Detection and Tracking of Dolphin Vocalizations

Xanadu C. Halkias

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2009

Abstract

Detection and Tracking of Dolphin Vocalizations

Xanadu C. Halkias

The field of audio engineering has predominantly been consumed by human centered applications. Many scientists have developed numerous successful algorithms that deal with important issues in speech and music processing. Automatic speech recognition systems, speech to text products or even music recommendation systems and simple transcription are just a few of the numerous research projects that have escaped research labs worldwide and have entered our every day lives.

However, audio processing has also affected the field of animal bioacoustics. Many audio engineers are now using their knowledge to advance our understanding of the world that surrounds us and especially that of animals. This work falls into that category, where the principles of signal processing, communication theory and machine learning are used to analyze the clandestine world of marine mammals and specifically dolphins. Although substantial efforts have been made in the scientific community, there is still a need for the creation of an automatic and robust system that will allow for the detection and tracking of dolphin vocalizations.

In this work, Chapter 1 provides a short description on the physiology of dolphins focusing on how they produce and use sound for their survival. Chapter 2 is an overview of the most popular systems in the field for the

analysis of marine mammal vocalizations and provides a framework for the task at hand.

Continuing, Chapter 3 describes the unique aspects of the data as well as the different features that are used in order to provide the proposed detection and extraction schemes.

In Chapter 4 several algorithms are proposed for the successful detection of dolphin calls in long recordings. Starting with the simple and widely used thresholding detectors, several advancements are proposed, based on the data, leading to more intricate classifiers like Support Vector Machines (SVM) that are known for their robustness.

Chapter 5 provides two systems for pitch extraction. the first system is based on a probabilistic framework and deals with the extraction of dolphin whistle calls while providing a first attempt on resolving simple overlaps. The second system assumes that the desired calls have already been detected and proceeds to identify the pitch for both whistle and burst calls using hierarchically driven Hidden Markov Models (HMM).

Finally, Chapter 6 presents the conclusions from the proposed methodologies and highlights both their advantages and disadvantages. Dolphins are intensely vocal mammals and although we are still unaware of the purpose or context of these vocalizations, I hope that by the end of this work we will have a better idea on how to uncover and approach their sounds, one of the few doors, that allow us to enter their world.

Contents

List of Figures

List of Tables

Acknowledgments

Special thanks to Robert Turetsky for his relentless support and for not only showing me where the traps are, but for helping me out when I would fall in them...

Also, to my advisor Prof. Dan Ellis, who was willing to go on this journey with me and taught me the principles of scientific research.

Also, to Prof. Diana Reiss for not only providing me with the data of captive dolphins, but for also sharing her passion for marine mammals.

Many thanks to my dissertation committee for honoring me with their time and constructive comments.

A big thank you to Prof. Ed Coffman for his beautiful teachings and for standing up for me...

My deepest appreciation to Laura Wechsler and Rosemarie Raffa who went above and beyond to help me and ensured that I would be able to finish.

To Azlyn, Elsa, Kevin, Michelle and everybody at the EE department, thank you for always being there with a smile and willingness to help.

To my father, Christos Halkias for setting the bar high...

To my mother, Christina Antonopoulou for instilling in me that there is no bar high enough...

and to my late stepfather, Takis Moshos who left us enduring so much pain and fear, and with that...

taught me how to live without them

Chapter 1

Introduction

"Which flutes' beloved sound
Excites to play,
Upon the calm and placid sea"
Pindar's words about the Dolphin found in Plutarch's Morals

In Greek mythology, dolphins first appear in the tale of the Tyrrhenian pirates and the god Dionysos. According to the myth, the pirates captured the god Dionysos who was traveling under disguise in the Aegean sea and were planning to sell him off as a slave. In anger, Dionysos filled the ship with spreading vines and snakes and when the pirates jumped over board to save themselves he transformed them into dolphins so that they would help those in need at sea.

Anecdotal stories about the friendly nature of dolphins as well as their close bond with humans are found throughout human history. Even in modern times people seem to have a special relationship with dolphins more than any

marine mammal, in fact. It is astounding how there exists an overall common notion when it comes to the abilities and intelligence of dolphins. Most people, believe that dolphins are highly intelligent and encompass human qualities. An indicator of their cognitive abilities and evolution can be found in the work of Reiss and Marino [44] on mirror self-recognition in the bottlenose dolphin. However, scientific research has yet to establish the range of their cognitive abilities with irrefutable evidence.

The first written record on dolphins is found in Aristotle's (384 BC-322 BC) *Historia Animalium* (History of Animals). Aristotle was the first to classify the dolphin as a mammal and to also give an estimate of its life span e.g. approximately 25-30 years, although now understood to be between $30 - 60$ years. His observations were later used and compared by modern scientists in order to get a better understanding of marine mammals. In recent years more attention has been given in the study of marine mammals. Several organizations have facilitated a more comprehensive research on whales and dolphins not only for conservation and population preservation purposes, but also for a well-rounded understanding of the abilities of these mammals through developmental and cognition studies.

In terms of their taxonomy, whales and dolphins are classified as cetaceans. Those can be further categorized into baleen whales (*mysticeti*) and toothed whales and dolphins (*odontoceti*). Dolphins belong in the family of *Delphinidae* (oceanic dolphins). There exist 33 species of dolphins and the scope of this work is based on the vocalizations of the bottlenose dolphin (*Tursiops truncatus*), as seen in Figure 1.1.

Figure 1.1: Bottlenose dolphin (*Tursiops Truncatus*)

The bottlenose dolphin is the most widely researched dolphin. Its abundance through out the world's seas as well as its proximity to humans have allowed scientists to study the physiology and behavior of this marine mammal. Dolphins are known for their cooperative and playful personality. There exist numerous anecdotal stories of them approaching and helping humans in various situations. They are social animals and they live in pods of multiple individuals, while creating strong bonds amongst themselves [51]. In terms of their physiology, we know that they don't posses a sense of smell, but do have a strong sense of taste and feel [41]. Although, dolphins have good eyesight both in and out of the water, they rely heavily on their hearing and sound production for their survival.

In the late 1960's, the first in depth experiments on dolphin communication were presented through the work of Dreher, Evans and Lilly [14, 29]. In these experiments, the researchers studied the vocalizations of dolphins attempting a first contextual analysis as well as an effort to identify and classify the different

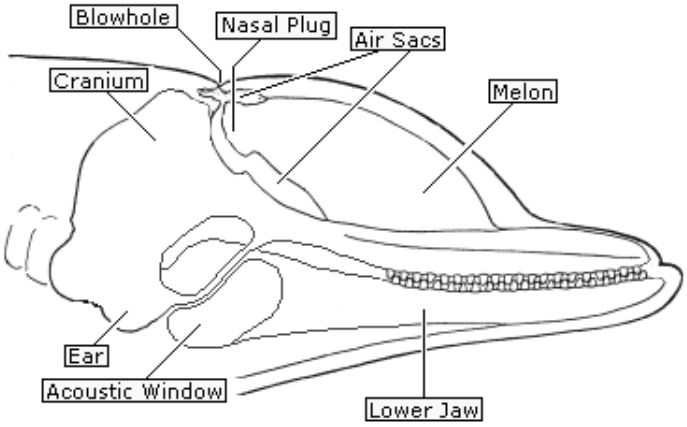


Figure 1.2: Dolphin sound production mechanism

types of vocalizations. Research on the understanding of marine mammals was further promoted with the enactment of the Marine Mammal Protection Act (MMPA) that came into effect in 1972. Much of the physiological work on the mechanics of sound production in dolphins was performed by Sam Ridgway [47] then employed by the Navy and is now continued by Cranford [12]. A simplified description of the mechanics of sound in dolphins is shown in Figure 1.2 [1]. Sound in dolphins is received through their lower jaw and transferred via surrounding fatty tissue to the middle ear.

Dolphins produce sound by passing air through air sacs that are located in their head. The sound is produced mainly through the blowhole and is guided with the help of the melon due to strong reflections within the skull [41].

Overall, scientists can consider two main types of vocalizations in the bottlenose dolphin based on their function:

- Navigational, Foraging Vocalizations: Echolocation/Clicks/Sonar

[1] Image from *Dolphin Acoustical Structure* (1991) Scheifele, P. M. NUSC TR 3080

- Social/Communication Vocalizations: Whistles, bursts/squeeks

These two broad categories have long been of interest in the scientific community. Although a lot of attention has been given to the echolocation abilities of dolphins due to their usefulness in the Navy, researchers have always been attracted and have tried to decipher dolphins' social vocalizations.

The origins and purpose of whistle calls has led to a widespread debate in the marine mammal community giving rise to two different camps concerning what is known as the "Signature whistle hypothesis" that was first proposed by Caldwell and Caldwell [7, 8]. Tyack [53] along with Caldwell and Caldwell, believe that dolphins possess an individual identifier, similar to a name in humans. On the other hand, Reiss and McCowan [45, 46] attempt to refute the findings of Tyack and the existence of a signature whistle, by showing that it is most likely that dolphins possess a shared whistle type each with individual distinctiveness. These whistles could give rise to different contextual instances e.g. alarm whistles, but overall appear to have similar contours across different pods. This could be analogous to human dialects that stem from the same language.

Although the exploration of the validity of the "Signature whistle hypothesis" is beyond the scope of this work, it is worth noting that in the analysis that will follow there is an inclination to approach whistle vocalizations as belonging to larger sets that are shared across all dolphins.

It is also worth noting that when it comes to dolphins' social vocalizations, research done by Caldwell et al [9] has indicated a change in their repertoire given their state of freedom, e.g. captive or wild. Apparently, dolphins have

a strong mimicking ability and when in captivity, usually in small groups of individuals, tend to alter their vocalizations in variety and frequency range to match those of their humans trainers. Unfortunately, most of the subsequent work on the analysis of whistles and bursts is based on captive dolphins, thus not allowing us to have a clear view of the full range and diversity of their calls. Although there is currently a huge interest in collecting sound recordings from the wild, scientists have yet to come up with solutions to several problems in acquiring the right data, e.g. right hardware, noise suppression, individual tagging. Reports of these efforts [24] however, indicate that wild dolphins have a much larger repertoire of whistles and bursts and are far more vocal and in wider frequency ranges than captive dolphins.

In this work there is an effort of trying to provide a generalized view of the analysis of social calls in dolphins. Recordings of dolphins both in captivity and in the wild are used and their differences are highlighted in order to get a better understanding of their interactions and increasing complexity.

In conclusion, I will try to provide a clear understanding of the tools that are needed in order to analyze and extract social vocalizations from long recordings. Several issues such as low Signal-to Noise ratio (SNR) and interferences classify this as a challenging task. There will be an effort to show case these highly modulated signals that are of great interest and provide comparative results that identify their variations while allowing the reader to get a good idea in recognizing these vocalizations.

Chapter 2

Technical Introduction

"Those images that yet
Fresh images beget,
That dolphin-torn, that gong-tormented sea".
Byzantium by William Butler Yeats

Since the 1950's when Ken S. Norris [38] first started studying whales and dolphins leading him to verify the existence of echolocation, several other researchers have been fascinated with the study of dolphin sounds. One needs to mention John Lilly [29] who, some might say, was responsible for the widely accepted view of dolphins in the public. Through his experiments, Lilly, pushed the intellectual limits and tried to provide a framework for the dolphin "language".

With the advances in technology and with the Navy showing extreme interest in dolphin research, scientists from different fields started converging towards the study of dolphin vocalizations. Engineers and mathematicians

started collaborating with marine biologists and animal cognitionists in order to provide a computationally feasible analysis of dolphin sounds; while exploring through several algorithms the existence or not of patterns within those sounds.

The ability to record dolphin vocalizations with the use of hydrophones led to the existence of massive quantities of data with diverse characteristics. The difficulty of indexing and analyzing it is clearly one of the major obstacles in this field of study. Without the use of automatic and robust algorithms it would be extremely difficult to provide an in depth analysis of dolphin vocalizations since the simple task of detecting the presence of a desired sound in a recording is really daunting when performed manually. It is also worth noting that as of now there is no standard in the field when it comes to obtaining underwater recordings. This leads to non-uniform recordings due to the use of different hardware which could have different sampling frequencies or lower tolerance in noise. With the analysis of dolphin recordings *in silico*, better insights can be obtained and large scale information on the different vocalizations can be extracted yielding meaningful conclusions.

As mentioned in Chapter 1 dolphin vocalizations can be distinguished into two broad categories depending on their functionality and context. Marine mammals and dolphins especially, utilize sound for two main purposes: navigation/foreaging and interaction/communication. Although the former has been of great interest in the scientific community due to the intricate mechanisms of echolocation (a.k.a sonar), this work focuses on the analysis of vocalizations that are used for inter and intra-species interaction.

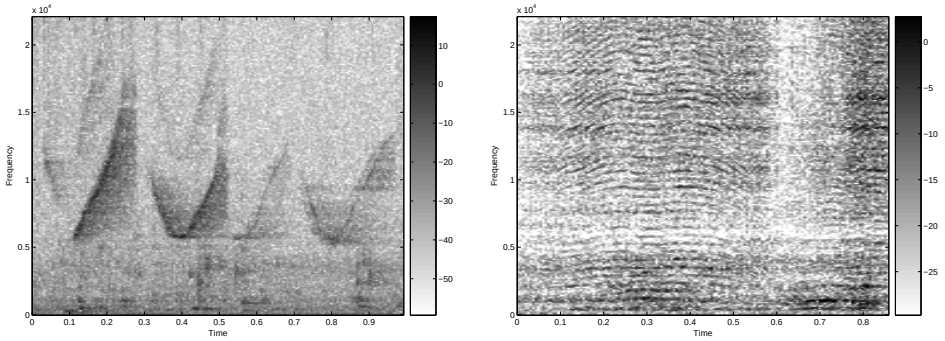
When interacting in a social context, dolphins use two main types of calls:

- Whistles: Whistle calls are highly AM-FM modulated signals. They are narrow band high-pitched signals and usually appear tonal. In the case of bottlenose dolphins the fundamental frequency is most commonly centered at around $7kHz$. However, accounting for all dolphin species, whistle calls can be found to have a range of about $80kHz$.

- Bursts: Burst calls are also AM-FM modulated signals. They are low-pitched and highly harmonic wideband signals. In the case of bottlenose dolphins the fundamental frequency is most commonly centered at around $700Hz$. Burst calls are often seen after a long string of click pulses (echolocation). In general their pitch can range anywhere from $50Hz$ to $1kHz$ throughout the different species.

It is worth noting that although the categories above describe the two main vocalizations of dolphins that serve a social function, researchers have also identified other types of wide band signals such as squeels, pops etc. However, scientists focus on whistles and bursts since they are more frequently used in dolphins' interactions. Figure 2.1(a) is an example of a dolphin's whistle call and figure 2.1(b) is an example of a burst call.

From Figures 2.1(a), 2.1(b) it is clear that there are inherent differences between these two types of calls. It is worth noting though that researchers have focused mostly on the analysis of whistle calls since they exhibit a wide variation on their contour indicating that there might be possible information encoding. This has led to a debate in the marine mammal community con-



(a) Whistle call (b) Burst call

Figure 2.1: Examples of a whistle and burst call

cerning the nature of the information present in these whistle calls. Tyack et al [53] support the theory that dolphins posses a "signature whistle" that they utilize as an identifier e.g. like humans use a name and that call remains the same throughout their life. On the other hand, Reiss et al [45, 46] argue that dolphins have a predominant type of whistle call is shared among individuals. Moreover, Dreher and Evans [14] show that there is a predominant whistle call used by several species of cetaceans. However, these whistle types exhibit slight differences across different pods that might indicate other type of information e.g. origin, species e.t.c. In this work, I tend to favor the latter theory since it provides a more general approach in the analysis of whistle calls. However, no conclusive decision is made since there needs to be specific data with contextual tags in order to explore the whistle hypothesis. Such data, unfortunately, is yet to be collected by researchers who are on the field.

This brings up a very important aspect when it comes to the collection of dolphin vocalizations in both captive and wild environments. There exist many researchers who employ their own hydrophones and recording equipment

in order to get sound recordings from dolphins. However, all the collected data remains unexploited since it requires some individual to hand label the presence of calls, a task that is understandably daunting. In order to alleviate this problem engineers who work on passive acoustics have employed several methodologies and created software packages that marine biologists can take with them in the field and use for their different needs.

These efforts are mainly supported by the Navy and especially the Office of Naval Research (ONR). Several software packages have emerged recently. They have as a primary goal to facilitate data collection and analysis for non-engineers. One of the most widely popular packages is Ishmael [35] by D. K. Mellinger. Ishmael is a visualization tool that allows on site recording and off line analysis. It provides semi-automatic detection based on the work by D. K. Mellinger [36, 37] on template matching of whistle calls. One of its most popular applications is the ability to provide automatic localization techniques on recordings of clicks. Overall, Ishmael is publicly available software package that allows for simple tasks and is also compatible with Matlab.

A more sophisticated package has been produced by Cornell's Ornithology lab based on the work by Chris Clark [11]. Raven: Interactive Sound Analysis Software, is a privately licensed software package that also provides the ability to record sounds and visualize them in real time. Due to their collaboration, Raven also includes similar call detection schemes like Ishmael. The main difference is that Raven is mostly focused on the acquisition and visualization of multiple signals instead of the processing and automatic extraction of the desired calls.

Another effort coming from the same lab at Cornell is XBAT: Extensible Bioacoustic Tool. This is also an open source project and is based on the Matlab platform. Once again this package does not provide sophisticated solutions for the automatic detection and tracking of dolphin vocalizations. However, its simplicity and extensibility allows for a growing number of not just users, but also developers.

Finally, it is worth noting the work of Serge Masse. His creation of Leafy Seadragon [33, 32], an open source software package for interactive dolphin communication research, actually adds an interesting component in the collection of dolphin vocalizations. His work aside from allowing the visualization of sound it also allows for an interaction between the user and dolphins with the playback of recorded sounds and thus allowing for the creation of contextual data.

This was just an overview of those packages that have gained a wide acceptance from the scientific community due to their applications and ease of use. There are several other attempts by researchers around the world. These efforts will eventually yield an all inclusive, robust and automatic tool for the real time recording, detection and analysis of dolphin vocalizations. However, most of the work when it comes to the intricate algorithms for the extraction of the desired calls, are in a primitive state. The field has yet to embrace the more advanced methodologies from machine learning and pattern analysis. Algorithms that have been extensively used in speech and music processing are slowly explored in the marine mammal community. This work focuses on applying several algorithms, that have not been used before, on dolphin

recordings.

Focusing on detection and tracking of dolphin vocalizations, several issues arise that highlight the difficulties of the task at hand. One of the most difficult problems to overcome is the existence of noise originating from different sources. Firstly, there is noise from the surrounding environment e.g. other marine life, bottom reflections. Also, there is hardware noise present from either the recording equipment or even worse the boat. This of course is true for recordings of wild dolphins. In the case of captive dolphins the conditions can be controlled slightly better. Overall, though we are dealing with signals of low Signal to Noise ratio (SNR). One more interesting aspect of dolphin vocalizations is that because they are social animals they tend to vocalize creating major overlaps. That increases the difficulty of tracking the calls of a single individual. This is not a problem that can be easily solved, but several approaches are proposed in this work that can at least resolve simple cases of overlaps.

So far, when it comes to detection of dolphin whistles and bursts in long recordings there exist two types of methodologies. Firstly, there is the approach of a simple energy threshold used on the spectrogram of the calls. This is popular due to its simplicity, but requires manual interaction for the fine tuning of the free parameter, the threshold. Secondly, there is a class of techniques based on template matching of several kernels. Both Mellinger [36, 37] and Clark [36, 37] are proponents of this method which implies the existence of a call "lexicon" in order to account for all the different types of calls. Once the right kernel is found then a simple autocorrelation with the target record-

ing will provide us with the matching calls. Clearly, this methodology has the drawback of requiring prior knowledge of the desired call in order to create the template/kernel.

Continuing, there hasn't been much advancement in the area of dolphin call tracking. This requires the use of both parametric and non-parametric techniques in order to achieve the best results given the inherent difficulty of the problem. So far, slight advances have been made to extract a desired frequency contour using cross-correlation procedures while once again requiring a semi-automatic way of capturing the desired calls.

2.1 Related work

As mentioned earlier, scientists have so far identified 33 species of dolphins. Although this work is based only on vocalizations of the bottlenose dolphin, thus rendering it species specific, it is important to note efforts that have been made by fellow scientists in the analysis of other species of dolphins.

Clearly, other species of dolphins have different vocal characteristics. However, several methodologies that have been proposed in the existing literature could be modified in order to account for the different types of calls of the bottlenose dolphin. Especially when dealing with recordings obtained in the wild one can assume that it is hard to isolate different species and thus it would be really useful to have an automatic species classifier. Extensive work has been done by Roch [48, 52] on classifying different types of dolphins e.g. Risso's dolphin and Pacific white-sided dolphin using features extracted from

their echolocation clicks. These methodologies are based on the use of machine learning tools on features extracted from the echolocation clicks.

In Chapter 1 a distinction was made between calls for social purposes and calls for navigation purposes. The latter, are of extreme interest in the scientific community for localization, survival and navigation tasks for these marine mammals. Echolocation clicks could provide species specific information and are considered one of the most intriguing acoustic abilities of dolphins and whales. Several methodologies based on localization techniques using inter-click times, have been proposed by Olivier and Glotin [19, 30] that will aid in the creation of tracking systems for pods in the wild. Also, Stylianou [28] has offered significant advancements with the use of phase information and energy operators in the task of detecting the onsets of these clicks. Continuing, I proposed a system for estimating the number of marine mammals by clustering features extracted from echolocation clicks [22]. All the above mentioned systems constitute the growing interest for the exploration of marine mammal vocalizations.

Finally, it is worth noting that in terms of interaction vocalizations in other species of dolphins, the work of Brown [4, 5] on the automatic classification of killer whale stereotypical pulsed vocalizations, is an indicator of the major improvements that can be accomplished in the field with the use of tools that are widely popular in speech and music processing. Dealing with such extreme phenomena as biphonation [6] e.g. two independent simultaneous sources or a single source producing a sound with two fundamental frequencies resulting in overlaps of pulsed calls, highlights the several approaches in the field that

will lead to a better understanding of marine mammals in general.

2.2 Summary of contributions

In this work I try to provide a variety of methods and approaches into dealing with the aforementioned problems when it comes to the detection and tracking of dolphin vocalizations. Several machine learning techniques are used and several informative features are extracted in order to achieve a generic and robust detector as well as a pitch tracker for both whistles and bursts. Specifically, Chapter 3 gives a detailed description of the data and the features that are used throughout this work. Chapter 4 focuses on the detection of whistles and bursts in long recordings. Several advancements are proposed with the use of optimization techniques such as gradient descent and also, other classifiers are explored, such as Gaussian Mixture Models (GMM) and Support Vector Machines (SVM). Continuing, Chapter 5 proposes two novel systems for pitch extraction of whistles and bursts and actually approaches one of the hardest problems in the field, overlap resolution. These systems have already been published in special issues of Canadian Acoustics [23] and Applied acoustics [21] respectively. Finally, Chapter 6 provides comparative conclusions and future attempts on all the proposed systems in this work.

Chapter 3

Data overview

> "Diviner than the dolphin is nothing yet created..."
>
> Halieutica by Oppian

The most important thing when dealing with the analysis of dolphin vocalizations is the type and quality of the existing data. As mentioned in Chapter 2 there exists a large volume of recordings. However, due to the lack of a standardized methodology when it comes to the collection of the data e.g. hardware specifications, species etc. researchers are yet to take full advantage of the possible insights the dolphin vocalizations might convey. Acquiring the right kind of data is not only a difficult task, but probably the most deciding factor when it comes to the success of any theory and methodology applied on said data.

Dolphin recordings can be classified into two broad categories given their environment:

- Captive dolphin recordings

- Wild dolphin recordings

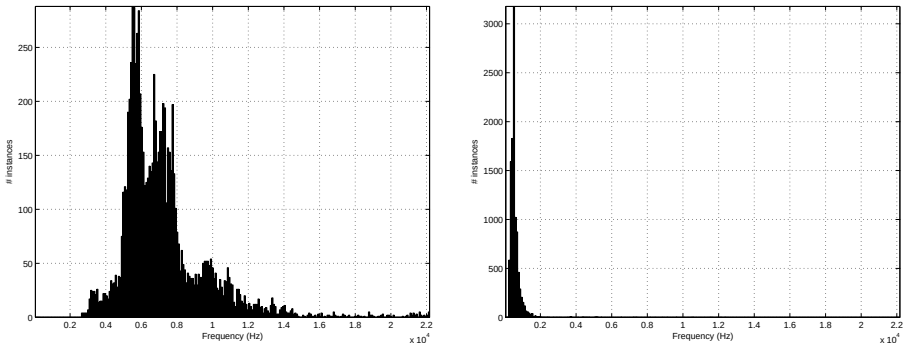
Each of these categories has both advantages and disadvantages as one could imagine. In the case of captive dolphin vocalizations there is an inherent advantage that arises from the observer's ability to control the environment. Labeling becomes easier as all individuals are visible thus favoring the use of data loggers or even bubble stream production approaches. Also there is a much better quality of the acquired signal, empirically of the order of 8dB. Continuing, the researcher can limit the number of individuals in the tank and control the different social interactions. However, the drawback of captive dolphin recordings is that the calls that are produced are biased from either the existence of trainers or other factors associated with dolphins in captivity.

On the other hand, wild dolphin recordings offer a huge diversity in their call repertoire and also provide a much better understanding of their interactions and use of sound. In wild environments dolphins are not isolated and field researchers are often witnesses to unknown behavioral patterns e.g. hostility. However, it is almost impossible for a field researcher who is located on a boat to be able to clearly observe and provide contextual as well as individual labels for what is happening in any situation. The lack of tagging in these situations is one of the biggest obstacles when it comes to applying machine learning techniques on the data since in order to be able to get a good discrimination training data and ground truth is needed.

Moreover, wild dolphin recordings suffer from really low SNR due to the existence of several noise sources. Interference from different species of marine life also constitute a type of noise since when trying to analyze and detect

the desired dolphin calls one needs to take into account the existence of these calls that might be similar to the ones we are trying to extract. Overall, although each type of data has its own characteristics it is fair to say that captive dolphin recordings appear to be slightly more manageable for audio processing techniques and machine learning algorithms. In this work, most of the methodologies are applied to captive dolphin recordings and a small comparison is obtained using a smaller set of recordings from wild dolphins.

I obtained captive dolphin recordings from Dr. Diana Reiss. These are recordings from the keyboard experiment [45] performed at Marine World-Africa aquarium in Vallejo California. There are two adult female dolphins present in the pools. The adult females were wild caught. Also there are two young male dolphins that were born in captivity. There is interaction with the researchers through the use of a sound emitting underwater keyboard that the dolphins are trained to use in order to obtain toys and/or food from the observers. All recordings were obtained with the use of a hydrophone. The recording is approximately 3 hours long of which approximately 40% is vocalized. The segments that contain whistles and bursts are diverse and of different levels of difficulty. In order to extract training data for the different methodologies applied, 100 whistle calls and 100 burst calls were manually extracted from the recording. A semi-automatic process was used in order to obtain the ground truth on a per frame level. This semi-automatic way of extracting the pitch ensured a better quality of the training data and reduced the time spent on the arduous process of labeling. I used de Chevigne's YIN [13] which is an easy time domain technique based on a modified and normalized autocor-



(a) Histogram of whistle calls (b) Histogram of burst calls

Figure 3.1: Histogram of dolphin calls

relation function. Once the pitch tracker was employed over the data, then I proceeded to manually inspect the results and correct the errors that were found.

Table 3.1 provides some of the basic statistics of the extracted calls that were used as training data. As seen, both the average whistle and burst have a duration of around $0.5sec$. The recording of captive dolphins has an average $12dB$ SNR. It is worth noting that the SNR was obtained by averaging the peak SNR computed at each frame of each call through the use of the short time autocorrelation function, as seen in Equation 3.1.

$$pSNR = dB(\frac{r(\tau = p)}{r(0) - r(\tau = p)}) \tag{3.1}$$

Where $r(0)$ is the energy of the signal plus the noise and $r(\tau = p)$ is the energy of the signal with period at lag $\tau = p$.

Finally as mentioned earlier the main difference between burst and whistles is the fundamental frequency. Most whistles have a fundamental frequency of

	Whistles	Bursts
Length	$0.55sec$	$0.62sec$
SNR	$12.4dB$	$12.8dB$
Pitch	$6.9kHz$	$713Hz$

Table 3.1: Average statistics for whistle and burst calls

approximately $7kHz$ while bursts have a fundamental frequency of around $700Hz$. This can be better visualized in Figures 3.1(a), 3.1(b) which show the histogram of the per frame pitch for the training data.

In order to obtain a better generalization as well as an understanding of the different systems created using captive dolphin recordings, I also applied the methodologies on a small set of wild dolphin recordings. These recordings have distinct differences not only from one another, but also from the ones the systems were created on.

There are two different sets of wild dolphin recordings. The first set was obtained from the Macaulay Library. Created by Cornell's Ornithology Lab it houses one of the biggest selection of natural animal sounds for the study of animal behavior. The sounds are accessible to the public and can be obtained directly from the library. There are six stereo recordings of wild bottlenose dolphins totaling approximately 1 hour and 7 minutes. These recordings were obtained in August of 2003 by Barlow Jay at Clipperton island in the North Pacific ocean. A Sony-TCD-D7 recorder was used along with a AN/SSQ-57 hydrophone. All recordings have a sampling rate of $48kHz$.

In these recordings there are multiple individuals vocalizing, but there is no knowledge of the size of the pod. In the recordings there are instances of whistles and clicks, but no bursts. This could be a characteristic of wild

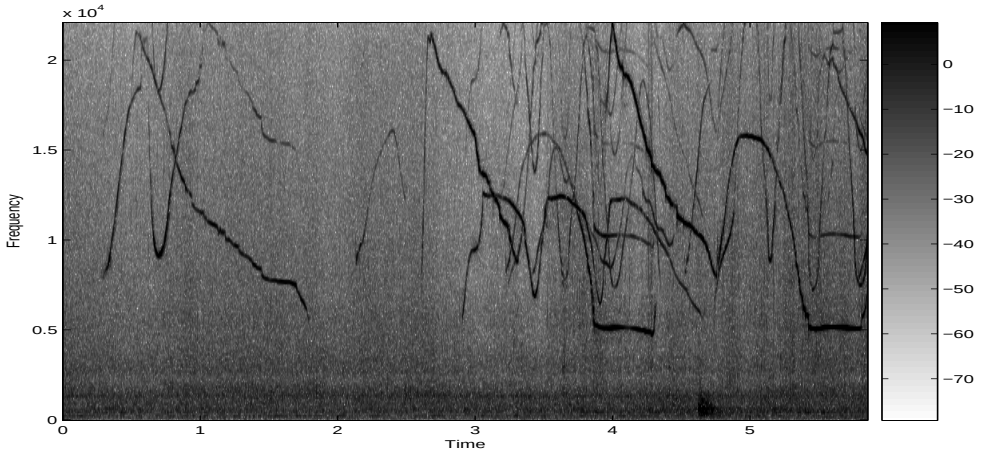


Figure 3.2: Example of wild dolphin recording

dolphins or it could be that the hardware was not sensitive enough to capture the low pitched sounds. The recordings have an SNR of 9dB An example of the data is seen in Figure 3.2. It is clear form the figure that the extraction of single whistles is not possible due to the severe overlaps that are present. it is worth noting that this is probably the hardest case that any automatic system for the detection and tracking of dolphin vocalizations would have to encounter.

In conclusion, it has been shown that there are inherent differences between wild and captive dolphin recordings. In order to create a robust and automatic system for the detection and tracking of the desired dolphin calls I chose to use the cleaner captive dolphin recordings. Although these recordings do not offer the realistic environment in which the system would need to operate in, they provide a good enough framework for the exploration and analysis of dolphin vocalizations. It is also worth noting that it is most likely to identify possible contextual vocalizations in a captive environment since it is easier to

control and monitor the different individuals. However, I also use wild dolphin recordings in order to test the limits of the systems that will be showcased in this work. That will allow for a better understanding of the task at hand and will highlight possible future recommendations.

3.1 Extracting Features from the Data

One of the most challenging tasks when dealing with machine learning algorithms is the extraction of suitable features from the data that will ensure the success of the algorithm. Finding good and descriptive features depends highly not only on the task at hand, but also on the nature of the data. Choosing the correct features will strongly alleviate possible issues when dealing with discrimination tasks, while providing a better understanding of the process that drives the data.

As mentioned in Chapter 2 the goal of this work is to create suitable algorithms for the automatic detection and tracking of the desired dolphin vocalizations. Although accuracy of detection is really important, one needs to also take into consideration the computational cost of extracting such features. Ideally there needs to be a balance between the two since the ultimate goal is for the system to be used on the field by non-engineers.

Dolphin vocalizations are highly AM-FM modulated signals described in the existing literature seen in Eq. 3.2.

$$x(t) = a(t)cos(2\pi f(t)) \tag{3.2}$$

Where $a(t)$ is the amplitude function and $f(t)$ is the modulating signal. In order to visualize these signals the 512 point spectrogram is used with a 50% overlap, yielding a frequency resolution of $86.13Hz$. Given the inherent time-frequency resolution trade-off in the creation of the spectrogram, these parameters provided a more balanced scheme for the dolphin calls. All recordings have a sampling frequency, $F_s = 44100Hz$

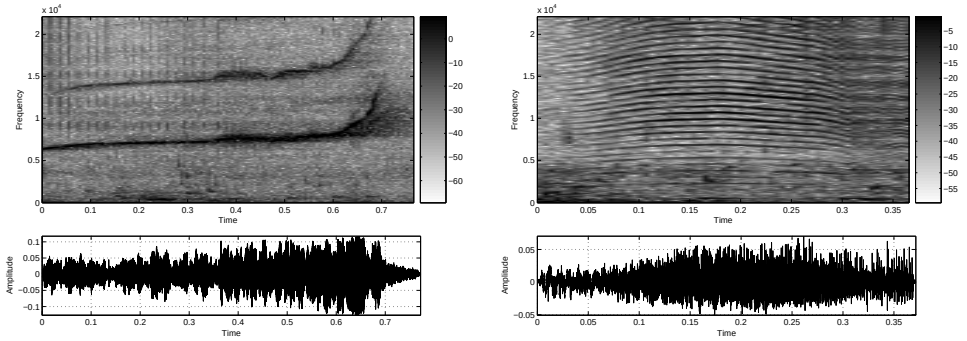
3.1.1 The Spectrogram

One of the easiest and most common features to use in audio processing is the raw spectrogram. The spectrogram is obtained form the short-time Fourier Transform (STFT) [39] and is derived through Eq. 3.3.

$$X[k,m] = \sum_{n=0}^{N-1} x[n]w[n-mL]\exp(-j\frac{2\pi k(n-mL)}{N}) \tag{3.3}$$

Where $x[n]$ is the time domain signal, $w[n]$ is the window, usually Hamming [39], L is the length of the window, k, m are the frequency and time indices respectively and N is the number of points for the calculation of the Fast-Fourier Transform (FFT).

Depicting the amount of energy at every frequency it provides a good starting point for discrimination tasks. One of the major drawbacks when using the spectrogram as a feature is that it usually suffers from being a high-dimensional space e.g. in our case 257-dimensional vector. That of course can lead to computation tractability issues although there are several ways to alleviate such problems. Spectrograms are often depicted using the logarithm



(a) Whistle call spectrogram (top) and waveform (bottom)

(b) Burst call spectrogram (top) and waveform (bottom)

Figure 3.3: Examples of dolphin calls

e.g. in dB of the absolute value. However, when using the spectrogram as a feature for a discrimination task there is an inclination to use its linear form e.g. simple magnitude, for several reasons such as better separation or creating a positive valued feature vector etc. Figures 3.3(a), 3.3(b) show examples of spectrograms as well as the time-domain waveforms for a whistle and burst dolphin call respectively.

3.1.2 Energy Summation

Originating from the spectrogram, many other features can be extracted. One of the most popular ones when it comes to detecting a desired audio clip in long recordings, is the summation of the amplitude/energy across all frequency channels. This is better shown in Equation 3.4

$$SE[m] = \sum_{k} |X[k, m]| \tag{3.4}$$

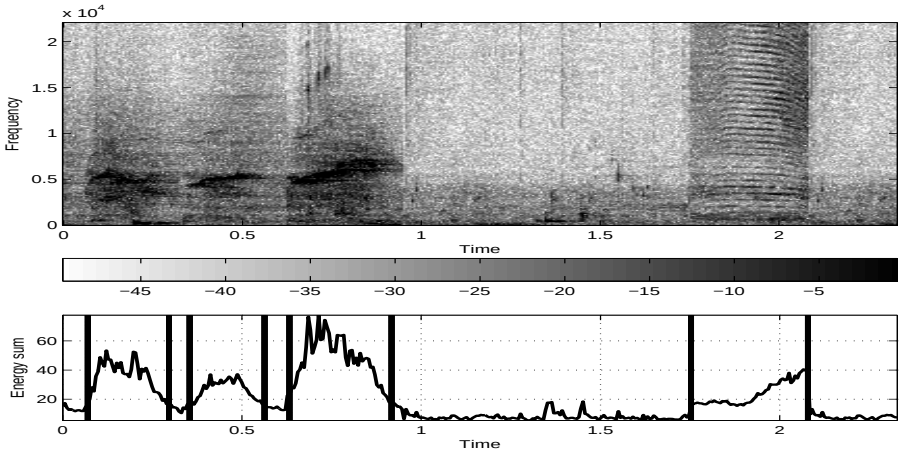


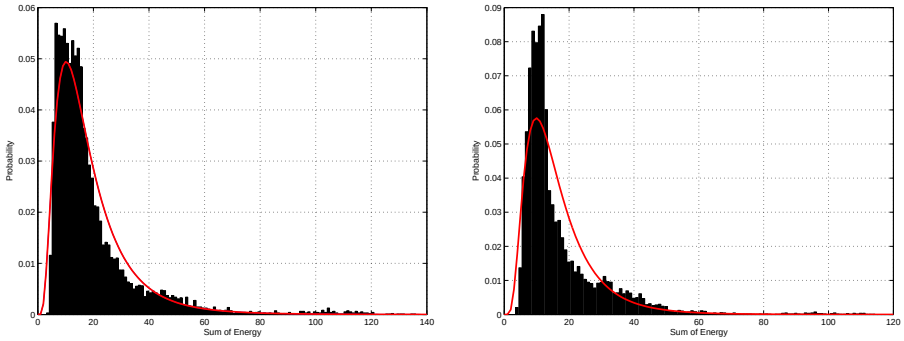
Figure 3.4: Example of energy summation feature

Once again the summation is computed on the magnitude of the STFT.

Energy summation is widely popular due to its simplicity and ability to discriminate between high and low energy signals. It has been used in many applications both in speech segmentation/detection as well as in previous attempts to detect marine mammal sounds. [37, 11]

Figure 3.4 gives a clear example of the spectrogram (top) and the energy summation (bottom) for a clip that includes both desired calls, bursts and whistles. The solid vertical lines show the boundaries of the calls that we want to detect. One can clearly see a distinction between the foreground, desired calls, and the background, ambient noise. Overall, the energy summation will adequately represent high energy signals

It is worth noting that there are several drawbacks when it comes to using the energy summation feature. As mentioned before, the success of the feature depends on the energy difference between the calls to be detected and the background. However, it will not provide a good discrimination between



(a) Energy distribution for whistle calls (b) Energy distribution for burst calls

Figure 3.5: Energy distribution for dolphin calls

two high energy signals. For example the use of the energy feature is not recommended for the distinction between burst and whistle dolphin calls. This is explained visually in Figures 3.5(a), 3.5(b) where the acquired distributions for both whistles and bursts are depicted. Clearly, there is no separation between the two different types of calls as the distributions are overlapping. Both distributions appear to be close to log-normal for the energy feature as shown with the solid red lines.

Continuing, another issue with the energy summation feature is its high sensitivity to existing background noise and/or interferences. As mentioned in Chapter 1 recordings of dolphin vocalizations suffer from a low SNR as well as the existence of several interferences. One of the most prominent ones being the echolocation vocalizations/clicks. These trains of pulses, that appear as energy across all frequencies in the spectrogram can lead to a number of false positives in the detection process. A simple example is shown in Figure 3.6. The spectrogram (top) and energy summation feature (bottom) are shown along with red arrows indication the location of clicks. Clicks appear

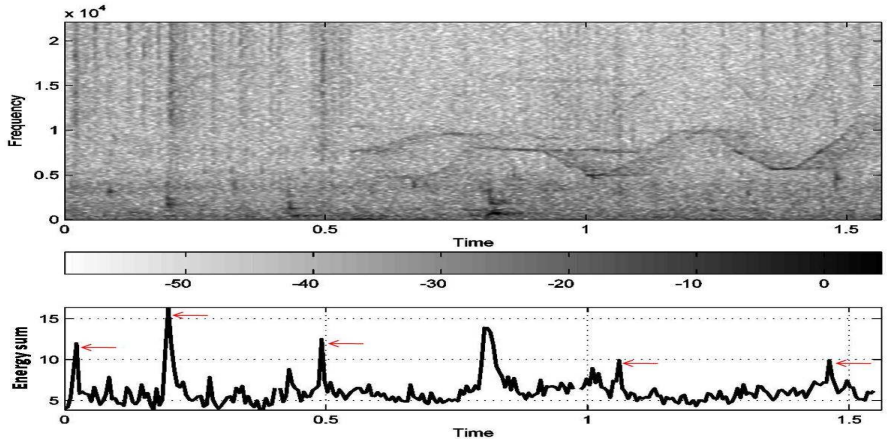


Figure 3.6: Example of energy summation feature with clicks shown in red arrows

as high peaks in the energy summation function and unless some type of post-processing manipulation is applied then a simple detection system will mistake these frames as desired call frames.

In conclusion, the use of the energy summation offers both advantages and disadvantages. It provides a really simple and easily computable feature that will adequately discriminate between simple cases of calls and noisy background. Unfortunately, its sensitivity to interferences indicates that it will not provide us with a good discrimination since most dolphin recordings are fairly complex. However, as it will be shown in the next Chapter, there are ways of adapting the energy summation feature in order to highlight the desired calls while suppressing the background noise.

3.1.3 Cepstral Features

Whistles and bursts are periodic signals. As mentioned earlier, one of the distinct characteristics that distinguish between the two types of calls is their period. Whistles are high-pitched sounds while bursts are low-pitched sounds, as shown in Figures 3.1(a), 3.1(b). When dealing with pitched signals, the cepstrum has long been used in speech and music processing in order to locate the pitch and compactly describe a given signal. First introduced in 1963 by Bogert et al [2] it is based on the source and filter model were the goal becomes to separate and identify the excitation/source from the resonance/filter. The real cepstrum, most commonly used in audio processing, is described in Equation 3.5.

$$c_n = \sum_l (\log |\sum_k x[n] e^{-j\frac{2\pi kn}{N}}|) e^{j\frac{2\pi ln}{N}} \tag{3.5}$$

Where $\sum_k x[n] \exp^{-j\frac{2\pi kn}{N}}$ describes the Discrete Fourier Transform (DFT). Basically the real cepstrum is the inverse Fourier transform of the logarithm of the absolute magnitude of a given signal's Fourier transform.

A better way of understanding Eq. 3.5 is to think about it as a deconvolution process, since the convolution of the source and filter, or in general of any two signals, is expressed as the addition of their cepstra. In general when computing the real cepstrum, the desire is to separate the resonance from the fine structure and actually reveal a pitch peak, which is due to the fact that both whistles and bursts are periodic signals. An example of computing the cepstrum is shown in Figure 3.7. The spectrogram and time slice are shown in vertical red line (top) along with the spectral magnitude of the time slice

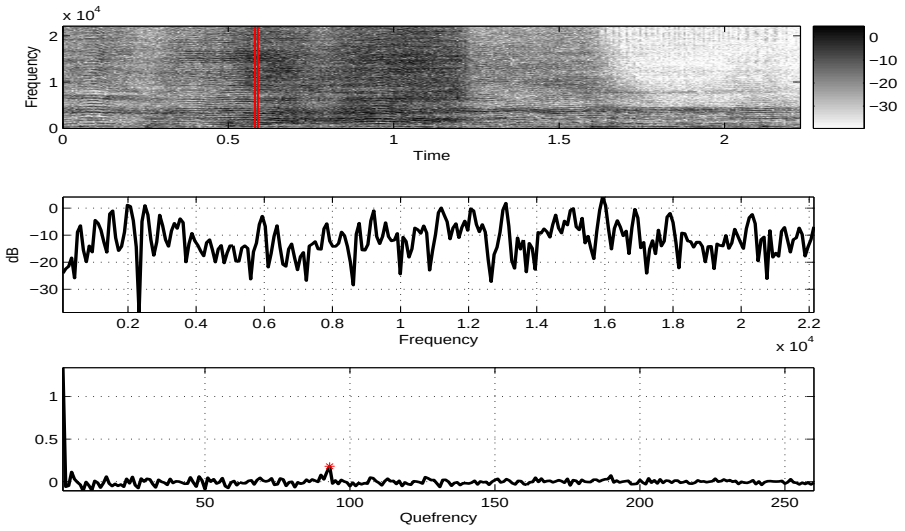


Figure 3.7: Example of the cepstrum for a burst call

(middle) and real cepstrum (bottom) with the pitch peak indicated by a red cross. In order to get a better visualization only one frame of a burst call is shown. From the ground truth I know that the pitch of the burst varies at approximately $480Hz$. This can be verified by the pitch peak which is located at quefrency 93 which translates to approximately $475Hz$.

When computing the cepstrum for machine learning applications it is customary to exclude the first cepstral coefficient, c_0, as it captures the average energy of the signal which reflects factors not relevant to classification such as the gain used in the particular recording device. Because the calculation of the cepstrum is a deconvolution process it offers a crude decorrelation of the data. This leads to better fits on the given data while minimizing the number of parameters that are used. Of course this also has an impact on the computational efficiency of any applied algorithm. Clearly, the number of cepstral coefficients that are chosen will determine the dimensionality of

the feature vector allowing the control of the trade off between accuracy and computation.

Figure 3.8 provides examples of the cepstral coefficients that were used as a feature vector for both burst and whistle calls. In the case of the whistle call the pitch peaks are evident in the low coefficients, as expected, since whistles are high-pitched signals. On the other hand, the pitch peaks appear in higher coefficients for a burst call given that these are low-pitched sounds. Although the cepstrum provides a highly compact way of extracting the pitch of a periodic signal, it is highly sensitive to possible constant background noise e.g. hardware noise. In practice, for interference with a stationary and sparse spectrum, it may be possible to employ a simple pre-processing filter to remove it from the recordings.

In conclusion, the cepstrum provides one of the most appropriate feature selections for the given data. Its ability to capture the different pitch and describe the underlying signal compactly, while possibly allowing for a dimensionality reduction of the feature vector, explains its wide popularity in speech and music signal processing.

3.1.4 Other features

Having seen and explored the most commonly used features for detection and tracking of periodic signals, it should be noted that there also exist several other variations of these features that can provide us with a good selection for the task at hand.

Continuing, a promising feature that has been used in speech processing

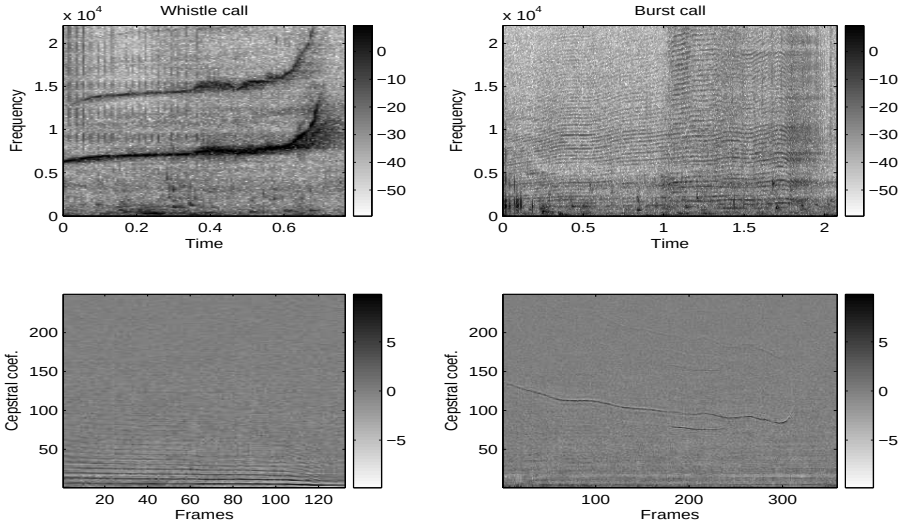


Figure 3.8: Example of spectrogram (top) and cepstral coefficients (botttom) for a whistle (left column) and burst call (right column)

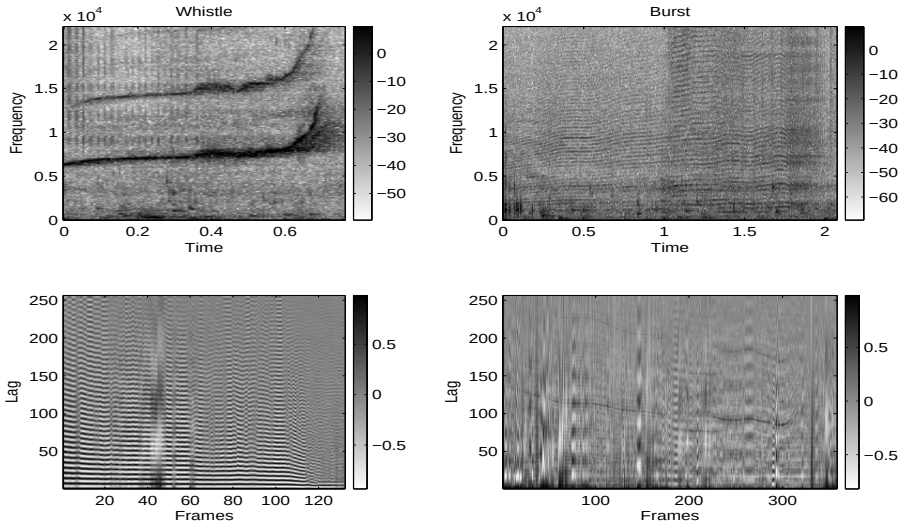


Figure 3.9: Example of spectrogram (top) and STAC (bottom) for a whistle (left column) and burst call (right column)

and provides information on the pitch of a periodic signal is the short-time autocorrelation function (STAC). A normalized version of the autocorrelation function is used in de Chevigne's YIN [13] that I will also apply on dolphin vocalizations. Equation 3.6 defines the short-time autocorrelation function where ℓ is the lag, n is the local time index, and N is the length of the signal.

$$R[\ell, n] = \sum_{m=0}^{N} x[m+n]x[m+n+\ell] \tag{3.6}$$

An example of the STAC is shown in Figure 3.9 for a whistle and burst call respectively. The differences in pitch between whistles and bursts are easily observed in the short-time autocorrelation function. By simple peak picking one can extract the pitch at a per frame level and actually track the frequency contour of each signal. The autocorrelation function is a suitable feature when there are no overlaps of periodic signals or other interferences e.g. clicks. However, there are several pre-processing normalization techniques that can be employed in order to enhance the signal.

In conclusion, a variety of features have been presented that will aid in the detection and tracking tasks of dolphin vocalizations. The usefulness of these features will be evaluated in the following chapters. They were chosen not only because of their ability to compactly describe the underlying data, but also because of their popularity in the speech and music processing community.

Chapter 4

Call detection

> "The voice of the dolphin in air is like that of the human
> in that they can pronounce vowels and combinations of vowels,
> but have difficulties with the consonants".
> Historia Animalium by Aristotle

Automatically detecting dolphin vocalizations in long recordings has long been a focus of research amongst engineers in the field. Given the amounts of data that exist from both captive and wild environments, scientists are faced with the difficult task of analyzing the data in an off line manner while relying on semi-automatic ways for the detection of the desired calls.

Several approaches have been explored by many researchers in order to get the most accurate detection system that will be able to adapt to the diversity of the recordings. As mentioned in Chapter 2, dolphin recordings suffer from low SNR as well as multiple interferences e.g. clicks, bottom reflection. These are attributes that can hinder any detection process leading to false positives.

In Chapters 1, 2, I discussed the basic differences between the two broad types of dolphin vocalizations. Throughout this work the term call detection refers to the detection of social vocalizations, which are comprised of whistles and bursts. Navigational vocalizations e.g. clicks are of no interest and are treated as an interference. Since detection is performed at a per frame level, a call is defined as a single, bounded element, that may or may not have harmonics. In the case of overlaps between calls, no attempt to separate the calls will be performed and the detection will be considered successful if the total length of the overlap is identified e.g. beginning of call A to end of call B.

The most common detection systems in the existing literature are based on two simple methodologies.

- Energy thresholding
- Kernel cross-correlation

Software packages like Ishmael, Raven and XBAT use energy thresholding in a semi-automatic way. The linear spectrogram is added across all frequency channels yielding the energy summation feature that was previously described in Chapter 3, and the user then chooses a threshold while visually inspecting the results.

On the other hand, kernel cross-correlation depends on the creation of synthetic kernels that resemble specific types of whistles or even the use of manually extracted whistles. These are then cross-correlated with the rest of the signal and possible other instances are therefore highlighted. This proce-

dure could be perceived as a simple template matching technique were a prior knowledge of the template is required.

Given that this work focuses on a robust automatic detection scheme that will not require the user's interaction, only energy thresholding techniques are taken under consideration. In the sections that follow several algorithms are proposed and compared in order to obtain the best detection scheme for dolphin vocalizations.

4.1 Detection using energy thresholding

In Chapter 3 I presented the training and testing data on which these algorithms are going to be employed. Also, a description of the energy feature was given in Eq. 3.4. Although in most recordings both whistles and bursts appear interchangeably, there is a need to separate the detection process between them in order to obtain a better insight to the limitations of the features as well as the classifier. Initially, results of the detection task will be presented solely on whistles, then on bursts alone, and finally on the combination of the two.

All results from the detection process will be presented through Receiver Operating Characteristic curves (ROC) [15]. These are obtained by altering a parameter of the system e.g. threshold and plotting the curve of the true positive rate (TPR) vs. the false positive rate (FPR) at each instance of the parameter. Continuing, the success of the classification/detection task will be reflected on the area under the ROC curve (AUC) [3]. This is considered to be

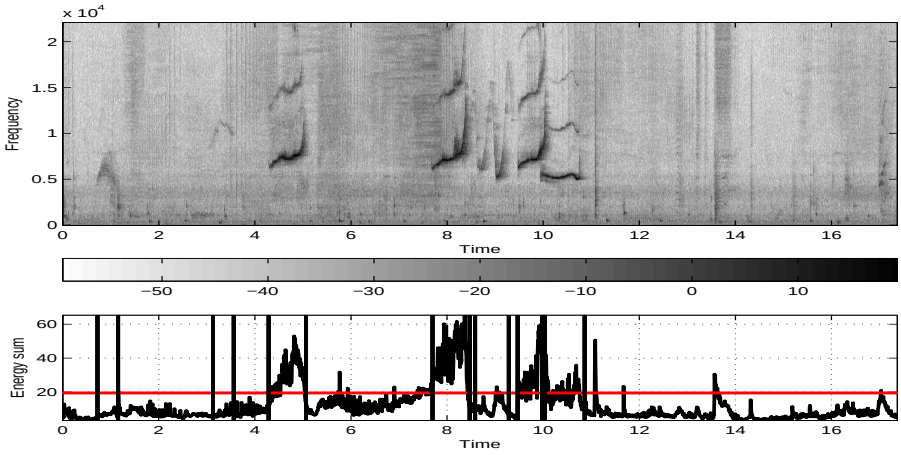


Figure 4.1: Whistle detection using energy thresholding

Energy feature	
Call Type	Threshold (ϑ_w)
Whistle	19.5 Arbitrary Units

Table 4.1: Energy threshold for whistle calls

a reliable metric that will allow us to directly compare the different proposed systems.

4.1.1 Energy thresholding for whistle calls

Whistles are high-pitched AM-FM modulated signals. They don't always have harmonics and are usually centered at around $7kHz$. Dolphins appear to use whistle calls more frequently than burst calls in their social interactions. Having manually extracted 100 whistle calls as the training data, I can obtain an initial simple threshold, Table 4.1, represented by the mean of the energy feature across the training data.

Once the threshold has been obtained the detection system simply classifies

Energy feature	
Call Type	AUC
Whistle	83.94%

Table 4.2: AUC for whistle calls

as call frames anything that is greater than ϑ_w and dismisses anything lower than that. Figure 4.1 shows an example of the threshold system for a dolphin clip that only includes whistle calls. Specifically, the spectrogram (top) and the energy summation feature (bottom) are shown along with the red horizontal line representing the threshold and the vertical lines indicating the location of the boundaries for every whistle present in the clip. It is clear from the figure that some clicks in the background will actually be misclassified as call frames. Also, the energy threshold cannot capture those whistles that are of low energy leading to false negatives. This can be seen in Figure 4.1 at around $3sec$ where the threshold fails to capture the whistle call present in the recording.

Figure 4.2 shows the ROC curve for a longer test clip that only has whistle calls. The desire in ROC curves is to have a high TPR for low values of FPR. The dashed line in the figure represents the boundary for random classification. If the ROC spans below the dashed line then the system has overall failed to correctly detect the whistle frames within the recording. Table 4.2 provides the overall AUC for the whistle detection task indicating the success of the system.

Clearly, the energy feature and threshold system appears to work fairly well for the detection of whistle calls in long recordings. If an error of approximately 20% is allowed then from Figure 4.2 one can see that a TPR of 70% is achieved.

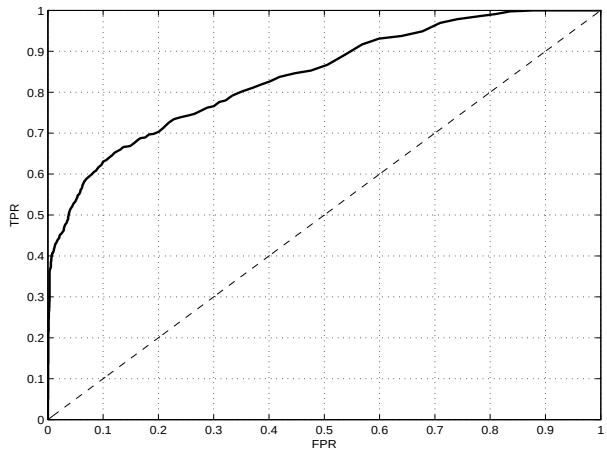


Figure 4.2: ROC for whistle calls using energy feature

As mentioned before, the reason behind the obtained results is that low energy whistles as well as click interferences are misclassified. It is likely that the use of a different feature could increase the system's accuracy.

4.1.2 Energy thresholding for burst calls

Bursts are low-pitched AM-FM modulated signals. They always have harmonics an their pitch is centered at around $700Hz$.Bursts are less frequently used by dolphins and are seen mostly in captive environments. As in the case for whistles, 100 bursts were manually extracted as training data. A similar energy threshold was obtained by taking the mean of the energy feature across all the training data. This is shown in Table 4.3.

Figure 4.3 depicts an example of the threshold system for a dolphin clip that only includes burst calls. Specifically, the spectrogram (top) and the energy summation feature (bottom) are shown along with the red horizontal line representing the threshold, ϑ_b, and the vertical lines indicating the location

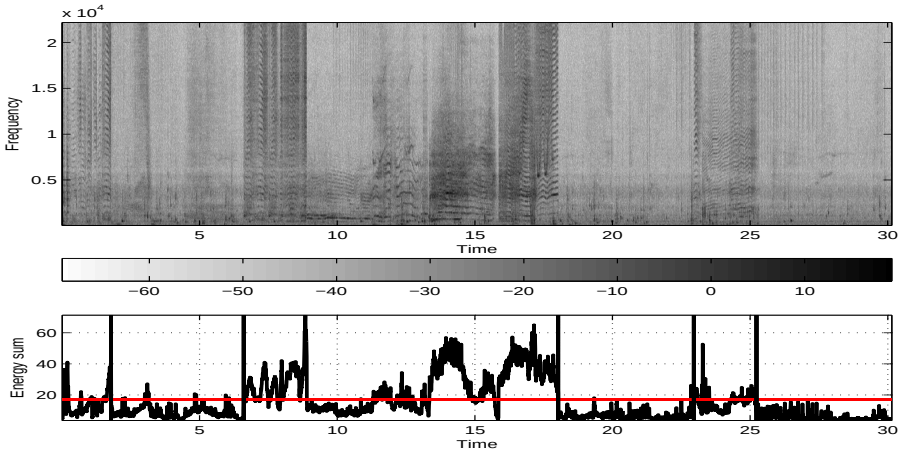


Figure 4.3: Burst detection using energy threshold

Energy feature	
Call Type	Threshold (ϑ_b)
Burst	17.07 Arbitrary Units

Table 4.3: Energy threshold for burst calls

of the boundaries for every burst present in the clip. Once again burst calls that are of low energy will not get correctly identified using the simple energy threshold and there are some instances of background noise being wrongly classified as call frames.

The system was applied to a test clip that only has burst calls. The resulting ROC is shown in Figure 4.4. Also, Table 4.4 provides the AUC metric for the classification process. Overall, it appears that the burst calls are better represented using the energy feature and threshold. If, for example, we allow for a 20% error then we get approximately 90% FPR. Interestingly, bursts appear to be better detected than whistles. This could be because burst calls have their energy spread out to multiple frequency channels, thus the energy summation will exceed the threshold. Another interesting factor is that bursts

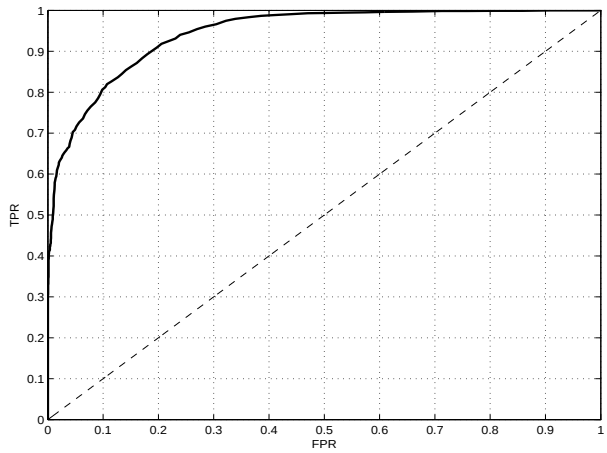


Figure 4.4: ROC for burst calls using energy feature

Energy feature	
Call Type	AUC
Burst	94.46%

Table 4.4: AUC for whistle calls

tend to have a less smooth distribution of their energy compared to whistles.

4.1.3 Energy thresholding for whistle and burst calls

Having shown the ability of the system on each of the calls individually, it is important to see how the energy feature and threshold perform on realistic recordings were both whistles and bursts are present and no distinction is made between the two. In this case, the recordings are more difficult to decipher since there is interference from clicks and overlaps between the calls. The energy threshold, ϑ_{bw} is, once again, obtained through the training data and is shown in Table 4.5. Continuing, an example of a typical clip that comprises both whistles and bursts is depicted in Figure 4.5. Specifically, the spectrogram

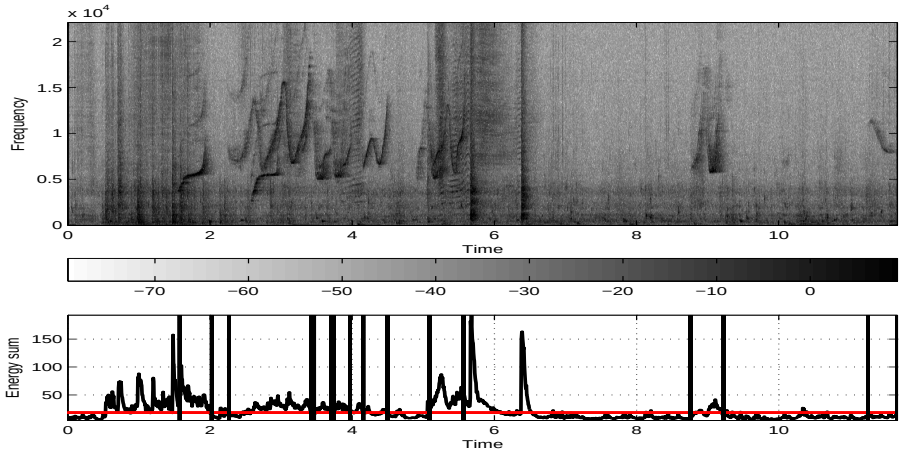


Figure 4.5: Whistle and burst detection using energy threshold

Energy feature	
Call Type	Threshold (ϑ_{bw})
Burst, Whistle	18.22

Table 4.5: Energy threshold for burst calls

(top) and the energy summation feature (bottom) are shown along with the red horizontal line representing the threshold, $\vartheta_b w$, and the vertical lines indicating the location of the boundaries for every burst present in the clip.

From Figure 4.5 it is evident that there will be instances where noise or click frames are misclassified as call frames. The overlaps between the calls tend to favor this method since it drives the energy to cross the threshold. However, the use of single detectors for each type of call is not recommended since they wouldn't offer any improvement in the overall detection.

The results of the energy detection system performed only on whistles or bursts appear to be very optimistic. When the system is applied to realistic clips where both types of calls are present and multiple overlaps occur then the simple energy threshold does not provide an adequate detection accuracy.

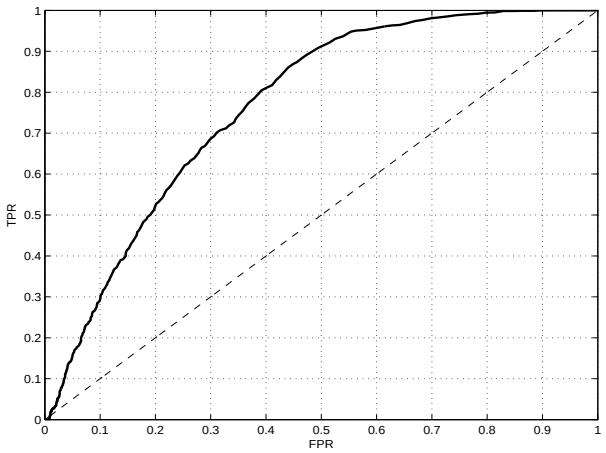


Figure 4.6: ROC for whistle and burst calls using energy feature

Energy feature	
Call Type	AUC
Whistles, Bursts	76.75%

Table 4.6: AUC for whistle and burst calls

It appears that applying the energy feature detection on test data consisting of a realistic vocal environment that includes other, non-communicative calls, a lower overall accuracy is obtained. This is shown better in Figure 4.6 and the overall AUC metric is given in Table 4.6.

Finally, when applying the energy feature detector on test data that represent a more realistic vocal environment, a lower overall accuracy is obtained. A closer look at the ROC reveals that for an allowable FPR of about 30% there is a TPR of approximately 70%. Table 4.7 shows the confusion matrix for the specific clip and it is evident that 30% of call frames get misclassified as noise frames, while another 30% of noise frames get misclassified as call frames. So the energy feature is biased towards low energy call frames and high energy noise frames. One could imagine that in wild dolphin recordings

Confusion matrix when allowing for a FPR of 30%

	Positive Result	Negative Result
Positive Label	68.69%	31.31%
Negative Label	30.01%	69.99%

Table 4.7: Confusion matrix

the energy feature and threshold detector will fail to correctly identify the desired calls. The usefulness of the energy detection scheme cannot be dismissed since it works adequately well for simple cases that have minimal overlaps or interference. Its popularity is justified by the fact that its mostly used in a semi-supervised manner allowing for a variable threshold set by the user.

4.2 Energy detection optimization

Previously, the use of the energy feature and threshold was explored for the detection of dolphin calls in recordings. The energy feature, Eq. 3.4, appears to perform well for simple cases where there are few overlaps and interferences. However, the energy feature tends to misclassify low energy calls or high energy background noise. One simplification contributing to these confusions is that each frequency channel is assumed to have equal importance when computing the feature. In Figures 3.1(a), 3.1(b) the pitch distribution of both whistle and burst calls is shown and one can see that they have distinct and characteristic pitches. This implies that the energy of bursts and whistles might be located in select frequency channels.

Clearly if one can indicate that most of the energy is located in specific frequency channels for both whistles and bursts then only those channels could

be considered when calculating the energy feature. That would lead to a better detection scheme since it would minimize misclassifications due to the noisy background.

In order to explore the above assertion, a scheme for measuring the classification "strength" of every frequency channel is explored. To achieve that, artificial clips of $30sec$ length are created from the manually extracted training data. ROC curves are computed for each of the frequency channels and finally, the AUC metric is obtained. This procedure will yield an AUC function across the different frequency channels and hopefully will highlight the "stronger" channels e.g. channels with high AUC value.

Figures 4.7, 4.8 show the AUC function, as described above, for whistles and bursts respectively. Both figures indicate that there are indeed dominant channels where most of the information is located. Taking advantage of this information might lead to a better detection system for dolphin calls. Interestingly, whistles appear to have a clear region of channels unlike bursts that appear to have a wider range of "strong" channels. This, of course, could be attributed to the fact that bursts have harmonics at multiples of the fundamental frequency and the energy spans across those channels as well.

The question now is, how does one choose the appropriate channels. I can arbitrarily assume that the channels that exhibit an AUC above 70% will provide the best set. However, that will introduce a variable parameter in the system and might lead to over-fitting of the data. In what follows I will consider less arbitrary approaches to optimizing the use of individual channels.

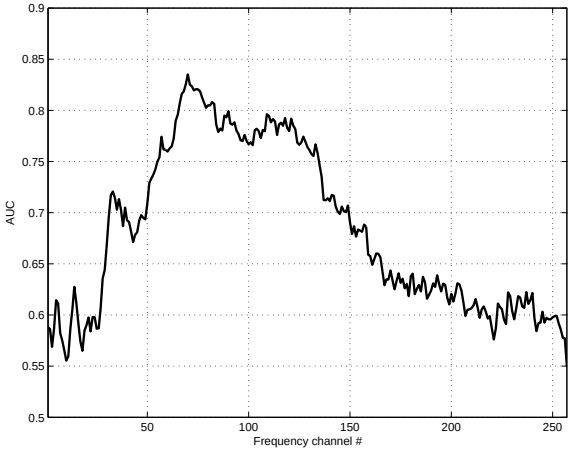


Figure 4.7: AUC function vs. frequency channels for whistle calls

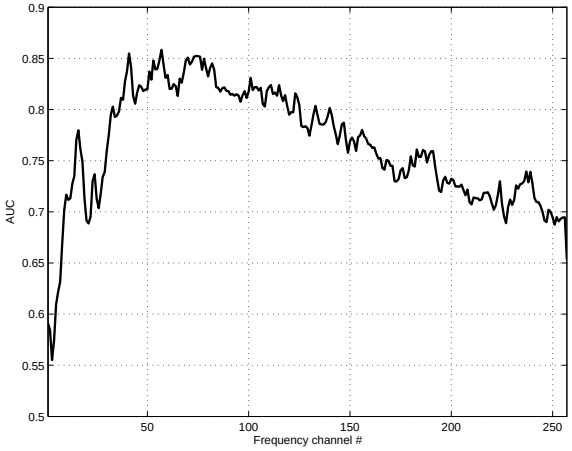


Figure 4.8: AUC function vs. frequency channels for burst calls

4.3 Energy detection using gradient descent

The AUC function indicated that there are predominant channels for both whistle and burst calls. Taking advantage of these channels will increase the detection accuracy of the energy feature system. However, choosing an arbitrary high enough value for the AUC as a threshold, that will define a suitable channel range, would add an unknown to the system. A better solution would be to theorize that given the AUC function there should be a set of weights, $\vec{w}$, that will offer the best possible combination of frequency channels so that the system can discriminate between call and noise frames.

This desired set of weights, $\vec{w}$, can be obtained using gradient descent [15, 1]. Gradient descent also known as steepest descent, is an optimization technique that became very popular with the use of neural networks. A simplistic explanation is given in Figure 4.9. Suppose we want to locate the minimum of an error function. In terms of neural networks this can be viewed as the error the network makes when classifying the training data as a function of the weights of the network. Ideally this error function needs to be minimized. Clearly, as seen in the Figure I want the error to move towards w^*. In order for that to happen we need to move along the slope of the error function, which is the path of steepest descent.

Analytically, if we consider the unthresholded perceptron [15, 1] then its output is shown in Eq. 4.1 where $\vec{x}$ is the input data.

$$o(\vec{x}) = \vec{w} \cdot \vec{x} \tag{4.1}$$

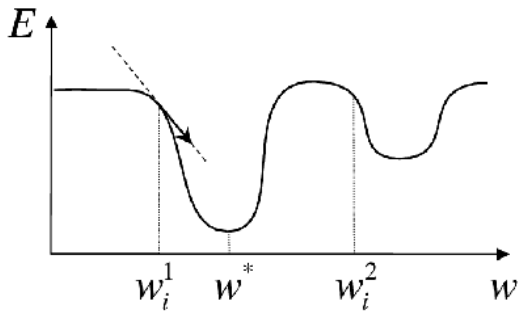


Figure 4.9: Gradient descent technique

We can consider a training error, E as the mean squared error between the target output and the actual output of the network, Eq. 4.2. That allows the error to be a function of the weights, $\vec{w}$.

$$E(\vec{w}) = \frac{1}{2}\sum_{k \in K}(t_k - o_k)^2 \tag{4.2}$$

Since E is a multi-dimensional function of $\vec{w}$ it can be viewed as a surface. The gradient of this surface, ∇E, represents a vector whose direction points to the greatest increase of the error function, E. Clearly, in order to achieve a decrease of the error I would want to move towards the opposite direction, $-\nabla E$. So basically I can obtain a new set of weights, $\vec{w}$ with the following updating rule, Eq.4.4.

$$\vec{w} \leftarrow \vec{w} - \eta \nabla E(\vec{w}) \tag{4.3}$$

This can be re-written as:

$$w_i \leftarrow w_i + \Delta w_i \tag{4.4}$$

Where $\Delta w_i = -\eta \frac{\partial E}{\partial w_i}$ and η is known as the learning rate and basically defines how big of a step is taken to reach the desired minimum. Continuing, from Equations 4.2, 4.4 one can get:

$$\begin{aligned} E(\vec{w}) &= \frac{1}{2}\sum_{k\in K}(t_k - \vec{w}\cdot\vec{x})^2 \\ \frac{\partial E}{\partial w_i} &= \sum_{k\in K}(t_k - \vec{w}\cdot\vec{x})(-x_{ik}) \Rightarrow \\ \Delta w_i &= \eta\sum_{k\in K}(t_k - o_k)x_{ik} \end{aligned} \tag{4.5}$$

It is interesting to note, though, that gradient descent has the inherent drawback of not being able to guarantee that it will converge to a global minimum. Because of that, the choice of both the learning rate, η, as well as the initialization of the weights become very important as to not "trap" the descent into a local minimum.

The algorithm described in Eqs. 4.1- 4.5 is based on the unthresholded perceptron implying that the data we are trying to manipulate is linearly separable and in this case, the globally optimal solution could be found by solving the Normal equation. However, linear separability is too limiting an assumption to make, especially when it comes to dolphin vocalizations where so far the feature distributions are overlapping indicating that there is no single surface that can successfully separate/detect the desired calls.

In order to account for the non-linearly separable case and thus create a more generalized system the threshold function of the perceptron is replaced by the sigmoid function,$\sigma(y) = \frac{1}{1+e^{-ky}}$. An interesting fact about the sigmoid

function is that $\frac{d\sigma}{dy} = k\sigma(1-\sigma)$. This of course alters the error function of Eq. 4.2. The new error function is shown in Equation 4.6.

$$E(\vec{w}) = \frac{1}{2}\sum_{k\in K}(t_k - \sigma(\vec{w}\cdot\vec{x}))^2 \tag{4.6}$$

In a similar way as described before the weight update rule is obtained and Δw_i is shown in Equation 4.7

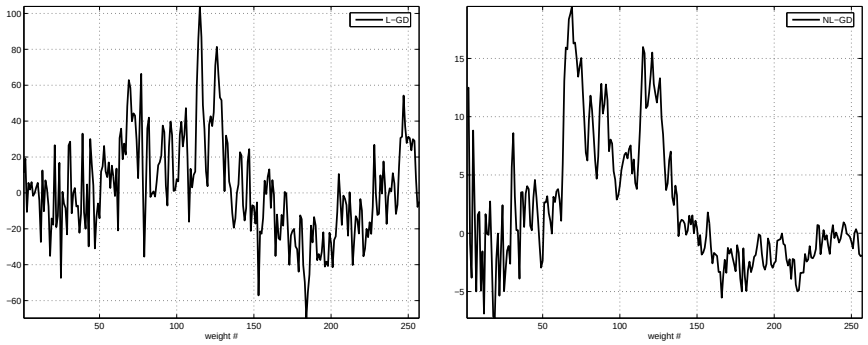
$$\Delta w_i = \eta\sum_{k\in K}((t_k - \sigma(w_k\cdot x_k))(1-\sigma(w_k\cdot x_k))\sigma(w_k\cdot x_k)\cdot x_k) \tag{4.7}$$

In the experiments that follow for the detection of whistle and burst calls, both linear (L-GD) and non-linear (NL-GD) gradient descent are used.

4.3.1 Whistle detection using gradient descent

Having extracted the whistle training data, gradient descent is applied in order to find a set of optimal weights, w, that will highlight the frequency channels that contribute the most in the detection process. Once the set of optimal weights are extracted and applied on the test data, a spectrogram based detection function similar to the energy feature is obtained, and the simple detection scheme that was introduced in the previous section is used.

In Figures 4.10(a), 4.10(b) the optimal weights, w_i are presented for both L-GD and NL-GD. These weights are for whistle calls only. In both cases the initial weights are randomly chosen and there are insignificant changes across multiple runs with different initializations. Continuing, Figure 4.11 shows the



(a) Optimal weights for whistles using linear gradient descent

(b) Optimal weights for whistles using non linear gradient descent

Figure 4.10: Optimal weights for whistle calls

spectrogram (top), and new energy feature when using the optimal weights obtained from L-GD (middle) and NL-GD (bottom) respectively. Once again the red horizontal line represents the detection threshold and the vertical lines represent the boundaries for the whistle calls. It is evident from the figures that the noisy background has been hugely suppressed when using the optimal weights. However, in light of the large number of parameters being set, there remains the possibility of overfitting the training data in the L-GD system.

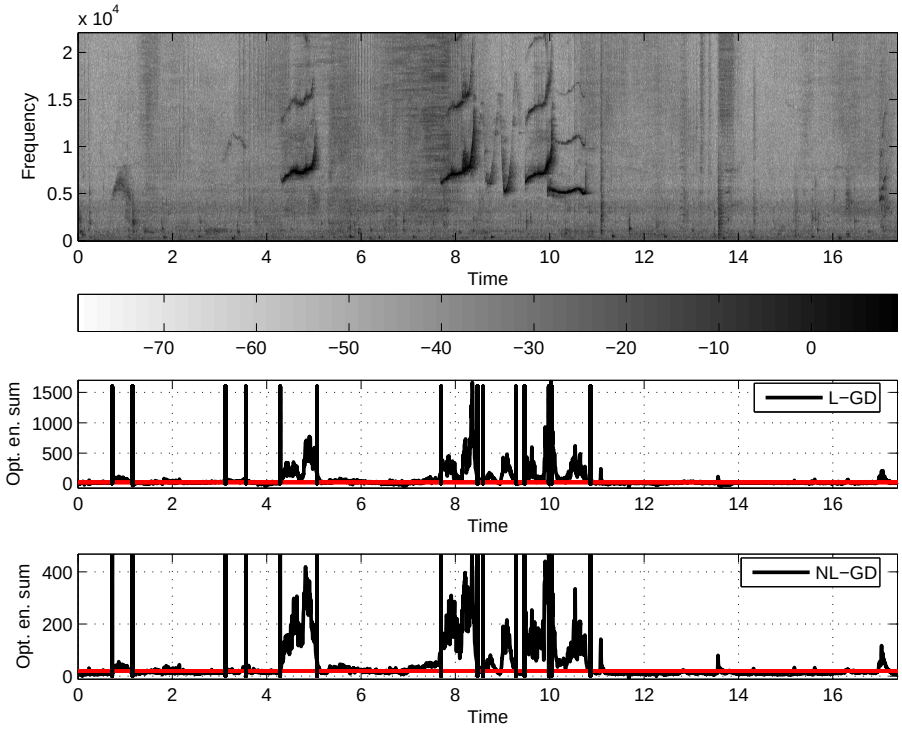


Figure 4.11: Example of spectrogram based detection function for whistles using gradient descent technique

Finally, Figures 4.12, 4.13 depict the ROC curves for a whistle clip when L-GD and NL-GD are used respectively. Table 4.8 provides the AUC metric for the two different systems.

It is evident from the computed results that both L-GD and NL-GD perform equally as well. There is a small variation of .34% that could be attributed to a statistical error e.g. weight initialization. However, one should keep in

Whistle Optimized Energy feature	
GD	AUC
Linear	94.08%
Non-linear	93.74%

Table 4.8: AUC for whistle calls using gradient descent

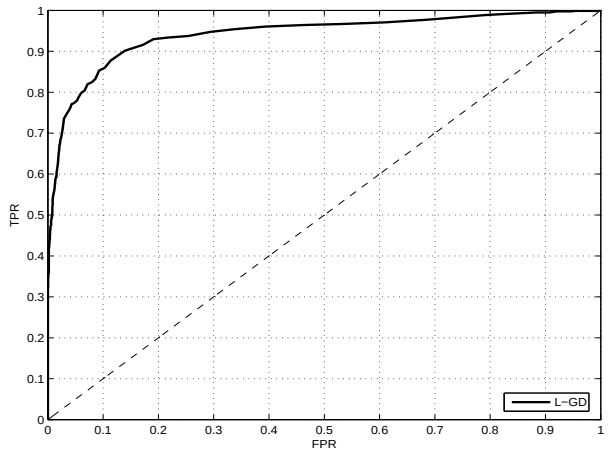


Figure 4.12: ROC for whistle calls using energy feature and L-GD

mind that the AUC measures the detection accuracy of whistles against noisy background and that the later addition of bursts might interfere with the linear separability of the data. Overall, the results for the simple energy feature detector and the optimized energy detector are similar. When looking closer to the ROC curve and if an FPR of 20% is allowed then the optimized weight system outperforms the vanilla energy system by approximately 2%.

4.3.2 Burst detection using gradient descent

The same experiments that were performed for the whistle calls are now depicted for the burst calls. A different set of optimal weights,w is extracted and the same analysis is depicted in the following figures.

In Figures 4.14(a), 4.14(b) the optimal weights, w_i are presented for both L-GD and NL-GD. These weights are for burst calls only. In both cases the initial weights are randomly chosen. Continuing, Figure 4.15 shows the spectrogram (top), and new energy feature when using the optimal weights obtained from L-

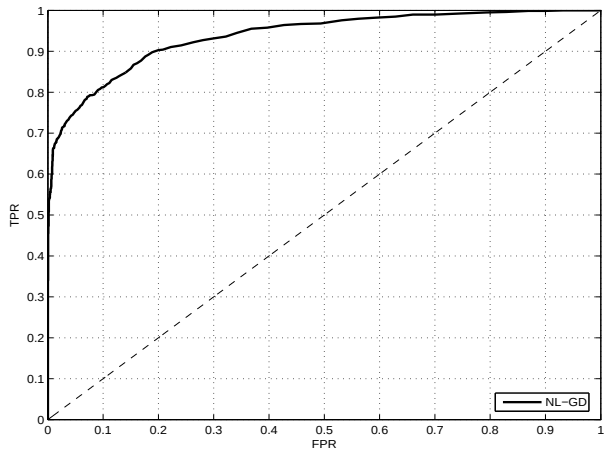


Figure 4.13: ROC for whistle calls using energy feature and NL-GD

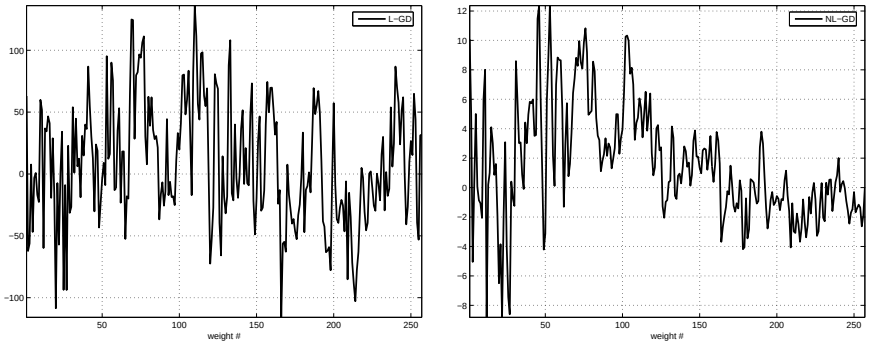
Burst Optimized Energy feature	
GD	AUC
Linear	92.58%
Non-linear	96.56%

Table 4.9: AUC for burst calls using gradient descent

GD (middle) and NL-GD (bottom) respectively. Once again the red horizontal line represents the detection threshold and the vertical lines represent the boundaries for the whistle calls.

Finally, Figures 4.16, 4.17 depict the ROC curves for a burst clip when L-GD and NL-GD are used respectively. Table 4.9 provides the AUC metric for the two different systems.

Unlike whistle calls, in burst calls the NL-GD outperforms the L-GD by 4%. Clearly the optimization technique favors their detection compared to the one of whistles, which can be attributed to the existence of harmonics. Overall, the gradient descent increases the detection accuracy by approximately 2%.



(a) Optimal weights for bursts using linear gradient descent

(b) Optimal weights for bursts using non linear gradient descent

Figure 4.14: Optimal weights for burst calls

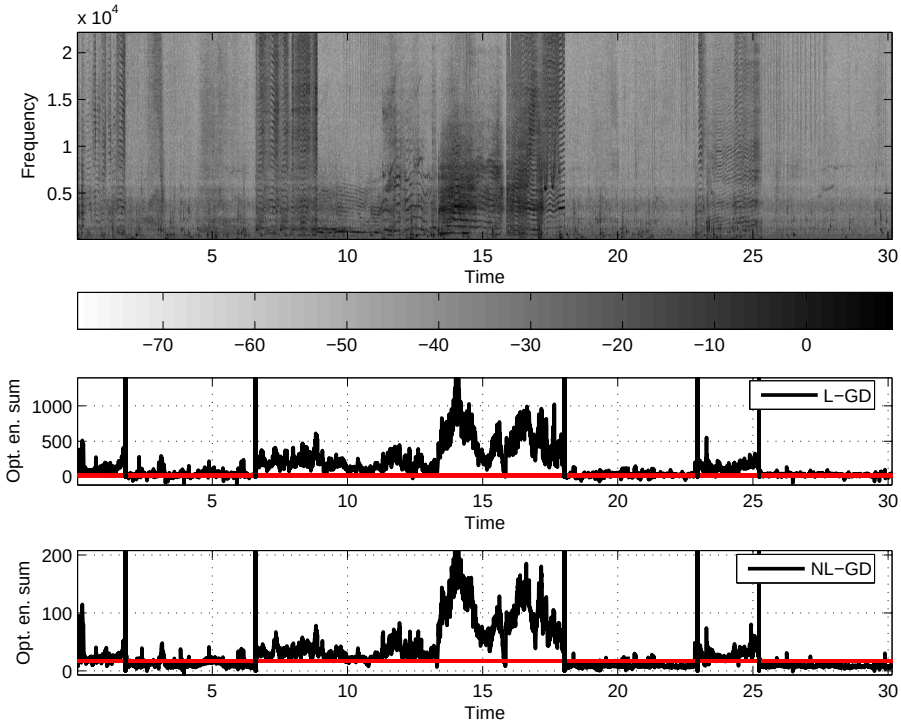


Figure 4.15: Example of spectrogram based detection function for bursts using gradient descent technique

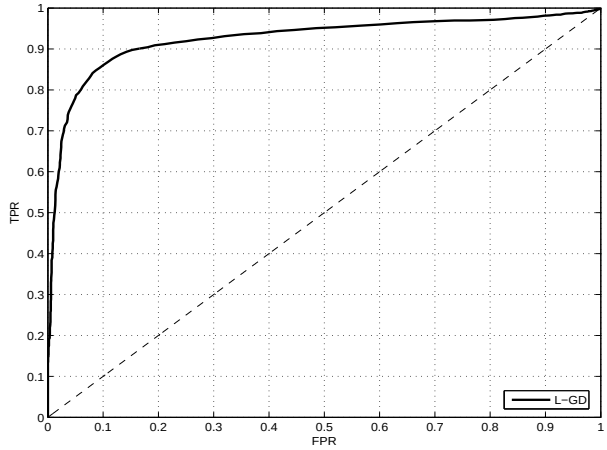


Figure 4.16: ROC for burst calls using energy feature and L-GD

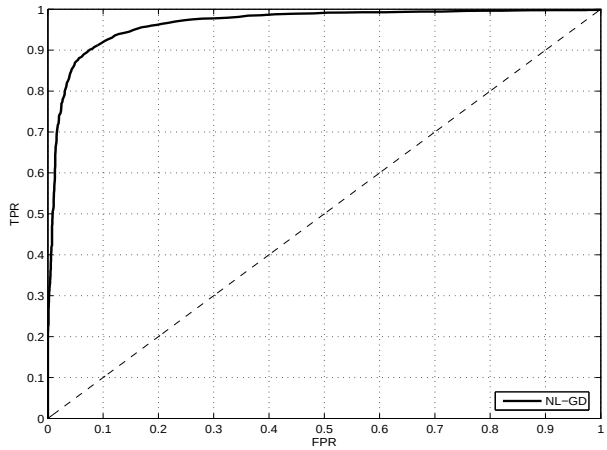


Figure 4.17: ROC for burst calls using energy feature and NL-GD

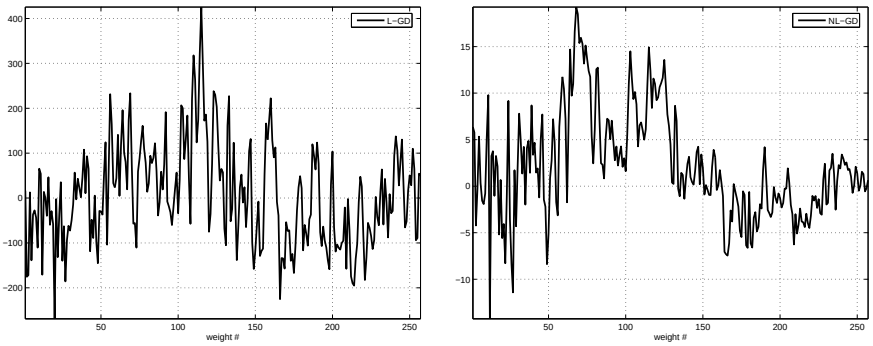
4.4 Energy detection using gradient descent for whistle and burst calls

In the previous sections it was shown that the optimization technique of gradient descent increased the detection accuracy for whistles or bursts when applied on the energy feature. The use of a set of weights, w that highlights those frequency channels with the most discriminatory strength allows for the suppression of the noisy background. However, in realistic recordings there are multiple interferences and whistles and bursts not only co-exist, but more often than not overlap. In order to explore how gradient descent performs in these cases, the same set of experiments is applied.

In Figures 4.23(a), 4.23(b) the optimal weights, w_i are presented for both L-GD and NL-GD. These weights are for burst and whistle calls. In both cases the initial weights are randomly chosen. Continuing, Figure 4.19 shows the spectrogram (top) and the new energy feature when using the optimal weights obtained from L-GD (middle) and NL-GD (bottom) respectively. Once again the red horizontal line represents the detection threshold and the vertical lines represent the boundaries for the whistle and burst calls.

Finally, Figures 4.20, 4.21 depict the ROC curves for a whistle clip when L-GD and NL-GD are used respectively. Table 4.10 provides the AUC metric for the two different systems.

As expected, the optimization technique of gradient descent increases the accuracy of the detection scheme. For example there is an approximate 2% accuracy increase when using L-GD and a 6% when using NL-GD. Clearly,



(a) Optimal weights for whistles and bursts using linear gradient descent

(b) Optimal weights for whistles and bursts using non linear gradient descent

Figure 4.18: Optimal weights for whistle and burst calls

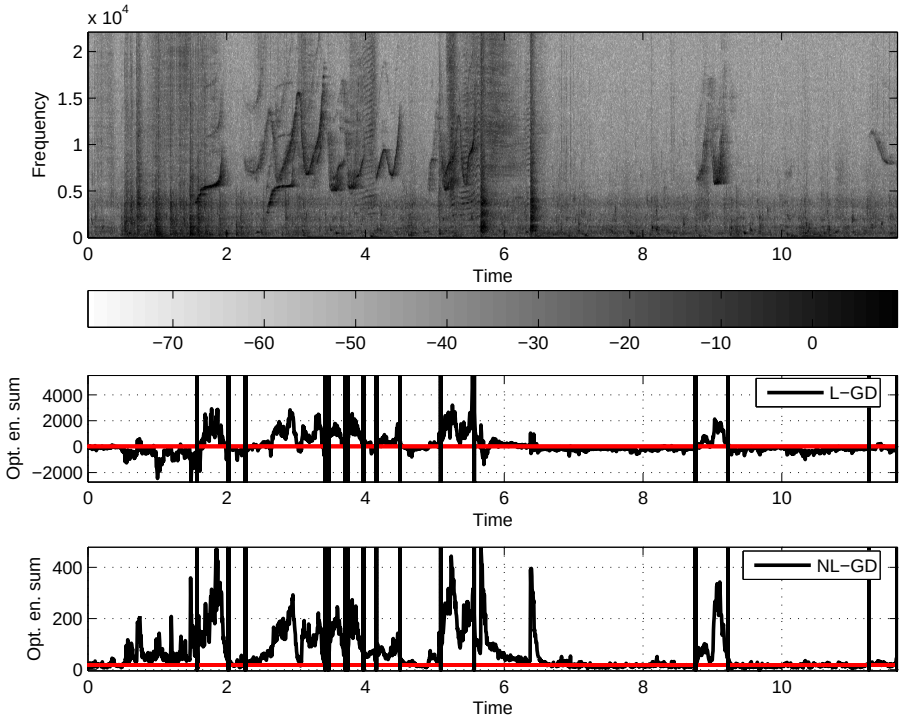


Figure 4.19: Example of spectrogram based detection function for whistles and bursts using gradient descent technique

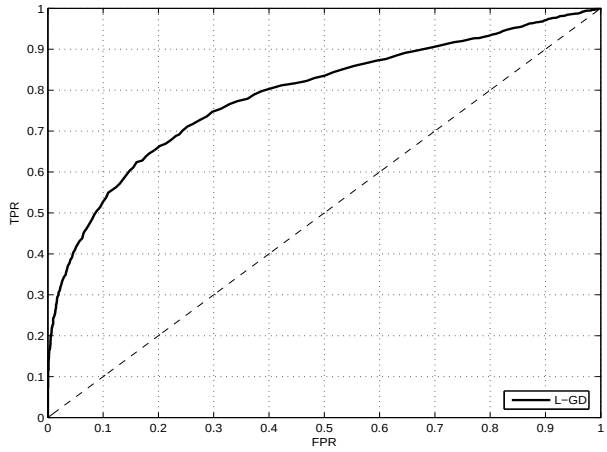


Figure 4.20: ROC for whistle and burst calls using energy feature and L-GD

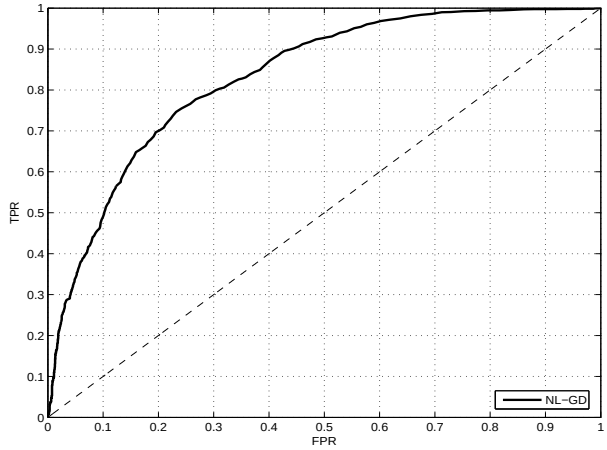


Figure 4.21: ROC for whistle and burst calls using energy feature and NL-GD

Whistle and burst Optimized Energy feature	
GD	AUC
Linear	78.94%
Non-linear	83.24%

Table 4.10: AUC for whistle and burst calls using gradient descent

the non-linearity of the optimal decision surface is now more prevalent which is why the NL-GD outperforms the L-GD.

4.5 Thresholding detection using cepstral features

The cepstrum was described in Chapter 3. It is computed as seen in Equation 3.5. The advantage of the cepstrum is that it provides pitch information for periodic signals. Based on the source filter model, it also allows for a crude decorrelation of the data. Since both whistle and burst calls are periodic signals I expect that the cepstrum will be able to effectively reveal this key, discriminating attribute of these signals, which remains hidden in the purely energy-based features used above.

In all the experiments presented in this section, the real cepstrum was used. The feature for the detection system was the summation of these coefficients in the same manner that the energy feature was extracted. Given that the optimization technique of gradient descent outperformed the use of simple features, both L-GD and NL-GD are used with the cepstrum.

For every frame of the data 50 cepstral coefficients were computed. The number of coefficients was empirically chosen because it provides a large dimensionality reduction while not compromising significantly the accuracy of the detection task. The above assertion can be better seen in Figure 4.22. Given the extracted training data, artificial clips were created and the AUC metric was obtained through the detection process for a different number of

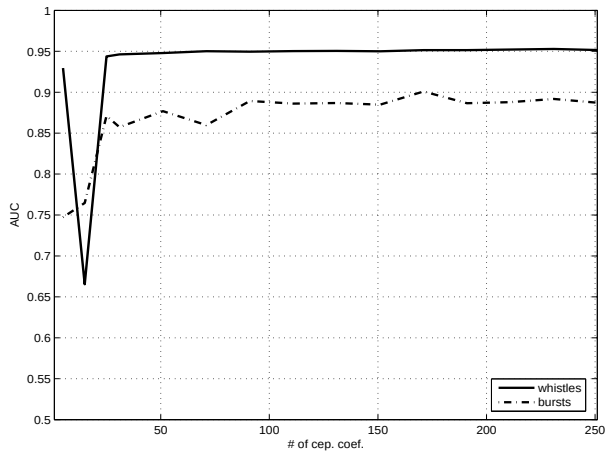


Figure 4.22: AUC vs. number of cepstral coefficients

cepstral coefficients. This was performed for whistles alone and bursts alone. In that manner an upper bound on the detection process was obtained.

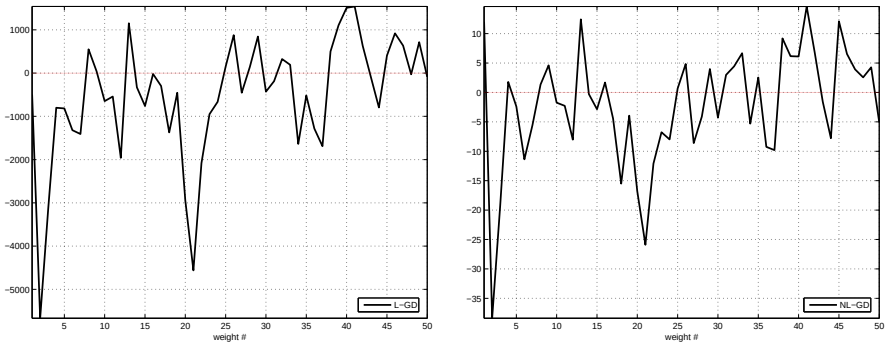
Clearly from the figure and in the case of whistle calls (solid line) there is a very small variation in the AUC when increasing the number of coefficients ($< 1\%$). In the case of burst calls the variation is in the order of 3%, but theoretically 50 coefficients will provide a detection accuracy of approximately 87.5%. However, it is interesting to note that the use of the cepstrun feature favors whistle calls rather than burst calls. This is mainly because burst calls evolve from clicks that get captured by the cepstrum.

Unlike before with the energy feature where results for bursts and whistles were given individually, to avoid redundancy with the cepstrum, only results performed on realistic clips are depicted. Once again, Figures 4.23(a), 4.23(b) show the optimal set of weights, w for both linear and non-linear gradient descent. Since I am only using 50 cepstral coefficients there are only going to be 50 weights. Continuing, Figure 4.24 provides an example of the spectrogram

(top) and optimized cepstral feature for L-GD (middle) and NL-GD (bottom). Once again, the red horizontal line represents the threshold for the detection process and the vertical black lines are the bounds for the calls in the recording.

From the figure one can see that the L-GD on the cepstral coefficients appears to lead to many false positives capturing the background noise. It is also worth mentioning that the threshold, simple average of the cepstral summation, for the detection task was obtained from the training data and was set at 16.

Finally, Figures 4.25, 4.26 depict the ROC curves for a whistle clip when L-GD and NL-GD are used respectively. Table 4.11 provides the AUC metric for the two different systems. It appears that the NL-GD on the cepstral features outperforms the L-GD by 4%. This is expected due to the fact that the cepstrum decorrelates the data allowing the underlying non-linearities to be highlighted. However, the use of the cepstral feature for simple detection of the desired calls does not offer an increased accuracy. On the contrary, there is an approximate 10% decrease on the AUC when comparing the performance of the cepstral features with the simple energy features. This can be attributed to the fact that the cepstrum is very sensitive to the type of noise and interference e.g. clicks present in the recordings. Overall, the use of the cepstrum for simple detection with a threshold is fair as it seems to not perform well for the case of burst calls.



(a) Optimal weights for whistles and bursts using linear gradient descent with cepstrum

(b) Optimal weights for whistles and bursts using non linear gradient descent with cepstrum

Figure 4.23: Optimal weights for whistle and burst calls using the cepstral feature

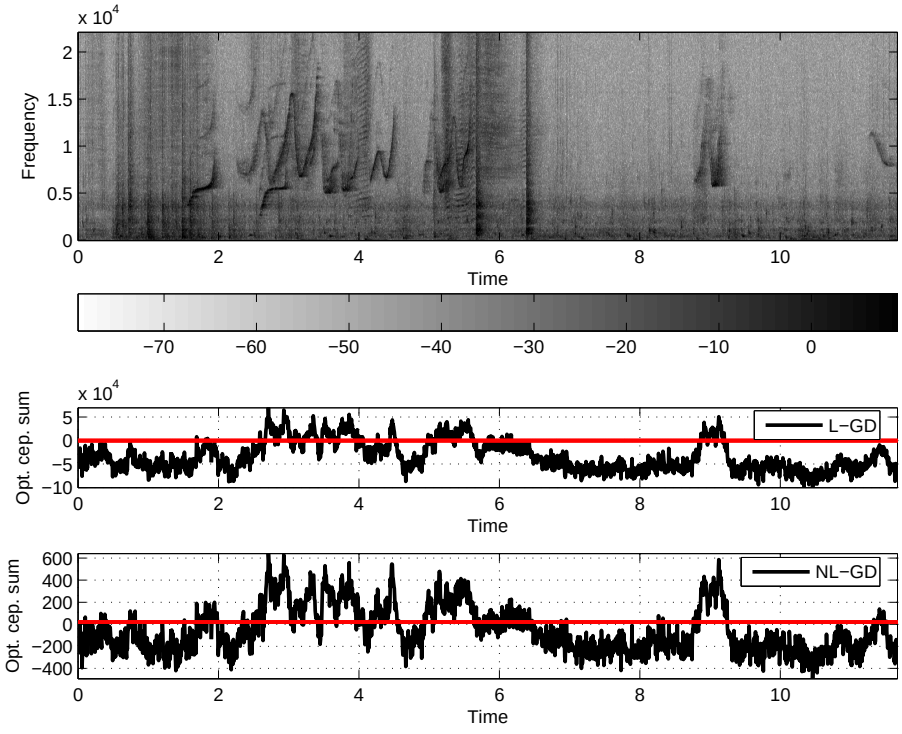


Figure 4.24: Example of cepstral feature using gradient descent technique for whistles and bursts

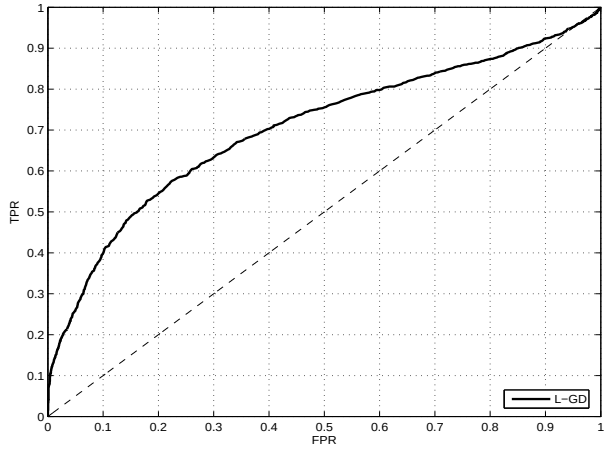


Figure 4.25: ROC for whistle and burst calls using cepstral feature and L-GD

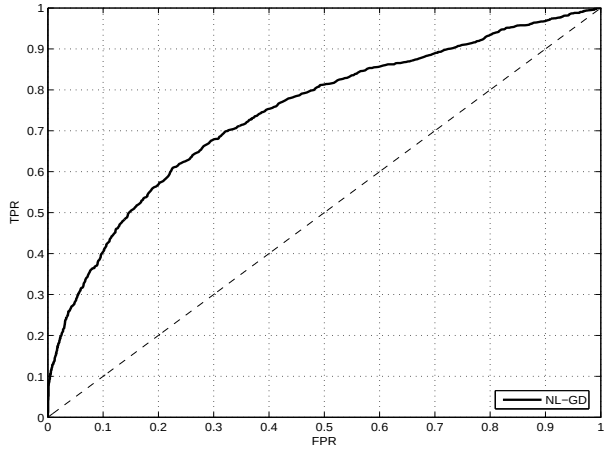


Figure 4.26: ROC for whistle and burst calls using cepstral feature and NL-GD

Whistle and burst Opt. Cepstral feature	
GD	AUC
Linear	70.41%
Non-linear	74.48%

Table 4.11: AUC for whistle and burst calls using gradient descent with cepstrum and best results from simple energy thresholding and optimized spectral feature system

4.6 Call detection using Gaussian Models (GM)

Gaussian Mixture Models (GMM) [15, 1, 26] are a semi-parametric clustering and density method that has long been used successfully to model the densities of a variety of signals. Based on the fact that most data found in nature can be modeled using Gaussian distributions and given that dolphin calls have a log-normal distribution as seen in Figures 3.5(a), 3.5(b), GMM's appear to be suitable to model the data.

The idea behind GMM's is that they provide the ability to model multi-modal distributions that might also have non-linear correlations. Basically , that kind of data can be modeled using a weighted sum of Gausssian distributions as seen in Equation 4.8 where c_k is a set of weights and $p(x|\theta_k)$ is a single Gaussian pdf component with parameters θ_k.

$$p(x) = \sum_k c_k p(x|\theta_k) \tag{4.8}$$

An easy way to view this is that every data point is drawn from the distribution $p(x|\theta_k)$ with probability c_k. The question becomes how to compute the parameters of the mixture model. The unknowns are the weights/priors, c_k and the mean and variance of the individual gaussian components, $\theta_k = \{\mu_k \Sigma_k\}$. In some cases we might know not only the number of classes, but also which data belongs to which class. In these cases, we can directly obtain the unknown parameters. When that information is not available then a general procedure is employed called expectation-maximization (EM) [1, 26, 20]. EM is an iterative algorithm based on maximizing the log-likelihood of the training

data. In order to achieve the maximization, the unknown parameters require to be updated until the algorithm converges. That of course implies that the initialization will define whether the method gets trapped in a local optimum or not. This is one of the drawbacks in most optimization techniques like gradient descent and EM, since none of them can guarantee that they will converge to the global optimum.

In the case of dolphin recordings, having extracted the training data manually, I have the labels for each of the classes so the Gaussian distributions can be fitted directly on the data. However, I also employed EM to verify the parameters of the distributions. Since the task at hand is to detect calls from a noisy background, I only need two components, one component to model call frames and another to model noise frames, as shown in Equation 4.9, where c designates the call model, n designates the noise model and N is the number of dimensions of the data.

$$\begin{aligned} p(x|c) &= \frac{1}{(2\pi)^{\frac{N}{2}}} \exp(-\frac{1}{2}(x-\mu_c)^\top \Sigma_c^{-1}(x-\mu_c)) \\ p(x|n) &= \frac{1}{(2\pi)^{\frac{N}{2}}} \exp(-\frac{1}{2}(x-\mu_n)^\top \Sigma_n^{-1}(x-\mu_n)) \end{aligned} \tag{4.9}$$

Once a model has been fit on the training data then Bayes rule [40, 20] can be used to classify all new data into one of the two components presented before. This is also known as maximum-likelihood (ML) [40, 26, 20] since a new data point is classified as a noise or call point according to the value of each posterior when evaluated for each of the components. Since I am only using a single Gaussian per class this process can be viewed as a more simplistic

approach of simple Gaussian modeling.

4.6.1 Call detection using Gaussian Models and the spectrogram feature

One of the most common methods for clustering in speech and music is the use of GMM's with the spectrogram magnitude as a feature. Given its popularity and success in several similar tasks the use of this system on dolphin calls might provide useful insights. Since the magnitude of the spectrogram is used as a feature, as described in Chapter 3, the feature space has 257 dimensions. However, in order to get the GM to work the data had to be pre-processed. This is due to the fact that the raw data is rank deficient, which led to non-positive covariance matrices. In order to avoid this problem the data was standardized by frame i.e. for each time slice, the features were scaled and offset to achieve zero mean and unit variance. This process provides a solution because it introduces a randomness in the data that will not allow the individual covariances go to zero.

Continuing, single components were used for each of the classes. In Figure 4.27 the parameters for the two components are shown with the top row depicting the mean and covariance matrix for the call class distribution and the bottom row showing the mean and covariance matrix for the noise class distribution.

Finally, Figure 4.28 depicts the ROC curve for a realistic recording of increasing difficulty where multiple overlaps and interferences exist. The re-

GM and spectral mag.	
Call	AUC
Whistle, burst	84.69%

Table 4.12: AUC for whistle and burst calls using GM's with spectral magnitude

sulting AUC metric can be seen in Table 4.12.

Overall, the use of the GMM with the spectral magnitude as a feature appears to outperform every other system so far. It has a better accuracy of approximately 10% compared to the use of cepstral features with the simple thresholding technique and almost 1% higher from the energy feature thresholding using NL-GD. Although that is encouraging for the use of GMM's in a wider scale when it comes to dolphin recordings caution needs to be taken when extracting and using the training data since without a suitable pre-processing technique there might be issues of non-positive definite covariance matrices.

4.6.2 Call detection using GMM's and the cepstral feature

Previously, the use of GMM's with the spectral magnitude indicated that this might be a suitable method for the detection of whistle and burst calls in dolphin recordings. Although, it offers the best accuracy so far the previous system faces rank deficiency issues with the use of the specific feature. One of the main reasons for this problem is the highly dimensional feature space that includes redundant and correlated dimensions. In order to avoid this problem the cepstrum is used as a feature, Eq. 3.5. In the same manner as before 50

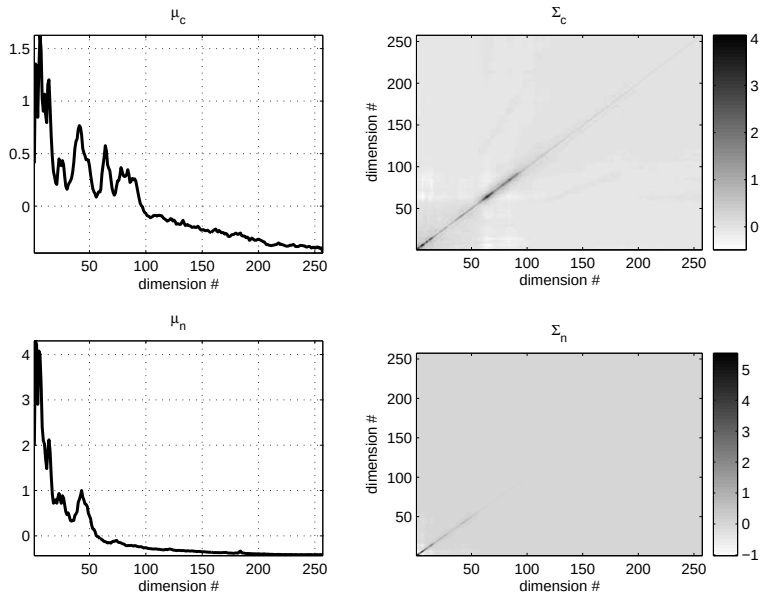


Figure 4.27: Parameters for GM with spectral features

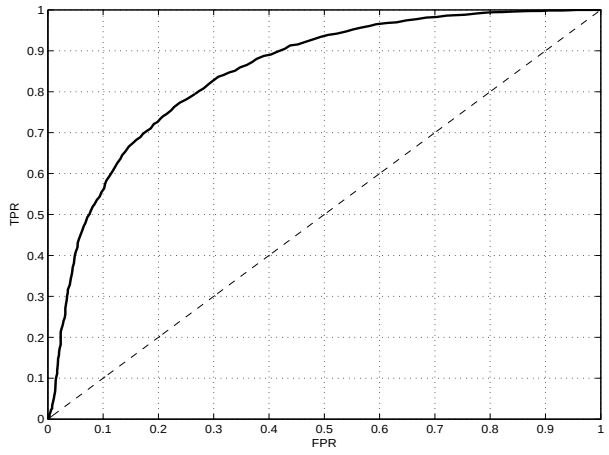


Figure 4.28: ROC for whistle and burst calls using spectral magnitude and GM's

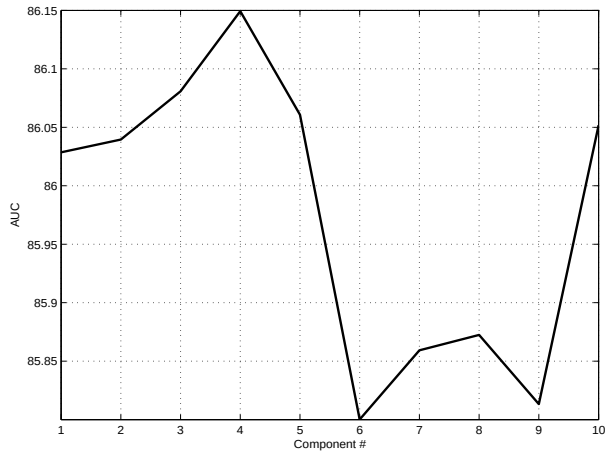


Figure 4.29: AUC as a function of the number of components used in a GMM (50 cepstral dimensions)

cepstral coefficients are extracted from every data frame and the first cepstral coefficient is excluded since experiments performed with the inclusion of the first cepstral coefficient didn't improve the resulting accuracy.

Since the dimensionality of the feature space has now been reduced to 50 dimensions instead of 257 and because of the underlying decorrelation performed by the cepstrum, there is no need to standardize the data. This allows the exploration of an increasing number of components. Figure 4.29 shows the AUC metric for an increasing number of call components. The difference between the different components is statistically insignificant and thus for comparative purposes a single component for the call distribution is used.

Continuing, Figure 4.30 shows the parameters for the class distributions where the top row depicts the mean and covariance matrix for the call class and the bottom row the mean and covariance matrix for the noise class. Finally Figure 4.31 depicts the ROC curve for a realistic recording of increasing

GM and cepstrum	
Call	AUC
Whistle, burst	86.03%

Table 4.13: AUC for whistle and burst calls using GM's with cepstral features

difficulty where multiple overlaps and interferences exist. The resulting AUC metric can be seen in Table 4.13. As expected the use of GMM's with cepstral features outperforms not only the previous GMM system by over 1% but also every previously employed detection system.

In conclusion, spectral and cepstral features are well modeled by normal distributions. However, given that GM's with cepstrum perform approximately 3% better from the second best system, energy thresholding using NL-GD, seen in Table 4.10 the idea of finding an optimal set of weights/hyperplane to separate the data needs to be explored further. Overall, in multiple experiments with GD the random initialization had statistically insignificant variations i.e. $< 2\%$ to the final accuracy, thus it is hard to know what level of difference will provide said statistical significance.

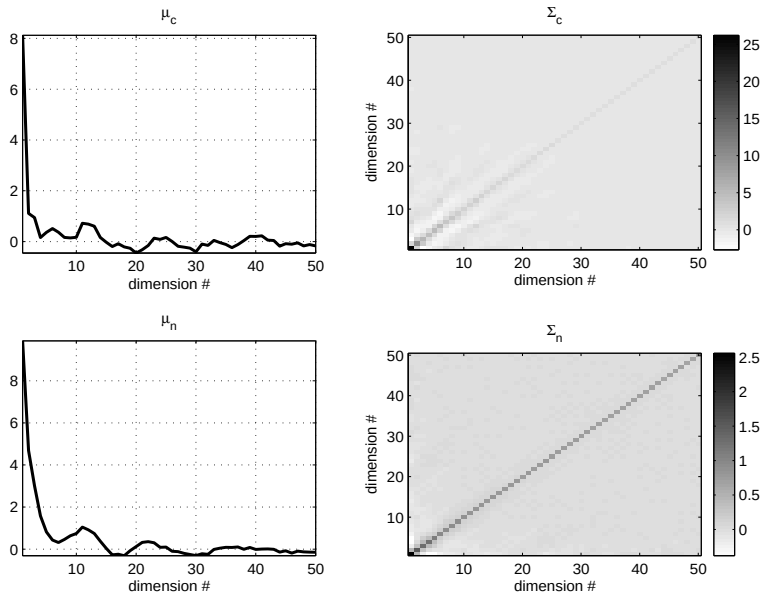


Figure 4.30: Parameters for GM with cepstral features

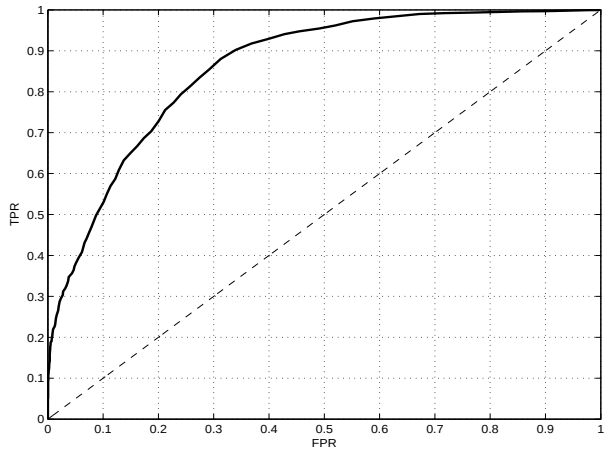


Figure 4.31: ROC for whistle and burst calls using the cepstrum and GM's

4.7 Call detection using Support Vector Machines (SVM)

Throughout this chapter I have described several methodologies and features that will be able to discriminate between noise and call frames in long dolphin recordings. In order to provide a more generalized system that will encompass the positive attributes of the previously described methodologies the use of Support Vector Machines (SVM) [10] is explored in this section. SVM's are considered one of the most powerful machine learning tools. They provide a robust and compact way of dealing with both linear and non-linear decision surfaces.

Unlike GMM's which can be considered as a generative process, SVM's belong to the category of discriminative methodologies. The main difference is that before, distributions were created over the input data trying to capture all aspects of the task at hand. However, for simple call detection one only needs to focus on the classification decision which is of course a binary decision.

SVM's offer that ability. They are binary classifiers based on the ideas of risk management. In simplistic terms and similar to gradient descent the idea is to minimize an expected error function e.g. risk. However, in order to avoid issues of over-fitting in SVM's there is an allowable margin of error by actually trying to minimize the bound of the error function. This idea, based on Vapnik's theorem [10] gives SVM's the flexibility they need to handle multiple types of data.

In their most simplistic way SVM's can be viewed as linear classifiers

that employ risk minimization on unseen test data by maximizing the margin on the training data. This can be analytically seen in Equation 4.10 where $(x_i, y_i), ...(x_N, y_N)$ with $x_i \in \Re^D$ and $y_i \in -1, 1$.

$$f(x; \theta) = sign(w^\top x + b) \tag{4.10}$$

The separating hyperplane/decision surface is clearly $w^\top x + b = 0$ and in order to maximize the margin, $\frac{2}{\|w\|}$ one needs to minimize $\|w\|$. So the SVM problem can be stated as, Equation 4.11.

$$\min \frac{1}{2}\|w\|^2 \text{ subject to } y_i(w^\top x + b) - 1 \geq 0 \tag{4.11}$$

The above formulation will handle binary cases of linearly separated data. However, SVM's have been extended to deal with non-linear data. The idea behind it is to map the input data into a higher dimension space and solve the classification problem in that space. In order to do that one can design different kernel functions that will map the data into that space and then solve the SVM. The new formulation is described in Equation 4.13.

$$K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j) \tag{4.12}$$

$$f(x, (\Lambda, b)) = \sum_i \lambda_i y_i K(x_i, x_j) + b \tag{4.13}$$

Where $x_{i,j}$ is the training data, λ are the Lagrange multipliers, y_i are the class labels and b is the bias.

Clearly, the use of the appropriate kernel, K, will define how well the clas-

sification task proceeds. Given that both whistle and burst calls appear to be modeled well by normal distributions I chose to use a general purpose classifier such as a Gaussian kernel, which represents a Radial Basis Function (RBF) classifier. The general formula for such a kernel is given in Equation 4.14.

$$K(x, y) = \exp(-\gamma \|x - y\|^2) \tag{4.14}$$

As seen in Equation 4.14 there is one more parameter that needs to be tuned by the user according to the available training data. That is the variance, σ^2 for the Gaussian kernel since $\gamma = \frac{1}{2\sigma^2}$. Also, the weights λ can be used either by solving the full QP or by a fairly new approach of SVM's known in the literature as Sequential Minimal Optimization (SMO) [42, 54]. This leads to the dual problem described in Equations 4.15- 4.17 where C are the box constraints.

$$\max_{\lambda} \sum_{i=1}^{M} \lambda_i - \frac{1}{2} \sum_{i,j=1}^{M} \lambda_i \lambda_j y_i y_j K(x_i, x_j) \tag{4.15}$$

$$\text{such that } \sum_{i=1}^{M} \lambda_i y_i = 0 \tag{4.16}$$

$$\text{and } 0 \leq \lambda_i \leq C \tag{4.17}$$

4.7.1 Call detection using SVM's and the spectrogram feature

Given that SVM's provide a combination of the classifiers that have been described in the previous sections the first approach would be to use the simplest

feature possible in order to get a baseline on the strength of the classifier. As described in Chapter 3 using the magnitude of the spectrogram is perceived as the most common feature. Of course the use of the spectral magnitude has the drawback of being a high dimensional feature space, thus increasing the computational complexity of the system.

In order to get the SVM to work several parameters need to be tuned. As seen in Equation 4.13 one needs to specify the variance, σ^2 for the RBF kernel that controls the width of the kernel. Continuing, another important parameter is the constraint on the Lagrange multipliers, λ, for solving the quadratic problem. This constraint, C is also known as the box constraint and it is responsible for the allowable "slackness" in the method. For example, a high value of C implies a larger margin and thus controls the number of outliers. On the other hand a small value of C implies a tight margin and less tolerance on the existence of outliers. In order to achieve the best accuracy with the SVM an initial wide search was performed for high values of (C, γ). According to performance that was narrowed down to a grid search varying their values $(C, \gamma) = (0.01...10, 1, ...3)$. The best pair was found by evaluating on the training data and it is $(C, \gamma) = (10, 1)$. All experiments for the SVM's were performed on WEKA (Waikato Environment for Knowledge Analysis) [25]. WEKA is a very popular machine learning software written in Java and is freely available. It contains a collection of visualization tools and machine learning algorithms.

The SVM is initially trained on artificial clips created by the manually extracted data from the recordings. Figure 4.32 depicts the ROC curve on a

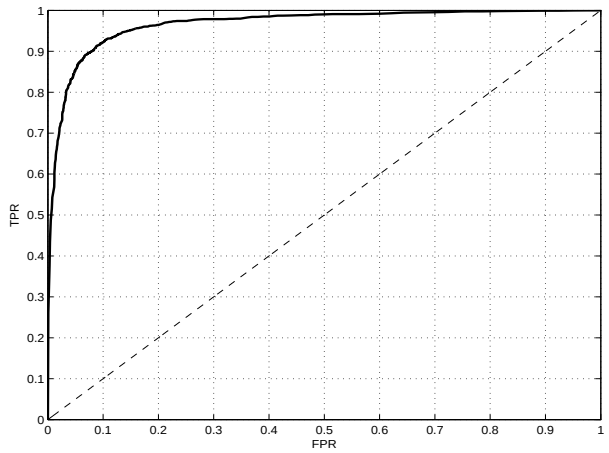


Figure 4.32: ROC for whistle and burst calls using spectral magnitude and SVM's

SVM and spectral mag.	
Call	AUC
Whistle, burst	96.65%

Table 4.14: AUC for whistle and burst calls using SVM's with spectral magnitude feature

long dolphin clip that is comprised of bursts and whistles and multiple interferences. Also Table 4.14 shows the AUC metric for the detection task using SVM's and the spectral magnitude feature.

From the resulting ROC curve and AUC metric it is evident that the use of the SVM is superior to any other classifier and feature combination used so far. It appears that the combination of the RBF kernel with the spectral magnitude feature allows for a really good classification accuracy since if an error of approximately 20% is allowed then the system can actually attain a TPR of more than 95%. In order to check for the generalization ability of the SVM's and given their really good performance I decided to apply the

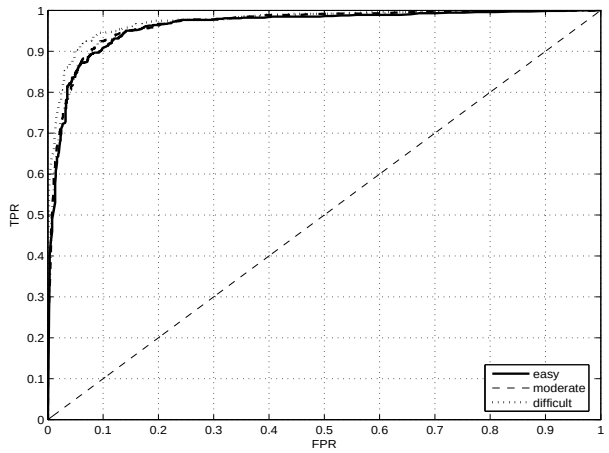


Figure 4.33: ROC for different clips using spectral magnitude and SVM's

SVM and spectral mag.	
Call	AUC
Easy	96.24%
Moderate	97.49%
Difficult	96.61%

Table 4.15: AUC for different clips using SVM's with spectral magnitude feature

same system on three different dolphin clips of increasing difficulty. The first clip having no interferences and the second and third clips having increasing difficulty with multiple overlaps. Figure 4.33 shows the ROC curves for the three different clips and Table 4.15 shows the AUC metric for each of the different clips.

Clearly, SVM's depict a superior performance in all three types of clips. That is a testament to their robustness and their generalization ability since they have an AUC of more than 95% for all types of clips.

SVM and cepstrum	
Call	AUC
Whistle, burst	96.08%

Table 4.16: AUC for whistle and burst calls using SVM's with cepstral features

4.7.2 Call detection using SVM's and the cepstral feature

The success of SVM's relies on solving a quadratic problem in a higher dimension space through kernel dot products. That, of course, is computationally expensive and can be exacerbated by the dimensionality of the input data/feature vector since the cost of computing the kernel distances is proportional to the number of dimensions. In order to compensate for computation complexity it is wise to use a feature space with lower dimensions. Using the cepstral coefficients as in previous implementations of other detection schemes will provide a lower computational cost and an insight on how pitch information might aid the detection task.

Computed as presented in Equation 3.5 and used in the same manner as described in previous sections, 50 cepstral coefficients are computed and used as the input data. The first coefficient is excluded since it does not offer any improvement to the overall accuracy. Figure 4.34 depicts the ROC curve for the SVM with cepstral coefficients on a realistic dolphin recording and Table 4.16 provides the AUC metric representing the classification accuracy.

Once again the use of SVM's with cepstral features highlights the success of this machine learning tool. In order to explore the robustness of the system

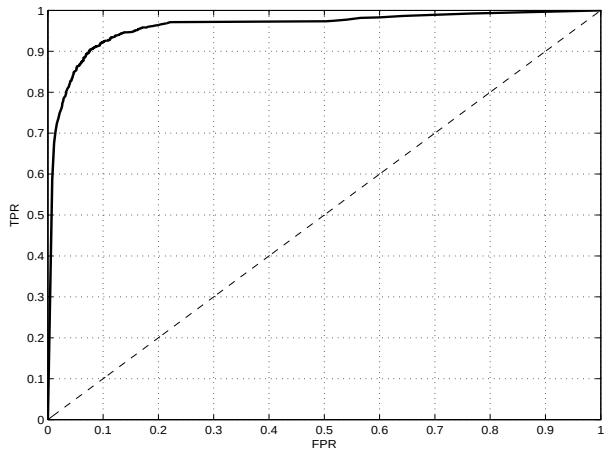


Figure 4.34: ROC for whistle and burst calls using the cepstrum and SVM's

SVM and cepstrum	
Call	AUC
Easy	95.89%
Moderate	96.92%
Difficult	95.75%

Table 4.17: AUC for different clips using SVM's with cepstral features

and compare its accuracy when using a different feature, the system is tested on three different clips of increasing difficulty. The resulting ROC curves are shown in Figure 4.35 and the respective AUC metrics are presented in Table 4.17.

From the above results it is clear that the use of SVM's seems to be favorable for the task of detecting whistle and burst calls in long noisy recordings. When using the cepstral features there is a small decrease, likely insignificant, in the overall accuracy of the detection. However, this slight decrease in accuracy is accompanied by a major decrease in the computational cost of training

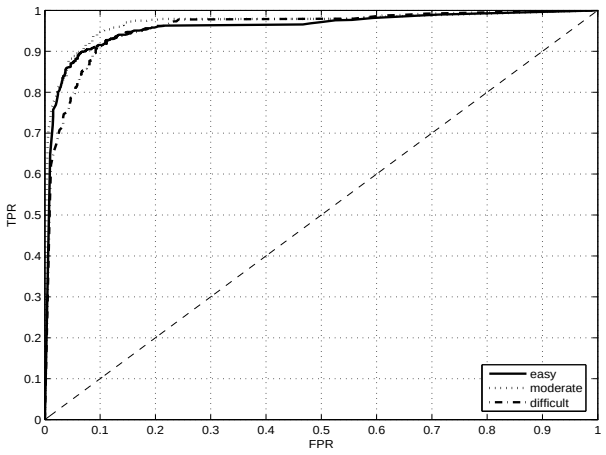


Figure 4.35: ROC for different clips using the cepstrum and SVM's

the SVM and extracting the support vectors. Due to the lower dimensionality of the feature space the computation is cut to half compared to the spectral magnitude features.

Finally, in this chapter several classifiers and features were used in order to explore the task of detecting whistle and burst calls in dolphin recordings. Simple features that have been widely used in speech and music processing along with classifiers that have successfully been used in the machine learning community. Throughout this chapter a better insight on the data and how they need to be modeled was attained. In Chapter 5 a different approach will be explored where the dual task of detecting and extracting the pitch is analyzed.

4.8 Conclusions on detecting dolphin calls

In this chapter several approaches were discussed for detecting whistle and burst calls in dolphin recordings. Starting from the simplest features and classifiers such as energy thresholding, the most popular detection scheme in the field, an in-depth analysis of the data revealed that the inherent characteristics of these calls dictate the evolution of the algorithms in such a way where the distributions of the features are modeled and the use of robust classifiers is preferred.

The first class of algorithms is based on simple thresholding as well as the addition of the optimization technique of gradient descent. The idea being that bursts and whistles have energy in distinctive frequency channels and thus as seen in Chapter 4 there should be an optimal set of weights that would be able to separate them. Two types of gradient descent were used, linear and non-linear, referring to the decision boundary. Figure 4.36 depicts the AUC results for simple thresholding, L-GD and NL-GD with the use of the energy feature for only whistle and burst calls. Clearly, the success rates can be considered high, but one needs to take into account that in realistic environments both types of calls are present along with interferences.

When the same algorithms are employed on realistic clips where there is a need to detect both whistle and burst calls the overall rates decrease. Figure 4.37 depicts the AUC's for the different algorithms as before when using the energy feature (white bar) and the cepstrum feature (black bar). It appears that in this class of algorithms the use of NL-GD is superior when

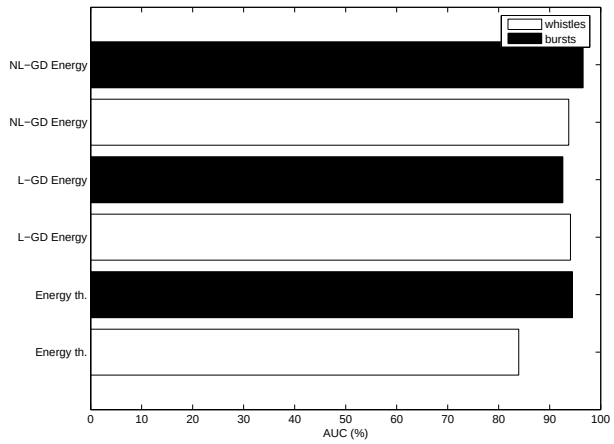


Figure 4.36: Comparative results for clips with only whistles or only bursts

used with the energy feature. It was expected that NL-GD would outperform the other algorithms since bursts introduce non-linearities in the data and the error surface is able to separate them. However, the cepstral features seem to capture enough background noise to introduce an approximate 10% decrease in the AUC metric. Results for simple thresholding on the ceptral features are not reported since they would be very similar to the ones obatined with the energy feature.

Continuing, and given that in Chapter 3 it was shown that the data is distributed in a log-normal fashion, a generative approach is applied on the detection by using GM's. This is a simple procedure widely used in speech and music and the idea is to capture the information and differences in the data. A much more powerful tool is also used to combine the ideas of finding a hyperplane in a higher dimension while modeling the data according to a normal distribution. SVM's offer the ability to separate non-linear data through the use of different kernels. Figure 4.38 provides the comparative

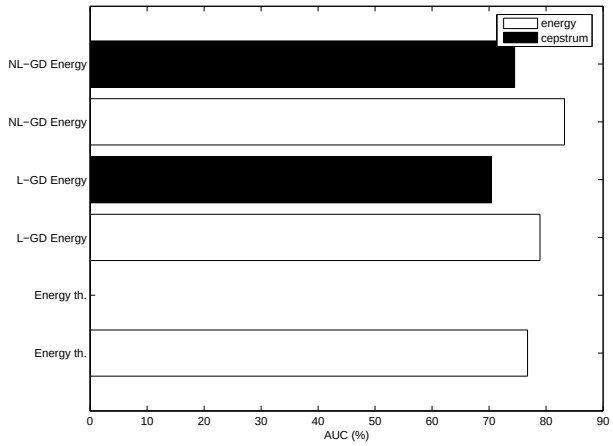


Figure 4.37: Comparative results for clips with whistles and bursts

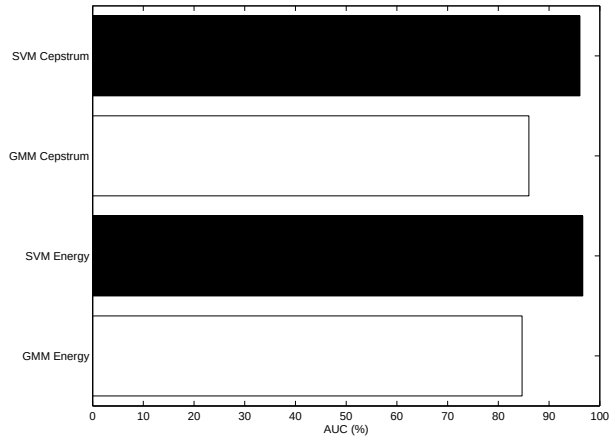


Figure 4.38: Comparative results for clips with whistles and bursts

results for GM's and SVM's on clips that include both whistle and burst calls. Both methodologies are employed using the spectral magnitude feature (white bar) and the cepstral features (black bar).

In conclusion, when it comes to the detection of dolphin calls in long recordings the use of SVM's with either the cepstral or energy features outperforms every other methodology. This was expected given the principles behind SVM's. Table 4.18 depicts the sorted results across all proposed methodologies. However, their success in inversely proportional to their computational cost. All algorithms presented, fall under the category of supervised classifiers requiring the use of training data. In order to use the classifiers on new test data there is a computational cost associated with their training. Table 4.19 presents, on average, the computational cost for employing the aforementioned algorithms on approximately $1min$ clips. It is worth noting that the training of the classifiers is only required once if there is a large training set at hand.

Detection Results

Classifier	AUC
SVM with spectrum	96.65%
SVM with cepstrum	96.08%
GM with cepstrum	86.03%
GM with spectrum	84.69%
NL-GD with spectrum	83.24%
L-GD with spectrum	78.94%
Energy Thresholding	76.75%
NL-GD with cepstrum	74.48%
L-GD with cepstrum	70.41%

Table 4.18: Results on detection algorithms

Computational Cost

Classifier	Time(minutes)
SVM with spectrum	120
SVM with cepstrum	45
NL-GD energy/cepstrum	4
L-GD energy/cepstrum	1.5

Table 4.19: Computation cost for detection algorithms

Chapter 5

Call pitch tracking

> "It is of interest to note that while some dolphins are reported to have learned English, up to 50 words used in correct context, no human being has been reported to have learned dolphinese."
> Carl Sagan

In Chapter 4 several systems were presented with the goal of detecting whistle and burst calls in noisy dolphin recordings. Given the copious amounts of available data, creating a robust and automatic call detector is an intricate part in the process of analyzing and understanding dolphin vocalizations. Once the desired calls have been detected then a more thorough statistical analysis can be performed.

In this work, I am interested in the analysis of dolphins' interaction calls. As seen in Chapter 3 whistles and bursts are periodic signals. The most important information one can extract from periodic signals is their fundamental frequency, f_0. In humans, periodic signals are often perceived as having a

pitch which is determined by their fundamental frequency.

When dealing with dolphin vocalizations there are several characteristics that need to be taken into consideration. In terms of whistle calls, they do not always have harmonics and therefore appear as single, modulated sinusoids. That implies that pitch extraction in these cases can be viewed as a frequency contour extraction. Also, in some cases the harmonics might not be visible due to the low sampling frequency used in recording the sounds. On the other hand when it comes to burst calls, harmonics are always present, but in most cases there is little or no signal energy visible at the fundamental frequency itself. This is a common problem in music although it is known that the lack of a fundamental frequency doesn't necessarily interfere with the perceived pitch. This phenomenon can be attributed to several factors. For example, burst calls are low pitched sounds e.g. centered around $800Hz$ and the low end of the spectrum is occupied by mostly noise in dolphin recordings e.g. tank reflections. Also, it could be part of the sound production system of dolphins or even a characteristic of the hardware used to record the sounds e.g. range of hydrophone.

In this work pitch extraction/tracking refers to the identification of the fundamental frequency, f_0 at a per frame level. Ground truth has been extracted in a semi-automatic way and due to the discrete nature of processing rounding errors are expected and permitted. In the following sections two systems will be presented for the task of pitch extraction in dolphin recordings. The first system assumes that the detection process has already been successfully employed and a proposed algorithm is compared to baseline methodologies for

the pitch extraction of whistles and bursts. The second system can be viewed as a dual system for both detection and pitch tracking focusing on whistle calls and providing a first attempt to resolve overlaps between them.

5.1 A comparison of pitch extraction methodologies for whistles and bursts

In this chapter the goal is to provide possible systems that will be able to resolve the problem of pitch extraction without manual interaction. Several methodologies that have been previously employed in speech and music processing can be utilized on dolphin recordings. However, these algorithms need to take into account the intricacies and differences that are present in dolphin recordings. As mentioned in Chapter 3 dolphin recordings have a low SNR requiring a careful selection of features. Continuing, the different vocalizing range of dolphins clearly limits the use of several off-the-shelf algorithms designed for speech. This wider range also implies that a high dimensional feature space might be needed to capture the information present in the recordings. One of the most unfortunate issues with dolphin recordings is that in many cases the frequency range of their calls exceeds the Nyquist rate of most common underwater recording devices leading to a loss of information in the high end of the spectrum.

In this section, a proposed algorithm is compared to widely used pitch extraction methodologies in speech and music. It is important to note that the proposed systems assume that the desired calls have already been detected and

Algorithm	Feature	Classifier
Cepstrum and HMM	256 cepstral coef	HMM
YIN	Autocorrelation	Local Min
get_f0	LPC residual	Dynamic programming

Table 5.1: Pitch extraction methodologies

segmented. One can imagine that this is possible by employing the different algorithms already described in Chapter 4.

Given that there are inherent differences in the frequency ranges of dolphin vocalizations i.e. the whistles and the bursts introduced in Chapter 3, it is worth exploring two-stage systems where the different calls are first clustered together using a different classifier. This implies the use of hierarchical systems that will have different parameters for bursts and whistles.

In order to provide a well rounded approach on pitch extraction of dolphin calls, three different algorithms are used and compared. The first algorithm is based on the use of Hidden Markov Models (HMM) [15, 43]. HMM's can be used either directly on the spectrogram or in combination with descriptive features. In this work the use of the cepstral coefficients is preferred. The second algorithm, YIN [13] has long been used successfully in speech processing for single pitch extraction and is based on a modified autocorrelation method and finally get_f0 [50, 18] a popular off-the shelf pitch tracker. Table 5.1 summarizes the algorithms that are used.

5.1.1 Cepstral coefficients with hierarchically driven Hidden Markov Models (HMM)

HMM's have been extensively used in many natural sequences such as speech, language and handwriting. They are a valuable tool for the analysis and extraction of information of time dependent data. The idea behind them is that the system being modeled is assumed to be a first order Markov process and the goal is to determine the hidden parameters based on the observations e.g. data. A simple Markov model is a finite state machine where the states are directly visible. In a hidden Markov model the states are not directly visible, but their dependence on the observations are known and thus some information can be extracted on the sequence of the states.

Since HMM's are a first order Markov process their behavior depends only on the current state and thus simplifications can be made due to independencies in order to compute the desired values. HMM's are specified by several parameters, Θ. Firstly, the states, q_i, the transition probabilities, $a_{i,j} = p(q_n^j | q_{n-1}^j)$ defining the probability of being at a specific state now given that I was in a different state in time $n-1$. Also, the emission distributions, $b_i(x) = p(x|q_i)$ representing the probability of seing a specific observation given that I am at state q_i and finally the initial state probabilities/priors, $\pi_i = p(q_1^i)$ [15, 43].

Once these parameters have been defined there are three main issues regarding the function of HMM's.

- The evaluation problem: Determining the probability that a particular

sequence of states was produced by the model obtained through the use of the Forward algorithm [15]

- The decoding problem: Given a set of observation determining the most likely sequence of hidden states that led to the observations, obtained through the use of the Viterbi algorithm [15]

- The learning problem: Given a set of training observations extract the parameters, Θ of the HMM, obtained through the use of the Forward-Backward algorithm or Baum-Welch [15]

HMM's can be used with a variety of features. For the task of pitch extraction I chose the use of cepstral features described in Chapter 3. This will ensure a more compact way of describing the data and will allow for a lower dimensionality in the feature space which will ease the computational cost.

Since as shown in Figures 3.1(a), 3.1(b) whistle and bursts occupy different regions of the frequency spectrum it can be stated that the data depicts a bimodality dependent on the call type. This information can be taken advantage with the use of a hierarchy in the system. In simple terms, the system can be described as follows. Initially, two HMM's are created with different number of hidden states corresponding to the different types of calls. For every input vector both HMM's are evaluated using the forward algorithm and the one that gives the highest likelihood is activated for the implementation of Viterbi decoding, thus obtaining the most likely path across the hidden states. Figure 5.1 provides a schematic overview of the system. It is clear that since there are two different HMM's there are two different sets of parameters

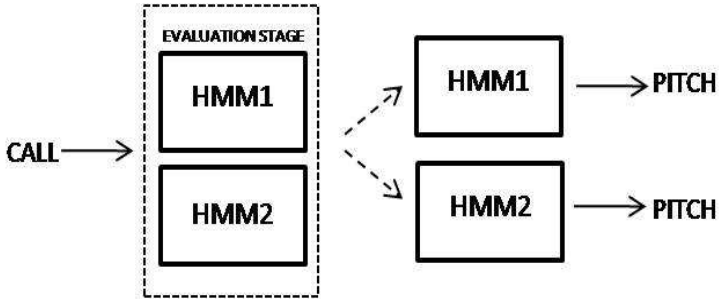


Figure 5.1: System overview

representing two different frequency ranges respectively.

Each HMM is defined through parameters, Θ that are extracted directly from the training data. Both HMM's are continuous, which means that each state q has an emission that is described by a single Gaussian probability density function. The first HMM is modeling whistle calls and its parameters are shown in Equation 5.2.

$$\begin{aligned} \Theta_1 &= \langle \pi_1, A_1, E_1 \rangle, \\ \pi_1 &= (\pi_{11}, \pi_{12}, ...\pi_{1N}), N = 1, 2, ...18 \\ A_1 &= \{\alpha_{1ij}\}_{i,j=1,2,...N}, \alpha_{1ij} = p(q_{1t} = j | q_{1t-1} = i) \\ E_{1N}(o) &= N(o; \mu_{1N}, \sigma_{1N}), N = 1, 2, ...18 \end{aligned} \tag{5.1}$$

In the same manner the second HMM is modeling burst calls and its pa-

rameters are shown in Equation 5.3.

$$\begin{aligned} \Theta_2 &= \langle \pi_2, A_2, E_2 \rangle, \\ \pi_2 &= (\pi_{21}, \pi_{22}, ...\pi_{2M}), M = 1, 2, ...52 \\ A_2 &= \{\alpha_{2ij}\}_{i,j=1,2,...M}, \alpha_{2ij} = p(q_{2t} = j | q_{2t-1} = i) \\ E_{2M}(o) &= N(o; \mu_{2M}, \sigma_{2M}), N = 1, 2, ...52 \end{aligned} \tag{5.2}$$

The states q_1, q_2 represent the frequency ranges of approximately $2.2kHz - 11kHz$ and $440Hz - 740Hz$ respectively. Also a noise state is added for each HMM in order to capture the lack of pitch in a particular frame. Each state q represents a call with a pitch delay/period in samples that can be directly mapped to a specific frequency. Also, π_1, π_2 define the priors on the states and are directly obtained from the statistics of the ground truth as described in Chapter 3. A_1, A_2 are the transition matrices for the state sets also obtained from the training data. Finally, E_1, E_2 are the emission distributions for each state set. These are single 256 dimensional Gaussian distributions obtained from the cepstral coefficients.

Once the parameters for each of the HMM's have been extracted then every call needs to be evaluated in order to identify its frequency range i.e. evaluating the best-path likelihood for both models. The last stage of the system is the application of Viterbi decoding yielding the most likely path across the evaluated state set, thus extracting the desired pitch at every frame. This is shown in Equation 5.3.

$$\begin{aligned} \text{if } p(Y_1) &> p(Y_2) \text{ then we want} \\ argmax_{q_{1\ldots T}} p(Q|Y) &= argmax_{q_{1\ldots T}} p(Y|Q)p(Q) \end{aligned} \tag{5.3}$$

Where Y_i, $i = 1, 2$ is a sequence of observations, Q_i, $i = 1, 2$ is a sequence of the hidden states, q_i, $i = 1, 2, ...T$ is the maximum probability state path.

5.1.2 YIN: A fundamental frequency estimator

The algorithm YIN [13] was briefly discussed in Chapter 3. Created by de Cheveigne and Kawahara it is widely used for the estimation of the fundamental frequency/pitch of speech or monophonic musical sounds. it is based on a modified autocorrelation method and it is extremely successful in extracting single pitches. It is a time domain based system and its simplicity and computational efficiency certainly add to its popularity.

Dolphin vocalizations are periodic signals, x_t with period T. YIN is based on the autocorrelation of the signal defined in Equation 5.4.

$$r_t(\tau) = \sum_{j=t+1}^{t+W} x_j x_{j+\tau} \tag{5.4}$$

Where $r_t(\tau)$ is the autocorrelation at lag τ calculated at time t and W is the integration window size. From the above equation it is easy to see that it still stands even if one takes the square and averages over a window, W. This implies that a difference function can be formed where an unknown period

may be found while searching for those lag values, τ for which the function is zero. This new function is seen in Equation 5.6.

$$d_t(\tau) = \sum_{j=1}^{W}(x_j - x_{j+\tau})^2 \tag{5.5}$$

$$d_t(\tau) = \sum_{j=1}^{W} x_j^2 + \sum_{j=1}^{W} x_{j+\tau}^2 - 2\sum_{j=1}^{W} x_j x_{j+\tau} \tag{5.6}$$

The main problem with Equation 5.6 is that it has a zero value at zero lag and that often it has a non-zero value at the lag corresponding to the period due to imperfections in the periodicity. However, it's worth noting that both $\sum_{j=1}^{W} x_j$ and $\sum_{j=1}^{W} x_{j+\tau}$ don't change much with τ, meaning that minima of d_t will occur where there are maxima in r_t. Clearly, the use of the difference function will fail to pick the correct pitch since it will always be minimal for the zero lag. In order to alleviate this problem, the authors propose the use of the cumulative mean normalized difference function as seen in Equation 5.7.

$$f(\tau) = \begin{cases} 1 & \text{if } \tau = 0 \\ \frac{d_t(\tau)}{[\frac{1}{\tau}\sum_{j=1}^{\tau} d_t(j)]} & \text{otherwise} \end{cases} \tag{5.7}$$

This function will actually take on the value of one at zero lag and will stay large at small lags. There are several more steps that can be employed to ensure a better pitch estimate. These steps are described in detail in [13]. Overall, the desired pitch can be obtained by picking the smallest value of the lag/pitch delay, τ that gives the minimum d. Figures 5.2, 5.3 show examples of whistle and burst call spectrograms (top) along with the distance function

proposed by YIN (bottom).

From the figures, one can clearly see the differences between whistle and burst calls when they are represented by YIN's distance function. Since the function measures in pitch delay e.g. samples,whistles that are high pitched sounds are going to appear with lots of peaks in the distance function. On the other hand, bursts that are low pitched sounds will have less peaks in the distance function. Once again, according to YIN the pitch at every frame is identified as the lag where the lowest peak is located.

5.1.3 Get_f0: A software package for pitch extraction in speech

Get_f0 is part of a widely used software package called Entropic Signal Processing Systems(ESPS) and Waves [18]. Used for pitch tracking it is quite popular amongst speech researchers. It is based on Doddington's and Secrest's 1983 algorithm [50] for pitch tracking in speech systems.

Their proposed algorithm uses the linear prediction coding (LPC) residual error signal in order to extract the desired pitch candidate. In the same way as the cepstrum, LPC is based on the source filter model described in Chapter 3. This indicates that it is theoretically expected that the residual signal will provide the excitation information.

In order to alleviate some problems of high frequency noise, the authors devise and employ a de-emphasis filter as a pre-processing tool. With that, they low pass filter the residual signal, s in the voiced regions and high pass

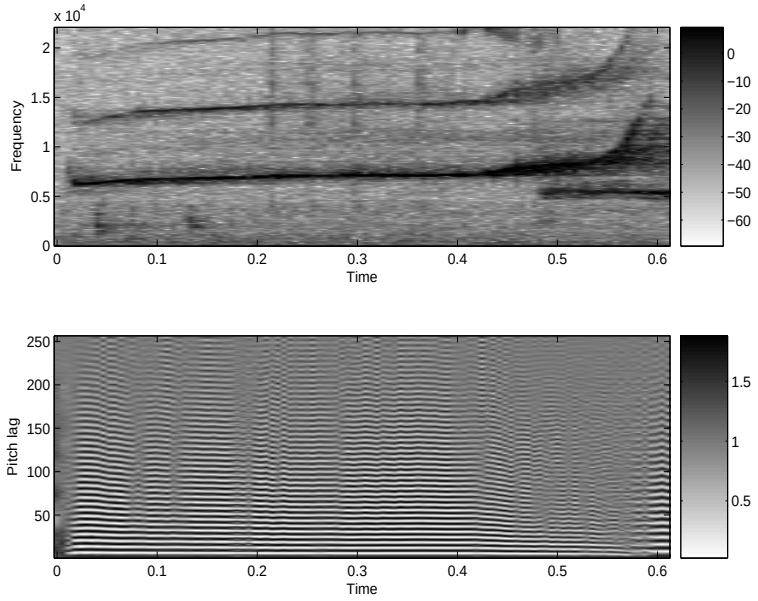


Figure 5.2: Whistle call and distance function from YIN

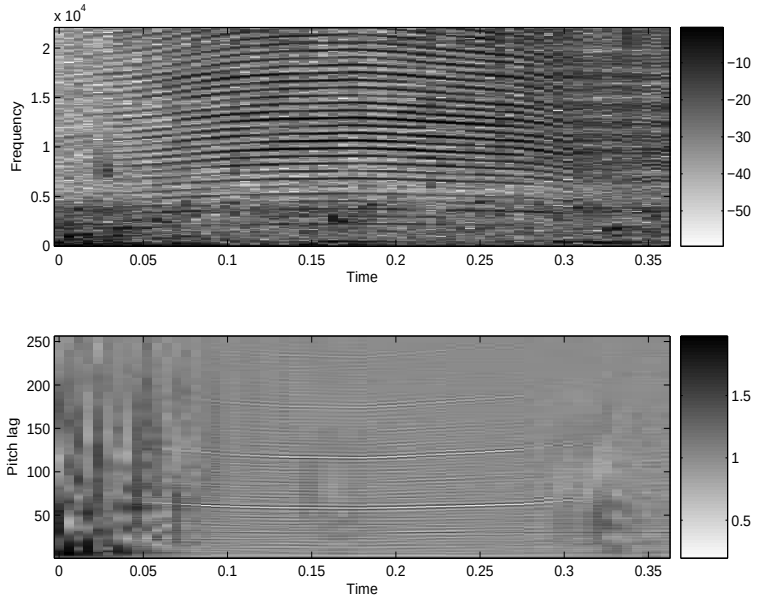


Figure 5.3: Burst call and distance function from YIN

filter in unvoiced regions. These filters are highly tuned for human speech and unfortunately do not work for dolphin vocalizations so the algorithm is not expected to perform well for the task at hand. In theory the filters could be modified to deal with dolphin ranges, but this was not easily accessible through the software package.

To extract the candidate pitch at each instance the peaks of the normalized cross-correlation are acquired as seen in Equation 5.8.

$$C(\tau) = \frac{\sum_{j=0}^{m-1} s(j)s(j-\tau)}{(\sum_{j=0}^{m-1} s(j) \sum_{j=0}^{m-1} s(j-\tau))^{1/2}} \tag{5.8}$$

Where τ is the lag and m is the number of samples to be correlated. As previously mentioned, the candidate pitch values are the lags at the peaks of C and the "goodness" measure is the corresponding value of C at those lags. Once these values have been extracted, dynamic programming [50] is performed in order to extract a smooth pitch contour. This requires the use of a penalty metric in order to decide what the best path amongst the candidates is. The cumulative penalty for each pitch candidate consists of a transition error in going from one frame to another. This methodology provides a good pitch extractor although it is highly specialized for speech.

5.1.4 Comparing the different algorithms

The algorithms described above are applied to the data that have been manually extracted. The data described in Chapter 3 are comprised of 200 calls, balanced number of whistles and bursts. No overlaps are present and these

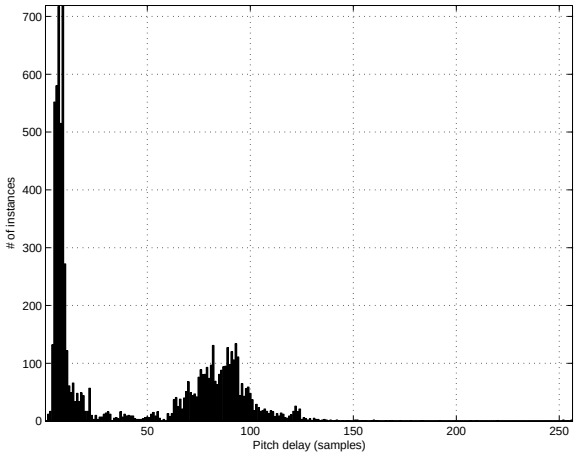


Figure 5.4: Data histogram and bimodality

are single calls. The statistics of the calls are provided in Table 3.1. In these experiments the ground truth is given in pitch delay/lag for every frame. As expected there are some errors with the extraction of the ground truth due to resolution and rounding limitations, thus incorporating bias in the final results. Once the ground truth had been extracted the analysis of the data indicated the existence of the bimodality. This is seen in Figure 5.4 and led to the choice of hierarchically driven HMM's. Two distinct frequency ranges are evident allowing the insertion of a decision level in the system. Arguably, one might explore the reasons for not choosing a single dynamic model. In several experiments, a single system suffered from erroneous "doublings" and/or "halvings" at a per frame level probably caused by the choice of the cepstral feature. Table 5.2 provides the comparative results for all three algorithms presented. In order to provide a good analysis of the results, three new metrics have been introduced. Firstly, the strict rate, which implies that the resulting pitch is an exact match with the ground truth. Secondly, the relaxed rate of

Average per frame accuracy (%)		
HMM cepstrum	Yin	getf0
Strict Rate (%)		
66.12	47.09	29.3
Relaxed Rate ±1 pitch delay (%)		
76.01	54.35	N/A
Relaxed Rate ±2 pitch delay (%)		
77.9	55.11	N/A

Table 5.2: Comparative results for different systems

±1 pitch delay (lag) and finally a relaxed rate of ±2 pitch delays (lag). Basically, this implies a soft boundary or range of acceptable error. The relaxed rates correspond to an approximate 1.5% and 3% deviation from the ground truth, which in many applications could be considered acceptable.

It is important to note that all results for the HMM were obtained using leave-one-out cross-validation i.e. 200 evaluations, otherwise known as round-robin. Although this procedure usually overestimates the rates that are obtained due to overfitting of the data, it is considered a popular way for testing on a moderate size data set.

In all cases the novel approach of using hierarchically driven HMM's with the cepstral features is superior to the baseline algorithms by over 10%. As expected the get_f0 package fails to give comparable results for the relaxed rates due to the fact that it is highly tuned for human speech and is not able to track the desired pitch in dolphin vocalizations, which exhibit a much wider frequency range.

Continuing, Figure 5.5 provides a scatter plot of the results on the individual calls for the proposed HMM with cepstrum and the Yin algorithm with

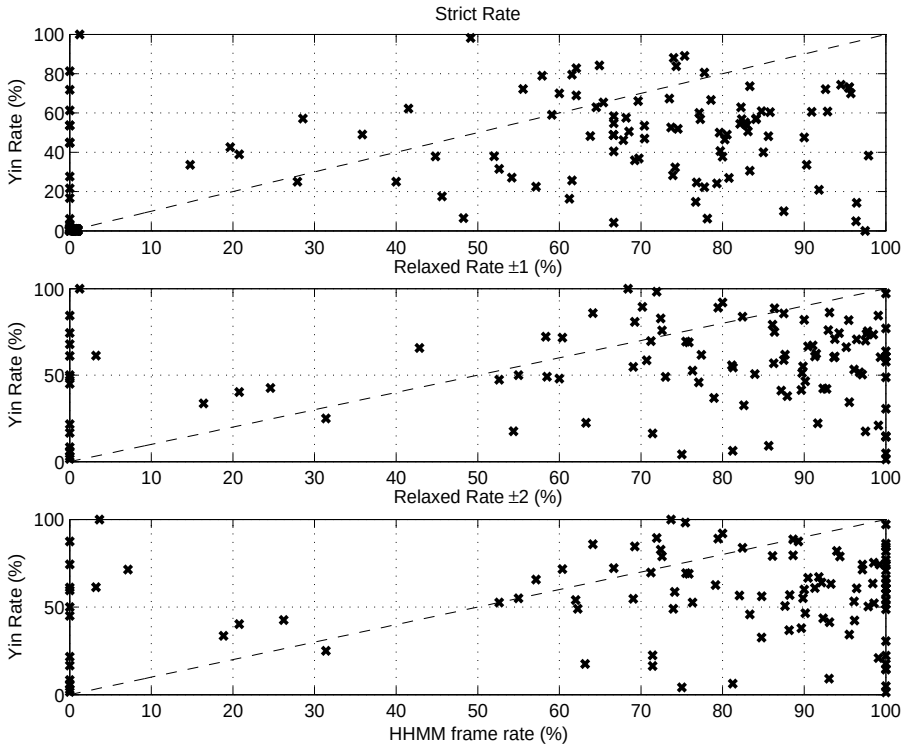


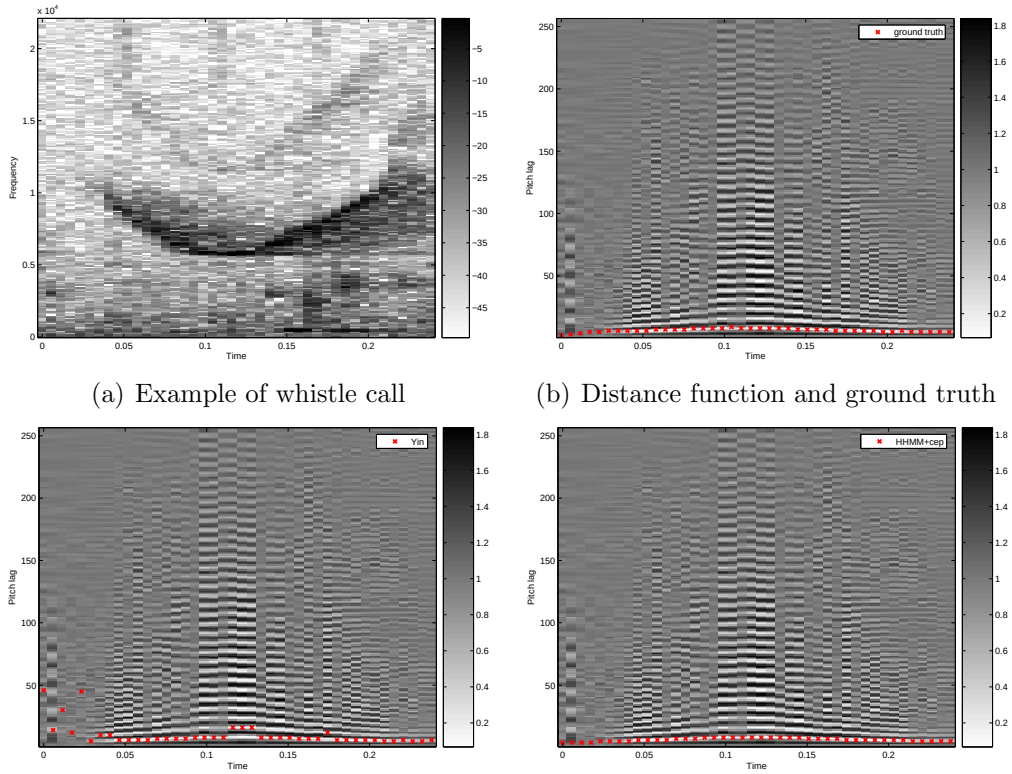
Figure 5.5: Yin frame rate vs. HHMM frame rate for every call

the strict rate (top), relaxed rate ±1 (middle) and relaxed rate ±2 (bottom). As seen in the figure there is a clear shift of the points towards the right side of the plots. This is indicative of the superiority of the system where there is a higher percentage of calls that are achieving above 80% frame accuracy. In addition there is an interesting fact that arises from the plots. There appears to be a constant number of calls giving a near 0% match. The discrepancy is caused due to the error that is introduced with the use of the hierarchy. Basically, these calls fail to get classified in the correct frequency range, thus the pitch extraction fails completely.

Finally, a closer look is presented for two calls when the HMM and Yin

algorithms are performed. Figure 5.6 depicts the resulting pitch extraction for a whistle call. It is evident from the figure that the proposed algorithm of the HMM and the cepstral features works better than simple YIN. Also in Figure 5.7 the HMM again depicts superior performance than YIN although it appears to misclassify several frames at the end of the call where even the ground truth is questionable indicating the errors of the semi-automatic way of labeling.

Overall, the proposed algorithm of the HMM with the cepstral features seems to outperform the baseline algorithms of Yin and get_f0. Although the existence of the bimodality in the data needs to be explored in much larger data corpuses it appears to be the key point in the success of the algorithm and also it offers an impressive reduction to the computational cost. The hierarchy level introduces and overall error of about 4% when it comes to deciding the range of every new call.



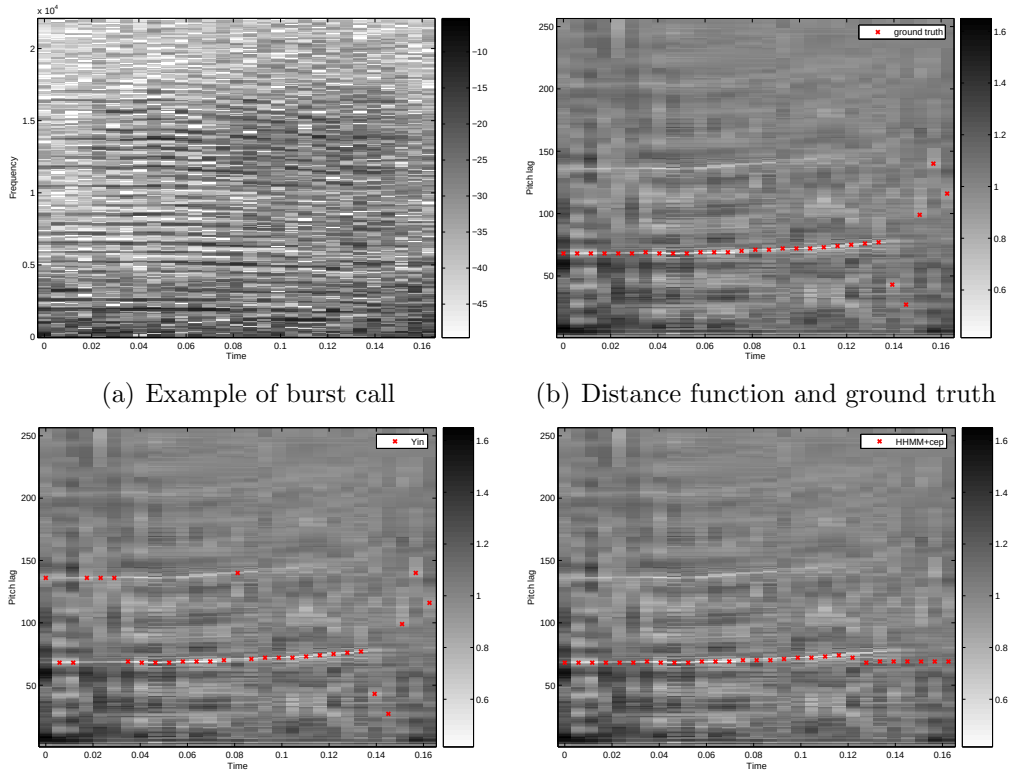
(a) Example of whistle call

(b) Distance function and ground truth

(c) Distance function and resulting pitch using YIN

(d) Distance function and resulting pitch using HMM and cepstrum

Figure 5.6: Example of pitch extraction for a whistle call



(a) Example of burst call

(b) Distance function and ground truth

(c) Distance function and resulting pitch using YIN

(d) Distance function and resulting pitch using HMM and cepstrum

Figure 5.7: Example of pitch extraction for a burst call

5.2 Pitch extraction using Bayesian inference

Dolphin vocalizations meant for interaction are classified into whistle and burst calls. The main differences of these types of calls have been described in Chapter 3. It is worth noting that through the examination of these recordings it has been found that when it comes to their social functionality whistle calls are more prominent than burst calls. Dolphins appear to vocalize whistles more than burst calls, thus adding to the variety of these calls. Researchers theorize that if there is some type of information encoding scheme in dolphin vocalizations then the most likely "carrier" of such information would be whistle calls.

Detecting and extracting the pitch of these whistle calls in an automatic way would offer a great amount of help for the understanding and description of these calls. Given the amounts of data accessible and their distinct characteristics a probabilistic approach is proposed based on Bayesian inference [15, 40, 26]. The novelty of the system is that it actually attempts to resolve overlaps between whistle calls. As described previously, the hardest problem when dealing with dolphin sounds is that due to their social behaviors e.g. existing in pods, dolphins tend to have multiple "conversations". This can be seen in Figure 3.2 where the difficulty of separating these overlaps becomes clear.

The proposed system [21] is based on a probabilistic framework with the goal being to detect and extract dolphin whistle calls. A schematic overview of the proposed system is depicted in Figure 5.8. Whistle calls are AM-FM

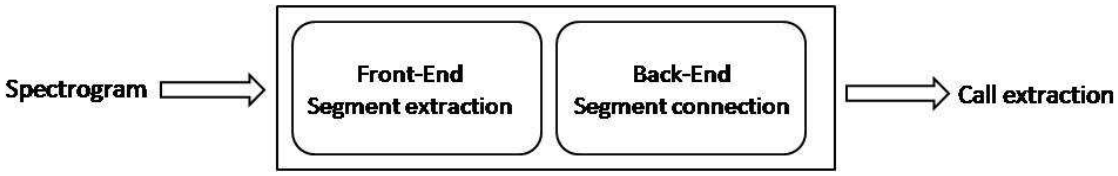


Figure 5.8: System overview for whistle extraction

signals that are described in Equation 3.2. The overall system is based on a Bayesian probabilistic framework that can be described as follows. The goal is to find the most probable call structure, C, given the observations, recordings, o. Through Bayes rule this can be obtained by Equation 5.9.

$$p(C|o) = \frac{p(o|C)p(C)}{p(o)} \tag{5.9}$$

Where $p(o|C)$ is the probability of seeing the actual observation given the hypothesized call parameters that define the call structure, and $p(C)$ is the prior of that call structure.

The observation signal, defined by the recordings, can be described as seen in Equation 5.10

$$o(t) = y(t) + n(t) \tag{5.10}$$

Where $o(t)$ is the observed signal, $y(t)$ is the ideal track waveform and $n(t)$ is the noise.This formulation allows us to define the likelihood shown in Equation 5.11, which implies that the observation can be modeled using a normal distribution with mean equal to the underlying call, and variance resulting from the noise.

$$p(o|y) = N(o; y, \sigma_n) \tag{5.11}$$

As mentioned before, C is the call structure that can be described through some global characteristics e.g. frequency range e.t.c. that will be presented later. One can then define $p(C)$ as the prior distribution on these parameters. This distribution can be modeled by a D-dimensional Gaussian given by Equation 5.13.

$$p(C) = N(\mu_D, \Sigma_D) \tag{5.12}$$

Continuing, the ideal waveform $y(t)$ is randomly related to the call parameters, C, which implies the existence of the $p(y|C)$ distribution and thus introducing it in the computations. Having described the main parameters in this formulation it is evident that the desired result is to obtain the probability of a specific call given the observations. The analytical formulation for that can be seen in Equation 5.13

$$p(C|o) \propto \sum_{y} \frac{p(o|y)p(y|C)p(C)}{p(o)} \tag{5.13}$$

In order to get the maximum likelihood inference of the call parameters, C one needs to maximize only the numerator since the denominator $p(o)$ does not depend on the call parameters, C. In Figure 5.8 the system is described as being comprised of two sub-systems. The front-end, responsible for the extraction of segments that belong to calls, and the back-end, responsible for connecting these segments in order to form the calls. In Equation 5.13 the sinusoidal modeling scheme [20, 17] reduces the search to only a few possible choices for y, namely those that consist of sinusoidal components from the front-end of the system which is reflected on the term $p(o|y)$. The back-end of

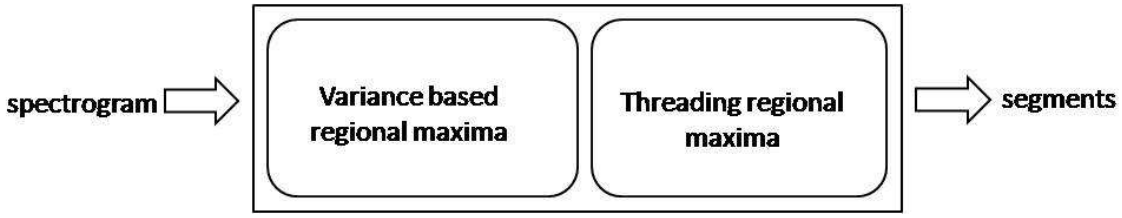


Figure 5.9: Front-end for whistle extraction

the system is reflected on the $p(y|C)p(C)$ term representing the formation of the calls based on the call parameters.

5.2.1 The front-end: Extracting sinusoidal segments

Sine wave modeling is based on modeling the sound as AM-FM sinusoids. This fits perfectly with the case of dolphin calls that are highly AM-FM signals. In practice, sine wave modeling can be performed directly on the spectral magnitude. Basically, one needs to find the local maxima of the spectral magnitude along all frequencies. Once the local maxima have been found, they are tracked along time while making sure to identify their boundaries.

The goal for the front-end of the system is to extract as many fragments of calls without having a high false positive rate e.g. extracting segments that might belong to noise background. Figure 5.9 provides a schematic description of the front-end of the system.

Initially, when applying sine wave modeling the spectrogram is scanned across all frequencies and the regional maxima are extracted. Regional maxima are defined as a set of connected points of constant value from which it is impossible to reach a point with a higher value without first descending.

However, if one extracts initial segments using the regional maxima then they will definitely encounter a large number of false positives. This is mainly due to the noisy recordings as well as the interferences that are present in dolphin recordings.

In order to alleviate the problem of false positives a hybrid regional maxima technique is employed. This is an iterative algorithm based on the variance of the spectrum at each time frame. The algorithm can be seen in Equation 5.14.

$$\begin{aligned} peak_{t,i} &= max(RegMax_{t,i+1}), \text{while } \vartheta < 0.65 \\ \vartheta &= \frac{var(RegMax_{t,i+1})}{var(RegMax_{t,i})} \\ RegMax_{t,i+1} &= \{RegMax_{t,i}\} \setminus \{peak_{t,i}\} \end{aligned} \tag{5.14}$$

Where $peak_{t,i}$ defines the extracted peaks from each time slice, t at every iteration, i. $RegMax_{t,i}$ are the maxima of that time slice, t at each iteration, i. $RegMax_{t,i+1}$ are the maxima of time slice, t and iteration $i+1$ excluding the extracted peaks at time slice t and iteration i.Continuing, ϑ is the threshold that determines the number of peaks that are extracted at each time slice. It basically measures the variance ratio at each iteration, i and it changes at every time slice, t and iteration, i. Empirically and through the training data the upper bound of the threshold has been found and it can be shown that the algorithm can extract the correct regional maxima when $\vartheta < 0.65$. The algorithm is initialized for $i = 1$ where the regional maxima of slice t are extracted and the first peak extracted is the global maximum for that time slice, t.

This hybrid methodology will separate the desired signal from its noisy background using the fact that the signal appears as accentuated peaks in the spectrum. These peaks will have a larger amplitude than that of the existing noise. Although the methodology cannot guarantee a complete lack of false positives it will significantly reduce their number.

Once the peaks have been extracted using the hybrid regional maxima technique, quadratic interpolation is performed on the amplitudes and corresponding frequencies [34]. This will provide a smoother effect on the extracted segments that will later be joined to create the desired calls.

Once the peaks have been extracted for a time slice, t there needs to be a decision regarding their connection with the peaks extracted in the subsequent time slices. That decision is based on two parameters:their frequency and their amplitude.

Each peak lies within a specific frequency channel and in order for it to be connected with a subsequent peak then they cannot be separated by more than a predetermined frequency "distance". Empirically, and given that whistle calls have steep frequency slopes the allowable frequency distance is set to 5 frequency channels/bins e.g. $5 \cdot \frac{44100}{512} = 430 Hz/msec.$ An exhaustive search is employed across the peaks and a preference is given to the ones that have the smallest distance.

Understandably, the above process is very likely to lead to several ties in the frequency domain. In order to resolve these instances, a second level of decision is employed through the use of the ratio of the amplitudes that cannot exceed a predefined amount.

The procedure is repeated at every time step, t, and a segment is considered extracted if there is no continuation found after a number of time steps, that are called dead steps. In all the experiments the number of dead steps was set to 3 time frames. Finally, once the segments have been extracted a post smoothing is employed and segments that don't satisfy a minimum length criterion are excluded, thus keeping only those segments that have more than two points.

Figures 5.10, 5.11 show examples of the front-end of the system. These sample recordings of whistles present an instance that would be considered simple with no overlaps. Specifically, the spectrogram (top) and segment extraction (bottom) with black circles indicating beginning of segments are depicted. The second clip shows an example of how the front-end would deal with overlaps. Specifically, the spectrogram (top) and segment extraction (bottom) with black circles indicating beginning of segments are depicted.

From the figures, it is evident that the majority of the call segments have been extracted and that there is a small percentage of error due to either the extraction of segments that are actually part of the noise background (false-positives) or because some of the extracted segments are excluded due to length requirements (false-negatives).

5.2.2 The back-end: Forming calls from segments

The goal for the back-end of the system is to connect the extracted segments from the front-end according to some decision made through maximum likelihood. Once the segments have been extracted using the algorithm described

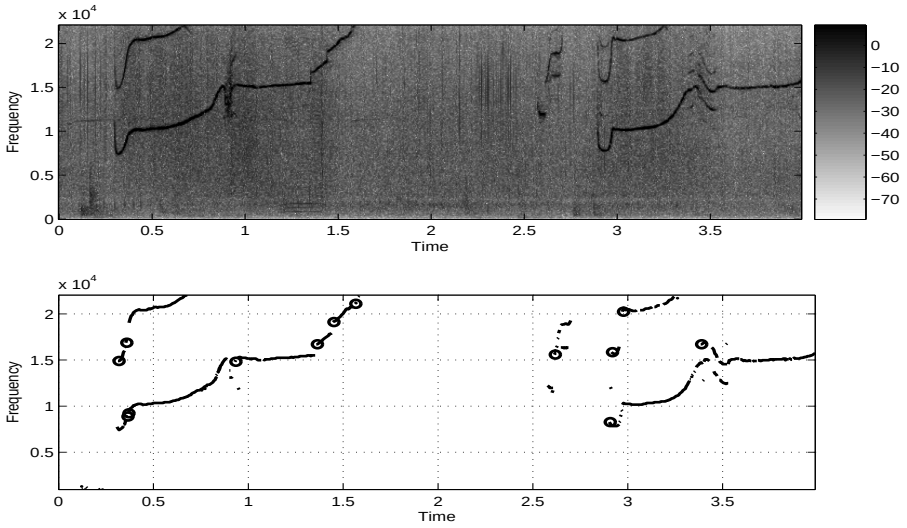


Figure 5.10: Example of the front-end of the system for a simple clip

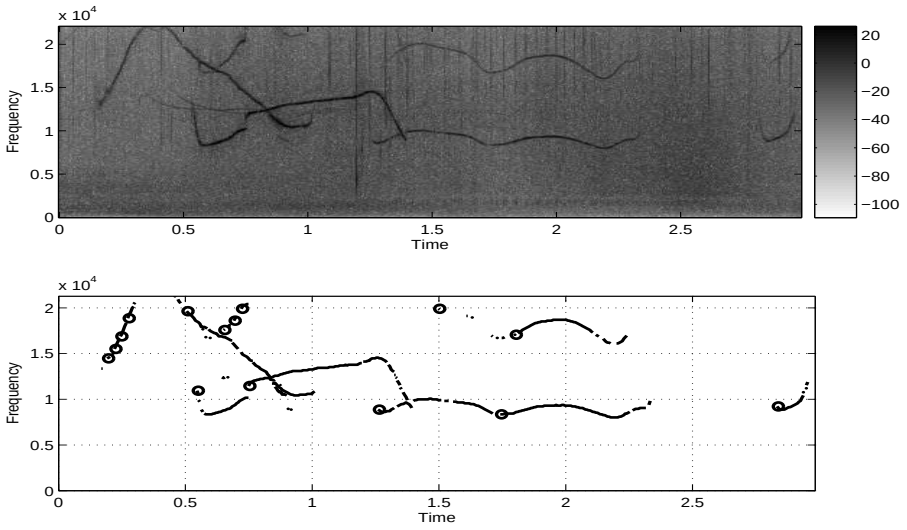


Figure 5.11: Example of the front-end of the system for a clip with overlaps

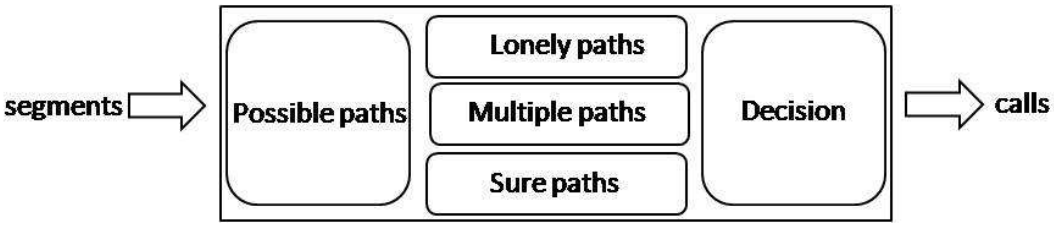


Figure 5.12: Back-end for whistle extraction

in the previous section the problem of connecting them can be formulated into two basic categories:

- Intra-call discontinuities
- Inter-call discontinuities

Intra-call discontinuities can be defined as connecting gaps that appear within the same segment. In order to connect these gaps a time gap parameter is defined that allows for the maximum gap within the segment. When such gaps are found linear interpolation of the edges is performed in both frequency and amplitude. This procedure will account for those calls that have a large amplitude difference within them which leads to gaps of fewer than the "dead steps" count of frames, where the parameter extraction for a single track was lost for one or two frames.

Inter-call discontinuities account for the main part of the back-end and basically refer to connecting a segment with a best choice out of a set of fragments. Figure 5.12 provides a schematic overview of the back-end of the system.

The first thing that needs to be defined in order to locate candidate fragments is a suitable search neighborhood. This neighborhood is adaptive ensur-

ing that every segment has a neighbor. The neighborhood can be perceived as a grid defined by two parameters, frequency and time. These parameters define the final boundaries of the search area e.g. ± 50 frequency bins, ± 10 time steps. It should be noted that each segment has two search areas, one for each tip and a restriction is placed so as to ensure that the segments are not overlapping in time, by having an asymmetry in time depending on the directionality of the tips i.e. $+10$ steps when looking forward in time, thus obtaining temporal consistency.

In order to decrease the size of the search neighborhood even more a tip directionality criterion is used. For each of the tips of each segment within the same search neighborhood its slope is measured by taking the average gradient over 5 time steps closer to the edges. From the training data normal distributions have been fitted on the slopes of instances of pairs of segments with an upward or a downward directionality that belong to the same track. There were 60 pairs for each directionality and they were obtained by running the front-end of the system on the training data and then manually inspecting and extracting each pair. The slope of the tips is used to figure out the directionality likelihood for a pair of segments. The segments that have the highest likelihood are the ones that are considered. This is more clearly seen in Equation 5.15.

$$\begin{aligned} p(sl_{up}|\theta_{up}) &= N(\mu_{x,y}, \Sigma_{2x2}) \\ p(sl_{down}|\theta_{down}) &= N(\mu_{x,y}, \Sigma_{2x2}) \\ \text{if } p(sl_{up}|\theta_{up}) &> p(sl_{down}|\theta_{down}) \Rightarrow \text{upward} \end{aligned} \tag{5.15}$$

Where $p(sl_{up}|\theta_{up})$ describes the likelihood of a pair of segments belonging to the same call and having an upward directionality. The distribution is approximated through a $2D$ Gaussian whose parameters are the slopes of pairs of tips that belong to the same track. These values were obtained from the training data. The same procedure is employed for the downward directionality thus providing a smaller set of neighbors to a segment. This is necessary when dealing with multiple overlaps in order to ensure that possible connections from false segments are limited in the next stage.

Once all the possible paths have been identified then it can be assumed that there are three kinds of possibilities for these paths.

- Sure path: A segment that only has a single connection
- Lonely path: A segment that has no possible connections
- Multiple paths: A segment that has multiple connections

The cases of sure and lonely paths are straightforward. However, in the case of multiple paths there needs to be a decision regarding which path will lead to the extraction of the correct call. In order to make the decision features extracted from the training data are used by forming a combination of scalar normal distributions. The features that are chosen are the smoothness in the curvature of the frequency and the change in energy as seen in Equations 5.16, 5.17.

$$F_{sm} = |mean(\frac{d^2 f}{dt^2})| \tag{5.16}$$

$$E_{sm} = |mean(\frac{de}{dt})| \tag{5.17}$$

Once the distributions have been fitted on values extracted from the hand-labeled training data, a decision can be made by using maximum likelihood since the likelihood of a path can be evaluated as seen in Equation 5.18.

$$p(\text{possible path}) = p(F_{sm}|\text{true path})p(E_{sm}|\text{true path}) \tag{5.18}$$

Also, in order to alleviate the overall computational costs of the system a greedy search amongst the eligible paths is performed choosing the one with the highest likelihood e.g. highest $p(possiblepath)$. The algorithm proceeds in that manner to connect the different segments and only keeps the paths whose likelihood does not drop more than a specific percentage e.g. 10%. Once the connections are established then the segments are merged according to their linearly interpolated value to provide a smoother contour. If for some reason to paths fail to meet the criterion then they are set aside.

Figures 5.13, 5.14 provides examples on how the back end of the system will actually merge the different segments in order to extract the correct calls. Specifically, the spectrogram (top) and call extraction (bottom) with black circles indicating beginning of calls are shown.

From the figures one can see that the majority of the calls in the recordings are correctly extracted by the system. In some cases, in Figure 5.13 there are some false negatives but overall the system performs well at a per frame level. In Figure 5.14 a more elaborate clip is presented with multiple whistle overlaps.

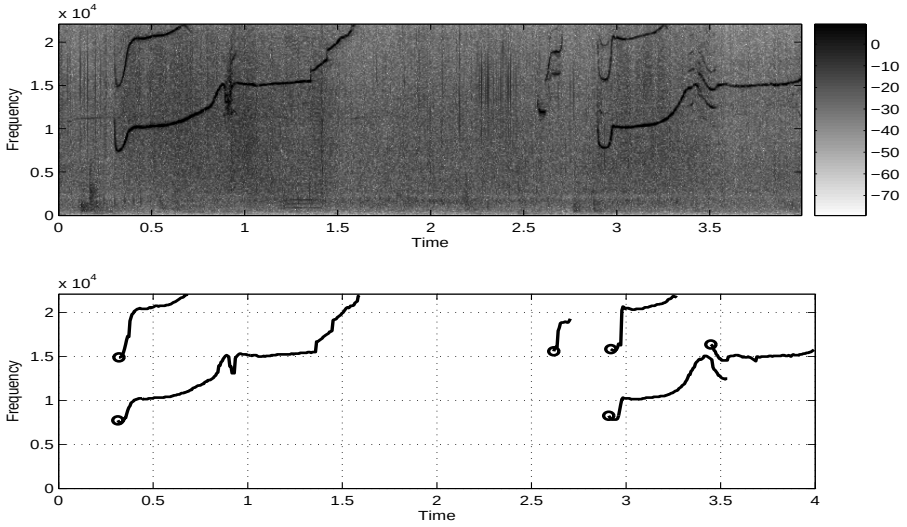


Figure 5.13: Example of the back-end of the system for a simple clip

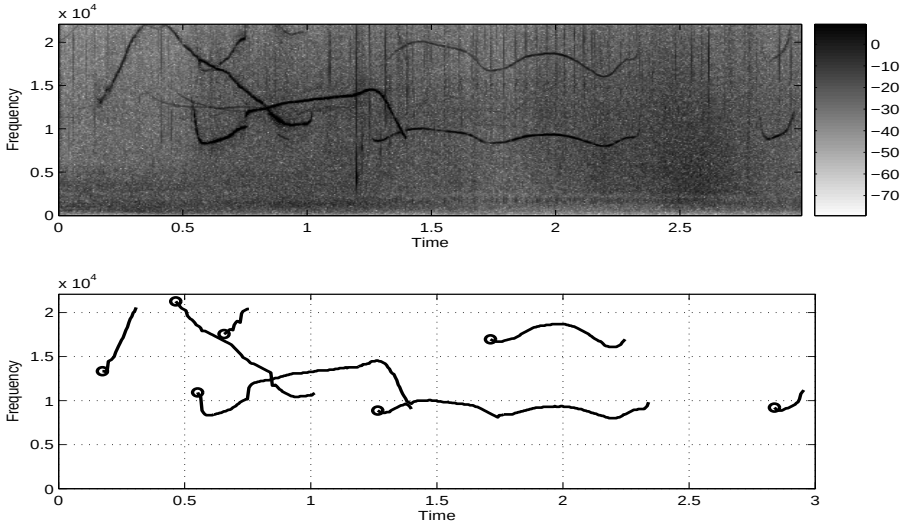


Figure 5.14: Example of the back-end of the system for a clip with overlaps

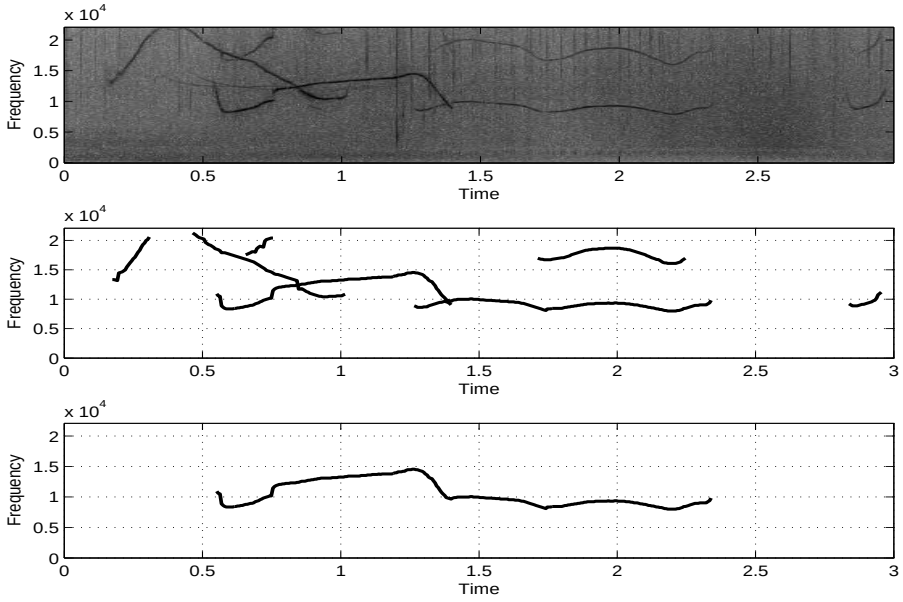


Figure 5.15: Resolving overlaps

Once again there are several false negatives here especially when it comes to harmonics. However, one of the most interesting points of this system is its ability to resolve simple whistle overlaps. In Figure 5.15 an example of a whistle with a double overlap is shown. Specifically, the original spectrogram (top), final extraction of calls (middle), and example of call with resolved and non-resolved overlap (bottom) are shown. The overlap is resolved in one instance, but not on the other instance. This is due to the allowable steps within a segment as well as due to the fact that there are click interferences in the background.

In Table 5.3 results are depicted on 5 min clips that were manually labeled. These recordings have 400 calls of moderate difficulty. The frequency contour of these calls was manually extracted via peak picking on the spectrum and all overlaps were visually inspected in order to be resolved. The front-end of the

system was used to obtain the segment slopes for the training data. Annotation was an arduous process given that there were several parameters that needed to be obtained in both a frame and segment level. The data used are wild dolphin recordings as obtained from Cornell University's MaCaulay Library. All recordings have a sampling frequency of $44100Hz$. It is worth noting that in order to obtain the rates that are presented, the performance of the system needs to be assessed. Given that the system is comprised of two different sub-systems, the overall success of the extraction algorithm depends not only on how well the front-end performs, since if a segment is not extracted then that call will not be represented, but also if the segments that are connected correspond to existing calls. It is necessary to explore other types of metrics for the evaluation of this system, but this would be suitable for future work.

Taking the above into consideration, an overall extraction rate is provided that is obtained on the frame level. Thus, for every track that is extracted from the back-end of the system, the number of correct points is measured and compared with the ground truth independent of which track they belong to. Also, a true positive and false positive rates are provided, but on the segment level. This is obtained by counting the number of correct and non-correct connections respectively and provides information on the success of the decisions of the system. The false positive rate is fairly high due to the fact that the system is created under the assumption that the extracted segments are more likely to be close to each other when belonging to the same call and their connection is desired.

Rates Bayesian Inference	
Frame Level	
Success Rate	82%
Segment Level	
FPR	5%
TPR	97%

Table 5.3: Rates for whistle system

Overall, this system performs well for easy and moderate clips of whistle calls. The big advantage of the system is that it can actually handle simple cases of overlaps, the most difficult problem in dolphin vocalization analysis. The main drawback is that it is a computationally expensive system comprised of two sub-systems thus allowing for the existence of errors and also it requires a lot of parameters to be tuned according to the different recordings.

5.3 Conclusions on pitch tracking of dolphin calls

This work has been divided into two major tasks consisting of the detection and pitch extraction of dolphins' social vocalizations. Considered one of the most difficult things in the field, pitch tracking of dolphin vocalizations presents challenging issues. In this chapter two different systems were proposed and compared with baseline algorithms that have been widely used in the audio processing community. These two systems are based on a probabilistic framework, incorporating time dependencies.

The first system implies that the detection task has successfully been per-

formed and one needs to automatically extract the pitch on the detected whistles and/or bursts. However, this system cannot handle overlapping calls. Once again the use of a probabilistic framework appears to favor the task of pitch extraction. Compared to other widely known algorithms in the field of audio processing such as YIN and get_f0 the proposed algorithm of hierarchically driven HMM's with the use of the cepstral features is far more superior. The key of the system is the exploitation of the bimodality present in the data as seen in Figure 5.4. A decision level is inserted that basically classifies an incoming test call as a whistle or burst according to its frequency range. Of course that will introduce an error for false classification, but in the experiments performed that error did not exceed 4%.

Figure 5.16 presents the resulting accuracies for the pitch detection task of the first system verifying the success of the proposed algorithm. If the system is allowed a soft margin of error of 3% then there is an accuracy rate of approximately 78%. Both Yin and get_f0 trail by a lot when it comes to pitch extraction. Overall the system behaves well and the use of cepstral coefficients aids in the description of the pitch while allowing a much needed flexibility in the dimensionality of the feature space, unlike the use of the spectral magnitude.

The second system uses Bayesian inference and only deals with the pitch extraction of whistle calls. Using specific characteristics of these calls, distributions are obtained and used in order to decide at a per frame level how and where a whistle call is formed. Given the nature of the system several issues arise such as the need for a large training set or the use of parameters that

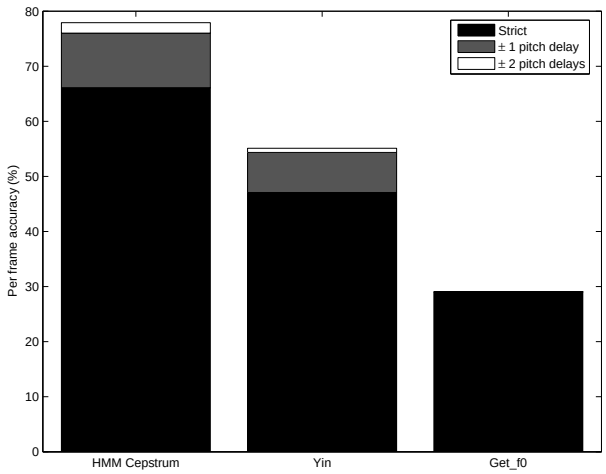


Figure 5.16: Comparative results for pitch extraction of whistles and bursts

are dependent on the different recordings e.g thresholds. However, this system provides the ability to resolve simple overlaps of whistles. The problem of overlaps in dolphin calls is considered as probably the hardest to solve. This is the first system in the field that will actually successfully extract simple overlapping whistles. As seen in Chapter 5 the system will actually perform fairly well by having a success rate at a per frame level of 82%. In order to create a more generalized approach one would need larger data corpuses as well as the use of other descriptive features unique to each type of call.

Overall, the task of pitch extraction is adequately represented in this work since the systems proposed offer the best accuracies in the field so far. The difficulties that are inherent in pitch tracking, especially for these signals that suffer from a low SNR as well as multiple overlaps and interferences, require the use of generative models. In order to account for the wide variety of dolphin calls it would probably be preferable to build advanced Bayesian networks that

will be able to handle the detection and pitch tracking while accounting for the intricacies of dolphin vocalizations. This work offers an insight on how to approach dolphin vocalizations and describes those methodologies that appear to be suitable for the detection and pitch extraction of whistles and bursts. Further analysis and experimentation is required in order to obtain a global automatic system able to deal with the variations found not only in different recordings, but also the different dolphin species that exhibit diversities in their vocalizations.

Chapter 6

Conclusions and future work

> "So long, and thanks for all the fish"
> Message left by the dolphins when they departed planet earth
> Douglas Adams, Hitchhiker's Guide to the Galaxy

Throughout this work the goal was to provide valuable tools for the analysis of dolphin vocalizations. Given that audio processing has been mostly identified through tasks performed on human speech and music analysis, most existing methodologies are highly tuned for those genres. It is understandable that most engineering researchers are inclined to tackle problems in speech and music due to their highly commercial applicability. Unfortunately, that has resulted to a slower advance rate in fields such as animal bioacoustics. Specifically, the field of marine mammal bioacoustics has not been explored in a more robust engineering manner until fairly recently.

Until the Marine Mammal Protection Act in 1972 was enforced by the Navy, little information was publicly known about the many abilities of dolphins

and their highly acoustic world. The Navy, which had been interested in the echolocation abilities of these mammals outreached to institutions and labs around the world in order to get more scientists to explore and analyze marine mammal vocalizations. That led to a massive attempt by marine biologists and animal cognitionists to approach and understand marine mammals. On site researchers started using specialized hardware in order to record the sounds made by dolphins, both in wild and captive environments. As a result, an immense amount of data was created, but the tools for analyzing said data were still in a primal state. Without the existence of automatic and robust software packages that would detect and extract the desired signals in long underwater recordings, scientists are unable to formulate new theories on the interactions of dolphins and provide an insight into their universe.

Audio and machine learning engineers are now realizing the intrinsic and exciting problems that comprise of the analysis of dolphin vocalizations. Dealing with underwater sound and sounds emitted by a different species demands the collaboration of different fields in order to extract meaningful conclusions.

It is my belief that in the work presented here I have been able to provide a better understanding of these exciting creatures while offering important suggestions and solutions to problems that have long been identified when it comes to creating automatic systems for the detection and tracking of dolphin vocalizations such as whistles and bursts.

Overall in this work, two major tasks have been described. Initially, the task of detecting dolphins' interactive calls e.g. whistles and bursts in long recordings, Chapter 4, and secondly the task of extracting the pitch of these

calls, Chapter 5.

In Chapter 4 a variety of methodologies are explored. Starting from the simple energy thresholding, that is widely used in the field, a closer look at the data uncovers that there seem to be "preferred" channels in both whistles and bursts. In order to exploit this information the optimization technique of gradient descent is used in both linear and non-linear manner. Continuing, more intricate models are also employed such as Gaussian models in an effort to provide a general fit of the data. Also, a more robust classifier, Support Vector Machines (SVM) is used given its strength in similar classification tasks in speech and music. SVM's outperform every other methodology regardless of which features are used i.e. spectral or cepstral.

Although SVM's outperform every other algorithm, one can imagine that the use of each algorithm is dependant on the nature of the task i.e. if the system is needed for on site detection we might need to sacrifice accuracy for computation, thus the use of GM's in an online fashion might be preferable. Also, the choice of the different proposed features during detection indicates that the success of the systems lies mostly on the choice of classifier.

In Chapter 5 two main systems were proposed for the pitch extraction of whistles and bursts. The first system could be considered as an add-on on the detection schemes since it assumes that all calls have already been detected and segmented. Based on hierarchically driven HMM's, each call is first classified as a whistle or burst according to its frequency range and then the pitch is extracted with the use of HMM's. The proposed algorithm is compared to YIN and get_f0 where its superior performance is highlighted.

The second system is based on a probabilistic framework and deals with the detection and extraction of whistle vocalizations. based on sinewave modeling and learned statistics directly from the training data, this system offers a first approach on handling the most difficult problem in the field, whistle overlaps.

At this point it is worth mentioning that several other approaches were considered for the analysis of dolphins' social vocalizations. Given that these sounds are produced through non-linear systems, one would argue that the straight forward linear approach widely used throughout this work might not be suitable. As presented in Chapter 3, the tool that I used for visualization and analysis of these calls is the spectrogram. The spectrogram is known for its inherent trade-off of frequency and time resolution. In order to explore a different non-linear transform that might provide better analysis, the Fan-Chirp [16] transform was used. The Fan-Chirp is a wavelet, whose mother function is a frequency modulated signal that can be adjusted to match desired signals. Although, the use of a wavelet eliminated the resolution issues, it showed a preference to specific signals based on the tuning of the modulation index. Clearly, in order to be able to take advantage of wavelets for dolphin calls there needs to be a predetermined set of whistles and bursts that need to be identified.

Also, non-linear features such as the fractal dimension, FD were explored in order to try and discriminate between whistles and bursts. Fractal dimension is based on the Minkowski-Bouligand dimension [49] and basically measures the fractal nature of a signal. Basically, for a signal to be considered as a fractal it's fractal dimension needs to be higher than it's topological dimension e.g.

for an audio signal to be a fractal it's FD needs to be above 1. The fractal dimension was measured for all 100 whistles and bursts comprising the training data. In the case of whistles the average FD is 1.45 and in the case of bursts the average FD is 1.54. Although both signals can be considered to belong in the fractal set, this feature is not enough to provide discriminatory information between the two types of calls.

Moreover, given the nature of these signals and their highly modulated character, I experimented with a non-linear demodulation algorithm as presented by Maragos [31] and known as Demodulation Energy Separation Algorithm (DESA). Based on the Teager-Kaiser [27] energy operator this methodology is able to extract the instantaneous frequency and amplitude of a signal. Basically, the energy operator can be viewed as a quadratic filter that is used to analyze the signal. Although, this method provides crucial information on whistles and bursts and is able to discriminate between the two on simple cases, problems arise when overlaps occur between the different calls or when there are multiple interferences from clicks. Because of the drawback, the more mainstream tools and machine learning algorithms were prefered in order to tackle the problems of detection and pitch extraction.

Overall, in this work, a broad understanding of how to manipulate and extract useful information from dolphin recordings has been presented. In order to be able to further analyze dolphin vocalizations there is a tremendous need for an automatic detector and pitch extractor for the desired calls. A variety of methodologies has been provided for both tasks giving insights on these signals. Future work might include the exploration of several other call spe-

cific features i.e. fractal dimension, harmonic mean etc.and more generalized classifiers such as Bayesian networks that will allow the exploration of different types of communication calls. One of the most important things when dealing with dolphin recordings is the need for the existence of standardized species specific data with behavioral labels that researchers can access and use for experimentation purposes. It is my belief that with time and further exploration we will be able to uncover the reasons behind dolphins' communication vocalizations.

Bibliography

[1] C. M. Bishop. *Neural Networks for Pattern Recognition.* Oxford Press, 1995.

[2] M. J. R. Bogert, B. P. Healy and J. W. Tukey. The quefrency analysis of time series for echoes: cepstrum, pseudo-autocovariance, cross-cepstrum, and shape cracking. *Proc. of the Symposium on Time Series Analysis*, pages 209–243, 1963.

[3] A. P Bradley. *Use of the area under the ROC curve in the evaluation of machine learning algorithms.* Pattern Recognition 30(7), 1997.

[4] J. C. Brown and P. J. O. Miller. Classifying killer whale vocalizations using time warping. *Acoustics Today*, pages 45–47, 2006.

[5] J. C. Brown and P. J. O. Miller. Automatic classification of killer whale vocalizations using dynamic time warping. *J. Acoust. Soc. Am.*, (**122)**:1201–1207, 2007.

[6] J. C. Brown and P. J. O. Miller. Mathematics of pulsed vocalizations with application to killer whale biphonation. *J. Acoust. Soc. Am.*, (**123**):2875–2883, 2008.

[7] Caldwell D. K. Caldwell, M. C. Individualized whistle contours in bottlenose dolphins (tursiops trunactus). *Nature*, (**207**):434–435, 1965.

[8] Caldwell D. K. Caldwell, M. C. Statistical evidence for indivindual signature whistles in pacific whitesided dolphins, lagenorhyncus obliquidens. *Cetology*, (**16**), 1973.

[9] D. K. Caldwell, M. C. Caldwell and Siebanaler J. B. Observations on captive and wild atlantic bottlenose dolphin, *Tursiops truncatus*, in the southeastern gulf of mexico. *LA County Mus., Contrib. in Sci.*, **91**:1–10, 1965.

[10] S.-T. J. Christianini. *An introduction to Support Vector Machines and other kernel-based learning methods.* Cambridge University Press, New York, NY, 2000.

[11] C. W. Clark and K. M. Fristrup. Signal processing techniques for passive acoustic location and tracking of animals. *J. Acoust. Soc. Am.*, **106(4)**:2188, 1999.

[12] T. W. Cranford and M. E. Amundin. Biosonar pulse production in odontocetes: The state of our knowledge. *Echolocation in Bats and Dolphins, The University of Chicago Press*, pages 27–35, 2003.

[13] A. de Cheveigne and H. Kawahara. Yin, a fundamental frequency estimator for speech and music. *J. Acoust. Soc. Am.*, **111**:1917–1930, 2002.

[14] J. J. Dreher and W. E Evans. *Cetacean Communication.* Marine Bioacoustics, Vol. 1, pp.373-399, Pergammon Press, Oxford, 1964.

[15] P. Duda, R. Hart and D. Stork. *Pattern Classification.* John Wiley & Sons, New York,NY, 2001.

[16] R. Dunn and T. F. Quatieri. Sinewave analysis/synthesis based on the fan-chirp transform. *Applications of Signal Processing to Audio and Acoustics, IEEE Workshop*, pages 247–250, 2007.

[17] D. Ellis. Sinewave and sinusoid+noise analysis/synthesis in Matlab, 2003. http://www.ee.columbia.edu/ dpwe/resources/matlab/sinemodel/.

[18] Entropic Signal PRocessing Systems. ESPS/WAVES. Online web resource/software package. http://www.speech.kth.se/software/.

[19] F. Glotin, H. Caudal and P Giraudet. Whale cocktail party: Real-time multiple tracking and signal analysis. *Canadian Acoustics/Aqoustique canadienne*, (**36)1**:139–145, 2008.

[20] B. Gold and N. Morgan. *Speech and Audio Signal Processing.* Jon Wiley & Sons, New York, 2000.

[21] X. C. Halkias and D. Ellis. Call detection and extraction using bayesian inference. *Applied Acoustics, special issue on Marine Mammal Detection*, **67(11-12)**:1164–1174, 2006.

[22] X. C. Halkias and D. Ellis. Estimating the number of marine mammals using recordings of clicks from one microphone. *Proc. ICASSP-06 Toulouse*, pages 769–772, 2006.

[23] X. C. Halkias and D. Ellis. A comparison of pitch extraction methodologies for dolphin vocalizations. *Canadian Acoustics/Aqoustique canadienne*, (**36)1**:74–80, 2008.

[24] L. M. Herman and W. N. Tavolga. *The communication systems of cetaceans*. John Wiley & Sons, Inc., New York, 1980.

[25] WEKA 3: Data Mining Software in Java. Online web resource/software package. http://www.cs.waikato.ac.nz/ml/weka/.

[26] M. I. Jordan. *Learning in Graphical Models, Adaptive Computation and Machine Learning*. MIT Press, Cambridge, MA, 1999.

[27] J. F. Kaiser. On a simple algorithm to calculate the.

[28] V. Kandia and Y. Stylianou. Detection of clicks based on group delay. *Canadian Acoustics/Aqoustique canadienne*, (**36)1**:48–54, 2008.

[29] J. C Lilly and A. M. Miller. Vocal exchanges between dolphins. *Science*, **134**:1873–1876, 1961.

[30] O. Motsch J-F. Lopatka, M. Adam and J. Zarzycki. Attractive time-variant orthogonal schur-like representation for click-type signal recognition. *Canadian Acoustics/Aqoustique canadienne*, (**36)1**:81–87, 2008.

[31] J. F. Maragos, P. Kaiser and T. F. Quatieri. Energy separation in signal modulations with application to speech analysis. *IEEE Transactions on Signal Processing*, (**41)10**:3024–3051, 1993.

[32] S. Masse. Source code for leafy seadragon. http://sourceforge.net/project/showfiles.php?group$_id$ = 28542.

[33] S. Masse. User guide for leafy seadragon. http://leafyseadragon.blogspot.com/2006/01/seadragon-2-user-guide.html.

[34] R. J. McAulay and T. F. Quatieri. Speech analysis/synthesis based on a sinusoidal representation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, (**34)4**:744–754, 1986.

[35] D. K Mellinger. Ishmael 1.0 user's guide. *NOAA Tech. report OAR-PMEL-120*, page 30, 2001.

[36] D. K. Mellinger and C. W. Clark. Methods for automatic detection of mysticete sounds. *Marine freshwater Beh. Physiol.*, **29**:163–181, 1997.

[37] D. K. Mellinger and C. W. Clark. Recognizing transient low-frequency whale sounds by spectrogram correlation. *J. Acoust. Soc. Am.*, **107(6)**:3518–3529, 2000.

[38] K. S Norris. *The Porpoise watcher. A Naturalist's experience with Porpoises and Whales*. W. W. Norton, New York, 1974.

[39] A. V. Oppenheim and R. W. Schafer. *Discrete-Time Signal Processing*. Prentice Hall, Englewood Cliffs, NJ, 1989.

[40] A. Papoulis. *Probability, Random variables, and Stochastic Processes*. MxGraw-Hill, New York, 1965.

[41] B Perrin, W. F Wursig and J. G. M. Thewissen. *Encyclopedia of Marine Mammals*. Academic Press, San Diego, 2002.

[42] J. Platt. *Fast training of Support Vector Machines using sequential minimizing optimization. Advances in Kernel Methods-Support VEctor Learning*. MIT Press, Cambridge, MA, 1998.

[43] L. R. Rabiner and B. H. Juang. An introduction to hidden markov models. *IEEE ASSP Magazine*, pages 4–15, 1986.

[44] D. Reiss and L. Marino. Mirror self-recognition in the bottlenose dolphin. a case of cognitive convergence. *Proceedings of the National Academy of Sciences of the United States of America*, (**98)10**:5937–5942, 2001.

[45] D. Reiss and B. McCowan. Spontaneous vocal mimicry and production by bottlenose dolphins (*Tursiops trunacatus*): Evidence for vocal learning. *J. Comp. Psych.*, **107**:301–312, 1993.

[46] D. Reiss and B. McCowan. Quantitative comparison of whistle repertoires from captive adult bottlenose dolphins(*Tursiops truncatus*): A reevaluation of the signature whistle hypothesis. *Ethology*, **100**:193–209, 1995.

[47] S. H. Ridgway. Dolphin hearing and sound production in health and illness. *Hearing and Other Senses: Presenations in Honor of E. G. Wever, Amphora Press*, pages 247–296, 1983.

[48] M. S. Hoenigman R. Wiggins S. M. Roch, M. A. Soldevilla and J. A. Hildebrand. Comparison of machine learning techniques for the classification of echolocation clicks from three species of odontocetes. *Canadian Acoustics/Aqoustique canadienne*, (**36)1**:41–47, 2008.

[49] M. Schroeder. *Fractals, Chaos, Power Laws: Minutes from an Infinite Paradise*. W. H. Freeman, New York, 1991.

[50] B. Secrest and G. Doddington. An integrated pitch tracking algorithm for speech systems. *Proc. on Acoustics and Speech and Signal Processing, IEEE ICASSP*, **8**:1352–1355, 1983.

[51] R. S. Shane, S. H. Wells and B. Wursig. Ecology, behavior and social organization of the bottlenose dolphin: A review. *Marine Mammal Science*, (**2)1**:34–63, 1986.

[52] M. A. Henderson E. E. Campbell G. S. Wiggins S. M. Soldevilla, M. S. Roch and J. A. Hildebrand. Classification of risso's and pacific white-sided dolphins using spectral properties of echolocation clicks. *J. Acoust. Soc. Am.*, (**124)1**:609–624, 2008.

[53] P. Tyack. Whistle repertoires of two bottlenose dolphins, *Tursiops truncatus*:mimicry of signature whistles? *Behav. Ecol. Sociobiol.*, **18**:251–257, 1986.

[54] I. H. Witten and Frank. E. *Data Mining: Practical Machine Learning Tools and Techniques*. Elsevier, 2005.