study with the Bombyx dataset, a similar one was conducted on fin whales of the Ligurian sea, again making use of the detection system designed for the GIAS buoy. The trained CNN described in section 5.2 was run over three available datasets : Boussole, Bombyx, and KM3Net (see section 3.2.1). This time, instead of presence monitoring, the study focused on fin whale song patterns, yet poorly documented in the Mediterranean sea. It is also subject to a journal publication [20], which results are reported here.

As other cetacean species, fin whales show geographical acoustic differentiation [83, 130, 29], hypothesised to be cultural in some cases [204]. The divergence of mysticetes songs in different populations is presumably a result of drifts emerging from the conformity and creativity constraints of song production [147]. Moreover, the character displacement theory with songs serving as a discrimination marker for allopatric populations has been hypothised for fin whales of the Northern Atlantic [42]. As for the Mediterranean population, it has been shown to be resident and genetically dissociated from the North Atlantic population [16], and their songs (especially the Inter Pulse Interval (IPI)) were shown to allow for their identification [29, 150]. The Mediterranean fin whales do not follow strict migration patterns or reproduction periods unlike their oceanic conspecifics [137], so their song can be heard all year round.

The base unit of the songs, the 20 Hz pulse, is shared by all fin whales. These pulses occur in sequences that typically last several hours [201], with highly regular pulse intervals between 10 and 40 seconds. The main differentiation of songs lies in the IPI and pulse spectra [194, 80]. Alike fin whales of the Pacific [204, 83], Mediterranean 20 Hz pulses fall into 2 distinct types, one with a slightly higher frequency content than the other [33, 171] (Fig. 7.2). These two categories are sometimes labelled 20 Hz pulse and back-beat, they will be referred to as type A and B for short, with A being the higher pitched pulse. Fin whales of the Pacific and Atlantic often exhibit sequences that alternate between A and B pulses. These are called doublet patterns, as opposed to singlets where only one of the pulse types occur. In doublets there is a strong relationship between IPI and pulse type: two different IPIs are found, one from A to B, and another one from B to A [138, 35, 68, 130, 83]. On the other hand, singlets also follow their own stereotypical IPI.

Mediterranean fin whale songs present more diversity in the consecution of pulse types than simple singlets and doublets (Fig. 7.2). Nonetheless, two studies present stereotypical IPIs. Based on recordings from 1999, Clark et al. [33] observe a link between pulse type and IPI in the Mediterranean sea for two bouts (about 100 pulses). About ten years later, Castellote et al. [29] observe a common IPI around 14.9 sec for that same population, but do not mention its relationship with pulse types.

Besides geographical variations, fin whale song structures also exhibit temporal variations, such as seasonal IPI increases [138, 130], and inter-annual variations of IPI and peak frequency [204]. PAM stations combined with automated analysis (template matching approach) have played a key role in revealing these long-term trends.

Until now, no large scale analysis has been conducted on Mediterranean fin whale songs that could reveal the long-term evolution of their vocal behaviour, which motivates the following study.

7.2.2 Method

Model inference

While the model was trained to detect pulse presence in 5-second segments, the convolutional stack is designed to maintain the temporal resolution of the predictions throughout the network. Discarding the max pooling layer at the end of the CNN, pulse times were retained as the highest predictions above a given threshold within sliding 4 second windows. These timings are approximate up to the size of the receptive field of the network (0.8 seconds).

Thresholds were set at the balance point of the ROC curves (equal sensitivity and specificity). This setting lead to sensitivities and specificities of 0.96 and 0.97 for the Bombyx and Boussole data respectively. For the KM3Net data, since the ROC curve is unknown (no annotation are available), a threshold of 0.12 was chosen so that there is approximately the same proportion of detections as in Bombyx and Boussole ($\approx$ 0.5%) .

Tab. 7.1 summarises the resulting detections, along with a calendar Fig. 7.1. Following Watkins et al. [201], pulses at a distance of less than 45 seconds were considered as being part of the same sequence, and sequences less than 2 hours apart were considered as being part of the same bout.

Data source	**Boussole** [107]	**Bombyx** [74]	**KM3Net** [1]	**Total**
Location	South of Sanremo	Port-Cros Island	Cap Sicié	
Recording year	2008-2009	2015-2018	2020-2021	
Recorded time (hours)	1,860	3,291	1,124	6,275
Detection threshold	0.15	0.68	0.12	
Pre-filtering detections	52,863	83,583	9,684	146,130
Detected pulses	1,647	2,827	657	5,131
Detected A pulses	1,411	2,554	322	4,287
Detected B pulses	236	273	335	844
Detected sequences	246	615	58	919
Detected bouts	51	214	11	276

Table 7.1: Summary of recording characteristics and automatic detections for each source of data.

Spectro-temporal pulse analysis

Following the detection process, a signal processing analysis was conducted to precisely describe each pulse (exact time position, center frequency, bandwidth and SNR). This yields the necessary data to search for song patterns, as shown in Figure 7.2.

For this analysis, an 8 sec window surrounding the prediction peak is selected ($T = [0, 8]$), band-pass filtered (Butterworth of order 3 between 10 Hz and 30 Hz), and resampled at 250 Hz. The STFT is then applied to the resulting signal (Hann window of 256, $NFFT = 1024$, and $hop = 8$) resulting in spectral and temporal resolutions of 0.24 Hz and 0.03 sec respectively.

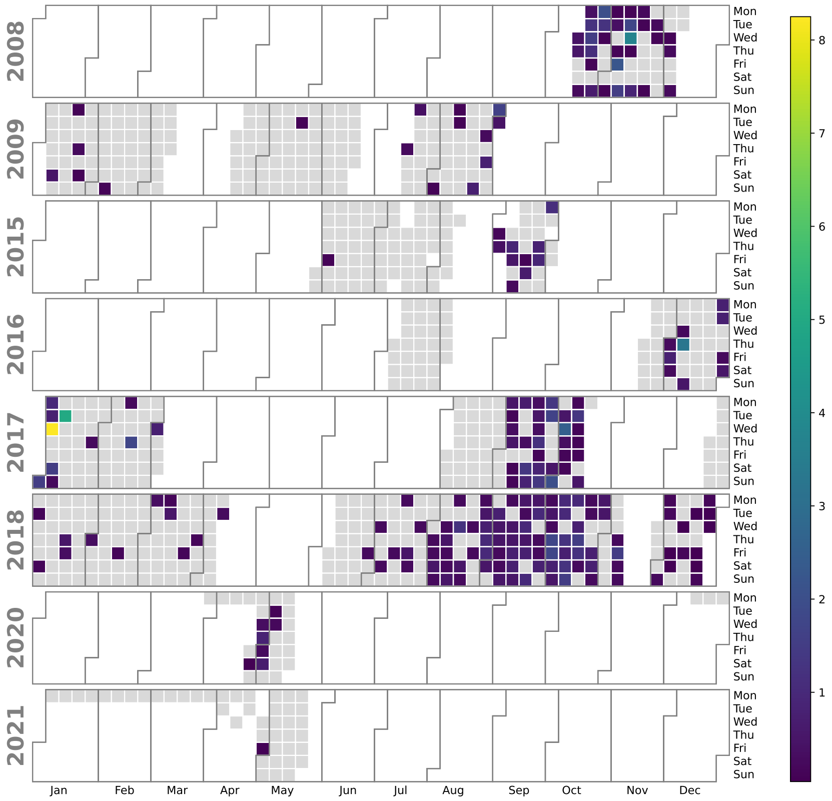


Figure 7.1: Number of detected sequences for each day with recordings, normalised by the amount of recorded hours. Grey cells denote days with recordings but no detection.

From this spectrogram, the precise time position of the pulse $\hat{t}$ is first estimated by selecting the column of the maximum value in the 18-22 Hz frequency band (Eq. 7.1). This value will be kept for IPI measurements.

$$\hat{t} = \underset{t \in T}{\operatorname{argmax}} \left(\max_{f \in [18,22]} (\mathbf{S}_{f,t}) \right), \tag{7.1}$$

To measure the spectral envelope of the pulse, a 1.2 sec window around $\hat{t}$ is max-pooled time wise. Background components are withdrawn (to focus on the pulse spectra only) by subtracting an estimate of the background spectrum: the median of each frequency bin within the window T (Eq. 7.2). Doing so, effects such as the impact of SNR on peak frequency and bandwidth (observed by Helble et al. [83]) are mitigated.

$$E(f) = \max_{t \in [\hat{t}-0.6, \hat{t}+0.6]} (\mathbf{S}_{f,t}\ ) - \underset{t \in T}{\operatorname{median}} (\mathbf{S}_{f,t}) \tag{7.2}$$

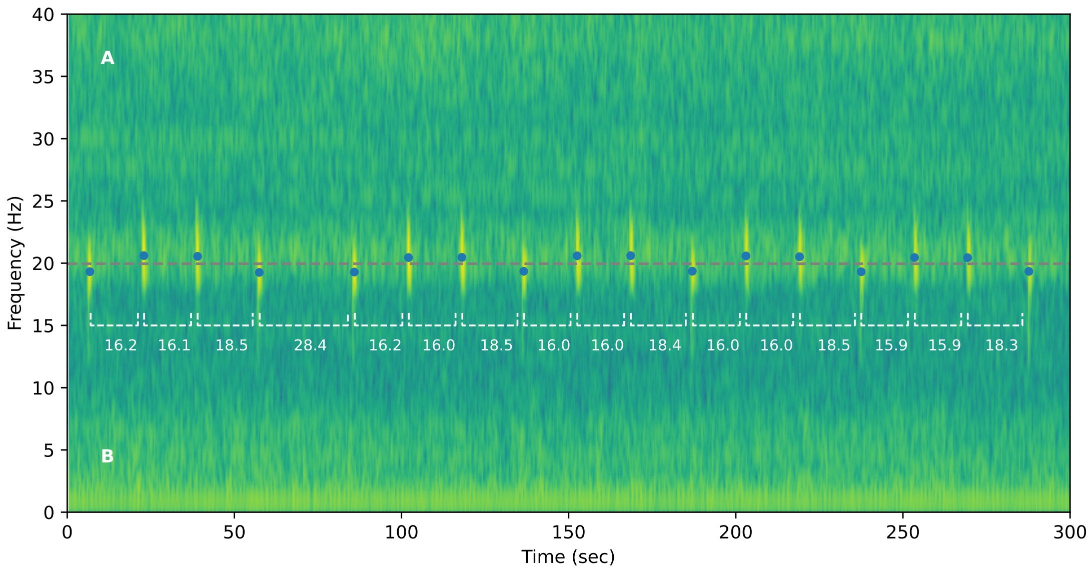


Figure 7.2: Spectrogram of a fin whale pulse sequence recorded by the Bombyx buoy in October 2018 ($f_s = 250$, $NFFT = 1024$, $hop = 8$, $padding = 75\%$). Dots show the center frequencies of the detected pulses, with white dashed lines showing IPIs. The grey dashed line denotes the discrimination threshold between type A and B pulses, at 19.88 Hz.

The resulting pulse envelope is used to compute the left and right boundaries of the pulse spectrum, with $\frac{\max E(f)}{4}$ as a threshold (equivalent to -6 dB). Left and right intersection frequencies are linearly interpolated to increase the precision of the estimate. This process yields the 6 dB bandwidth (width between the boundaries), and the center frequency (mid-point between the boundaries) of the analysed pulse.

For later filtering by pulse quality, the SNR is also computed following Eq. 7.3 (pulse energy as the maximum of its envelope and background energy as the median of the spectrogram surrounding the pulse).

$$\begin{aligned} E_{Background} &= \underset{\substack{f\in[15,25] \\ T\setminus[\hat{t}-1,\hat{t}+3]}}{\text{median}} \mathbf{S}_{f,t}, \\ E_{Pulse} &= \max_{f} E(f), \\ SNR &= 10\log_{10}\left(\frac{E_{Pulse}}{E_{Background}}\right). \end{aligned} \tag{7.3}$$

The pulse spectral characteristics of mysticetes are often described using the frequency of maximum energy (peak frequency) or the spectrum weighted mean (centroid frequency) [204, 121]. Here, the center frequency was chosen, as it appeared

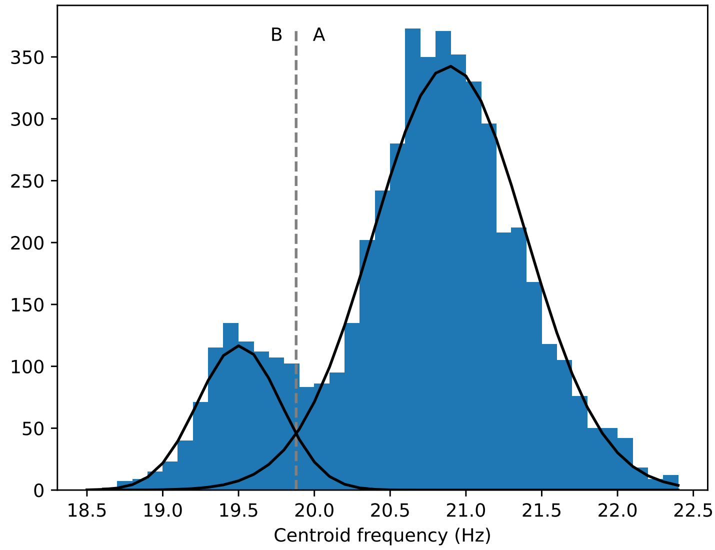


Figure 7.3: Histogram of the center frequencies of the detected pulses. Black lines denote the fitted GMM.

to be better suited for the discrimination between the two pulse types. In fact, when modeling the distribution of peak frequencies using a Gaussian mixture model, the two components (emerging from the two types of pulses) overlap more than when using center frequencies (the Kullback-Leibler divergence between the Gaussian components in center frequency is significantly higher than that of peak frequencies, with 113 nats and 30 nats respectively).

Pre-analysis filtering

To filter out false positives, only pulses with a bandwidth below 10 Hz and a center frequency within $[18.5, 22.5]$ were retained. Besides, only sequences with a mean SNR of at least 8 dB, and with at least 3 pulses were kept for the following analysis. Sequences containing IPIs below 10sec or above 45sec were discarded as well. The resulting number of registered pulses and sequences are shown in a calendar Fig. 7.1 and in Tab. 7.1.

To classify between A and B types, a two component GMM was fitted on the center frequency data (Fig. 7.3) using the Expectation Maximisation (EM) algorithm. This lead to a threshold of 19.88 Hz to discriminate between the two types. Even though the center frequency is found to evolve over time, the change is sufficiently small to not interfere with the categorisation (see Fig. 7.6).

7.2.3 Results

Stereotypical IPI

The time between a pair of consecutive pulses in a sequence (the IPI) appears to be strongly determined by their type (see Fig. 7.4). The typical interval for an ‘AB’ bi-gram is 2sec longer than that of ‘AA’ or ‘BA’. On the other hand, the ’BB’ pairs (less frequent but still commonly found) are 11sec longer on average, but present larger variability than the others.

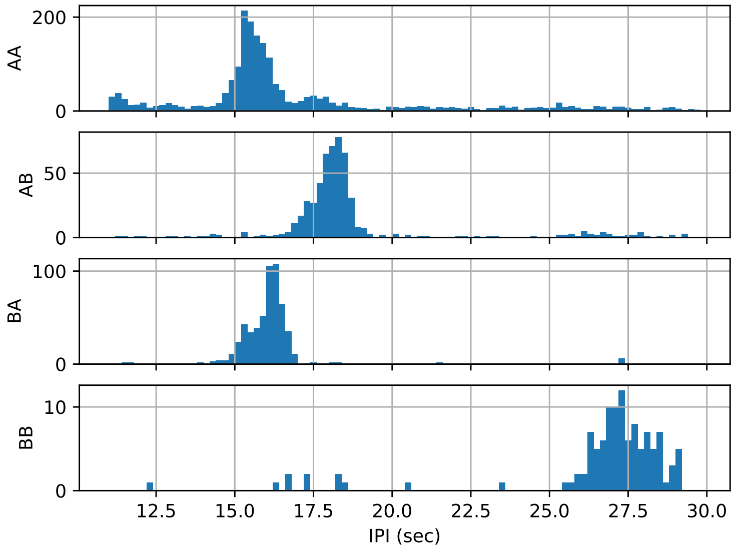


Figure 7.4: Histogram of the IPI for each type sequence (bi-gram).

Figure 7.5 shows how these intervals have changed over the course of two decades, following an approach similar to Weirathmueller et al. [204]. For each month and pulse type pair, points denote the most frequent IPI (quantised with a resolution of 0.1sec). For months containing more than 100 bi-gram occurrences, the most frequent IPI was retained only if representing at least 5% of it. Points measured in previous studies of the same population were also added : in 1999 by Clark et al. [33] (the only study to our knowledge that references IPI depending on type sequence in the Mediterranean sea), and in 2008 by Castellote et al. [29] (assuming it describes the most common pair ‘AA’, as it was not specified). The ‘BB’ sequence did not provide enough occurrences for the statistical tests to be relevant.

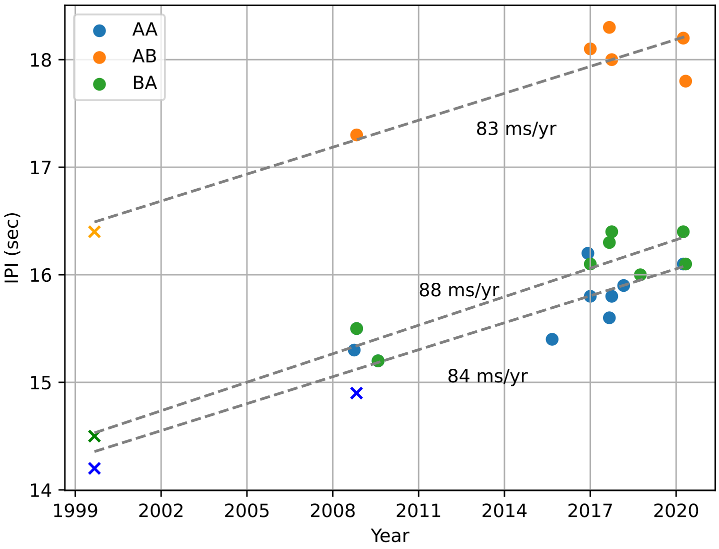


Figure 7.5: Scatter plot of the most frequent IPI per month for each type sequence. Fitted linear models are shown as grey dashed lines. Points extracted from Clark et al. [33] and Castellote et al. [29] appear as crosses.

For sequences ‘AA’, ‘AB’, and ‘BA’, fitted linear models are plotted (their coefficients of determination are 0.83, 0.89, and 0.91 respectively). The p-values for the null-hypothesis that the slopes are not significantly different from 0 are all inferior to 0.01. The estimated slopes for the ‘AA’, ‘AB’, and ‘BA’ bi-grams are 84, 83, and 88 respectively (in milliseconds/year).

Center frequency

In a similar fashion, temporal trends of pulses’ spectral characteristics were analysed. This revealed an intra-annual decrease in pulse center frequency between the months of August and May (Fig. 7.6). On the other hand, no inter-annual shift was observed (Pearson analysis yields a correlation coefficient of -0.06 between pulse absolute dates and their center frequency).

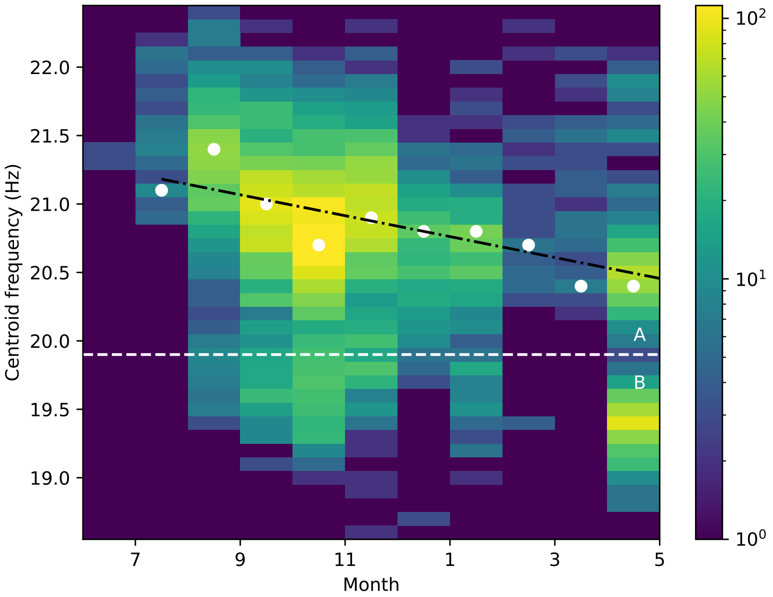


Figure 7.6: Bi-histogram of the center frequencies against months of the year. The horizontal line shows the separation between type A and type B pulses. The fitted linear model is shown as a black dashed line.

For this statistical analysis, center frequencies were quantised to a resolution of 0.1 Hz and grouped by months. Center frequencies with the most occurrences were kept, if among groups (months) of at least 50 pulses. Fitting a linear model on the retained points yields a coefficient of determination of 0.73, with an estimated slope of -0.08 Hz/month) (for the null-hypothesis that the slope is not significantly different from 0, the p-value is below 0.01).

For comparison with other previous studies, the same analysis was ran using peak and centroid frequencies. The slope of the observed intra-annual trends are similar for all metrics (-0.09 Hz/month, -0.08 Hz/month, and -0.11 Hz/month for peak, center, and centroid frequencies respectively) and p-values for the null-hypothesis that the slope is not different from 0 are all below 0.01.

Correlation between center frequency and IPI

With the observation of synchronous inter-annual shifts of both IPI and center frequencies in Pacific fin whales, the hypothesis of a link between the two arose. Weirathmueller et al. [204] states that the augmentation of the IPI through the years could be explained by the simultaneous decrease in pulse peak frequencies (lower frequency pulses presumably requiring a bigger effort to produce, a bigger gap between them could be needed). The observed stereotypical IPIs of Mediterranean fin whales also support this idea (sequences towards A pulses show lower IPIs). This hypothesis was thus further tested by analysing the correlation between IPI and center frequency (for pulses with IPIs between 14 and 20 seconds).

To dissociate this analysis from the link between pulse types and IPI, 3 component Gaussian mixture model was fitted on the bi-dimensional representation of pulses (center frequency versus time until the next pulse). This enabled to group the different pulse bi-grams (‘AA’, ‘AB’, and ‘BA’), and conduct a correlation analysis on each group independently. Figure 7.7 shows the scatter plot of the pulses with their assignation to each mixture component. For each of the latter, the Pearson correlation coefficient was computed, yielding -0.37, -0.22, and -0.35 for ‘BA’, ‘AB’, and ‘AA’ respectively (all p-values are below 0.01).

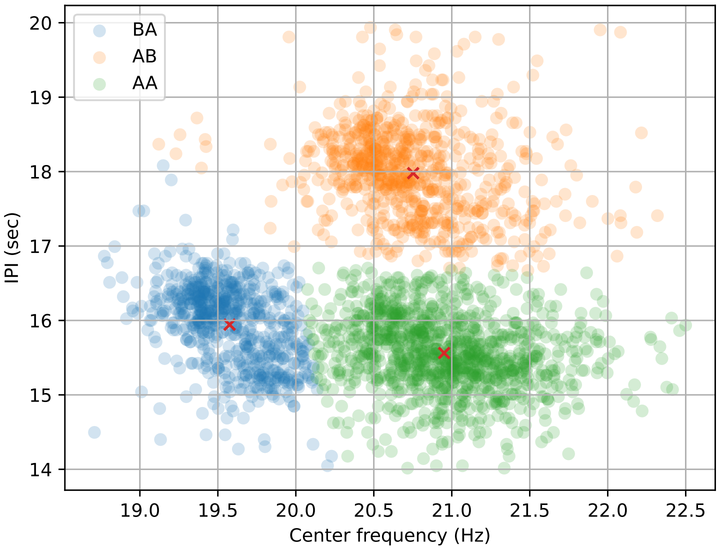


Figure 7.7: Scatter plot of pulses center frequency against the time until the next pulse (IPI). Colours denote the GMM assignations, whose means are marked with crosses.

7.2.4 Discussion

Mediterranean sea stereotypical IPIs

The present study led to the confirmation of the local stereotypical IPIs being determined by pulse bi-grams. These results were previously shown on relatively a small corpora of around 100 pulses [33], they are hereby confirmed with a corpus larger by an order of magnitude, and over a span of 10 years.

Moreover, two temporal trends were observed. They are put in relation to other fin whale song studies in Tab. 7.2 and discussed in the following sections.

		Inter-annual		**Intra-annual**	
Study	**Location**	**Frequency**	**IPI**	**Frequency**	**IPI**
Weirathmueller et al. [204]	N.E. Pacific	-0.17 Hz/yr	0.5-0.9 sec/yr	-	-
Oleson et al. [138]	N. Pacific	-	-	-	+7.5 sec
Leroy et al. [112]	Indian	-0.21 Hz/yr	-	∼ -0.1 Hz/mth	-
Helble et al. [83]	N. Pacific	-	0.6-1.3 sec/yr	-	-
Morano et al. [130]	N.W. Atlantic	-	* 0.5 sec/yr	-	+5.5 sec
Watkins et al. [201]	N.W. Atlantic	-	-	-	+6 sec
Širović et al. [178]	Gulf of California	-	∼ 1 sec/yr	-	∼ +8 sec
Furumaki et al. [68]	Chukchi sea	-	∼ 0.5 sec/yr	-	∼ +1 sec
Wood and Širović [206]	W. Antarctic	-0.2 Hz/yr	0.1 sec/yr	-	-
self	W. Mediterranean	-	0.1 sec/yr	-0.1 Hz/mth	-

Table 7.2: Summary of song pattern trend studies. For intra-annual IPI shifts, since trends are not linear, we report the difference between low IPI season and high IPI season (summer vs winter). The inter-annual IPI shift for Morano et al. [130] (see '*') is reported between two consecutive years only.

IPI trends

Mediterranean fin whale stereotypical IPIs are shown to evolve over the years, following a linear growth of approximately 0.1 sec/year over the past 20 years. Such trends had already been observed in the songs of North-East Pacific [204] and Central-North Pacific [83] fin whales.

Inter-annual shifts in IPI are rather recent and poorly documented. Weirathmueller et al. [204] state that the increasing IPI might be linked to the downward frequency shift, lower frequency pulses potentially being more demanding in energy. As for the present data, a low correlation coefficient was measured between the two variables, and no evidence of any inter-annual center frequency decrease was found. These observations thus go against this hypothesis, but more data is required to draw firm conclusions.

As for the IPI shift slopes, it seems plausible that the differences between Pacific and Mediterranean populations arise culturally. Whether they are originally caused by the same factors or not, the singing patterns drift independently, with song conformity only taking place within a given population. If environmental or physiological factors alone were responsible for such patterns, they would have to be present both in the Pacific and in the Mediterranean sea, but operating at different rates. The hypothesis of the post-whaling population recovery (increasing density and animal sizes) explaining those trends suits the latter conditions, as recovery rates could differ between Mediterranean and Pacific waters. On the other hand, cultural features such as contact rate between individuals could explain slope differences as well, regardless of the root cause of the shift.

On the other hand, studies of Atlantic and Pacific fin whales [130, 201, 138, 204] point to IPI increases during winter, before dropping back to autumn values. These trends are hypothesised to be directly linked to the reproductive season [138] (due to hormonal activity or progressive dilution of the competition for instance). No such trend was observed in the present data, but the irregular data sampling through seasons might create an observational bias in that sense.

Pulse frequency trends

Inter-annual shifts in vocalisation frequencies were already documented in blue whales [123, 121, 159], and bowhead whales [191]. Fin whales also showed similar trends in the Pacific [204] (for 20 Hz pulses, -0.17 Hz/year) and in the Indian Ocean [112] (for 99 Hz pulses, -0.21 Hz/year). Numerous hypotheses have been formulated for the cause of this phenomenon, such as the increase in population density or body sizes (following the cessation of commercial whaling), the increase in calling depth [71], the augmentation of noise from melting icebergs [112], or the acidification of the oceans affecting sound propagation [86] (among others).

No inter-annual frequency shift was found in the analysed data. Mediterranean fin whales could thus be an exception to this widespread trend. Nonetheless, an intra-annual decrease in center frequencies was observed (-0.08 Hz/month). Such phenomenon was previously observed in large mysticetes of the Indian Ocean including fin whales [112]. The latter study hypothesised pulse frequencies to follow seasonal ambient noise level variations (notably due to melting ice). Such phenomenon does not apply to the Mediterranean sea.

7.3 Orca call sequences

7.3.1 General context and objective

As previously mentioned, part of mysticete communication systems have been characterised as songs for being associated with courtship. No such phenomenon has been observed in odontocetes. Nonetheless, toothed whale communication has been studied extensively, especially with bottlenose dolphins and orcas. Their vocal displays (whistles and pulsed calls) have been suggested to serve social signaling and bonding purposes [92]. Bottlenose dolphins use individual specific signature whistles [196], whereas orcas use community specific pulsed calls [65] (the set of call types are specific at several levels such as clans and pods).

For orcas, the observation of stereotyped calls in relation to behavioural states has suggested no strict relationship between the two, but rather a group identification function. Ford [66] has manually analysed 20 thousands calls from 43 days of boat observation from 1978 to 1983, and reported call type bi-gram distributions (Fig. 7.8). Some call type distributions differed across activities, especially when involving multiple pods. Filatova et al. [58] have manually analysed 32 hours of recordings for calls to be assigned among 4 categories, and showed that activity did not affect proportions of occurrence but multi-pod interactions did.

Given the available 5 years of continuous recordings from the OrcaLab observatory (section 3.2.1), the following study will focus on the NRKW population of British Columbia. First, the detection CNN presented in section 5.3.1 was run on the summers from 2015 to 2020 (season of presence of the NRKW), detecting

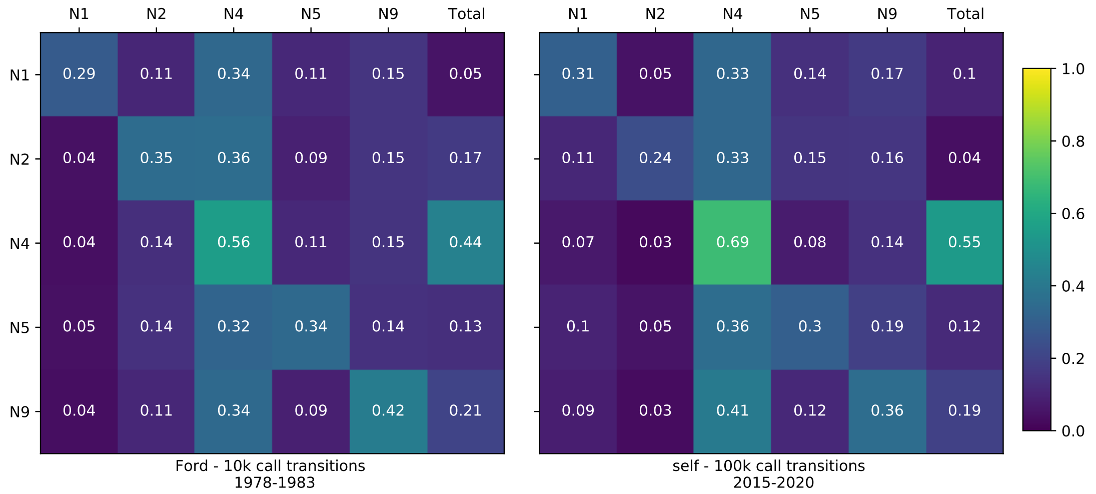


Figure 7.8: Comparison between the transition matrix from Ford [66] (left) and the present study (right). The ‘Total’ columns denotes the proportion of each call in the dataset.

more than 300 thousands calls. Then, the classification CNN presented in section 5.5 allowed to automatically recognise 7 common call types, and to tell when other calls are encountered.

The intent of this work is to study the potential structure in the sequences of call types, trying to make the most out of the large but blind corpus at hand (no information is available on associated behaviour or on the individual that emitted a call). We start by estimating the repertoire complexity following the Zipf power law coefficient approach. Then, to question the randomness / predictability of the sequences of types, we compare the occurrence of specific events with random simulations.

For the following analysis, sequences of calls were extracted from the CNN predictions (detections are located at confidence peaks that are above 0.8 and between 0.4 sec and 2 sec long). Calls were considered as being part of the same sequence if separated by less than 3 seconds. Sequences with at least 3 calls, and with no call labelled as ‘other’ were kept. This yielded 15,305 sequences with a total of 77,202 calls (Fig. 7.9).

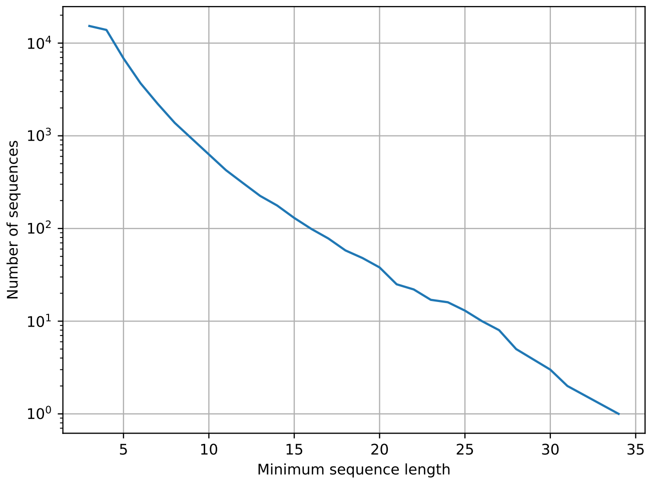


Figure 7.9: Log-survivor plot [52] of the extracted sequences' lengths.

7.3.2 Zipf's law and call type repertoire

Context

In several studies, Zipf's Law [218] has been used to quantitatively evaluate animal communication system repertoires (for humans [210] and non-humans [122, 99]). Such analysis rely on the estimation of the Power Law Coefficient (PLC) which reflects the relationship between a word's rank r (for the most frequent word $r = 1$ and so on) and its frequency f, following Eq. 7.4.

$$f = \alpha \times r^{\mathrm{PLC}} \tag{7.4}$$

The PLC denotes how stereotyped a system is, $PLC = 0$ meaning a uniform distribution, and PLC $<< -1$ meaning a high predictability. Zipf [218] states that for a system that follows constraints of efficiency ("least effort"), the PLC would converge to -1. This is supported by the fact that most human languages have a PLC close to -1 [210]. A PLC would thus be a necessary condition for a communication system to be language-like [99, 122].

Method

If enough data is available, a straightforward linear regression suffices to estimate the PLC of a repertoire (Fig. 4 of Kershenbaum et al. [99]). Eq. 7.5 shows the

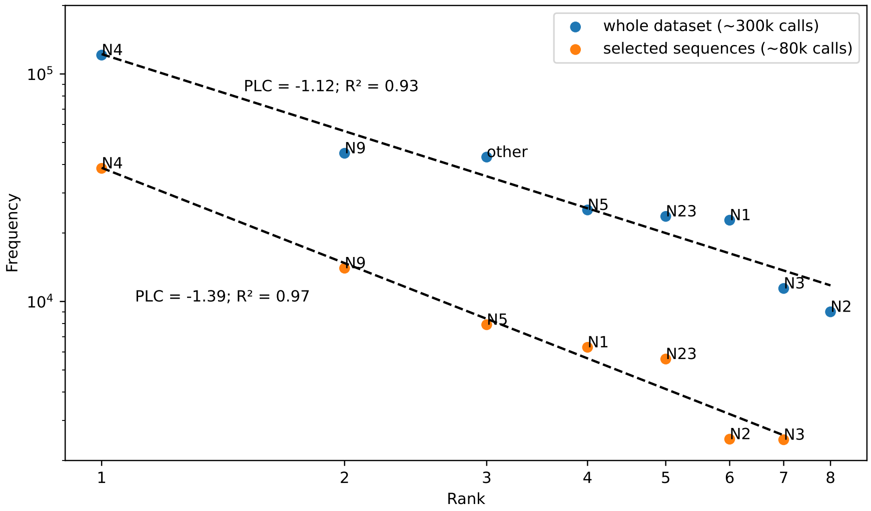


Figure 7.10: Zipf's law analysis for the whole repertoire of detected calls and for the repertoire of calls from extracted sequences. PLC are reported along with associated coefficients of determination for the linear regressions.

logarithm applied to Eq. 7.4 that allows for a linear regression. Figure 7.10 shows the resulting linear fits (via least square) in a log-rank vs log-frequency plot.

$$\log(f) = -\mathrm{PLC} \times \log(r) + \log(\alpha) \tag{7.5}$$

Discussion

The estimated PLC from the whole dataset (-1.12) lies close to the one estimated by Kershenbaum et al. [99] for a repertoire of the same species but with a much smaller dataset (with 773 calls, $PLC \approx -1.1$))[1].

The large gap between the PLCs from the whole dataset and that of the selected sequences demonstrates the significant impact that data sampling has on the such estimates, even with large datasets. These results do not refute the potential for orca call sequences to be language-like, but do not prove it either.

[1] The estimated PLC of Kershenbaum et al. [99] varies between -1 and -1.5 depending on the method employed, but the method that yielded -1.5 also showed a large error during the verification showed in Fig. 7 of the paper.

7.3.3 Span of correlation in sequences

Context

An approach to measure the randomness or predictability of sequences is to measure the MI between call types. The MI measures the KL divergence between marginal and joint probability distributions of two random variables. Applying this concept to dolphin whistle sequences, Ferrer-i Cancho and McCowan [57] propose to measure the MI between calls X and Y at a distance d (Eq. 7.6). The distance here is measured in number of calls that separate a pair, $d = 1$ denoting a consecutive pair.

$$\begin{aligned}&\mathrm{I}(X;Y|D=d) = \\ &\sum_{x,y} p(X=x, Y=y|D=d) \log\Big(\frac{p(X=x, Y=y|D=d)}{p(X=x|D=d)p(Y=y|D=d)}\Big)\end{aligned} \tag{7.6}$$

Method

To have a reference against which measures of $\mathrm{I}(X;Y|D=d)$ can be compared, we can randomly generate call pairs and measure their own $\mathrm{I}(X;Y|D=d)$. For that purpose, Ferrer-i Cancho and McCowan [57] propose two randomisation methods:

- **Global randomisation**: shuffle the concatenation of all sequences before recreating sequences of the same size to count call pairs,
- **Local randomisation**: shuffle the concatenation of all pairs at distance d before extracting pairs from the resulting vector.

In each of these methods, we generate as many pairs as observed in the data. They are more or less equivalent to generating sequences via a first order Markov model (taking into account only the probability of occurrence of a call). I propose to rather use a second order Markov model (or bi-gram model) to generate sequences and extract call pairs. Doing so, we integrate the propensity of consecutive calls to be of the same type, as observed by Ford [66] and shown by Fig. 7.8.

Using these randomisation methods, we can generate call pairs (as many as in the real data), measure $\mathrm{I}(X;Y|D=d)$, and compare it to that of the real data.

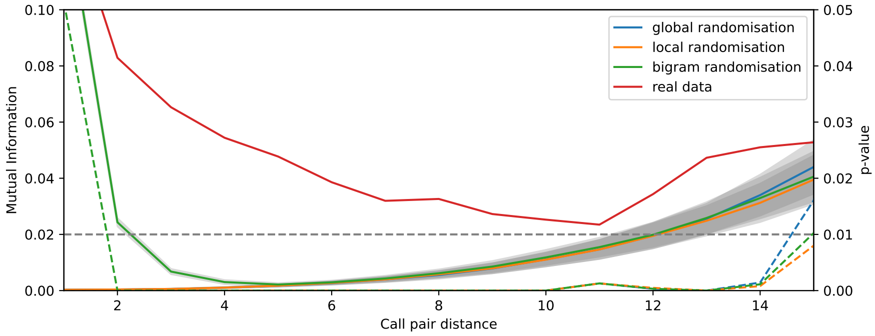


Figure 7.11: (solid lines) MI between pairs of call types depending on their relative distance d. For randomised MI (10,000 trials for each method), the mean ± standard deviation is given. (dashed lines) associated p-value for the null-hypothesis that the randomised pairs have a higher MI than the observed ones.

Doing so, and for each distance d, we can count the number of times the random pairs show an MI higher than the real ones (in this case out of 10,000 trials). Again following Ferrer-i Cancho and McCowan [57], we estimate the *p-value* of the null-hypothesis that random pairs have a higher MI than real ones, defined as the number of trials with a higher MI divided by the total number of trials.

Discussion

Figure 7.11 shows the evolution of the MI with a growing distance between calls. The fact that less long sequences are available (Fig. 7.9) might explain why the MI grows for $d > 11$. Non-surprisingly, the bi-gram generated pairs have the same MI than real ones at $d = 2$. Nonetheless, the MI of the observed pairs is significantly higher than the global, local and bi-gram randomisation, showing that single and second order Markov models do not suffice in modelling the observed sequences.

7.3.4 Propensity for specific patterns

Method

This sections studies the propensity for patterns in consecutive calls of observed sequences. It focuses on patterns of 4 consecutive call types (tri-grams) noted $f_r(X = x, Y = y, Z = z)$. Following a similar approach than in the last section, we generate as many tri-grams as observed in the data using a second-order Markov model. Doing so, we can compare the frequencies of generated tri-grams ($f_g(X = x, Y = y, Z = z)$) against real ones. Repeating this procedure for several trials (1,000 in this case), we can count the number of times a tri-grams appears more in the generated data than in the observed data, noted $N_g(X = x, Y = y, Z = z)$ (Eq. 7.7).

$$N_g(X = x, Y = y, Z = z) = \sum_{i=1}^{1,000} \begin{cases} 1 & \text{if } f_r(X = x, Y = y, Z = z) \leq f_g(X = x, Y = y, Z = z) \\ 0 & \text{otherwise} \end{cases} \tag{7.7}$$

N_g allows to test the null-hypothesis that the frequency of a tri-gram is explained by the bi-gram distribution alone. If N_g is below 0.01, the tri-gram appears significantly more than expected with the bi-gram model. Conversely, if N_g is above 0.99, tri-gram appears significantly less than expected with the bi-gram model.

Discussion

As stated by Kershenbaum et al. [98], "The most common application of the Markov model is to test whether or not units occur independently in a sequence". We make follow this incentive and compare randomly simulated n-gram frequencies with observed ones. Doing so, we can test whether an smaller order model suffices in explaining a higher order one. 54 tri-grams appeared less than expected, and 52 tri-grams appeared more (out of 343 possible ones).

7.3.5 Propensity for specific patterns

Method

This sections studies the propensity for patterns in consecutive calls of observed sequences. It focuses on patterns of 4 consecutive call types (quadri-grams) noted $f_r(X = \{a, b, c, d\})$. Following a similar approach than in the last section, we generate as many quadri-grams as observed in the data, this time using a third-order Markov model. Doing so, we can compare the frequencies of generated quadri-grams ($f_g(X = \{a, b, c, d\})$) against real ones. Repeating this procedure for several trials ($n = 1^5$ in this case), we can count the number of times a quadri-gram appears more in the generated data than in the observed data, noted $N_g(X = \{a, b, c, d\})$ (Eq. 7.8).

$$N_g(X = \{a,b,c,d\}) = \\ \frac{1}{n} \times \sum_{i=1}^{n} \begin{cases} 1 & \text{if } f_r(X = \{a,b,c,d\}) \leq f_g(X = \{a,b,c,d\}) \\ 0 & \text{otherwise} \end{cases} \tag{7.8}$$

N_g allows to test the null-hypothesis that the frequency of a quadri-gram is explained by the bi-gram distribution alone. If N_g is below 0.01, the quadri-gram appears significantly more than expected with the tri-gram model. Conversely, if N_g is above 0.99, the quadri-gram appears significantly less than expected with the tri-gram model.

Discussion

As stated by Kershenbaum et al. [98], “The most common application of the Markov model is to test whether or not units occur independently in a sequence”. We follow this incentive and compare randomly simulated n-gram frequencies with observed ones. Doing so, we can test whether a smaller order model suffices in explaining a higher order one.

For instance, out of 100,000 trials (each generating 31,287 quadri-grams using a third order Markov model) the quadri-gram ‘N4 N9 N9 N4’ appeared 243 times in average ($std = 15$). On the other hand, this quadri-gram was observed 324 times in the real data, and therefore none of the random generations yielded a higher frequency of occurrence for it. This means that if a ‘N4’ precedes ‘N9 N9’, there is a higher chance for a ‘N4’ to follow than expected in average after ‘N9 N9’.

Considering 54 tri-grams appeared less than expected, and 52 tri-grams appeared more (out of 343 possible ones).

Table 7.3: Quadri-grams appearing significantly more or less than expected with a third-order Markov model. The most frequent quadri-gram starting with the same tri-gram is also given (right column).

Quadri-gram	N_g	**Most frequent**
Observed more than in random generations		
N23 N23 N23 N23	0.0048	N23 N23 N23 N23
N4 N4 N4 N4	0.0	N4 N4 N4 N4
N4 N5 N5 N4	0.00016	N4 N5 N5 N4
N4 N9 N9 N4	0.0	N4 N9 N9 N4
N5 N5 N5 N5	0.0016	N5 N5 N5 N5
N9 N4 N4 N9	0.0	N9 N4 N4 N4
N9 N4 N9 N9	0.00736	N9 N4 N9 N4
Observed less than in random generations		
N1 N4 N4 N4	0.9968	N1 N4 N4 N4
N4 N4 N4 N23	0.99984	N4 N4 N4 N4
N4 N4 N4 N5	0.99664	N4 N4 N4 N4
N5 N4 N4 N4	0.99536	N5 N4 N4 N4
N9 N5 N4 N4	0.99872	N9 N5 N4 N4
N9 N9 N4 N4	0.99952	N9 N9 N4 N4

8 Conclusion and perspectives

8.1 Thesis contributions

This thesis demonstrates several PAM use cases, revolving about the use of ANNs to accelerate data analysis. It lies between a tutorial on how to use ANNs for PAM, an empirical study of what works and what doesn't, and the demonstration of the wide potential ahead of this approach. It is motivated by the following problematic: how to best use ANNs for cetacean vocalisation detection ? This thesis answers the latter in 3 folds : data annotation, architecture design and training regularisation, and detection exploitation for biological insights.

Methods in annotation Robust detection systems are needed to save analysis time on long term PAM recordings. ANNs offer an opportunity for this, but demand annotations to be trained and evaluated on. In the absence of already available robust analysis systems (detection or classification) and annotated databases, I proposed several procedures to enhance annotation efficiency, making the most out of recording characteristics and prior knowledge on target signals.

The proposed procedures where illustrated with several use cases starting from raw recordings, yielding 6 annotated databases (5 for detection and 1 for classification).

Training procedures Given annotated databases, training ANN allowed to solve the detection tasks for 12 target signals (5 from custom annotated databases, and 7 from the Antarctic mysticetes database). For signals with a limited variability such as sperm whale clicks and fin whale 20 Hz pulses, relatively small (three depth-wise convolution) networks yield satisfactory performances, improved compared to previous handcraft algorithms.

As for detecting the more variable orca calls, systematic searches and heavier models also yield satisfactory performances. Several insights arise from the exploration of network frontends, architectures and hyper-parameters, but they might be task specific.

On the other hand, heavier models can also serve the detection of several target signals with a shared set of weights, as shown with Antarctic mysticete calls. In this context, performance metrics are discussed and an interpretable metric for PAM uses is proposed.

Eventually despite efforts in using unlabeled data for self supervised representation learning and semi-supervised learning, the regular supervised approach appeared to be the most efficient for the orca call type classification task.

Applications Perhaps the most ambitious objective of this thesis was to bridge the gap between training deep learning algorithms and their application to long term bioacoustic surveys. This was conducted for the study of 3 species: sperm whales, fin whale and orcas. For each of them, different orientations were taken for the analysis. Sperm whale presence was studied in relation to anthropogenic noise, the fin whale song structure was described by long-term trends, and sequences of orca call types were analysed in search of specific patterns and dependencies.

8.2 Future work

Frontend experiments PCEN is a promising frontend but does not lead to a systematic performance gain. Work should be oriented towards understanding better why it might be detrimental, especially when fixing its smoothing and compression parameters.

In addition, to advance on embedded capacities for real time alert systems, analog feature extraction (stack of band-pass filters) should be experimented with. This would be relevant to tackle the main computational bottleneck of embedded bioacoustic analysis: the STFT.

Integration of spatial information The data available at DYNI has the potential to numerous other studies than the ones conducted so far. Work on the spatialisation of acoustic sources could be conducted on the data from KM3Net and OrcaLab data. This would allow to add a new dimension of analysis when processing vocalisation sequences.

Intra call modulations The analysis of orca call sequences presented in this work was subject to the prior discretisation by types. Some information is presumably lost in this process, such as within call variations. Li et al. [114] propose a deep learning based whistle contour extraction procedure, which seems robust to low SNR and overlapping calls. Experiments with this approach would be relevant to the analysis of orca call sequences.

Using ANNs for sequence modelling Modern day language modelling is often conducted with ANN based methods, especially with the recent boom of Transformer architectures [44]. These models could be trained on orca call sequences and yield a notion predictability and / or perplexity more reliable than with n-gram models.

Appendices

Bibliography

[1] S Aiello, A Albert, S Alves Garre, Z Aly, A Ambrosone, F Ameli, M Andre, G Androulakis, M Anghinolfi, M Anguita, et al. The km3net potential for the next core-collapse supernova observation with neutrinos. *The European Physical Journal C*, 81(5):1–19, 2021.

[2] Mark A Aizerman. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and remote control*, 25:821–837, 1964.

[3] Mebarka Allaoui, Mohammed Lamine Kherfi, and Abdelhakim Cheriet. Considerably improving clustering algorithms using umap dimensionality reduction technique: a comparative study. In *International Conference on Image and Signal Processing*, pages 317–325. Springer, 2020.

[4] Ann N Allen, Matt Harvey, Lauren Harrell, Aren Jansen, Karlina P Merkens, Carrie C Wall, Julie Cattiau, and Erin M Oleson. A convolutional neural network for automated detection of humpback whale song in a diverse, long-term passive acoustic dataset. *Frontiers in Marine Science*, 8:165, 2021.

[5] Masao Amano, Aya Kourogi, Kagari Aoki, Motoi Yoshioka, and Kyoichi Mori. Differences in sperm whale codas between two waters off Japan: possible geographic separation of vocal clans. *Journal of Mammalogy*, 95(1):169–175, 2014.

[6] Tomas Arias-Vergara, Philipp Klumpp, Juan Camilo Vasquez-Correa, Elmar Nöth, Juan Rafael Orozco-Arroyave, and Maria Schuster. Multi-channel spectrograms for speech processing applications using deep learning methods. *Pattern Analysis and Applications*, 24(2):423–431, 2021.

[7] Whitlow WL Au. Echolocation in dolphins. In *Hearing by whales and dolphins*, pages 364–408. Springer, 2000.

[8] Meghan G Aulich, Robert D McCauley, Benjamin J Saunders, and Miles JG Parsons. Fin whale (Balaenoptera physalus) migration in australian waters using passive acoustic monitoring. *Scientific reports*, 9(1):1–12, 2019.

[9] Randall Balestriero, Romain Cosentino, Hervé Glotin, and Richard Baraniuk. Spline filters for end-to-end deep learning. In *International conference on machine learning*, pages 364–373. PMLR, 2018.

[10] Pedro Bonito Baptista and Cláudia Antunes. Bioacoustic classification framework using transfer learning. *Modeling Decisions for Artificial Intelligence*, page 35, 2021.

[11] Valentin Barchasz, Valentin Gies, Sebastian Marzetti, and Hervé Glotin. A novel low-power high speed accurate and precise daq with embedded artificial intelligence for long term biodiversity survey. In *Proc. of the Acustica Symposium*, 2020.

[12] Mark F Baumgartner and Sarah E Mussoline. A generalized baleen whale call detection and classification system. *The Journal of the Acoustical Society of America*, 129(5):2889–2902, 2011.

[13] Christian Bergler, Manuel Schmitt, Rachael Xi Cheng, Hendrik Schröter, Andreas Maier, Volker Barth, Michael Weber, and Elmar Nöth. Deep representation learning for orca call type classification. In *International Conference on Text, Speech, and Dialogue*, pages 274–286. Springer, 2019.

[14] Christian Bergler, Hendrik Schröter, Rachael Xi Cheng, Volker Barth, Michael Weber, Elmar Nöth, Heribert Hofer, and Andreas Maier. Orca-spot: An automatic killer whale sound detection toolkit using deep learning. *Scientific reports*, 9(1):1–17, 2019.

[15] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[16] Martine Bérubé, Alex Aguilar, Daniel Dendanto, Finn Larsen, Giuseppe Notarbartolo Di Sciara, Richard Sears, Jóhann Sigurjónsson, Jorge URBAN-R, and PJ Palsbøll. Population genetic structure of North Atlantic, Mediterranean sea and sea of cortez fin whales, Balaenoptera physalus (linnaeus 1758): analysis of mitochondrial and nuclear loci. *Molecular ecology*, 7(5):585–599, 1998.

[17] Paul Best, Maxence Ferrari, Marion Poupard, Sébastien Paris, Ricard Marxer, Helena Symonds, Paul Spong, and Hervé Glotin. Deep learning and domain transfer for orca vocalization detection. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE, 2020.

[18] Paul Best, Sebastian Marzetti, Marion Poupard, Maxence Ferrari, Sébastien Paris, Ricard Marxer, Olivier Philipe, Valentin Gies, Valentin Barchasz, and Hervé Glotin. Stereo to five-channels bombyx sonobuoys: from four years cetacean monitoring to real-time whale-ship anti-collision system. In *e-Forum Acusticum 2020*, 2020.

[19] Paul Best, Ricard Marxer, Paris Sébastien, and Hervé Glotin. Representation learning for orca calls classification. In *Workshop on Vocal Interactivity in-and-between Humans, Animals and Robots (VIHAR), Paris*, 2021.

[20] Paul Best, Ricard Marxer, Sébastien Paris, and Hervé Glotin. [in review] temporal evolution of the Mediterranean fin whale song. *Scientific reports*, 2022.

[21] Paul Best, Sébastien Paris, Marxer Ricard, and Hervé Glotin. Deep learning for Antarctic blue and fin whale vocalization detection. In *The 9th International Workshop On Detection, Classification, Localization, And Density Estimation Of Marine Mammals Using Passive Acoustics*, 2022.

[22] Léa Bouffaut, Richard Dréo, Valérie Labat, Abdel-O Boudraa, and Guilhem Barruol. Passive stochastic matched filter for Antarctic blue whale call detection. *The Journal of the Acoustical Society of America*, 144(2):955–965, 2018.

[23] Jane Bromley, James W Bentz, Léon Bottou, Isabelle Guyon, Yann LeCun, Cliff Moore, Eduard Säckinger, and Roopak Shah. Signature verification using a “siamese” time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(04):669–688, 1993.

[24] Alexander Brown, James Montgomery, and Saurabh Garg. Automatic construction of accurate bioacoustics workflows under time constraints using a surrogate model. *Applied Soft Computing*, 113:107944, 2021.

[25] Judith C Brown and Paris Smaragdis. Hidden markov and gaussian mixture models for automatic call classification. *The Journal of the Acoustical Society of America*, 125(6):EL221–EL224, 2009.

[26] Judith C Brown, Andrea Hodgins-Davis, and Patrick JO Miller. Classification of vocalizations of killer whales using dynamic time warping. *The Journal of the Acoustical Society of America*, 119(3):EL34–EL40, 2006.

[27] Melba C Caldwell and David K Caldwell. Individualized whistle contours in bottle-nosed dolphins (Tursiops truncatus). *Nature*, 207(4995):434–435, 1965.

[28] Transport Canada. Speed restriction measures in the gulf of st. lawrence, 2022. URL `https://tc.canada.ca/en/marine-transportation/marine-safety/ship-safety-bulletins/protecting-north-atlantic-right-whale-speed-restriction-measures-gulf-st-lawren`

[29] Manuel Castellote, Christopher W Clark, and Marc O Lammers. Fin whale (Balaenoptera physalus) population identity in the western Mediterranean sea. *Marine Mammal Science*, 28(2):325–344, 2012.

[30] Northwestern center for robotics and biosystems. Pic32mx: Benchmarking mathematical operations, 2010. URL `http://hades.mech.northwestern.edu/index.php/PIC32MX:_Benchmarking_Mathematical_Operations#Overview`.

[31] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[32] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.

[33] Christopher W. Clark, J. F. Borsani, and G. Notarbartolo-Di-sciara. Vocal activity of fin whales, Balaenoptera physalus, in the ligurian sea. *Marine Mammal Science*, 18(1):286–295, 2002. ISSN 1748-7692. doi: https://doi.org/10.1111/j.1748-7692.2002.tb01035.x.

[34] Kevin R Coffey, Russell G Marx, and John F Neumaier. Deepsqueak: a deep learning-based system for detection and analysis of ultrasonic vocalizations. *Neuropsychopharmacology*, 44(5):859–868, 2019.

[35] Alexandra N Constaratas, Mark A McDonald, Kimberly T Goetz, and Giacomo Giorli. Fin whale acoustic populations present in new zealand waters: Description of song types, occurrence and seasonality using passive acoustic monitoring. *Plos one*, 16(7):e0253737, 2021.

[36] Donald A Croll, Christopher W Clark, Alejandro Acevedo, Bernie Tershy, Sergio Flores, Jason Gedamke, and Jorge Urban. Only male fin whales sing loud songs. *Nature*, 417(6891):809–809, 2002.

[37] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pages 702–703, 2020.

[38] Alejandro Cuevas, Alejandro Veragua, Sonia Español-Jiménez, Gustavo Chiang, and Felipe Tobar. Unsupervised blue whale call detection using multiple time-frequency features. In *CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, pages 1–6. IEEE, 2017.

[39] James D Darling and Martine Berube. Interactions of singing humpback whales with other males. *Marine Mammal Science*, 17(3):570–584, 2001.

[40] Volker B Deecke and Vincent M Janik. Automated categorization of bioacoustic signals: avoiding perceptual pitfalls. *The Journal of the Acoustical Society of America*, 119(1):645–653, 2006.

[41] Volker B Deecke, John KB Ford, and Paul Spong. Dialect change in resident killer whales: implications for vocal learning and cultural transmission. *Animal behaviour*, 60(5):629–638, 2000.

[42] Julien Delarue, Sean K Todd, Sofie M Van Parijs, and Lucia Di Iorio. Geographic variation in Northwest Atlantic fin whale (Balaenoptera physalus) song: Implications for stock structure assessment. *The Journal of the Acoustical Society of America*, 125(3):1774–1782, 2009.

[43] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE conference on computer vision and pattern recognition*, pages 248–255, 2009.

[44] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[45] Dexin Duan. Detection method for echolocation clicks based on LSTM networks. *Mobile Information Systems*, 2022, 2022.

[46] Paul Nguyen Hong Duc, Maëlle Torterotot, Flore Samaran, Paul R White, Odile Gérard, Olivier Adam, and Dorian Cazau. Assessing inter-annotator agreement from collaborative annotation campaign in marine bioacoustics. *Ecological Informatics*, 61:101185, 2021.

[47] Parsons ECM. Impacts of navy sonar on whales and dolphins: Now beyond a smoking gun? *Frontiers in Marine Science*, 4:295, 2017.

[48] Christine Erbe, Colleen Reichmuth, Kane Cunningham, Klaus Lucke, and Robert Dooling. Communication masking in marine mammals: A review and research strategy. *Marine pollution bulletin*, 103(1-2):15–38, 2016.

[49] Christian D Escobar-Amado, Mohsen Badiey, and Sean Pecknold. Automatic detection and classification of bearded seal vocalizations in the northeastern Chukchi sea using convolutional neural networks. *The Journal of the Acoustical Society of America*, 151(1):299–309, 2022.

[50] Mahdi Esfahanian, Hanqi Zhuang, and Nurgun Erdol. On contour-based classification of dolphin whistles by type. *Applied Acoustics*, 76:274–279, 2014.

[51] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, pages 226–231, 1996.

[52] R. M. Fagen and D. Y. Young. Temporal patterns of behaviors: durations, intervals, latencies, and sequences. *Quantitative Ethology*, pages pp. 79–114, 1978.

[53] A Fais, Mark Johnson, M Wilson, N Aguilar Soto, and PT Madsen. Sperm whale predator-prey interactions involve chasing and buzzing, but no acoustic stunning. *Scientific Reports*, 6(1):1–13, 2016.

[54] Max Ferguson, Ronay Ak, Yung-Tsun Tina Lee, and Kincho H Law. Automatic localization of casting defects with convolutional neural networks. In *IEEE international conference on big data*, pages 1726–1735. IEEE, 2017.

[55] Maxence Ferrari. *Study of a biosonar based on the modeling of a complete chain of emission-propagation-reception with validation on sperm whales.* PhD thesis, Amiens, 2020.

[56] Maxence Ferrari, Hervé Glotin, Ricard Marxer, and Mark Asch. Docc10: Open access dataset of marine mammal transient studies and end-to-end CNN classification. In *International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2020.

[57] Ramon Ferrer-i Cancho and Brenda McCowan. The span of correlations in dolphin whistle sequences. *Journal of Statistical Mechanics: Theory and Experiment*, 2012(06):P06002, 2012.

[58] OA Filatova, MA Guzeev, ID Fedutin, AM Burdin, and Erich Hoyt. Dependence of killer whale (Orcinus orca) acoustic signals on the type of activity and social context. *Biology bulletin*, 40(9):790–796, 2013.

[59] Olga A Filatova, Filipa IP Samarra, Volker B Deecke, John KB Ford, Patrick JO Miller, and Harald Yurk. Cultural evolution of killer whale calls: background, mechanisms and consequences. *Behaviour*, 152(15):2001–2038, 2015.

[60] Cristina Fiori, Luca Giancardo, Mehdi Aïssi, Jessica Alessi, and Paolo Vassallo. Geostatistical modelling of spatial distribution of sperm whales in the Pelagos sanctuary based on sparse count data and heterogeneous observations. *Aquatic Conservation: Marine and Freshwater Ecosystems*, 24(S1):41–49, 2014.

[61] NOAA Fisheries. Environmental impact statement: Reducing ship strikes to North Atlantic right whales. Technical report, NOAA, 2012.

[62] NOAA Fisheries. Marine mammal protection act (mmpa), 2013.

[63] Frederic B Fitch. Mcculloch warren s. and pitts walter. a logical calculus of the ideas immanent in nervous activity. bulletin of mathematical biophysics, vol. 5, pp. 115–133. *Journal of Symbolic Logic*, 9(2), 1944.

[64] W Tecumseh Fitch. The biology and evolution of music: A comparative perspective. *Cognition*, 100(1):173–215, 2006.

[65] JKB Ford. A catalogue of underwater calls produced by killer whales (Orcinus orca) in British Columbia (canadian data report of fisheries and aquatic sciences no. 633). *Fisheries Research Branch, Pacific Biological Station*, 1987.

[66] John KB Ford. Acoustic behaviour of resident killer whales (Orcinus orca) off Vancouver island, British Columbia. *Canadian Journal of Zoology*, 67(3): 727–745, 1989.

[67] Kaitlin E Frasier. A machine learning pipeline for classification of cetacean echolocation clicks in large underwater acoustic datasets. *PLOS Computational Biology*, 17(12):e1009613, 2021.

[68] Shiho Furumaki, Koki Tsujii, and Yoko Mitani. Fin whale (Balaenoptera physalus) song pattern in the southern Chukchi sea. *Polar Biology*, 44(5): 1021–1027, 2021.

[69] Ellen C Garland and Peter K McGregor. Cultural transmission, evolution, and revolution in vocal displays: insights from bird and whale song. *Frontiers in psychology*, page 2387, 2020.

[70] Ellen C Garland, Anne W Goldizen, Melinda L Rekdahl, Rochelle Constantine, Claire Garrigue, Nan Daeschler Hauser, M Michael Poole, Jooke Robbins, and Michael J Noad. Dynamic horizontal cultural transmission of humpback whale song at the ocean basin scale. *Current biology*, 21(8):687–691, 2011.

[71] Alexander N Gavrilov, Robert D McCauley, and Jason Gedamke. Steady inter and intra-annual decrease in the vocalization frequency of Antarctic blue whales. *The Journal of the Acoustical Society of America*, 131(6):4476–4480, 2012.

[72] Shane Gero, Hal Whitehead, and Luke Rendell. Individual, unit and vocal clan level identity cues in sperm whale codas. *Royal Society Open Science*, 3 (1):150372, 2016.

[73] Douglas Gillespie, DK Mellinger, JONATHAN Gordon, David Mclaren, PAUL Redmond, Ronald McHugh, PW Trinder, XY Deng, and A Thode. Pamguard: Semiautomated, open source software for real-time acoustic detection and localisation of cetaceans. *Journal of the Acoustical Society of America*, 30(5): 54–62, 2008.

[74] H. Glotin, P. Giraudet, J. Ricard, F. Malige, P. Patris, V. Roger, J.-M. Prévot, M. Poupard, O. Philippe, and P. Cosentino. Projet VAMOS : Visées aeriennes de mammifères marins jointes aux obervations acoustiques sous-marines de la bouée BOMBYX et antares. Technical report, Pelagos, 2017. URL `https://www.sanctuaire-pelagos.org/fr/tous-les-telechargements/etudes-scientifiques-studi-scientifici-studies/etudes-francaises/789-14-037-vamos`.

[75] Hervé Glotin, Maxence Ferrari, Paul Best, Marion Poupard, Nicolas Thellier, Audrey Monsimer, and Pascale Giraudet. Carimam report bioacoustic data processing. Technical report, DYNI LIS, 2021.

[76] Jack Goffinet, Samuel Brudner, Richard Mooney, and John Pearson. Low-dimensional learned feature spaces quantify individual and group differences in vocal repertoires. *Elife*, 10:e67855, 2021.

[77] Thomas Grill and Jan Schlüter. Two convolutional neural networks for bird detection in audio signals. In *25th European Signal Processing Conference (EUSIPCO)*, pages 1764–1768. IEEE, 2017.

[78] Jan-Eric Grunwald, Sven Schörnich, and Lutz Wiegrebe. Classification of natural textures in echolocation. *Proceedings of the National Academy of Sciences*, 101(15):5670–5674, 2004.

[79] Graham Harris. *Phytoplankton ecology: structure, function and fluctuation*. Springer Science & Business Media, 2012.

[80] L.T. Hatch and C.W. Clark. Acoustic differentiation between fin whales in both the North Atlantic and North Pacific Oceans, and integration with genetic estimates of divergence. *Unpublished paper to the IWC Scientific Committee*, 2004.

[81] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[82] Tyler A Helble, Glenn R Ierley, Gerald L D'Spain, Marie A Roch, and John A Hildebrand. A generalized power-law detection algorithm for humpback whale vocalizations. *The Journal of the Acoustical Society of America*, 131(4): 2682–2699, 2012.

[83] Tyler A Helble, Regina A Guazzo, Gabriela C Alongi, Cameron R Martin, Stephen W Martin, and E Elizabeth Henderson. Fin whale song patterns shift over time in the central North Pacific. *Frontiers in Marine Science*, 7: 907, 2020.

[84] Louis M Herman. The multiple functions of male song within the humpback whale (Megaptera novaeangliae) mating system: review, evaluation, and synthesis. *Biological Reviews*, 92(3):1795–1818, 2017.

[85] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al. CNN architectures for large-scale audio classification. In *2017 ieee international conference on acoustics, speech and signal processing (icassp)*, pages 131–135. IEEE, 2017.

[86] Keith C Hester, Edward T Peltzer, William J Kirkwood, and Peter G Brewer. Unanticipated consequences of ocean acidification: A noisier ocean at lower ph. *Geophysical research letters*, 35(19), 2008.

[87] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. doi: 10.1109/MCSE.2007.55.

[88] Yeongtae Hwang, Hyemin Cho, Hongsun Yang, Dong-Ok Won, Insoo Oh, and Seong-Whan Lee. Mel-spectrogram augmentation for sequence to sequence voice conversion. *preprint arXiv:2001.01401*, 2020.

[89] Max Jaderberg, Valentin Dalibard, Simon Osindero, Wojciech M Czarnecki, Jeff Donahue, Ali Razavi, Oriol Vinyals, Tim Green, Iain Dunning, Karen Simonyan, et al. Population based training of neural networks. *arXiv preprint arXiv:1711.09846*, 2017.

[90] Yannick Jadoul, Bill Thompson, and Bart De Boer. Introducing parselmouth: A python interface to praat. *Journal of Phonetics*, 71:1–15, 2018.

[91] Ali Jahangirnezhad and Afra Mashhadi. Deep embedded clustering for bioacoustic clustering of marine mammal vocalization. In *AI: Modeling Oceans and Climate Change Workshop at ICLR*, page 7, 2021.

[92] Vincent M Janik. Cetacean vocal learning and communication. *Current opinion in neurobiology*, 28:60–65, 2014.

[93] Vincent M Janik and Peter JB Slater. Context-specific use suggests that bottlenose dolphin signature whistles are cohesion calls. *Animal behaviour*, 56 (4):829–838, 1998.

[94] Xu Ji, Andrea Vedaldi, and João F Henriques. Invariant information clustering for unsupervised image classification and segmentation. In *CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South)*, pages 9864–9873, 2018.

[95] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

[96] Ibrahem Kandel and Mauro Castelli. The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset. *ICT express*, 6(4):312–315, 2020.

[97] V Kandia and Y Stylianou. Detection of sperm whale clicks based on the Teager–Kaiser energy operator. *Applied Acoustics*, 67(11-12):1144–1163, 2006.

[98] Arik Kershenbaum, Daniel T Blumstein, Marie A Roch, Çağlar Akçay, Gregory Backus, Mark A Bee, Kirsten Bohn, Yan Cao, Gerald Carter, Cristiane Cäsar, et al. Acoustic sequences in non-human animals: a tutorial review and prospectus. *Biological Reviews*, 91(1):13–52, 2016.

[99] Arik Kershenbaum, Vlad Demartsev, David E Gammon, Eli Geffen, Morgan L Gustison, Amiyaal Ilany, and Adriano R Lameira. Shannon entropy as a robust estimator of zipf's law in animal vocal communication repertoires. *Methods in Ecology and Evolution*, 12(3):553–564, 2021.

[100] Darlene R Ketten. Functional analyses of whale ears: adaptations for underwater hearing. In *Proceedings of OCEANS'94*, volume 1, pages I–264. IEEE, 1994.

[101] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[102] Diederik P Kingma and Max Welling. An introduction to variational autoencoders. *arXiv preprint arXiv:1906.02691*, 2019.

[103] Gregory Koch, Richard Zemel, Ruslan Salakhutdinov, et al. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille, 2015.

[104] Scott D Kraus, Robert D Kenney, Charles A Mayo, William A McLellan, Michael J Moore, and Douglas P Nowacek. Recent scientific publications cast doubt on North Atlantic right whale future. *Frontiers in Marine Science*, 3: 137, 2016.

[105] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

[106] Anders Krogh and John A Hertz. A simple weight decay can improve generalization. In *Advances in neural information processing systems*, pages 950–957, 1992.

[107] Sophie Laran, Manuel Castellote, Frédéric Caudal, Alexandre Monnin, and Glotin Hervé. Suivi acoustique des cetacés au nord du sanctuaire Pelagos. Technical report, Pelagos, 2009. URL `https://www.sanctuaire-pelagos.org/en/tous-les-telechargements/etudes-scientifiques-studi-scientifici-studies/etudes-francaises/70-08-048/file`.

[108] Mario Lasseck. Large-scale identification of birds in audio recordings. In *CLEF (Working Notes)*, pages 643–653, 2014.

[109] Mario Lasseck. Acoustic bird detection with deep convolutional neural networks. In *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, pages 143–147, 2018.

[110] Yann LeCun, John Denker, and Sara Solla. Optimal brain damage. *Advances in neural information processing systems*, 2, 1989.

[111] Yann LeCun, Yoshua Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10), 1995.

[112] Emmanuelle C Leroy, Jean-Yves Royer, Julien Bonnel, and Flore Samaran. Long-term and seasonal changes of large whale call frequency in the southern Indian Ocean. *Journal of Geophysical Research: Oceans*, 123(11):8568–8580, 2018.

[113] Liam Li, Kevin Jamieson, Afshin Rostamizadeh, Ekaterina Gonina, Jonathan Ben-Tzur, Moritz Hardt, Benjamin Recht, and Ameet Talwalkar. A system for massively parallel hyperparameter tuning. *Proceedings of Machine Learning and Systems*, 2:230–246, 2020.

[114] Pu Li, Xiaobai Liu, KJ Palmer, Erica Fleishman, Douglas Gillespie, Eva-Marie Nosal, Yu Shiu, Holger Klinck, Danielle Cholewiak, Tyler Helble, et al. Learning deep models from synthetic data for extracting dolphin whistle contours. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–10. IEEE, 2020.

[115] Bela G Liptak. *Instrument Engineers' Handbook, Volume One: Process Measurement and Analysis*. CRC press, 2003.

[116] Maciej Lopatka, Olivier Adam, Christophe Laplanche, Jan Zarzycki, and Jean-François Motsch. An attractive alternative for sperm whale click detection using the wavelet transform in comparison to the Fourier spectrogram. *Aquatic Mammals*, 31(4):463–467, 2005.

[117] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[118] Vincent Lostanlen, Justin Salamon, Andrew Farnsworth, Steve Kelling, and Juan Pablo Bello. Robust sound event detection in bioacoustic sensor networks. *PloS one*, 14(10):e0214168, 2019.

[119] Tao Lu, Baokun Han, and Fanqianhui Yu. Detection and classification of marine mammal sounds using AlexNet with transfer learning. *Ecological Informatics*, 62:101277, 2021.

[120] Shyam Madhusudhana, Yu Shiu, Holger Klinck, Erica Fleishman, Xiaobai Liu, Eva-Marie Nosal, Tyler Helble, Danielle Cholewiak, Douglas Gillespie, Ana Širović, et al. Improve automatic detection of animal call sequences with temporal context. *Journal of the Royal Society Interface*, 18(180):20210297, 2021.

[121] Franck Malige, Julie Patris, Susannah J Buchan, Kathleen M Stafford, Fannie Shabangu, Ken Findlay, Rodrigo Hucke-Gaete, Sergio Neira, Christopher W Clark, and Hervé Glotin. Inter-annual decrease in pulse rate and peak frequency of southeast Pacific blue whale song types. *Scientific reports*, 10(1): 1–11, 2020.

[122] Brenda McCowan, Laurance R Doyle, Jon M Jenkins, and Sean F Hanser. The appropriate use of zipf's law in animal communication studies. *Animal Behaviour*, 69(1):F1–F7, 2005.

[123] Mark A McDonald, John A Hildebrand, and Sarah Mesnick. Worldwide decline in tonal frequencies of blue whale songs. *Endangered species research*, 9(1):13–21, 2009.

[124] Leland McInnes, John Healy, and James Melville. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[125] David K Mellinger and Christopher W Clark. Recognizing transient low-frequency whale sounds by spectrogram correlation. *The Journal of the Acoustical Society of America*, 107(6):3518–3529, 2000.

[126] Eduardo Mercado. The sonar model for humpback whale song revised. *Frontiers in Psychology*, 9, 2018.

[127] Nathan D Merchant, Philippe Blondel, D Tom Dakin, and John Dorocicz. Averaging underwater noise levels for environmental assessment of shipping. *The Journal of the Acoustical Society of America*, 132(4):EL343–EL349, 2012.

[128] Brian S Miller, Naysa Balcazar, Sharon Nieukirk, Emmanuelle C Leroy, Meghan Aulich, Fannie W Shabangu, Robert P Dziak, Won Sang Lee, and Jong Kuk Hong. An open access dataset for developing automated detectors of Antarctic baleen whale sounds and performance evaluation of two commonly used detectors. *Scientific Reports*, 11(1):1–18, 2021.

[129] Bertel Møhl, Magnus Wahlberg, Peter T Madsen, Anders Heerfordt, and Anders Lund. The monopulsed nature of sperm whale clicks. *The Journal of the Acoustical Society of America*, 114(2):1143–1154, 2003.

[130] Janelle L Morano, Daniel P Salisbury, Aaron N Rice, Karah L Conklin, Keri L Falk, and Christopher W Clark. Seasonal and geographical patterns of fin whale song in the western North Atlantic Ocean. *The Journal of the Acoustical Society of America*, 132(2):1207–1212, 2012.

[131] Philipp Moritz, Robert Nishihara, Stephanie Wang, Alexey Tumanov, Richard Liaw, Eric Liang, Melih Elibol, Zongheng Yang, William Paul, Michael I Jordan, et al. Ray: A distributed framework for emerging AI applications. In *13th USENIX Symposium on Operating Systems Design and Implementation (OSDI 18)*, pages 561–577, 2018.

[132] Daniel T Murphy. Analysis of residual neural networks for marine mammal classification using multi-channel spectrograms. Master's thesis, University of New Orleans, 2021.

[133] Loris Nanni, Gianluca Maguolo, and Michelangelo Paci. Data augmentation approaches for improving animal audio classification. *Ecological Informatics*, 57:101084, 2020.

[134] Steven Ness. *The Orchive: A system for semi-automatic annotation and analysis of a large collection of bioacoustic recordings.* University of Victoria (Canada), 2013.

[135] Kenneth S Norris and George W Harvey. A theory for the function of the spermaceti organ of the sperm whale. *Animal orientation and navigation*, pages 393–417, 1972.

[136] Giuseppe Notarbartolo-di Sciara, Tundi Agardy, David Hyrenbach, Tullio Scovazzi, and Patrick Van Klaveren. The Pelagos sanctuary for Mediterranean marine mammals. *Aquatic Conservation: Marine and Freshwater Ecosystems*, 18(4):367–391, 2008.

[137] Giuseppe Notarbartolo-Di-Sciara, Margherita Zanardelli, Maddalena Jahoda, Simone Panigada, and Sabina Airoldi. The fin whale Balaenoptera physalus (l. 1758) in the Mediterranean sea. *Mammal Review*, 33(2):105–150, 2003. ISSN 1365-2907. doi: https://doi.org/10.1046/j.1365-2907.2003.00005.x.

[138] Erin M. Oleson, Ana Širović, Alexandra R. Bayless, and John A. Hildebrand. Synchronous seasonal change in fin whale song in the North Pacific. *PLoS ONE*, 9(12):e115678, Dec 2014. ISSN 1932-6203. doi: 10.1371/journal.pone.0115678.

[139] Clare Owen, Luke Rendell, Rochelle Constantine, Michael J Noad, Jenny Allen, Olive Andrews, Claire Garrigue, M Michael Poole, David Donnelly, Nan Hauser, et al. Migratory convergence facilitates cultural transmission of humpback whale song. *Royal Society open science*, 6(9):190337, 2019.

[140] Adam A Pack and Louis M Herman. *Dolphins can immediately recognize complex shapes across the senses of echolocation and vision.* PhD thesis, Acoustical Society of America, 1996.

[141] Dimitri Palaz, Mathew Magimai Doss, and Ronan Collobert. Convolutional neural networks-based continuous speech recognition using raw speech signal. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4295–4299. IEEE, 2015.

[142] Simone Panigada, Giovanna Pesante, Margherita Zanardelli, Frédéric Capoulade, Alexandre Gannier, and Mason T Weinrich. Mediterranean

fin whales at risk from fatal ship strikes. *Marine Pollution Bulletin*, 52(10): 1287–1298, 2006.

[143] Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*, 2019.

[144] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *International Conference on Machine Learning*, pages 4055–4064. PMLR, 2018.

[145] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`.

[146] G. Pavan, C. Fossati, M. Manghi, and M. Priano. Passive acoustics tools for the implementation of acoustic risk mitigation policies. *European Cetacean Society Newsletter No*, pages 52–58, 2004.

[147] Katherine Payne. The progressively changing songs of humpback whales: a window on the creative process in a wild animal. *The origins of music*, pages 135–150, 2000.

[148] Roger Payne and Scott McVay. Songs of humpback whales. *Science*, 173 (3997):585–597, 1971.

[149] Roger Payne and Douglas Webb. Orientation by means of long range acoustic signaling in baleen whales. *Annals of the New York Academy of Sciences*, 188 (1):110–141, 1971.

[150] Andreia Pereira, Danielle Harris, Peter Tyack, and Luis Matias. Fin whale acoustic presence and song characteristics in seas to the southwest of Portugal. *The Journal of the Acoustical Society of America*, 147(4):2235–2249, 2020.

[151] Hieu Pham, Zihang Dai, Qizhe Xie, and Quoc V Le. Meta pseudo labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11557–11568, 2021.

[152] Marion Poupard, Paul Best, Jan Schlüter, Helena Symonds, Paul Spong, Thierry Lengagne, Thierry Soriano, and Hervé Glotin. Large-scale unsupervised clustering of orca vocalizations: a model for describing orca communication systems. In *2nd International Workshop on Vocal Interactivity in-and-between Humans, Animals and Robots*, 2019.

[153] Marion Poupard, Maxence Ferrari, J Schluter, Ricard Marxer, Pascale Giraudet, Valentin Barchasz, Valentin Gies, G Pavan, and Hervé Glotin. Real-time passive acoustic 3d tracking of deep diving cetacean by small non-uniform mobile surface antenna. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8251–8255. IEEE, 2019.

[154] Marion Poupard, Helena Symonds, Paul Spong, and Hervé Glotin. Intra-group orca call rate modulation estimation using compact four hydrophones array. *Frontiers in Marine Science*, page 1383, 2021.

[155] Marion Poupard, Maxence Ferrari, Paul Best, and Hervé Glotin. Passive acoustic monitoring of sperm whales and anthropogenic noise using stereophonic recordings in the Mediterranean sea, North West Pelagos sanctuary. *Scientific reports*, 12(1):1–13, 2022.

[156] Jean-Francois Puget. STFT transformers for bird song recognition. In *Working Notes of CLEF*, 2021.

[157] Mirco Ravanelli and Yoshua Bengio. Speaker recognition from raw waveform with sincnet. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 1021–1028. IEEE, 2018.

[158] Luke E Rendell and Hal Whitehead. Vocal clans in sperm whales (Physeter macrocephalus). *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 270(1512):225–231, 2003.

[159] Ally Rice, Ana Širović, John A Hildebrand, Megan Wood, Alex Carbaugh-Rutland, and Simone Baumann-Pickering. Update on frequency decline of Northeast Pacific blue whale (Balaenoptera musculus) calls. *PloS one*, 17(4): e0266469, 2022.

[160] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[161] Marie A Roch, Melissa S Soldevilla, Rhonda Hoenigman, Sean M Wiggins, and John A Hildebrand. Comparison of machine learning techniques for the classification of echolocation clicks from three species of odontocetes. *Canadian Acoustics*, 36(1):41–47, 2008.

[162] Marie A Roch, Scott Lindeneau, Gurisht Singh Aurora, Kaitlin E Frasier, John A Hildebrand, Hervé Glotin, and Simone Baumann-Pickering. Using context to train time-domain echolocation click detectors. *The Journal of the Acoustical Society of America*, 149(5):3301–3310, 2021.

[163] Robert C Rocha, Phillip J Clapham, and Yulia V Ivashchenko. Emptying the oceans: a summary of industrial whaling catches in the 20th century. *Marine Fisheries Review*, 76(4):37–48, 2014.

[164] R Cotton Rockwood, John Calambokidis, and Jaime Jahncke. High mortality of blue, humpback and fin whales from modeling of vessel collisions on the us west coast suggests population impacts and insufficient protection. *PLoS One*, 12(8):e0183052, 2017.

[165] Rosalind M Rolland, Susan E Parks, Kathleen E Hunt, Manuel Castellote, Peter J Corkeron, Douglas P Nowacek, Samuel K Wasser, and Scott D Kraus. Evidence that ship noise increases stress in right whales. *Proceedings of the Royal Society B: Biological Sciences*, 279(1737):2363–2368, 2012.

[166] Miriam Romagosa, Sergi Pérez-Jorge, Irma Cascão, Helena Mouriño, Patrick Lehodey, Andreia Pereira, Tiago A Marques, Luís Matias, and Mónica A Silva. Food talk: 40-Hz fin whale calls are associated with prey biomass. *Proceedings of the Royal Society B*, 288(1954):20211156, 2021.

[167] Tara Sainath, Ron J Weiss, Kevin Wilson, Andrew W Senior, and Oriol Vinyals. Learning the speech front-end with raw waveform CLDNNs, 2015.

[168] Laela Sayigh, Mary Ann Daher, Julie Allen, Helen Gordon, Katherine Joyce, Claire Stuhlmann, and Peter Tyack. The watkins marine mammal sound database: an online, freely accessible resource. In *Proceedings of Meetings on Acoustics 4ENAL*, volume 27, page 040013. Acoustical Society of America, 2016.

[169] Jan Schlüter. *Deep learning for event detection, sequence labelling and similarity estimation in music signals*. PhD thesis, Universität Linz, 2017.

[170] Tyler M Schulz, Hal Whitehead, Shane Gero, and Luke Rendell. Overlapping and matching of codas in vocal interactions between sperm whales: insights into communication function. *Animal Behaviour*, 76(6):1977–1988, 2008.

[171] Virginia Sciacca, Francesco Caruso, Laura Beranzoli, Francesco Chierici, Emilio De Domenico, Davide Embriaco, Paolo Favali, Gabriele Giovanetti, Giuseppina Larosa, Giuditta Marinaro, Elena Papale, Gianni Pavan, Carmelo

Pellegrino, Sara Pulvirenti, Francesco Simeone, Salvatore Viola, and Giorgio Riccobene. Annual acoustic presence of fin whale (Balaenoptera physalus) offshore eastern Sicily, central Mediterranean sea. *PloS one*, 10(11):e0141838, 2015.

[172] Maxime Sèbe. *An interdisciplinary approach to the management of whale-ship collisions*. PhD thesis, Brest, 2020.

[173] Yu Shiu, KJ Palmer, Marie A Roch, Erica Fleishman, Xiaobai Liu, Eva-Marie Nosal, Tyler Helble, Danielle Cholewiak, Douglas Gillespie, and Holger Klinck. Deep neural networks for automated detection of marine mammal species. *Scientific reports*, 10(1):1–12, 2020.

[174] Malene Simon, Kathleen M Stafford, Kristian Beedholm, Craig M Lee, and Peter T Madsen. Singing behavior of fin whales in the Davis Strait with implications for mating, migration and foraging. *The Journal of the Acoustical Society of America*, 128(5):3200–3210, 2010.

[175] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[176] Ana Širović, John A Hildebrand, and Sean M Wiggins. Blue and fin whale call source levels and propagation range in the Southern Ocean. *The Journal of the Acoustical Society of America*, 122(2):1208–1215, 2007.

[177] Ana Sirovic, Erin M Oleson, John Calambokidis, Simone Baumann-Pickering, Amanda Cummins, Sara Kerosky, Lauren Roche, Anne Simonis, Sean M Wiggins, and John A Hildebrand. Marine mammal demographics of the outer washington coast during 2008-2009. Technical report, Scripps of institution oceanography LA Jolla CA, 2011.

[178] Ana Širović, Erin M Oleson, Jasmine Buccowich, Ally Rice, and Alexandra R Bayless. Fin whale song variability in southern California and the Gulf of California. *Scientific reports*, 7(1):1–11, 2017.

[179] Joshua N Smith, Anne W Goldizen, Rebecca A Dunlop, and Michael J Noad. Songs of male humpback whales, Megaptera novaeangliae, are involved in intersexual interactions. *Animal Behaviour*, 76(2):467–477, 2008.

[180] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in Neural Information Processing Systems*, 33:596–608, 2020.

[181] Panu Somervuo. Time–frequency warping of spectrograms applied to bird sound analyses. *Bioacoustics*, 28(3):257–268, 2019.

[182] SOOS. Atwg terms of reference, 2009. URL `https://www.soos.aq/images/soos/activities/cwg/AT/ATWG-TOR.pdf`.

[183] Paul Spong and Helena Symonds. Orcalab, 1970. URL `https://orcalab.org/`.

[184] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[185] Elsa Steinfath, Adrian Palacios, Julian Rottschaefer, Deniz Yuezak, and Jan Clemens. Fast and accurate annotation of acoustic signals with deep neural networks. *bioRxiv*, 2021.

[186] Dan Stowell. Computational bioacoustics with deep learning: a review and roadmap. *arXiv preprint arXiv:2112.06725*, 2021.

[187] Dan Stowell and Mark D Plumbley. Framewise heterodyne chirp analysis of birdsong. In *Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pages 2694–2698. IEEE, 2012.

[188] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147. PMLR, 2013.

[189] Michael J Tetley, Gill T Braulik, Caterina Lanfredi, Gianna Minton, Simone Panigada, Elena Politi, Margherita Zanardelli, Giuseppe Notarbartolo di Sciara, and Erich Hoyt. The important marine mammal area network: a tool for systematic spatial planning in response to the marine mammal habitat conservation crisis. *Frontiers in Marine Science*, page 321, 2022.

[190] Anshul Thakur, Daksh Thapar, Padmanabhan Rajan, and Aditya Nigam. Deep metric learning for bioacoustic classification: Overcoming training data scarcity using dynamic triplet loss. *The Journal of the Acoustical Society of America*, 146(1):534–547, 2019.

[191] Aaron M Thode, Susanna B Blackwell, Alexander S Conrad, Katherine H Kim, and A Michael Macrander. Decadal-scale frequency shift of migrating bowhead whale calls in the shallow Beaufort Sea. *The Journal of the Acoustical Society of America*, 142(3):1482–1502, 2017.

[192] Len Thomas and Tiago A Marques. Passive acoustic monitoring for estimating animal density. *Acoustics Today*, 8(3):35–44, 2012.

[193] Mark Thomas, Bruce Martin, Katie Kowarski, Briand Gaudet, and Stan Matwin. Marine mammal species classification using convolutional neural networks and a novel acoustic representation. *arXiv preprint arXiv:1907.13188*, 2019.

[194] Paul O Thompson, Lloyd T Findley, and Omar Vidal. 20-Hz pulses and other vocalizations of fin whales, Balaenoptera physalus, in the Gulf of California, Mexico. *The Journal of the Acoustical Society of America*, 92(6):3051–3057, 1992.

[195] Irina Tolkova, Brian Chu, Marcel Hedman, Stefan Kahl, and Holger Klinck. Parsing birdsong with deep audio embeddings. *arXiv preprint arXiv:2108.09203*, 2021.

[196] Peter L Tyack. Review of the signature-whistle hypothesis for the Atlantic bottlenose dolphin 10. *The bottlenose dolphin*, page 199, 2012.

[197] Maria Florencia Noriega Romero Vargas. *Revealing structure in vocalisations of parrots and social whales*. PhD thesis, Georg-August-Universität Göttingen, 2017.

[198] Heike Vester, Sarah Hallerberg, Marc Timme, and Kurt Hammerschmidt. Vocal repertoire of long-finned pilot whales (Globicephala melas) in northern Norway. *The Journal of the Acoustical Society of America*, 141(6):4289–4299, 2017.

[199] Yuxuan Wang, Pascal Getreuer, Thad Hughes, Richard F Lyon, and Rif A Saurous. Trainable frontend for robust and far-field keyword spotting. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5670–5674. IEEE, 2017.

[200] William A Watkins. Activities and underwater sounds of fin whales. *Sci. Rep. Whales Res. Inst*, 33:83–117, 1981.

[201] William A. Watkins, Peter Tyack, Karen E. Moore, and James E. Bird. The 20-Hz signals of finback whales (Balaenoptera physalus). *The Journal of the Acoustical Society of America*, 82(6):1901–1912, Dec 1987. ISSN 0001-4966. doi: 10.1121/1.395685.

[202] Stephanie L Watwood, Patrick JO Miller, Mark Johnson, Peter T Madsen, and Peter L Tyack. Deep-diving foraging behaviour of sperm whales (Physeter macrocephalus). *Journal of Animal Ecology*, 75(3):814–825, 2006.

[203] Linda Weilgart and Hal Whitehead. Coda communication by sperm whales (Physeter macrocephalus) off the Galapagos Islands. *Canadian Journal of Zoology*, 71(4):744–752, 1993.

[204] Michelle J Weirathmueller, Kathleen M Stafford, William SD Wilcock, Rose S Hilmo, Robert P Dziak, and Anne M Tréhu. Spatial and temporal trends in fin whale vocalizations recorded in the NE Pacific Ocean between 2003-2013. *Plos one*, 12(10):e0186127, 2017.

[205] Hal Whitehead and Luke Rendell. *The cultural lives of whales and dolphins*. University of Chicago Press, 2015.

[206] Megan Wood and Ana Širović. Characterization of fin whale song off the Western Antarctic Peninsula. *PloS one*, 17(3):e0264214, 2022.

[207] XFCE Desktop Environment. Thunar, 2007. URL `https://www.xfce.org/`.

[208] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. *arXiv:1511.06335 [cs]*, May 2016.

[209] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Unsupervised data augmentation for consistency training. *arXiv preprint arXiv:1904.12848*, 2019.

[210] Shuiyuan Yu, Chunshan Xu, and Haitao Liu. Zipf's law in 50 languages: its structural pattern, linguistic interpretation, and cognitive motivation. *arXiv preprint arXiv:1807.01855*, 2018.

[211] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*, 2021.

[212] Neil Zeghidour, Olivier Teboul, Félix de Chaumont Quitry, and Marco Tagliasacchi. Leaf: A learnable frontend for audio classification. *arXiv preprint arXiv:2101.08596*, 2021.

[213] Feiyu Zhang, Luyang Zhang, Hongxiang Chen, and Jiangjian Xie. Bird species identification using spectrogram based on multi-channel fusion of DCNNs. *Entropy*, 23(11):1507, 2021.

[214] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[215] An Zhao, Krishna Subramani, and Paris Smaragdis. Optimizing short-time fourier transform parameters via gradient descent. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 736–740. IEEE, 2021.

[216] Ming Zhong, Manuel Castellote, Rahul Dodhia, Juan Lavista Ferres, Mandy Keogh, and Arial Brewer. Beluga whale acoustic signal classification using deep learning neural network models. *The Journal of the Acoustical Society of America*, 147(3):1834–1841, 2020.

[217] Ming Zhong, Maelle Torterotot, Trevor A Branch, Kathleen M Stafford, Jean-Yves Royer, Rahul Dodhia, and Juan Lavista Ferres. Detecting, classifying, and counting blue whale calls with siamese neural networks. *The Journal of the Acoustical Society of America*, 149(5):3086–3094, 2021.

[218] George Kingsley Zipf. Human behavior and the principle of least effort: An introduction to human ecology. *Addison-Wesley Press*, 1949.