INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY
CAMPUS ESTADO DE MEXICO

TECNOLÓGICO
DE MONTERREY®

Bird Song Formal Language Modeling Based on Acoustic Syllable Detection

Dissertation Proposal to Opt for the Degree of Doctor in Computer Science

Ivan Alejandro Escobar Broitman

Supervisor:	Dra. MARÍA DE LOS ÁNGELES JUNCO REY
External Supervisor:	Dr. CHARLES TAYLOR
Dissertation Committee:	Dr. EDGAR EMMANUEL VALLEJO CLEMENTE
	Dr. MIGUEL GONZÁLEZ MENDOZA
	Dr. SAUL LOZANO FUENTES

Atizapán de Zaragoza, Edo. Méx., February 2007.

Table of Contents

List of Figures

List of Tables

Abstract

INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY
CAMPUS ESTADO DE MÉXICO

Author: Ivan Alejandro Escobar Broitman

Title: Bird Song Formal Language Modeling Based on Acoustic Syllable Detection.

Date: February 2007 **Number of Pages:** 75

Supervisor: Maria de los Angeles Junco
External Supervisor: Dr. Charles Taylor

Birdsong has been regarded as a biological model of human language, especially because of the similarity in developmental processes [24]. The main goal of the project will be to efficiently classify different species by means of generating a formal language, which can describe in a more efficient and clear manner the information, gathered from their songs. The study of birdsong involves many different disciplines that converge amongst each other when interpreting bird songs and calls. We will develop computational tools in order to enhance and simplify the knowledge of ornithologists, biologists, computer scientists and electrical engineers in the study of bird song and their behavior. These tools will help understand the structure of a bird's song and contribute to the interpretation of them. This will be done by means of gathering enough acoustical information in order to perform a syllable extraction analysis of bird songs to construct an alphabet of their sounds. In order to perform this classification, we must analyze in detail the fundamental building blocks of bird songs, which are notes and syllables and find their natural groupings. This information will be extracted from songs recorded in their natural habitats. We must define boundaries and efficient methods for extracting these syllables in order to process the song data and to convert bird songs into strings of symbols, which will model the bird song language. The general idea is to use language theory in order to contribute to the understanding of the structure of bird songs at different sublevels, such as lexical, syntactical and semantical. This work is part of the collaboration between UCLA and ITESM in the ongoing project "Sensor Arrays for Acoustic Monitoring of Bird Behavior and Diversity"[39], whose specific goal is to monitor different bird species from the ecological reserves in California, USA and Chiapas, Mexico.

Keywords: Data mining, Grammatical Analysis, Bird Song, Species Recognition, Feature Extraction, Language Generation, Language Analysis.

1 Introduction

The monitoring of animal behavior and diversity over a variety of spatial and temporal scales poses many challenges to human observers. A significant amount of knowledge on bird diversity and their interpreted behavior is the result of field observations made by expert ornithologists. Bird species identification and the study of their interactions have been developed by means of the visual and acoustic abilities of these experts. Over the years, much of the information gathered from bird songs has been analyzed in order for experts to effectively classify different species through their sounds. This has been done in a great manner through manual work with minimal aid from computer equipments. In the last few years, great advances have been developed both in the fields of computer science as well as in electronics, which can be applied to fields such as biology and ornithology, in order to aid and automatize much of the work performed in the field. Nevertheless, much of the hard breaking work is still being done in a primitive or semi primitive manner by experts using field recordings and their knowledge to classify, locate and interpret bird songs, species and behavior.

These kinds of limitations have caused our knowledge about some bird species with complex societies to be very scarce. We also have very little information about the interactions among species and the influence that environmental factors such as rain, earthquakes, predator invasions, etc., have on the behavior of a particular group of birds. Climatic changes can also affect these bird societies and impact their behavior, population and nest locations. Other factors also interfere with the reliability and accuracy of the information, such as human error and the animal's behavioral changes induced through their interactions with humans.

In this context, the goal of sensor networks [19,38,39] is to introduce in a natural environment a certain number of small sensors or motes, in order to acquire data from their surroundings (figure 2). A wireless sensor network is an autonomous ad hoc system consisting of a collective of networked sensor nodes or motes designed to intercommunicate via wireless radio. The technology of distributed sensor networks also allows us to detect unusual events that occur in a certain environment, such as the presence of rare or endangered bird species, their social interactions and communications without human intervention.

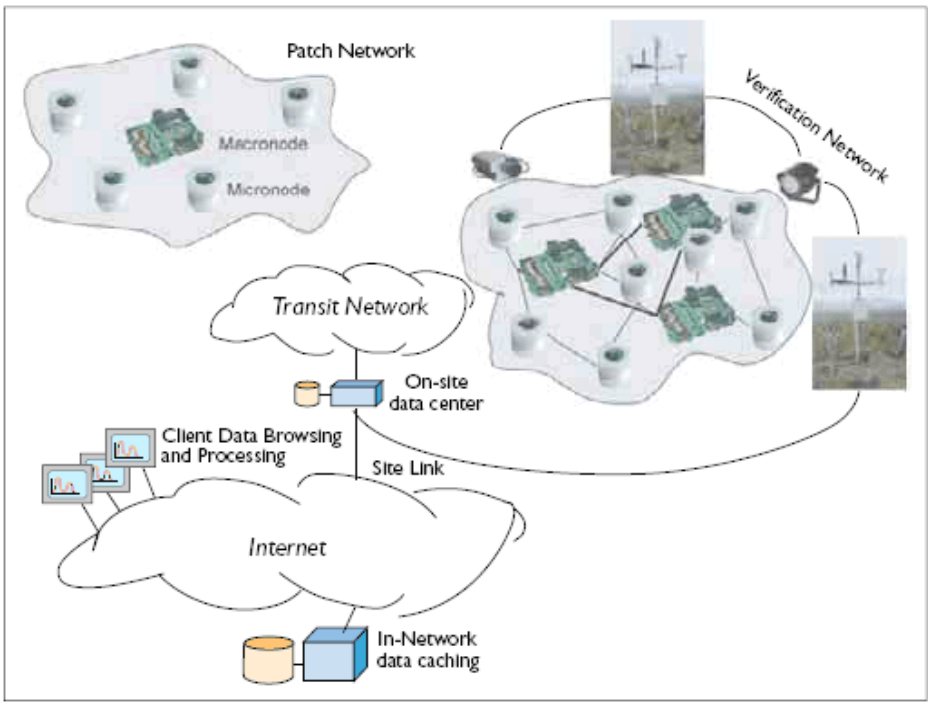


Figure 1:Sensor Network Deployment[38]

A typical deployment of a sensor network is shown in figure 1. We can observe that sensors are placed evenly spaced in the area that is going to be monitored. It is the job of the sensors to do the actual data collection. Samples originate at the sensor nodes, which typically involve heterogeneous sensing capability, processing power and storage. These sensors are typically deployed in dense patches, where each patch corresponds to a particular slice of habitat of interest. The individual patches are often separated in order to perform a more efficient monitoring. The data from the various patches flows through the transit network to an on-site data center.

Figure 2:Sensor mote in plastic cover

Sensor nodes are small battery powered equipment that are installed in the areas of interest, in order to do the monitoring (see figure 2). A typical micro node is usually built around a lower power consumption microcontroller running at a few MIPS with only a few kilobytes in RAM. The different sensing elements take the form of probes connected to a general-purpose signal acquisition board integrated with the microcontroller and a wireless connection kit.

Certain applications such as acoustic bird monitoring require more capable sensors with more computing power and storage. All nodes in a patch form a routing tree to control collect and process biological data. This tree is routed at the gateway node, which provides access to the transit network. Finally, the data produced by the sensor network gains scientific validity through a process of verification and corroboration by means of a verification network.

To be able to explore the full potential of distributed sensor networks in order to analyze bird diversity and behavior we must deal with complex processing challenges. Bird songs, when converted to the frequency domain by means of Fourier transforms, produce large amounts of data that requires processing and discrimination. Implicit relationships must be found; data must be sorted and cleaned. Here is where we can apply what has been done previously in this matter by means of applying different dimensionality reduction techniques such as principal component analysis (PCA) [15] and data mining [45,46,50] to help us alleviate the computational cost that is required to process the raw data. This will help us to incorporate these processing technologies into the existing energy and processor constrained platforms such as nodes in a sensor network. Once we obtained the desired data from the sensor networks we can start to analyze each individual sample in order to construct a simple grammar that models the songs. It will be very significant to obtain a simple but exact representation of the songs.

For this work the sensor network platform that is used is based on the current Intel Stargate platform, with custom build equipment from the Center for Embedded Networked Sensing (CENS) in UCLA as seen in figure 3.

Figure 3: CENS custom-built sensor arrays

Tools to remotely sense, record and automatically analyze acoustical behavior would be of enormous help for the studies of ecology, biodiversity and behavior. Such tools would also be a major step towards realizing the full potential of embedded, networked sensor arrays [39]. If each sensor sees and understands part of the situation that its analyzing depending on its own location and perceptions, each individual sensor abilities will merge in a sensor network to form a coherent view of the situation in hand which would help the processing and understanding of bird vocalizations to be able to effectively classify, locate and distinguish between different bird species [39]. In order to achieve this goal, we must concentrate on the

preprocessing of information and to use these results in order to feed a more stable and efficient classifier by means of expanding current statistical and acoustic analysis. We must create tools to simplify the current processing techniques and to be able to understand the composition and structure of bird songs.

1.1 Problem Description

This work is part of the collaboration between UCLA and ITESM in the ongoing project "Sensor Arrays for Acoustic Monitoring of Bird Behavior and Diversity"[39], whose specific goal is to monitor different bird species from the ecological reserves in California, USA and Chiapas, Mexico. This project proposes to develop sensor arrays in order to find useful observations and to perform efficient analysis of bird diversity and behavior. The focus will be on the sensors and their abilities in order to obtain robustness and adaptability.

The collaboration project has developed four levels of complexity to attack during the five-year process of the project. As mentioned in [39], the levels are:

- Develop filters to identify species and individuals in noisy settings.
- Develop software tools to determine if species/individuals are present in the acoustic neighborhood of the sensor network and correlate that presence with simultaneous temporal or abiotic information.
- Develop software tools to localize individual birds in real settings.
- Employ these tools for study of social interactions among birds.

Software now exists that can accomplish each of these tasks in isolation and in simplified environments, such as a laboratory. The challenge is to extend that to real, complex environments. [39] All of the collaboration project efforts are redirected to the development of software tools and methods in order to use custom built hardware by the UCLA Center for Embedded Networked Sensing (CENS) to perform the remote sensing and analysis.

The goal of a sensor network is to obtain enough data in order for an efficient post processing of the information. For this work, the sensor networks will collect acoustic data in order to achieve a first level of species classification. In a near future, the sensor network should be able to perform minor data mining algorithms in order to simply extract the needed features for more complex high level processing algorithms instead of feeding the actual recorded songs to the on-site data center. Once we have the acoustic data we can proceed to analyze it and use it as a preprocessing stage to create a low-level language model of the bird songs.

Previous work on acoustic analysis applied to bird species recognition, has managed to detect that some of the components of a sound signal that can be specifically tied to different species. These components can be mined from large databases of acoustic signals and used in conjunction with other traits in order to perform species classification. A major problem with these findings is that they can vary significantly depending on the quality of the recorded songs. Another factor to consider is that

depending on the bird species to study, different traits help classification while others hinder it, making very difficult to contrast the different results obtained previously on this matter. This means that there is a need for a more efficient classification system, which can use the preprocessed information gathered from the acoustical analysis, but which can give more efficient results that don't depend on the quality of the recordings.

1.2 Proposal

During the development of this work, we have observed several limitations from the previous work analyzed, which have hindered the production of automatic systems to distinguish and classify bird species by their sounds. These limitations will be shown later on in this document. It is important to note that even though those previous lines of research have strong limitations, they are a fundamental background and the main inspiration for this work. Some of the results obtained by those research scientists are pretty impressive and very useful for the first processing stages of our work. The problem in hand is quite complex since it involves research from different areas such as biology, electronics and computer science, therefore a more objective perspective is required when proposing or modifying current work. The most relevant background work for this document will be analyzed.

Most working systems nowadays strongly rely on the quality of the recordings as well as a strong cleaning and processing stage. Usually sounds are cleaned, reorganized, filtered and then processed through computer algorithms in order to recognize patterns of each individual species. If more than one bird species is being targeted, this can actually make this work pretty complicated and obviously inefficient if we want to take it into the field and do streaming analysis. Field equipments have low power processors, storage and performance constraints that limit the amount of processing that they can do. Also field equipments must withstand climatic changes and must run without human intervention for a minimum period of a few months, limiting their power outputs.

We also have to consider that some of the previous lines of research manage to get pretty impressive results when classifying bird species through their songs. Unfortunately, the computational cost of these results makes those processes useless for field analysis, especially if the goal is to do recognition with streaming audio in power limited platforms.

An important subject to take into consideration is that these preprocessing stages are quite useful and can help build a small database of signal characteristics per species which can help future classifiers do a better job, without having to redo the strong computational processing of this previous stage. For this work we will take as preprocessing stage the work done previously in [9,45,46] using both feature extraction and data mining in order to obtain the important features of each bird species and to do an attribute reduction for the future application in sensor networks. These extracted attributes will constitute a species selection database that will be used to configure and develop the syllable extraction software and classification systems. These systems will aid in the monitoring of bird behavior and diversity.

1.3 Birdsong and Language

Birdsong has been regarded as a biological model of human language, especially because of the similarity in the developmental processes [24]. Birdsong shares another exciting aspect with human language: its syntactical organization. Human language is a hierarchically organized syntactical behavior [24]. It is important to note that birdsong shares developmental characteristics with the human language, therefore giving a more general insight to our research. It is very important to take into consideration these types of relationships when we develop our bird song classifier since the language generation will be closely tied with the natural development of bird songs and their intrinsic relationships with the human language.

The study of birdsong exemplifies a neuroethological approach to understanding brain function, in which a detailed knowledge of naturally occurring behaviors can inform and guide the search for underlying neural mechanisms [4]. Behavioral tests suggest that birds share a sensory phase with humans in which they learn the basic constructs of their language. If we carefully analyze these common factors, we might be able to process in a more efficient manner the extracted syllables from out bird songs and be able to not only classify correctly the different species but also to interpret up to a certain manner the information we can gather. We can also comprehend more about the interactions of different individuals and form the general rules of what is to become the structure of an avian language. It will be very important to understand the general construction of this type of language since it shares many similarities with the human language.

Human language and speech are unique accomplishments. Nevertheless, they share a number of characteristics with other systems of communication, and investigators have thus compared them to birdsong and the vocal singing of nonhuman primates. Particular interesting parallels concern the development of singing and speaking. Birsong is an excellent biological model for memory research and also an appropriate system for the study of evolutionary strategies in a very successful class of organisms [43]. It is also a unique challenge to study how language develops in other species.

A language is a set of strings over an alphabet. The alphabet is the set of symbols of the language and a string over the alphabet is a finite sequence of symbols from the alphabet [36]. A finite language can be explicitly defined by enumerating its elements. It is also know as a *formal language*. When formal languages can be described using finite state automata's, they are called *regular languages*

1.3.1 Hypothesis

Is it possible that by means of feature analysis and through the generation of a simple syllable based language, to generate a semi automatic bird song classifier that will be able to discriminate correctly different species of birds from the formal language representation of their songs?

1.3.2 Objectives

In this work we propose to use the acoustic analysis as a first step in the development of an automatic classifier (see figure 4). This classifier will be based not only on acoustic information but also on the development of a formal language that can model bird songs and their interactions.

Finite automata and their probabilistic counterparts *Markov chains* are useful tools when attempting to recognize patterns in data [31]. These tools are normally used for speech recognition. It is important to correctly identify key elements from bird song recordings and divide them into their simplest form, in order to recognize patterns and construct a language from the bird songs.

It is very important to be able to define from our bird songs, the set of elements that will build the alphabet, in order to create a set of strings that will help constitute the language.

Once we have constructed the set of strings, we will proceed to define a finite state automaton that will be used to recognize these defined strings. We will try to contribute in the induction of such automata, taking into consideration that it will have to be very efficient with our data sets. We will explore different methodologies to induce this finite state automaton, like MDL [15] and evolutionary programming. When we've constructed our regular language by means of a finite state automaton, we will proceed to generalize it, so it can accept more string combinations outside those that were used for its construction. In order to generalize it we will perform different compression techniques. Compression not only allows large amounts of information to be stored more efficiently, but can also enable the system to generalize [41]. Finally we will try to associate meanings to the generated strings (semantic grounding) and generate test string vectors that will be translated back to an acoustic signal and played to the actual bird species to measure their response.

Our purpose here is to be able to model each individual species with this language, perform classification and ultimately contribute to the understanding of the structure and function of bird songs.

For this purpose we must carefully analyze the family of birds that we are trying to identify and apply existing speech recognition techniques in order to model these techniques to the specific species we are working with. The goal is to adapt the knowledge we already have on spoken language recognition to create a bird song

language that will not only aid in acoustic recognition but that will also give us some insights on their communications and behavior

The birds selected for this work, are part of the family of birds known as the Antbirds. The Antbirds are a large family of the passerine bird species of the subtropical and tropical Central and South America. They are forest birds that feed on insects on the ground. Minorities of their species specialize in following columns of army ants to eat the small invertebrates that leave hiding to flee the ants. This is the origin of their family name.

The Antbirds are birds that only sing innate songs. They do not learn songs nor try to imitate songs from other bird species. Through the inspection of their song's spectrograms, we can observe that this type of birds produce repetitive songs which might suggest us to believe that these songs can be modeled as a regular language [28,29]. A language is called a *regular language* if some finite automation recognizes it [31] . A finite automata is defined as follows:

A ***finite automaton*** is a 5-tuple (Q,Σ,δ,q_0,F)

where:

- Q is a finite set called the ***states***.
- Σ is a finite set called the ***alphabet***.
- $\delta : Q x \Sigma \longrightarrow Q$ is the ***transition function***.
- $q_0 \in Q$ is the ***start state***.
- $F \subseteq Q$ is the ***set of accept states***.

It is our goal to extract from the bird's song, syllables which will lead us to create an alphabet in order to develop a regular language that can help us classify and interpret bird songs (see figure 4). Once constructed, we will use the finite state automaton to perform classification, since its computational requirements meet the limitations provided by our sensor network platform, considering that some of these tasks will be performed with streaming data. We also considered the use of a probabilistic approach such as Hidden Markov Models [25], but for the moment we will only use them as a comparison technique for our language, since they require a more robust and heavy power platform than the one we have for our project.

The general objectives of this work are as follow:

- To use the relevant information collected from previous work to reduce the computational cost required to analyze the data.
- To efficiently extract syllables from bird songs applying different feature extraction techniques.
- To develop semi-automatic syllable extraction software that will use a bird song database to gather species-specific parameters for configuration.
- Use existent techniques to view the natural groupings of these extracted syllables in order to interpret and understand their natural relationships.
- Build an alphabet based on the extracted syllables.

- Use the alphabet in order to generate strings that will model the bird songs.
- Using the constructed strings, to generate a regular language that will be able to identify not only the generated sets of strings but also to generalize its results and recognize other input strings from testing samples of those bird families.

- To develop and enhance the current syllable extraction software in order to use it as the basic foundation of an automatic classifier based on a lexical and grammatical analysis performed to the generated syllable data set.
- To generate a language that can model bird songs. This language will be used to interpret and classify birds by means of constructing string and performing a grammatical analysis of the modeled bird songs.

1.3.3 Social Impact

With the development of more efficient tools to classify bird species and understand their behavior, several different social applications can be addressed in future work. It will be very important to develop tools that not only can classify different bird species, but that they can also distinguish between different individuals from the same species. It will be part of our goals to advance in individual recognition so that our work will serve others as a foundation for bird individual recognition. Our syllable bird song classifier will not only classify between different species but also will have generated a set of strings that will construct a bird song language that can give us many insights of their behavior and population.

One possible application of this knowledge might be to monitor different individuals from the same species to see how they shift throughout their territories and how abiotic factors can influence their life. It could be possible to monitor them and see how they shift their territories due to the intervention of human kind and how important aspects such as climate and population shifts affects their songs. The work that will be developed for this project will not only aid engineers and biologists for species classification but will also aid the casual ornithologist and will provide governments with social insights to avian behavior that can ultimately help minimize the deforestation and human intervention that can cause some species to become extinct.

A very important benefit from our work, will be that the acoustic information gathered from the spectral analysis will be deployed in a species selection database that will be available, in a certain part, to the general public. This database will have sufficient acoustic data on our target species so that others will be able not only to replicate our work, but also to carry on and enhance our findings. Furthermore, the database will contain general information about our target species, so that it can be used in conjunction with other databases as educational material for students in high school that might be interested in ornithology.

1.4 Developmental Process

1.4.1 Syllable Analysis

In order to generate this language and to do a grammatical analysis, we must first understand the natural organization and grouping of these bird syllables. To achieve this purpose, we will use several computer algorithms and techniques in order to study and obtain the necessary knowledge from the bird songs. Such algorithms include data mining, clustering, self-organizing maps, and principal component analysis, among others. Once we understand the natural groupings of these extracted syllables, we will proceed to construct a small alphabet to perform a lexical analysis. Later on, we will develop a simple analyzer that will classify bird species based on a string representation of their songs, which were obtained through the syllable analysis.

When dealing with unknown or uncharted territory like it is the complex analysis of bird songs, we have to face ourselves with different questions. Experts can answer some of them, while others are still open questions to all of us. Some of the questions in hand are:

- Which is the best representation for the given data?
- Can we use symbols in order to characterize the basic elements in composing a bird's song?
- How can we know how many different syllables we can find in bird songs? Is this distinction per species or does it vary depending the different individuals or ages?
- Can we detect through the acoustical analysis of bird songs the relationship between parents and siblings?
- Where do we cut in a song in order to represent a single syllable?
- How complex (Chomsky's Hierarchy) is the syntactical structure of bird song compositionality?

Most of the answers to these questions are still unknown to us, but it is a motivation for this work to contribute to answer some of them. Some experts might say that the bird songs are less complex than the human language; therefore the number of syllables found should be less than those found in our language [40]. It will be shown later on this document that during the experiments performed up to date, we were able to find a range of the number of syllables from the recorded species. This could give us a clearer idea in order to answer some of these questions.

An important factor to consider, in a syllable-based analysis for species classification is regularity. Sometimes when analyzing bird songs we obtain parameters, which describe the fundamental characteristics of a bird from a selected species. These parameters might vary considerably with other members of the same species, especially if the techniques and tools used for their extraction are not constant and require manual modifications. A problem we encountered which was also observed in the studied literature, is that the feature extraction and the data mining

classification are very sensitive procedures that require manual adjustments even when dealing with different individuals from the same species.

When preprocessing the recorded data using feature extraction, we have to do manual adjustments on some of the important parameters of the software being used, in this case Sound Ruler [35] or Raven [26] because of the minor differences that might arise from individual to individual of the same species. This is a strong limitation if our main goal is to take this feature extraction and data mining to the field aiming to do it automatically in each sensor node. These minor differences cause some changes in the data mining stage, which will influence in the accuracy given by most classifiers.

On the other hand the syllable extraction analysis gives us promising results that can overtake the limitations introduced into this work by the feature extraction. Syllables can be seen as more elementary building blocks of bird vocalization [1] and may therefore be more suitable for automatic identification of bird species than song patterns [33,34]. Some of our initial tests show that once the key features of each species can be identified, the syllable extraction software can handle every type of song or call from different individuals without requiring manual adjustments or human intervention. This is a very important factor for this project, the automation of acoustic recognition in sensor nodes.

1.4.2 Language Generation Analysis

Other questions in hand rise from the different areas of research that were explored during the development of this project. While studying and performing field experiments with the Antbirds, we corroborated the fact that they have a very limited repertoire of songs, mostly consisting of repetitive sequences. Contrasting this information with the research performed in [20,28,29], songbirds have a more complex song system with a more varied repertoire. Songbirds are able to imitate, learn and mature their songs through time. They can be trained to acquire complex recursive grammars such as A^nB^n language. [11,20]. This can lead us to believe that a universal bird song language might not be possible, or might involve more work in species-specific analysis, before making general assumptions. Considering these facts, new questions arise which are shown below:

- Can Antbird songs be classified as monotonous repetitive chunks that can be represented by regular languages?
- Do Songbirds have a more varied repertoire than Antbirds making their songs more complex?
- Can Songbirds learn to evolve their songs into more complex languages?
- Will regular languages be able to model all bird songs or just Antbirds and similar species?
- Do we need to consider bird songs as complex as the human language, therefore justifying to use context-free languages [31] to model bird songs?

1.4.3 Organizational Block Diagram

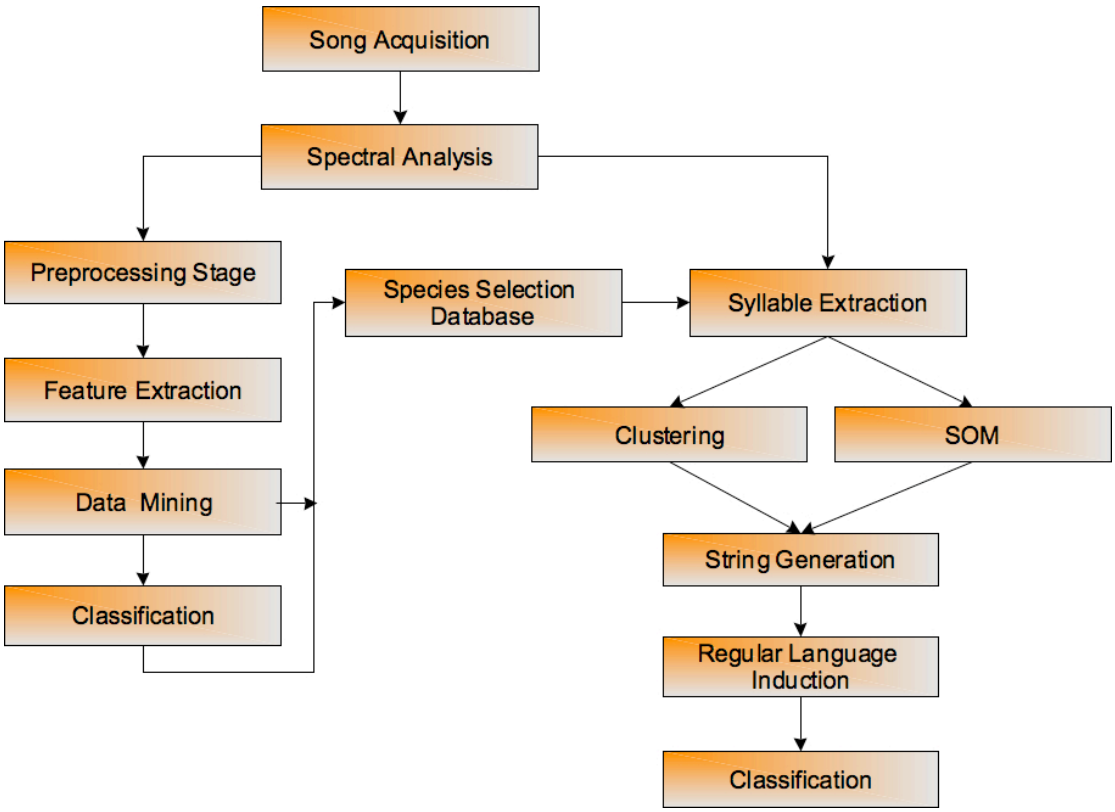


Figure 4: Project's Block Diagram

2 Background

2.1 Introduction

The main motivation for this work is to incorporate the advances and knowledge of computer science into different areas in order to achieve the project's main goal, to correctly and automatically classify different bird species through their songs. This project is part of an international collaboration between the UCLA and ITESM CEM and will join many different areas. Some of the areas that this project covers are, computer science, electrical engineering, digital signal processing, ornithology, linguistics and biology. It is important to keep in mind that this work will mainly focus in the area of computer science but will also deal with electronics, biology and linguistics.

A very important goal to keep in mind is to be able to research from each of these individual areas and to incorporate the knowledge into the work being performed from the computer scientist's perspective. In order to develop an efficient classifier based on bird songs and syllable detections, we must incorporate as much knowledge as possible from these different areas. This section contains the background information necessary in order to fully understand what is being developed in this project and to gain the necessary knowledge to comprehend the projects goals and the preliminary results that are being developed.

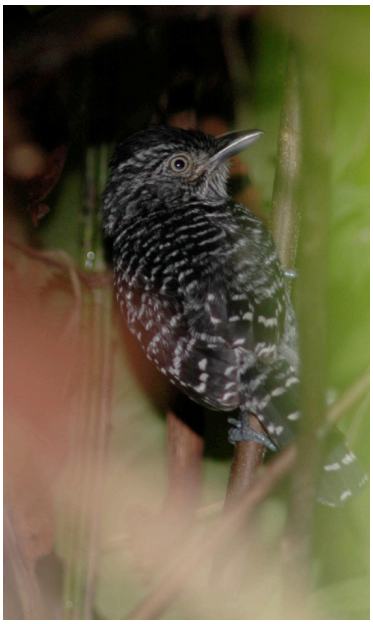
2.2 Bird Songs

Bird songs and calls for this work were obtained from two different sources. The first of the sources was through the Cornell Lab of Ornithology, Macaulay Library [7]. We gathered samples from their collection and did the initial testing for our preprocessing acoustic analysis phase. The second set of bird songs were collected in the field during our February 2006 trip to the "Reserva de la Biosfera de Montes Azules", in Chiapas Mexico. We gathered samples from the species to study through digital recorders powered by high capacity microphones. We used these songs to validate some of our previous results, and to further enhance the acoustic database built for our experiments.

We focused on songs from three species of Antbirds: *Great Antshrike*, Taraba Major (49 song files); *Dusky Antbird*, Cercomacra Tyrannina (79 song files); and *Barred Antshrike*, Thamnophilus Doliatus (76 song files) as seen in figure 5. Each song file has from a few seconds to several minutes of bird calls, with either one, two or more birds singing on it.

Sub Fig. 1 *Taraba major* (Great Antshrike).

Sub Fig. 2 *Cercomacra tyrannina* (DuskyAntbird).



Sub Fig. 3 *Thamnophilus doliatus* (Barred Antshrike).

Figure 5: Antbirds used for this work [40]

The reason to choose these species is because they are part of the Antbirds that are abundant in Montes Azules, Chiapas, the ecological reserve where the sensor network will be deployed in the near future and that these tropical bird species do not learn songs, which makes the job of acoustic recognition easier, since they only sing innate songs.

The easiest way to study bird songs is by visually analyzing them. It might be contradictory to say that a visual aid will help to analyze a sound. It has been proven both by amateurs and experts in the field, that in order to start analyzing, to comprehend and recognize different birds by their sounds, the first stage is to analyze their sounds visually. Seeing bird sounds as we hear them greatly helps us appreciate the details in the sounds and the differences among them [17,30]. For this, experts use sonograms. Sonograms are formally called spectrograms. A spectrogram is the result of calculating the frequency spectrum of windowed frames from a compound signal

in order to view changes of frequency over time. It is shown as a three-dimensional plot of the energy signal as it changes over time. Spectrograms are usually used to analyze phonetic sounds, which in our case is fundamental for species recognition.

Bird vocalizations can be divided into songs and calls. The distinction is both traditional and arbitrary, but as these terms are still retained in the literature some clarification must be made [6]. Bird songs are normally more musical and complex than calls. Males usually produce them and they are associated with breeding. Calls tend to be shorter, simpler and produced by both sexes throughout the year. Unlike songs, calls are less spontaneous and usually occur in particular contexts [6]. Birds use calls to communicate things to each other and between members of a flock or family.

Birds can have a dozen or more distinct calls, which they may use in specific ecological circumstances. For example, alarm calls signal danger, contact calls locate other individuals within a territory and flight calls are used to keep the flock together. As with other forms of communication, specific bird songs can be intended for more than one audience. For example a territorial song might send the message of stay away to other species and at the same time it might be attracting potential mates.

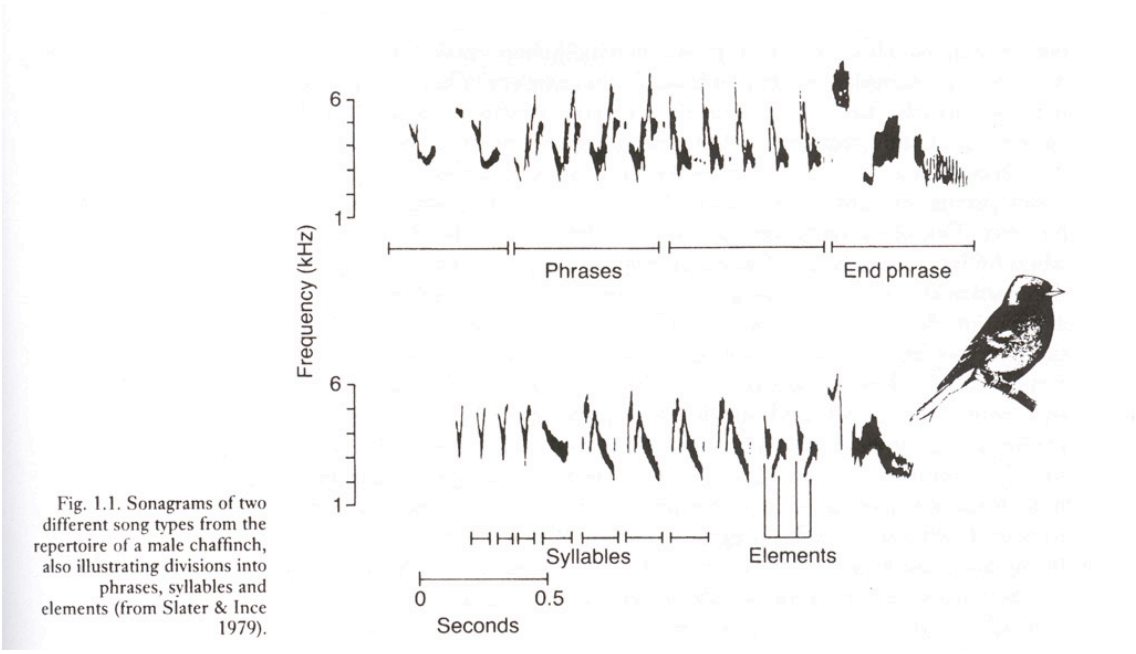


Fig. 1.1. Sonagrams of two different song types from the repertoire of a male chaffinch, also illustrating divisions into phrases, syllables and elements (from Slater & Ince 1979).

Figure 6: Elements of a bird's song [6]

Most birds have more than one single version of their species song. Each different version is called a *song type* and each male is considered to have a *repertoire* of song types. If we take our analysis a step further, we can observe that each song consists of different sections. These sections are called *phrases* and each phrase is also subdivided into a series of simpler units arranged together into a particular pattern, see figure 6. Sometimes the units are all different in a single phrase, other times they repeat themselves. These units are usually referred as *syllables*. Syllables can be very simple or quite complex in their structure. When complex, they are constructed from several of the smallest building blocks of all, called *elements or notes*. An element is a continuous line in a sonogram [6].

These different elements are the ones we have to consider in order to perform an analysis of bird songs. In particular different approaches have been taken working with phrases and songs. Some of the disadvantages found from these approaches are that both songs and phrases rely heavily on the quality of the recordings and on the fact that there has to be only one single bird singing at the same time. If more than one bird sings for a sample of song, these types of classifiers suffer greatly due to the lack of techniques to identify at a phrase or song level the different individuals singing on it. For our purposes we will rely on the smaller building blocks of these songs, which are the syllables. We will concentrate our efforts on building efficient classifiers with them, since syllables are easier to recognize even with different birds singing at the same time, as it will be shown later on this document.

2.3 Signal Processing

2.3.1 Introduction

In order to perform an efficient preprocessing analysis of the recorded bird songs, we must do a careful analysis of the components of these songs in order to determine which of them will be useful for the preprocessing feature extraction and data mining stage. During the data mining stage the relevant information will be separated from the rest of the signal information in order to feed the more complex syllable extraction software. It is very important to know the key elements of these sound waves in order to fully understand the process and to give the syllable extraction software exact parameters in order to segment the songs and calls into the appropriate syllable elements. This is a critical stage for the development of an efficient classifier since the actual results can heavily rely on the separation of the sound segments into syllables and careful research must be done in this stage in order to guarantee satisfactory results.

2.3.2 Sound Analysis

Sound is a longitudinal pressure wave, formed from compressions and rarefactions of air molecules, in a direction parallel to that of the application of energy. [Huang Spoken language processing] Compressions are zones were air molecules are packed tight together due to the application of energy. Rarefactions are zones were the air molecules are not so tightly packed. Two important parameters, amplitude and wavelength are normally used to describe the sine waves that form through rarefactions and compressions, sound. These parameters are often used in different research areas to model bird songs. Amplitude is a nonnegative scalar measure of a wave's magnitude of oscillation, that is the magnitude of the maximum disturbance in the medium during one wave cycle [51]. The wavelength is the distance between the repeating units of any wave pattern. It is the distance that a sound wave travels during a period.

When birds sing, they apply different levels of pressure to produce their songs and calls. Some variations on their species defined songs can occur depending on the level of sound pressure they emit. It is important to analyze and comprehend these small changes since they can alter defined characteristics of the sound signal.

Sound pressure is the pressure variation from the local ambient pressure caused by the movement of sound waves. The pressure variation may be either positive or negative. A sound source generates pressure variations that travel away from the source in all directions. Such traveling pressure variations are called *sound waves*. A sound wave typically travels at speeds of 340m/s. The sound pressure level (SPL) is a measure of the absolute sound pressure P in decibels (db) as given by the formula:

$$SPL(db) = 20\log_{10}\left(\frac{P}{P_0}\right) \quad (1)$$

where the reference of 0 db corresponds to the threshold of hearing , which is $P_0 = 0.0002\mu bar$ for a tone of 1kHz.

Sound data is defined as the recorded songs to be used transformed to digital signals. The first step in making a digital representation of a signal is sampling. For speech signals the most important quantity of interest is air or sound pressure. We do not measure air pressure; we convert it into a changing voltage and then let electronic equipment do the measuring. The sampling theorem is defined as follows:

If *f* is the frequency of a signal, the sampling rate must be at least *2f*.

For this work most of the sound files are already recorded in digital format using professional Sennheiser microphones and a Marantz PMD-670 solid-state recorder, which stores the audio directly in wave files. A digital recording is a sequence of sound pressure data represented in digital format. Two parameters will determine the quality of the recorded data; they are the *sampling rate* and the *sampling frequency*. The sampling rate determines how many samples of sound data are collected per second Hz. An example of sampling rate is *CD quality recordings*, which are normally performed in 44100 Hz. The accuracy of the digital representation of each sound pressure data is measured in bits. For example 16 bits means that each sample of sound is represented by one of 2^{16} = 65536 possible values.

In order to perform an efficient feature extraction, we must also define several spectral parameters, which are going to be used and extracted through the use of the software Sound Ruler [35]. In order to understand how those parameters are generated we must comprehend some basic mathematical concepts. They are listed in Appendix 1 of this document.

2.3.3 Filtering

Field recordings can be extremely noisy, especially when taken from tropical rainforests. In these types of forests, the vegetation is densely packed causing sound reverberations; there are many different bird species interacting and a huge amount of other animals producing harsh noises. The climate is also a crucial factor, rain can cause significant interference and wind causes leaves to fall and interfere through most of the acoustic frequencies. All of these factors limit the quality of the sound recordings making the automatic bird species recognition a more complicated process and requiring the introduction of different filtering techniques in order to obtain suitable results.

Filtering is a process through which some frequencies of an audio signal are removed in order to prevent them from interfering with the actual signal that is being studied. Filtering can be performed at either hardware or software level. Filtering is used to pass certain frequency components in a signal through the desired system without distortion and to block the non-desired frequency components. The system that

implements this operation is called a *filter*. The filter allows a certain range of frequencies to pass, called the *passband.* The range of frequencies that the filter stops from passing is called the *stopband*. Various types of filters can be defined and constructed depending on the nature of the filtering operation.

In most cases the filtering operations for analog signals are linear, and they are described by the convolution integral:

$$y(t) = \int_{-\infty}^{\infty} h(t-\tau)x(\tau)d\tau \qquad (2)$$

where *x(t)* is the input signal and *y(t)* is the output filter characterized by an impulse response *h(t)*. Lowpass filters allow all low-frequency components below a certain specified frequency f_c called the cutoff frequency to pass and they block all high frequency components. On the other hand, highpass filters allow all high frequency components above a certain cutoff frequency fc to pass and block all low frequency components. Bandpass filters allow to pass all frequency components that are between two margins usually represented by two cutoff frequencies.

All of our field recordings were done without the use of hardware filters. The recordings obtained from the Macaulay Library [7] were also unfiltered at the hardware level.

3 Previous Work

3.1 Introduction

Several authors have analyzed bird songs in the past, ranging from typical theories in sound recognition such as applying HMM's as it is done for speech recognition, to taking different approaches considering the high temporal variability of the sounds produced by different bird species. In this work we will analyze the previous background work or state of the art, which significantly contributes to this proposal and also contrasting it with other significant approaches taken such as the classical application of HMM's, statistical analysis, neural networks and the application of data mining techniques to do an efficient first stage classification of species.

3.2 Syllable processing

The set of background work, which is directly related to this proposal, is based on a first research paper published by professor Aki Harma from the Helsinki University of Technology and part of the Avesound project [2]. In [13], syllables are considered to be the elementary building blocks of bird songs. Several others have considered syllables to be very important in the detection of bird species [1]. The work presented in [13] is an important first step for the processing of bird songs and their analysis is based on a sinusoidal representation of bird song syllables. An alternative method for recognizing bird songs is to consider song melodies instead of using a syllable analysis approach, but some species produce high regional variability songs, which limits this approach.

The temporal resolution of bird hearing is extremely high as well as their temporal variability in the song production, which leads to the need of a high temporal resolution in the range of a few milliseconds to do the analysis of bird songs. The typical duration of a syllable ranges from a few milliseconds to a couple hundred milliseconds. The spectrum energy in bird songs is typically concentrated on a very narrow area in the range of 1 to 6kHz and the sound is often composed of a single or small number of sinusoidal components [13]. This work [13] therefore proposed to use the simple method of sinusoidal modeling in order to represent bird songs and extract syllables to perform the actual classification of species. To obtain a sinusoidal model from a signal we use the following formula:

$$s(t) = \sum_{r=1}^{R} A_r(t)\cos[\theta_r(t)] + e(t) \tag{3}$$

$A_r(t), \theta_r(t)$: is the instantaneous amplitude and phase of the r^{th} sinusoid.
e(t): is the residual component.

The idea of sinusoidal modeling is to use sinusoids with time-varying frequencies and amplitudes to represent harmonic signals. To obtain these sinusoids from an original signal, their short-time spectra is analyzed from the original signal by taking the discrete Fourier transform of the windowed signal to locate the prominent peaks in the amplitude spectra [47]. A peak is defined as a local maximum in the magnitude spectrum.

This work [13] proposed to do the sinusoidal modeling analysis since it considered easier to recognize individual syllables from a crowded sample of bird songs than to recognize individual songs from these samples considering that more than one bird was singing at the same time. The general idea was to decompose a bird's song into a set of frequency and amplitude-modulated pulses. Each pulse would represent one individual syllable. Although this is a practical technique, it is an over simplified model to represent the complexity of a bird's song. It might work efficiently if the number of species to distinguish is limited and they are of the same family or similar families.

When different families are involved, there is a high risk of misclassification, especially when one family has a complex vocabulary. Also the degree of correct classifications is low, usually around 30%. Another problem found with this approach is that the analysis of sinusoidal modeling is reasonable only for stable periodic signals because the number of required sinusoids would be the same as the number of partials in the sound, which for our case is a simplification of the problem since it assumes that all bird songs are stable periodic signals.

A proposed solution to these problems is to include in the modeling analysis more information from the signal, such as those produced by the first three harmonics. Other solutions might be viable; such as to include statistical information or to do a previous feature extraction for each family to observe which representation of the signal might work best for each family, sinusoidal, parametric or statistical.

From this paper we can observe that a mayor drawback and problem is the actual determination of the syllables. It is very important to define how to segment a song and determine exact matching syllables since they are the building block for the rest of the research. If you cut or segment in one section of song you would get a set of syllables while if you do it in another it will vary. Some questions that arise from this research are:

- How can we determine the right place to segment a song into syllables?
- Can we measure efficiently these proposed cut sections?
- Is there any way to obtain previous knowledge of the song or call in order to do a proper syllable segmentation of it?

This first paper from the Acoustic Laboratories in Finland motivated further research. In [33], they propose a variation of the method used for classification. This second work still uses sinusoidal modeling as the basis for their syllable segmentation but proposes a more complex model for classification. They propose to represent variable-length syllables as a fix dimensional feature vector. For this purposes they are going to base their classification techniques on syllable pair histograms and perform a nearest neighbor classification. A histogram is a graphical display of

tabulated frequencies. It is the graphical display of a table, which shows what proportion of cases fall into each of several or many specified categories [51].

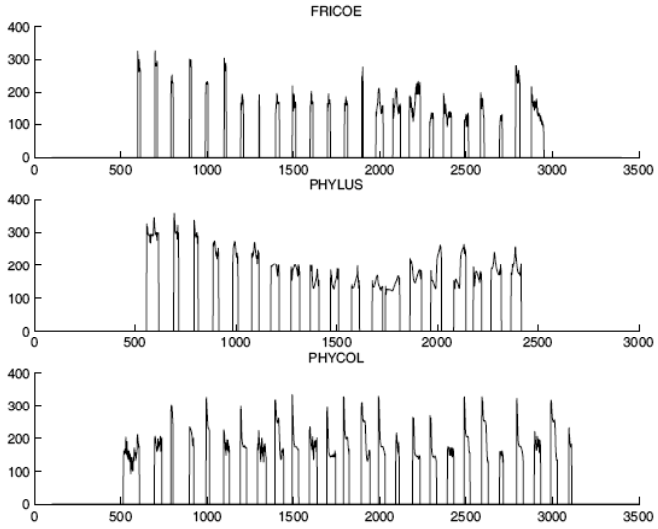


Figure 7: Examples of bird songs after sinusoidal modeling [33]

The aim of the authors was to show in a more graphical display the organization of the syllables extracted from the birds songs in order to perform a more efficient classification. Many bird species have a high individual and regional variability in their phrases and song patterns. This can cause several problems when trying to identify between different species. A histogram collected from single syllables might not help distinguish between these temporal changes in the structure of the songs as shown in figure 7. To try to correct this, they decided to use syllable pairs since they will reveal a more temporal structure of the song than by doing the analysis on single syllables.

The representation of a bird's song for this work was based on the study of syllable histograms. In order to form the histograms, the syllable space was divided into bins [33]. Obtaining a set of syllable prototypes and finding dissimilarity measures amongst them did this. For this purpose, the dissimilarity between the syllables was obtained using the Dynamic Time Warping Algorithm.

Dynamic time warping (DTW) is an algorithm for measuring the similarity between any two sequences that vary in either time or speed. DTW pattern matching may be efficiently used for speech recognition applications where we have a small vocabulary stored into prototype enunciations. Since this algorithm performs efficient comparisons for speech recognition, many authors chose it as a useful tool for bird song recognition.

Histograms were constructed from consecutive syllables (pairs) since some of the songs obtained for this analysis have a small number of syllables and the authors considered important to do a proper smoothing of the samples. For this reason they used Gaussian prototypes instead of a simple vector quantization and they chose an arbitrary number of consecutive syllables, which for this study was $N=2$ (pairs).

The results they obtained were much more significant than their previous approach, getting classification efficiencies from 76 to 80%. Comparing these results with the ones obtained from their previous work, we can see that there are great advances and that further research is needed in order to efficiently organize the extracted syllables.

A disadvantage found from this article is that they did not consider the silence intervals between syllables in order to determine syllable boundaries. They based their syllable recognition on their previous approach of sinusoidal modeling. This can hinder performance since it was found that this type of modeling might be too simple in order to perform accurate representations of bird songs.

Since one of the mayor drawbacks of these two articles is that they model bird songs using single time varying sinusoids, they decided to go one step further and to include signal information from the first few harmonics that were generated in each sampling period. In [14] the authors proposed to analyze a small database of syllables extracted from bird songs in order to find an alternate method to extract bird syllables.

They propose to model syllables using parametric line spectrum estimation, which is referred as Analysis-By-Synthesis/Overlap-Add [12]. They estimate that this technique, which now includes harmonic information of the signal, will improve their classification results. In this new document, the authors propose that there are four different classes of syllables found in bird songs. The first class includes almost pure sinusoidal information; class two includes mostly sinusoidal information plus data from the first harmonic. Class three includes basically most of the information of the signal from the first harmonic, while class four is purely concentrated by syllables whose information is produced with the second harmonic.

From their experiments they conclude that most of the signal information from a recorded bird song lies between the fundamental frequency and the second harmonic of the signal. From their tests they found out that around 60% of the extracted syllables are purely sinusoidal and that their first approach is a good step for a first stage bird song classifier.

On the other hand, 60% is not a reliable approximation and we must consider the rest of the information that is being produced in the bird's song signal, therefore harmonic structures must be used. Their experiments also show that the second class to produce enough information for classification is class four, or the pure second harmonic of the song. This second harmonic information can actually model or describe around 14% of their extracted syllables. This corroborates the work done in the preprocessing stage and reported in [9,45,46] which mentions that some of the most important features to distinguish birds are the fundamental frequency and the second harmonic of the signal.

It is important to note that their tests did not reflect exact procedures and the syllable database they used was not large enough to generalize their findings. Although the findings from this research helped confirm our own findings, their classification results were not included in the document. It would have been useful to observe individual tests and see how the inclusion of harmonic information in the syllables helped to distinguish between different species. The only clue the authors give is that

the improvement was around 5 to 20 percent in classification. This is a mayor drawback since we cannot replicate the experiments nor compare our results.

A fourth document in the series of work from the Avesound [2] project in Finland produced very interesting results. In [10] the same researchers of the previous three articles investigated the effects of not only modeling bird sounds as simple sinusoidal waves, but also as inharmonic sounds. They considered that some bird species might not have such colorful songs as the ones they had previously analyzed, the songbirds. Such is the case of the species of our work, which don't learn songs and don't have such complex vocabularies as songbirds and we can relate our own experiments to most of the ones performed in this article as background work. For these species in hand, they propose to consider their songs as inharmonic sounds. They consider inharmonic sounds as those that feature irregular pitch patterns. They also consider that even though songbirds produce mostly harmonic sounds (sinusoidal), sometimes they can also produce sounds which have a complex spectrum and temporal envelope, which makes them fall into the category of inharmonic sounds.

In this article, they mention that if the likelihood of a song to belong to a pure sinusoidal class is less than 60%, the syllables are labeled to be of inharmonic nature. This is clearly a source of future classification problems, since some harmonic sounds will clearly be labeled and treated as inharmonic ones, but for the practicality of their experiments they decided to use it as a general rule. In this research article, they introduce a method to measure feature importance for classification and they try to find out individual species-specific features sets in order to build a more robust classifier. We can directly relate this approach to our own, since most of our preprocessing stage experiments have the same goal, to find species-specific feature sets in order to construct an automatic bird recognition system, based on syllable extraction software for classification.

The importance of distinguishing these species-specific feature data sets is invaluable. If we can know which features to look for a certain species, we can narrow down the feature extraction stage and directly implement it on a sensor network, reducing computational costs as well as energy resources. Unfortunately, they do not present these feature-sets neither in their results section nor in their conclusions or future work, since it would be interesting to compare their species-specific feature-sets to the ones we have obtained in our preprocessing stage.

In order to maintain a scientific objectivity, this article presents results of species that are considered to produce inharmonic sounds as well as some songbirds to contrast their results. They also divide their system into three different components:

- The division of recorded bird songs into syllables using the algorithm vaguely presented in [14].
- A set of parametric representation is computed from each syllable to obtain feature vectors.
- Classification is performed and contrasted using two different methods to represent sounds to their *k* Nearest-Neighbor classifier.

The authors decided to compare two different ways of representing a sound signal to their classifier. Their two different methods were, using low-level descriptive

parameters and to extract the Mel-frequency cepstral coefficients (MFCC) of the syllables in order to feed those to the classifier. For the first method, the authors used the FFT in order to obtain spectral information from the signal. It is unknown to us why the decided to focus only on 7 parameters from the signal and then use the mean and variance of these 7 parameters as actual information values fed into the classifier. It is obvious that they made some sort of discrimination since performing the fast Fourier transform of a sound signal gives you roughly around 70 different parameters that can describe a sound from a bird's song. The idea of calculating the mean and variance of these 7 selected parameters was interesting since they decided to use the average values from each syllable, not only using the initial and final values.

For their second method, the authors decided to use the cepstral coefficients of the signal because these coefficients are a very convenient way to model spectral energy distribution from a signal. They decided to use twelve cepstrum coefficients to model the precise spectrum of the signal. Using this type of analysis, the more coefficients you use the most accurate precision of the signal you can acquire. A drawback from using purely cepstral coefficients is the linear frequency scale. Perceptually, the frequency ranges of 100-200 Hz and 10kHz to 20kHz should be approximately equally important. The standard cepstral coefficients do not take this into account. It would be better to use a logarithmic frequency scale to mimic perception. For this matter the authors decided to use the Mel scale to compute Mel-frequency cepstral coefficients whose complete process is shown in figure 8. The Mel scale is a perceptual scale of pitches judged by listeners to be equal in distance from one another. Using this, small changes in the feature vector will represent small perceptual changes and vice versa, making more accurate the vector representation of the signal.

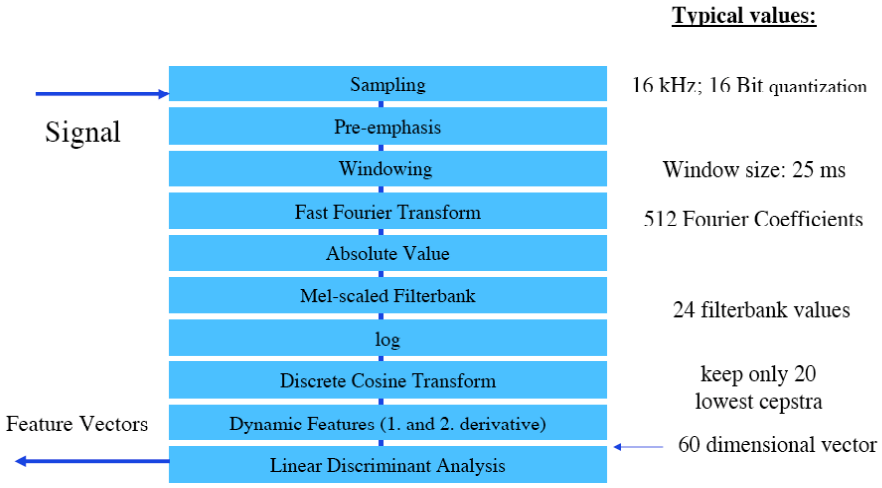


Figure 8: Complete Pipeline for MFCC [http://www.lsv.uni-saarland.de/]

Contrasting this preprocessing stage with the one performed in our own work, we can observe that our aproach does consider all of the spectral parameters from the signal and to reduce dimensionality we used data mining in order to obtain the most significant parameters used for classification from the complete data set of parameters. In this paper, they used a similar method to obtain the most important parameters but only for their use of MFCC's. This is a disadvantage since they don't mention at all how did they select these first seven parameters they used for their low-level descriptive parameter analysis, which will definitely affect their classification results.

The authors reduced their MFCC feature sets by using linear discriminant analysis (LDA). This method is used in machine learning to find the linear combination of features which best separate two or more clusters of objects. It is a method used to decrease the size of a vector and to maximize the severability of class regions.

The results shown in this document are very important since they demonstrate that the use of MFCC and LDA can help in the significant reduction of their feature set vectors without loosing too much classification efficiency. They state that their classification results using these methods obtain approximately a 74% efficiency using the complete data set or a reduction of the data set to 6 features. This is very important to our line of work since we have been working with a reduced feature set of vectors with 15 features. If we can apply this second method of presenting data to a classifier without loosing our classification efficiency it would really advance our own work in terms of real time analysis and field tests. On the other hand the pure signal parameter analysis they performed produced poorer results ranging from a 49% up to a 71% of accuracy in classification. The authors also failed to give theories on why these results were so low, but it can be observed that their results were low due to their seven initial parameter selection that was not mentioned in the document.

As a final remark, this document contains valuable information that can corroborate some of the results of our work, and interesting approaches when using k-nearest neighbor classifiers to do the actual classification of the species in hand. It is important to note that it would have been much more efficient if we could directly compare their attribute selection process with ours but the document doesn't give many hints on how they actually selected them.

A final document from this research group focuses more on the low level visualization of the data space than on the actual classification of the different species [34]. Although classification is important for them, it is more important on this document to find natural groupings of the syllables and to try to interpret their visual arrangements. They used the Self Organizing Map to represent a low-level visualization space for the data. Their main emphasis is not recognition but the organization and representation of the syllables they collected. Interesting results from their document suggest that a syllable-based approach to classification might prove useful if we can manage to properly interpret the organization of their components. Unfortunately on this document they still used an oversimplified model to represent the bird song syllables as seen in [13].

Another key aspect to mention is that actual bird song syllable extraction is omitted. We consider this to be a decisive step in the extraction of valuable data, since if there

are inconsistencies at this low level, it is most probable that the results obtained from high-level language classifications will be inaccurate. Unfortunately most of the research presented in this series of documents don't specify how the extract and build their syllable database.

An interesting approach they consider is how to compare actual extracted syllables and they apply two different methods. The first is to use Euclidean distances to compare the dissimilarities of two different extracted syllables. The second method they present is to use *Dynamic Time Warping* to achieve the same goal. Their results show that *Dynamic Time Warping* is clearly better than Euclidean distance, and that if they introduce slope constraints into the DTW algorithm, they can actually achieve higher results. It is shown again in this document that DTW is the algorithm to choose when you want to compare dissimilarities between syllables, and it is an algorithm we are going to explore in future work.

By using two different approaches to represent Self Organizing Maps (SOM), the authors managed to show us how data can be presented to these modeling tools. The first approach they took was to submit a fix length vector that was represented by the eigenvalue decomposition of their parameters. They obtained from a 1000 x 1000 syllable matrix 1000 samples of 7 parameters that could represent their data. We must note that it would have been interesting to compare those 7 parameters with our own, but there was no mention of them in the document. These 1000 x 7 vectors were used to train the SOM. The second approach was based on online learning of variable length sequence prototypes. Rows from the distance matrix were used ad feature vectors for the SOM. In both cases, the SOM was able to help in the identification of syllables from both different species as well as for individuals. It is important to note that by using the Self Organizing Maps to represent the syllable data, they were able to identify that different individuals produced some of their collected syllables. This type of advancement can clearly help us to achieve one of our secondary goals, the identification of individuals.

As a conclusion for this document and this series of syllable extraction documents, it is worth mentioning that it's a fairly new approach that has been barely tested by different research groups. This area of research is definitely worth looking into, since it proposes a new method to classify bird songs that can effectively eliminate the temporability of data that is used by some of the previous lines of research. Our current work is based on these approaches and will we will consider the excellent findings of our colleagues as well as their limitation points in order to construct a more robust and effective syllable based classifier.

3.3 Traditional Approaches

The traditional approaches were selected because they involve background work that can be partially directed to our investigation. This background work can be directly related to some of the key factor of our investigation while totally omitting others. The first set of these traditional approaches, treat bird songs as more complex units, only distinguishing between calls, songs and phrases, totally omitting the more simple constructing elements of these songs, and clearly omitting how they actually gather their samples if they ever use them to test their experiments. The second set of them, apply different recognition techniques that have been useful human language recognition, into the complex process of bird song recognition. They clearly omit the basic building blocks of bird songs and are not interested in building structures that can model these songs.

We will first analyze in this section a set of papers that consider the fact that bird songs can not only describe a language, but that their language can actually be represented by grammars and show how selection and evolution has helped birds to actually develop more complex grammars from their languages.

One of the classic documents for this area is the one from Kazuo Okanoya [24]. In this work they study how indirect sexual selection might influence in the song production of Bengaleese finches when compared to their direct ancestors, the white rumped munia. They found that the domesticated version of the species, the Bengalese Finch, had a high degree of complex song production that resemble finite state syntaxes, while the wild ancestor, the white-rumped munia sang very stereotyped linear songs.

This document foretells that birdsong has been regarded as a biological model for the human language. This is a very important justification for the study of birdsong and a motivation for our work. They also show that there is a high correspondence in both species, birds and humans, when regarding to the developmental processes. Another important factor that they share is the syntactical organization. Human language is a hierarchically organized syntactical behavior. Phonemes are formed into a word, words into a sentence and sentences into speech [24].

A very important matter to notice from this work is that bird languages can be more complex than expected due to the evolution of their species. Some species might represent linear songs while others might actually represent more complex languages. It is important to note that sexual selection and pressure are an important factor in the developmental process of bird songs. It is also important to note that there are some similarities between bird song evolution and the human language evolution, of course, without elevating both to a syntactical level, which in the case of bird's songs, it would not exist.

Also it is very important to mention that the basic building blocks of bird songs remain the same in either the white munia songs as well as in the Bengalese Finch's songs. This means that even though the song complexity of one is more elevated than

of the other, their basic building blocks are the same, which in our case means that the syllables are the appropriate elements to work with when attempting to construct a bird song language. If both the wild ancestors and the domesticated species share the same building blocks, we might be able to construct more descriptive strings from the syllable based representations of the birds songs in order to model their communications and to be able to correctly classify between different species.

The research in [28,29] proposes that a domesticated type of songbird, the *Bengalese finch (Lonchura striata)*, starts its evolutionary process with a grammar that can be modeled by a finite state automata, and that through sexual selection it can actually evolve its song into a more complex form. They specifically propose that males start singing with simple regular grammars and as mating selection begins, they evolve into more complex grammars in order to attract the females. An important fact to note since the beginning of their document, is that this paper totally omits the fact that bird songs are actually constructed from simple elements such as notes or syllables, and treats bird songs as either phrases, calls or songs using models to represents these and to tests their hypothesis.

A problem with this assumption is that they never consider that songs and phrases vary considerably from individual to individual and also because of their habitat and location. Also, songs and phrases are also harder to use as recognition elements since the quality of the recording has to be much higher than for simpler elements and the recording periods have to be stable. The answer to this omission by the authors is simple; they do not use actual bird songs to do their experiments. They base their work on previous material from [24] and model bird songs by constructing finite state automata representing bird and female songs. They suggest that females can actually discern grammatical features such as recursive arrangements of song elements in order to distinguish interesting songs from monotonous random ones. They assume that males can actually modify their singing abilities from generation to generation in order to increase their mating capabilities.

To test their assumptions the authors construct artificial birds as asymmetric finite state automata (FAs). One type of FAs is used only for song generation while the other is used for listening. Then they introduced the communication interaction between males and females in order to model the sexual selection process. The female interjects in synchrony with the male song, measuring how many interjections succeed according to her preferences before she evaluates her satisfaction with the song [28,29]. With this model, they demonstrate the co-evolution of male song grammars and female preferences.

Another very interesting application of bird songs looking to answer the open question of the language and its origins was taken into consideration very recently as a letter that appeared in [11,20]. In this work, the authors analyze the possibility of finding recursive syntactic pattern learning techniques applied to songbirds. The general questions that can be perceived from this work are:

- "Can other species besides the human species recognize syntactically well-formed strings, including those that use a recursive centre-embedded rule?"
- "Is recursive syntactic pattern processing unique to the human species?"

The findings from this particular work [11,20] showed that at least one non-human species, the European Startling (*Sturnus vulgaris*) can be trained to acquire complex recursive grammars such as the A^nB^n using their own songs and calls as an appropriate language. Motifs were selected from different bird song segments that were classified from the spectrogram analysis that was performed. A vocabulary of motifs was selected in particular two clearly identifiable patterns from their repertoire, nicknamed *rattle* and *warble*. Eight different rattles and eight warbles were selected without repetition in order two form two distinct grammars. Finite state grammars and context free grammars.

Half of the eleven Startlings used for this study learned to respond to sequences defined by the CFG grammar, positive stimuli, while withholding responses for the FSG. The other half took the opposite learning method. If the songs followed a certain pattern the birds were expected to react and if they did they were rewarded with food. Nine out of the eleven birds leaned to distinguish between FSG and CFG grammars even though several control tests were performed to rule out memorization, growth capacity and approximation probabilities. All of these nine birds efficiently recognized the patterns with over 90% accuracy.

To conclude this section, we will briefly mention below some other documents considered in this literature review and some of their contributions. The ones presented before, were considered to be of vital importance to this work. We are conscious that there are many others that can help us; therefore we will briefly mention some of them, which were also reviewed in this selection process.

In [1], DTW and Hidden Markov Models were used for automatic recognition of songs from Zebra Finches and Indigo Puntings. The authors decided to represent syllables as spectrograms and compared them in order to perform classification experiments and recognition. Their tests proved to be computational demanding and complex, since they constructed a template database that was compared to actual samples using spectrogram data. Some of their recordings contained acoustical information that was not necessary for their purposes but filtering was not included in their research. In [23], they noticed that signal features have different classification abilities in context of different species. They used canonical discriminant analysis to determine and select features that maximized the recognition results. They obtained 14 different acoustic features that were used for classification. This result can be compared to some of our current work, since we are using a reduced set of 15 acoustic features. One of their most important features was the frequency modulation rate that would contrast with the one we obtained, pulse dominant frequency.

3.4 Conclusions

The literature review helped us focus our efforts and gave us further motivation, since we were able to observe that different research groups needed similar tools and were aiming towards similar goals. We were able to observe how different areas from this project have been dealt with in the past and to learn from their findings. We were also able to contrast some of our work with the ones that were done by different research groups and notice that they were encountering similar results. Such is the case of the work group from the Avesound project in Finland [2]. From their findings we can observe that an accurate representation of the signal is needed and that harmonic information should not be discarded. We can also relate some of our current work with Self Organizing Maps to the one they presented in their work and contrast our strict feature extraction techniques with the more relaxed sinusoidal modeling that they were taking into account.

One of the key factors that will make our project stand out from what we've seen so far, is that we are focusing on the syllable extraction using previous classification and extraction techniques, which incorporates most of the hard breaking work done in bird song recognition. A key factor of this work is to develop automatic syllable extraction software that will incorporate data from a bird song database and automatically adjust its extraction parameters on a species basis. This will outstand from what we've seen so far because most software is dependant on manual adjustments. This will let us concentrate on more specific needs, such as an automatic bird song classifier that will be more robust than actual implementations. It is important to note that our goal with the syllable based classifier is to develop a language that can model bird songs, and by means of this language to not only classify different species but also to be able to represent some of their behavior with it.

Finally it was of great satisfaction to observe that most of the work performed in our project is new in the area of computer science and that key references are still to become available. It was motivating to observe and to be able to interact with some of the other authors and share experiences and information. Most of the research performed in this area is still on a developmental process, which further motivates us to further research and experimentation.

4 Methods: Preprocessing Stage

4.1 Introduction

An important part of the analysis to be performed requires working with data obtained from field recordings. Unfortunately the birds we are dealing with live in habitats where there is a lot of background noise. Filtering and sound cleaning are vital to obtain clean recordings but we need to know a bit more about the signal we are dealing with. For these purposes we use feature extraction. With feature extraction, we apply Fourier analysis in order to approximate these signals with a sum of sinusoids each at a different frequency. The more sinusoids included in the sum, the better the approximation.

For this work it is very important to obtain the syllable information from the bird songs as clearly as possible. It is a complicated process since there is no natural or defined place to make the cuts in the song. Some authors have decided to do this song segmentation based on a time scale or based on what they can observe in the spectrograms of the songs, therefore an important subject in hand is the processing of the raw data and the filtering of unwanted signals.

It is important to remove unwanted frequencies from the spectrum without actually loosing relevant data. Some of the unwanted features can be directly recognized as noise and be easily removed with software filters, while others are not and might not contribute useful information but still be present in the signal. In order to know which are the key elements of the acoustic signal we performed data mining of these extracted elements in order to obtain the most important acoustic features that can help species classification. Also by applying several data mining algorithms we are able to obtain a first level classification of these species that can give us a certain orientation or idea for future and more exact classifications.

The information we obtained from this stage (see figure 4) will help in the operation and design of a more complex syllable analyzer that will effectively extract syllables from bird songs. The idea behind the processing stage (figure 4) is to perform this only once in order to obtain the necessary information for the selected species to be able to design and adjust the syllable analyzer without having further needs of adjustment.

Once we have identified the vital elements for classification, we can incorporate the following techniques directly into the sensor network by means of the data reduction obtained from the data mining. This will increase the sensor network's capacity to interpret data and it will feed our next stage with appropriate results.

4.2 Sound Cleaning

Once we obtained all the recordings, we proceeded to do a frequency analysis of the songs. Most of the times it is easy to remove unwanted noise from the recordings using software filters, since they occur at higher or lower frequencies than the bird songs we are studying. In order to know at which frequencies do these three bird species sing, we process the audio files using the Adobe Audition software. We carefully analyze the spectrograms of the audio files and notice each single species visual representation, as it can be seen in figures 9 to 11.

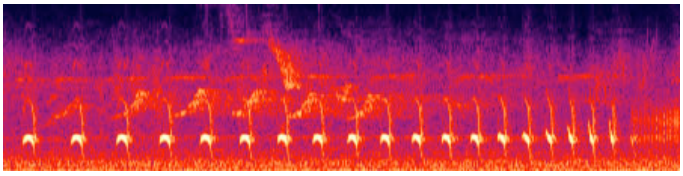


Figure 9: Great Antshrike Spectrogram [9]

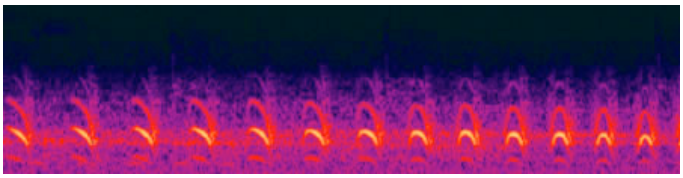


Figure 10: Barred Antshrike Spectrogram [9]

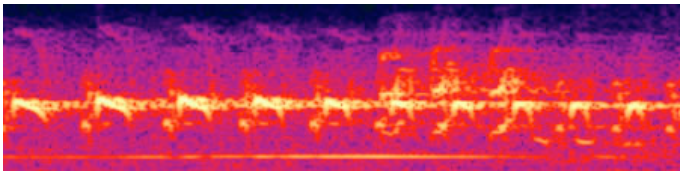


Figure 11: Dusky Antbird Spectrogram [9]

Using this information we can isolate their songs by using a combination of low and high pass software filters that can by applied with most audio editing software. When we finish the cleaning process we can actually measure different audio components of each species and record them in a small database so we can use them later on to perform an accurate feature and syllable extraction from the sound recordings.

Sound cleaning is a very important part for this work since most of the recordings we have contain different background noises that sometimes can affect even the most accurate classifiers. Sometimes we may have more than one bird species singing at the same time, for example when pigeons sing, they actually interfere the classifiers and have to be removed by means of filtering. Fortunately their sounds occur at different frequency levels and are easily cleaned out. One of the most annoying sounds you can find in any tropical rain forest recording, are the sounds produced by crickets. Their sound signals occur at high frequency which means that they can also be cleaned out by means of filtering.

4.3 Feature Extraction

Once we cleaned the songs and converted them to *.wav* format files, we loaded them into the computer software, Sound Ruler [35]. With this software, we are able to see the oscillogram and spectrogram of the signal and within the oscillogram we are able to locate each call from the recording and each pulse within a call.

Spectrograms are used to identify phonetic sounds and analyze the bird songs. They are the result of calculating the frequency spectrum of windowed frames from a compound signal, a three-dimensional plot of the energy of the frequency content of a signal as it changes over time.

It's important to remember that these songs were only preprocessed through low and high pass filters to facilitate an accurate call and pulse recognition. These filters are species dependant as we can see in Table 1.

	Taraba Major	**Cercomacra Tyrannina**	**Thamnophilus Doliatus**
Low-pass filter	3597 Hz	4200 Hz	3597 Hz
High-pass filter	517 Hz	920 Hz	686 Hz

Table 1: Low-pass and High-pass filters per species

The process performed with Sound Ruler [35] is done almost automatically although we need to manually indicate the beginning and end of the first call and we also have to indicate where the second call starts. After that, we must manually set the best settings to find each species specific calls (trial and error will lead to the right values). The settings that must be adjusted are: smoothing, resolution, amplitude peak +/- proportion, maximum silence between pulses in milliseconds, and time displayed around the call. Once this process is done, the automatic recognition stage begins by clicking on "sample" button. Even though by adjusting these parameters the recognition is quite accurate, it does commit some errors, which have to be corrected manually. It is important to mention that these settings have to be adjusted for each bird species and sometimes even for different individuals of the same species. A great advantage we have is that later on for the syllable analysis we already know the exact parameters for each species and the syllable analyzer can be fed with these parameters so no further adjustments have to be made.

Once every call is recognized correctly within a recording, we must analyze each pulse is also being recognized correctly. To do this, we must go to the call list and check the oscillogram displayed for each call by clicking it. In case there are one or more pulses missing in the recognition, we must add them manually.

The pulse-by-pulse analysis results were saved as comma delimited files. These files contain the 71 attributes of each pulse from the processed samples, representing the bird's song data. The resulting datasets' size is as follows: Taraba Major – 21,360 pulse samples, Cercomacra Tyrannina – 5373 pulse samples, and Thamnophilus Doliatus – 911 pulse samples.

4.4 Data Mining

When working with bird songs, we unfortunately have to deal with information that is represented as raw data. This information may contain valuable records that may be hidden from the naked eye. We have to apply different computational tools in order to extract the information we require from the raw data. The approach we took was to apply different data mining techniques in order to obtain the most relevant information from the raw data. This information will be used with the syllable detection software in order to calibrate each individual species parameters and to accelerate the processing of data, since all the hard work has been previously done out of the field by these data mining algorithms.

"Data mining is the extraction of implicit, previously unknown, and potentially useful information from the data"[50]. Once the important data is extracted, we can use only the significant information to feed our classifying algorithms in the sensor nodes in order to recognize different bird species based on their song and call production.

During the preprocessing stage of this work, several data mining algorithms were studied and some were considered and applied to the data obtained from the song samples. The algorithms selected were the decision tree based ID3 and J4.8, the probabilistic classifier Naïve-Bayes, vector quantization and association rules.

Decision tree based algorithms were chosen to reduce the dimensionality of the problem and eliminate data set redundancy. Naïve-Bayes was chosen because its ability to tell the percentage of accuracy of a classified instance and because of its affinity with non-redundant, independent data sets, such as the one produced after the reduction with decision tree algorithms' execution.

Vector quantization was chosen in order to convert our original numeric data set into nominal data, which is a requirement to run the ID3 and the association rules algorithms. By using this algorithm combination, we will be able to compare the full attribute data set classification with the reduced attribute data set classification in order to improve it while reducing the processing power required for use in sensor networks.

The classification improvement on the reduced attribute data set is caused by the attribute dependency elimination by means of the decision tree algorithm. Data mining provides us with the key elements from the raw data that will aid in the deployment on the sensor networks.

4.4.1 Vector Quantization

This algorithm was implemented because of the ID3's and association rules lack of numeric support. Quantization [27] is a process in which numeric to nominal data conversion is possible. The algorithm takes an original numeric vector and returns a quantized equivalent numeric vector, which can be easily represented by nominal values.

The quantization process calculates two intermediate vectors, partition and codebook. The partition vector is ordered and contains the minimum and maximum values from the original vector plus intermediate values calculated from adding the increase factor

to the minimum value of the vector up to the maximum value from the vector. Increase factor is calculated as follows:

$$Increase = \frac{\max(vector) - \min(vector)}{2^{bits-1} - 1} \quad (4)$$

The codebook vector is also ordered and includes values from zero to $2^{bits-1} - 1$ in increments of one. The partition's vector size is one element lesser than codebook's vector size. Finally we take each value from the original vector and check in which partition's vector interval it falls and map it with the corresponding codebook vector's value for that position. The easiest way to pass these quantized values to nominal values is to set a character equivalent for each codebook value so that you can map them directly. An example of this would be to have the next codebook for a 3 bit quantization: [0, 1, 2, 3, 4, 5, 6, 7] and map it directly with the following vector: ['0', '1', '2', '3' '4', '5', '6', '7']. As we can see, the "labels" contained in the last vector are equivalent to the values in the codebook vector. In this way, we obtain nominal representations from numeric values for any set of quantities making it possible to run the ID3 algorithm and association rules with them. A step-by-step example of the process of vector quantization is showed in figure 12.

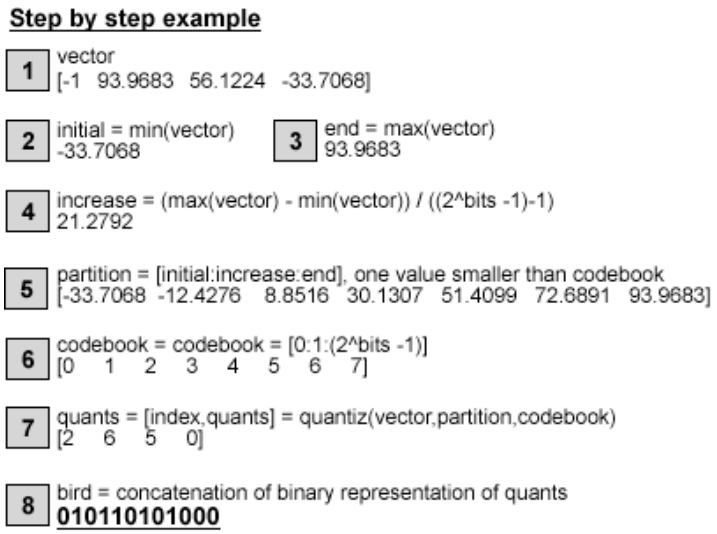


Figure 12: Quantization Example

In Figure 13 we present a plot comparison from a full original signal with values from 0 to 5000 approximately versus a quantized signal with values from 0 to 6. We can clearly appreciate how the relationship among the attribute values is preserved in the quantized set, even though we can appreciate some information loss.

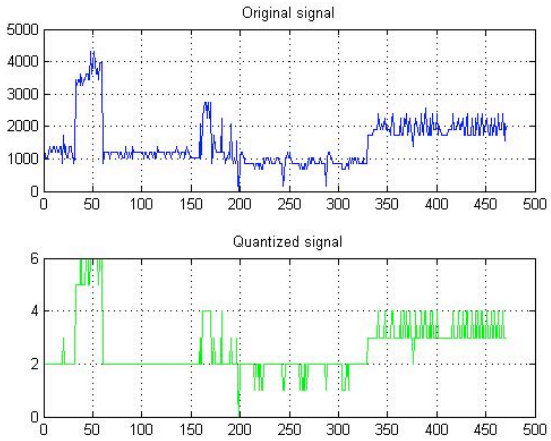


Figure 13: Original vs. quantized signal

4.4.2 ID3 Algorithm

Once we converted the entire species data sets into quantized data, we proceeded to process the information with a decision tree algorithm. Decision tree algorithms use full binary trees that represent the comparisons between elements that are performed by a particular sorting algorithm operating on an input of a given size [50]. The ID3 algorithm was used to generate the decision tree with Weka [48] software. ID3 is a decision tree algorithm that takes all unused attributes and counts their entropy concerning the test samples to be used. We define entropy as:

$$Entropy(p1, p2,...pn) = \sum_{i=1}^{n} - pi \log_2 pi \qquad (5)$$

where $pi = \frac{p_{inst}}{p} g$ and information gain is:

$$Gain(P, xP) = Entropy(P) - Entropy(x \mid P) \qquad (6)$$

The algorithm calculates the class's and attribute's entropy and performs a system gain. Then it compares the sample entropies and chooses the one with the maximum information gain or smallest entropy to be the next center node. When the tree is completed, the resulting nodes will be the most significant attributes used to classify the different instances or bird species (the leaves of the tree).

Once we obtained the corresponding decision tree, we only preserved in our data set the attributes that were used in the nodes of the tree (an attribute can be repeated in many nodes). This reduced data set will be used to attempt a reliable classification with the Naïve-Bayes algorithm.

4.4.3 J4.8 Algorithm

This algorithm is an extension of the ID3 algorithm, which solves some deficiencies that the original ID3 algorithm had. Some of the improvements are that J48 avoids over-fitting, uses a reduced-error pruning focus that is based on the consideration that each node of the tree is a prune candidate reducing this way the error, rule post-pruning to find high precision hypothesis and numeric attribute handling. The two main advantages that made us select this algorithm are the computational cost savings and the numeric attribute handling. Weka [48] was used to test this algorithm with our original data sets. The surviving attributes in the reduced data sets were also used to attempt a reliable classification with the Naïve-Bayes algorithm.

4.4.4 Naïve-Bayes Algorithm

We decided to introduce the Naïve-Bayes algorithm usage as a final classifier because of the main disadvantages that decision tree algorithms have. One if them is that they are unstable. Slight variations in the training data can result in different attribute selections at each choice point within the tree. The effect can be significant since attribute choices affect all descendent sub-trees. Another important disadvantage with decision trees is that trees created from numeric data sets can be quite complex since attribute splits for numeric data are binary.

Naïve-Bayes was executed in Weka [48], for the original, post-ID3 and post-J4.8 datasets. It is a statistical method based on Bayes rule that naively assumes independence. The Bayes rule says that if you have an hypothesis *H* and an evidence *E* then:

$$\Pr[H \mid E] = \frac{\Pr[E \mid H]\Pr[H]}{\Pr[E]} \tag{7}$$

Numeric values are handled by this algorithm assuming they have a “normal” or Gaussian probability distribution:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{8}$$

The mean and standard deviation are calculated for each class and each numeric attribute.

We know that it is only valid to multiply probabilities when the events are independent. The assumption that attributes are independent in real life certainly is a simplistic one [49]. In this work, we attempt to eliminate redundancy or dependency in data by means of decision trees (ID3 and J4.8). We use only its surviving attributes to construct the data set that will be fed into Naïve-Bayes trying to assert that we are working only with independent attributes and thus assuring that the learning process is being skewed as less as possible by redundancy and that the maximum efficiency is being obtained.

4.5 Preprocessing Stage Results

In figure 14 we can see that the most accurate algorithm is J4.8 (without Naïve-Bayes) obtaining a 98.39% of accuracy. The original attribute number was 71, which this algorithm reduced to 47. We can also appreciate that regarding Naïve-Bayes, the reduced data sets produce a slightly better performance, up to 4.5% improvement [9].

Besides the reliable accuracy preservation, the required processing power is also directly affected from the attribute reduction, since the number of calculations needed to classify in a smaller data set is lower and so is the power consumption.

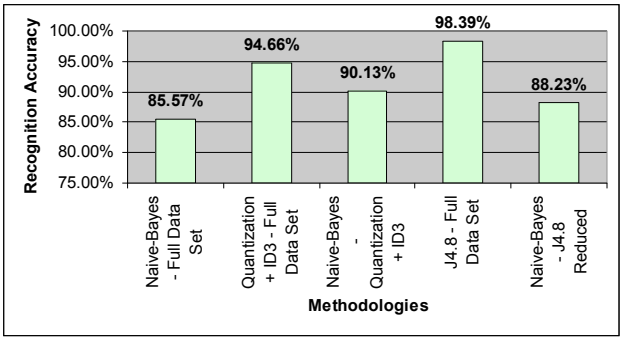


Figure 14: Preprocessing Stage Algorithm Comparison Results

In the J4.8 tree, the main attribute was pulse dominant frequency, the root of the tree. In the next level we find the width of the dominant frequency peak at half of its height divided by the frequency of the peak. One more level down, we find the maximum of dominant frequency in the pulse, the total number of pulses in the call and the dominant frequency at final 50% peak amplitude. These five attributes which J4.8 identified as the main ones, contrast with the song duration, number of phrases and number of notes identified by Nelson [23] and the speed, duration, frequency range, and center frequency identified by Bard [3]. The reasons of these disagreements are probably the usage of songs from different bird species and different algorithms for attribute selection, such as canonical discriminant analysis.

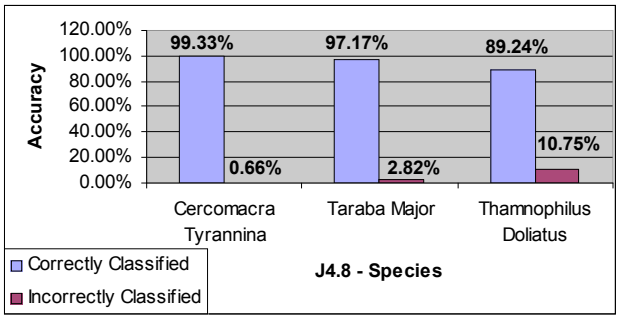


Figure 15: Preprocessing Stage J4.8 Results

In figure 15 we can observe the results obtained from the J4.8 algorithm on each of the individual species tested. It is clear that some problems were encountered with the third species in hand *Thamnophilus Doliatus*. Further testing is needed to confirm these results, mainly because with this particular species we had problems obtaining abundant recordings and we had the less amount of samples for the tests we performed. We can predict that the error percentage will decrease if we increase the amount of samples used for training.

It is clear that the best results were obtained from the decision tree algorithm J4.8. This algorithm not only gave us the most important attributes for classification, but also it was able to correctly classify most of the test instances we used. We can also observe that our reduced data set produced encouraging results that can motivate us to keep working on reduction techniques in order to include this preprocessing stage in the current sensor networks, so that the results obtained from them can be directly fed into the syllable processing algorithms.

4.6 Preprocessing Stage Conclusions

After obtaining the desired results we must find a way to simplify this process as much as possible in order to expand the classification techniques to include more Antbirds and to expand to other species. One important factor to consider is that we plan to do semi automatic classification using the sensor nodes, so we have to find more robust algorithms in order to obtain reasonable results. This is were our syllable based classifier can come into action since it wont depend as much on parameter manipulation as the data mining stage did will benefit from the results obtained from this stage.

An interesting approach that we are going to explore for future work in this stage is the use of *wavelets* to transform our bird songs to the frequency domain for the component analysis. Wavelets are the representation of a signal in terms of a finite length oscillating waveform. An advantage they pose over Fourier analysis is that you can have different window sizes when performing the transformation into the frequency domain. Also wavelets give a more accurate detail when representing signals using multiresolution analysis. Another difference they have with Fourier analysis is that wavelets are localized in both time and frequency whereas the standard Fourier transform is only localized in frequency. For some of our tests we used both the FFT and the STFT (Short-time Fourier transform), which is also localized both in time and frequency.

Another important method to consider for future work for this preprocessing stage is the unsupervised learning approach of *principal component analysis (PCA)*. PCA [8,15] is an approach to obtain the right features from the data. PCA searches for a projection that best represents the data in a least-squares sense. In other words, it reduces multidimensional data sets (such as our data set), to lower dimensions for analysis.

5 Methods: Current Work

5.1 Syllable Extraction

Our goal during this stage is to observe how the extracted syllables form natural groups in order to interpret their meanings and to form a tokenized representation of the data. We expect that these observations will lead us to generate a grammar that can describe the bird's song. With this syllable analysis and by means of generating a regular language, we expect to be able to efficiently classify different bird species, to be able to recognize up to a certain level of accuracy individuals and hopefully to interpret the communication birds have through their songs.

Once we have a small database of features from each of the bird species obtained through the preprocessing stage (see figure 4) we proceed to extract syllables from the bird's song by means of the syllable extraction software *Sound Analysis Pro (SA+)* [42]. SAP is an integrated system for studying animal vocalizations. The heart of the software is its digital signal-processing (DSP) engine called ztBirdEngine and its syllable identification algorithms that are based on hard-core mathematical foundations. It is the only software in the market developed specifically to extract and model syllables from recorded sounds. For these initial stages of this work, we are using this software as it comes directly from the source, but for the final stages of this investigation we plan to design a Matlab [21] syllable extraction software based on the source code of SA+. In order to perform correct syllable detections with this software we must calibrate it on a per species basis. For this purpose we use the results obtained from the preprocessing stage were we located the central frequencies of the signal for each species and the main components of their acoustic signals.

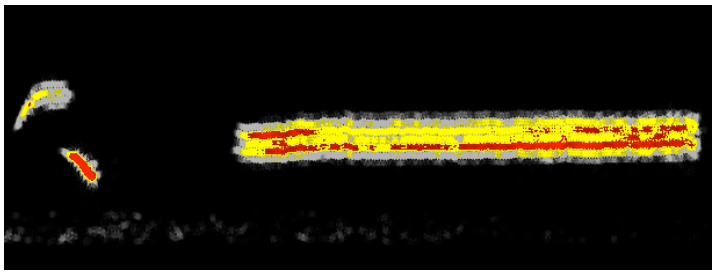


Figure 16: A multitaper sonogram of a bird song segment[42]

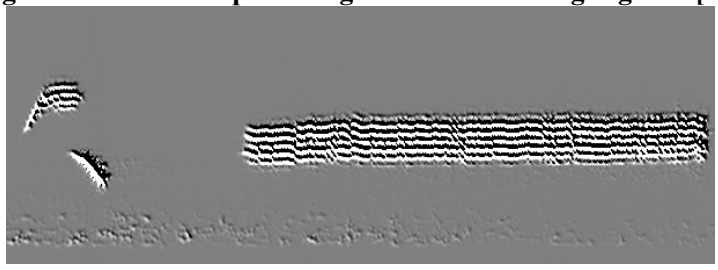


Figure 17: Spectral derivatives of the same bird song segment[42]

How can SA+ extract syllables in a better way than by doing a time analysis and a visual inspection of sonograms? The answer is simple, it uses a mathematical concept

known as *spectral derivatives* [42]. A traditional sonogram represents the power of sound in a time-frequency plan, while spectral derivatives represent *the change in power*. For each point of the two-dimensional time-frequency plan of a sonogram, one can measure change of power from left to right (on time), from bottom to top (on frequency) or at any arbitrary direction [42]. Spectral derivatives are derivatives of the spectrogram and are not artificially broadened. SA+ uses them to track frequency changes, providing greater detail than traditional spectrograms as seen in figures 16 and 17.

The formal definition of spectral derivatives is shown next. "Estimates of frequency and time derivatives of the spectrum may be robustly obtained using quadratic inverse techniques.

These estimates have the general form:

$$\sum_{k,k'} A_{k,k'} \tilde{x}_k(f) \tilde{x}^*_{k'}(f) \tag{9}$$

An Approximation of the above matrix:

$$\text{define } z(f,\theta) = \sum_{k=1}^{k-1} \tilde{x}_k(f) \tilde{x}_{k+1}(f) e^{i\theta} \tag{10}$$

$$\text{Empirically: } \frac{\partial S(f,t)}{\partial S} \propto \text{Re}(z(f,\theta)) \tag{11}$$

where: $\partial S(f,t)/\partial S$ is a directional derivative of the spectrogram in the time-frequency plan, the direction being specified by the angular parameter θ. In particular, the time and frequency derivates of the spectrogram may be obtained by setting, $\theta = 0, \pi$" [42].

The most difficult part of the process of extracting the syllables from the .wav files is the appropriate setting of the parameters to detect vocalization to segment the sound into syllable units. Deciding where to segment a song is a very critical process. In order to diminish errors that might appear in this stage, we use the information we obtained from each individual song during the preprocessing stage. With this information and with most relevant components of bird songs that were used for an initial classification and dimensionality reduction obtained through data mining we can adjust the syllable extraction software.

For SA+ the sound detection process is based on three parameters that can be adjusted knowing the most relevant parameters of each species as found in the preprocessing stage. These parameters are:

- The amplitude threshold within a frequency range.
- The Weiner entropy or spectral shape threshold within a frequency range
- Noise detection and rejection based on a power analysis.

In the program you have two sliders to control amplitude and spectral shape as seen in figure 18. For this program amplitude is measured in decibels as shown in the following formula:

$$Amp(dB) = 10\log_{10}\sum_{f} P_f - baseline \qquad (12)$$

where: P_f is the power at any frequency and baseline is set as an arbitrary value that for our purposes will be set at 70dB.

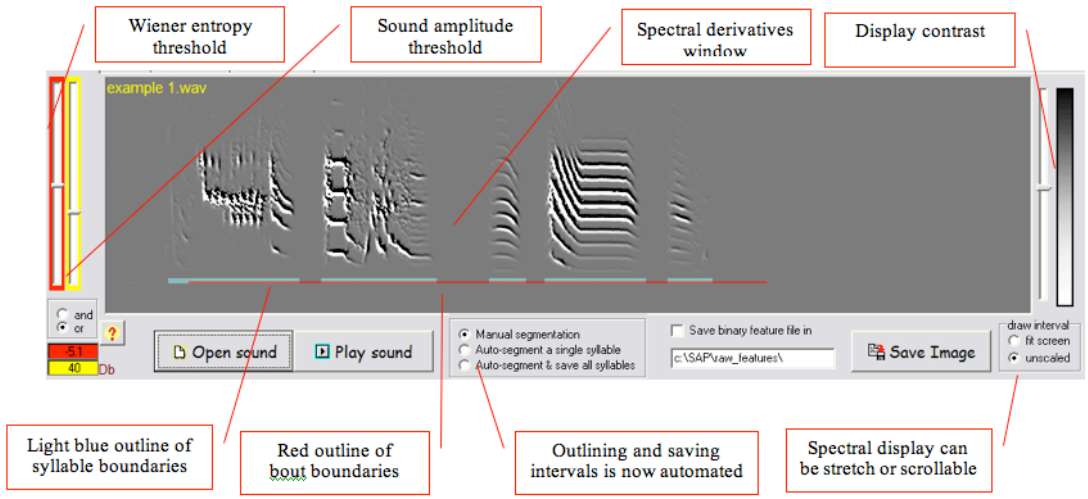


Figure 18: SA+ detection screen [42]

Weiner entropy is a measure of the width and uniformity of the power spectrum [42]. Weiner entropy is measured as a pure number; it is measured on a scale from zero to one. White or thermal noise has entropy of 1 while pure tones have entropy of 0. Syllables that are very tonal will have a very low Wiener entropy (highly predictable) whereas those that are more related to white noise will have a very high Wiener entropy (highly unpredictable). The formal definition of the Weiner entropy is shown below [42]:

Wiener entropy is a pure number defined as the ratio of geometric mean to arithmetic mean of the spectrum:

$$W = \log\left(\frac{\exp[\int dfLog(S(f))]}{\int dfS(f)}\right) \qquad (13)$$

Finally for noise detection and rejection we have a box on the program were we specify which frequencies to cut off. You can apply both high and low pass filters using this parameter to clean the signal.

Once we adjust the program parameters, the advantage of this syllable detection software is that it can detect syllables from all samples of the same species without having to do any major changes or adjustments. Finally we just have to open up a

new song from the species and it will do an efficient job detecting the syllables as we can observe in figure 19.

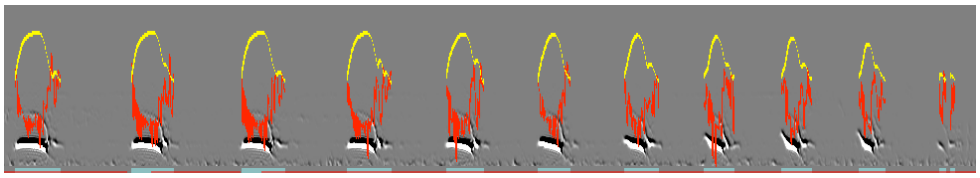


Figure 19: Taraba Major syllable analysis produced by SA+

In this figure we can observe each individual syllable as shown by the spectral derivative of the song. A small cyan box underlines each syllable. The yellow lines represent the amplitude of the signal while the red lines represent the Weiner entropy.

Sound Analysis Pro measures 15 different parameters from each of these individual syllables and places them as individual vectors in a MySQL database. Most of these parameters were the same as the ones we obtained from the data mining stage. The most important parameters for syllable edge detection are the following:

- Amplitude
- Mean Frequency
- Pitch
- Weiner Entropy
- Goodness of Pitch
- Frequency Modulation
- Amplitude Modulation

From the MySQL database constructed by SA+ we can export the information either to Microsoft Excel or directly into Matlab [21]. The process of extracting syllables is delicate and time consuming since we need at a great number of syllable samples per species in order to perform any type of analysis and the process for obtaining them varies depending on the size of the song. A typical song might contain from 5 to 15 syllables.

Using this syllable extraction process we obtain data-vectors that contain acoustic features, which represent the syllables extracted from the bird songs. These vectors form a data matrix as output, whose dimension is either 100 x 15 for the 163-sample data set and 1000 x 15 for the 1000 sample data set. It is important to note at this moment that the high dimensionality of these matrices will be a serious problem for data visualization and that reduction techniques like PCA or Sammon [15] mapping must be used in order to plot our results. Once we obtain enough information, we can proceed to visually analyze them in order to find natural relationships amongst the elements. This analysis can give produce framework in order to design a language to model the songs.

5.2 Syllable Interpretation

In order to generate tokens for each individual syllable and to create a grammar based on our assumption, that in its simplest form, bird songs must adapt to regular languages, we must first understand the natural groupings of these building block elements in order to work with them. We must also take into consideration all the questions that were generated during the proposal and to identify at least in a range value, the number of different syllables that these bird species have in order to start modeling the language. For this purpose, the next stage in the development of this work is to find the natural classification of these elements. Since we don't have much previous knowledge on the subject, and since most of the background work reviewed doesn't consider this critical aspect, we decided to apply unsupervised learning techniques in order to find the natural groupings of the syllables.

Unsupervised learning is a method of machine learning where a model is fit to observations. It is distinguished from supervised learning by the fact that there is no apriori output [51]. Unsupervised learning methods are usually appropriate for large data mining applications. It's a method that can give us some sort of background relations from unknown or disorganized data for which we have no clue on how the different elements are arranged together. Lastly, in the early stages of an investigation it may be valuable to perform exploratory data analysis and thereby gain some insight into the nature or structure of the data [8]. The discovery of distinct subclasses-clusters or groups of patterns whose members are more similar to each other than they are to other patterns-or of major departures from expected characteristics may suggest we significantly alter our approach to designing the classifier [8].

Our first approach to view and comprehend the natural associations and groupings in our data set is to apply *cluster analysis*. Cluster analysis, also called data segmentation, has a variety of goals. All relate to groupings or segmenting a collection of objects into subsets or "clusters", such that those within each cluster are more closely related to one another than objects assigned to different clusters [15]. Clustering algorithms can be hierarchical or partitional. Hierarchical algorithms attempt to find successive clusters from the previously established ones. On the other hand partitional algorithms determine all clusters at once. Since we don't have any previous knowledge on the natural groupings of bird syllables, we have to use the approach taken by partitional algorithms in order to perform our clustering. All of the visualization techniques used for this work were implemented using Matlab [21].

5.3 Clustering Algorithms

5.3.1 K-means

The first approach we took was to use one of the most popular iterative descent clustering methods available, known as the *k-means clustering algorithm* [8,15]. It is an algorithm that assigns each point to a cluster whose center is the nearest. The center is determined as the average point in the cluster. This type of clustering algorithm is intended for situations in which all the variables are of a quantitative nature and where squared Euclidian distance is chosen as the dissimilarity measure [15] as shown below:

$$d(x_i, x_{i'}) = \sum_{j=1}^{p} \left(x_{ij} - x_{i'j} \right)^2 = \left\| x_i - x_{i'} \right\|^2 \qquad (14)$$

The *k*-means algorithm is as follows [15]:

- For a given cluster assignment C, the total cluster variance is minimized with respect to $\{m_1, \ldots, m_k\}$ yielding the means of the currently assigned clusters.
- Given a current set of means $\{m_1, \ldots, m_k\}$, is minimized by assigning each observation to the closest (current) cluster mean. That is,

$$C(i) = \arg\min_{1 \le k \le K} \left\| x_i - m_k \right\|^2 \qquad (15)$$

- Steps one and two are iterated until the assignments do not change.

After applying *k*-means with our demo data set (33 samples per species) we concluded that there would be some problems in our classification tests that would have to be considered. The first problem detected was that k-means uses fixed centroids to divide the data into groups and them in a bi-dimensional scale and our input data has a 15-dimension complexity. Once the data is divided they either pertain to a centroids or to another, but we do not know up to what percentage the data actually fits into each group. The problem with this concept is that our data is unknown to us at this stage and we need to have an algorithm that can give us a specific degree of pertinence of each syllable to each group. In order to enhance the classification and hinder the effects of this problem the *fuzzy c-means* algorithm was selected.

Another problem we face is that most partitional algorithms require us to fix initial cluster centroids in order to do the partitioning of the data, and to determine exactly how many clusters do we pretend to use. In our case, we do not know how many different syllables can be obtained from each species or in general from bird songs. This is one of our central questions and we need a tool or technique to be able to predict these parameters. One idea is to do different tests manually and visually inspect the results in order to determine how many syllables (cluster centers) can be obtained from the bird songs. The other is to apply a clustering technique that doesn't

need a fixed cluster number value. The algorithm to use for this purpose is called *subtractive clustering*.

5.3.2 Subtractive Clustering

Subtractive clustering [21,37] is a fast one pass algorithm designed to estimate the number of clusters and their corresponding centers for a given data set. The algorithm works in the following way. The object in the data set with the highest potential (P_1^*) is selected as the first cluster center. Next, the potential of each one of the elements or objects in the data set is reduced proportional to the degree of similarity with this previous cluster center. Therefore, there is a larger subtraction in potential of objects that are closer to this specific cluster center compared to those that are farther away. After this subtractive step, the object (x_i) with the next highest potential (P_i) is selected as the next candidate for cluster center. This procedure is constantly repeated until there is a convergence.

The tests performed with this algorithm were compared to the ones performed manually. It is very important to note that the results obtained were similar which give a degree of objectiveness to the manual testing performed. In figure 20 we can see the results obtained by applying this algorithm to our data set. It is important to note that we used two different data sets, one with 33 syllable vectors per species and another one with 333 syllable vectors per species.

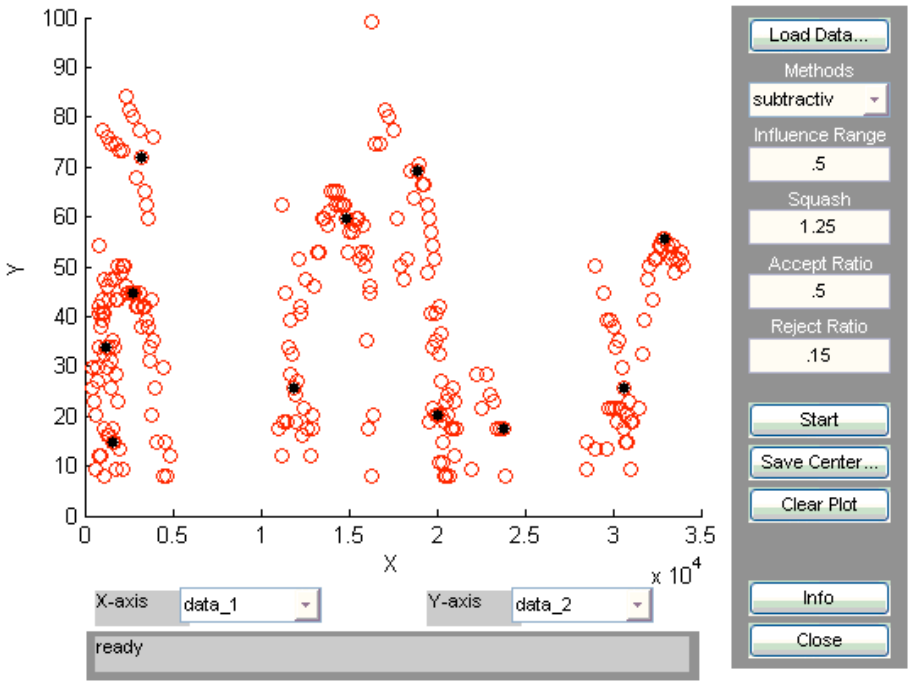


Figure 20: Subtractive Clustering with 100-syllable data set

From this figure we observe that the natural grouping of the data predicted around 10 different centroids or syllables, as noted by the concentrated black dots. Our manual testing took us to decide that for the three species in hand we had an approximate range of 8 to 12 different clusters. This can take us to speculate and determine that

there must be around 10 different syllables from these three species. Two different problems arise with this statement. The first, will these results hold with more data from these species? Are these results valid for each individual species of the family of the Antbirds? We performed similar tests with the 1000 syllable vector data set as seen in figure 21.

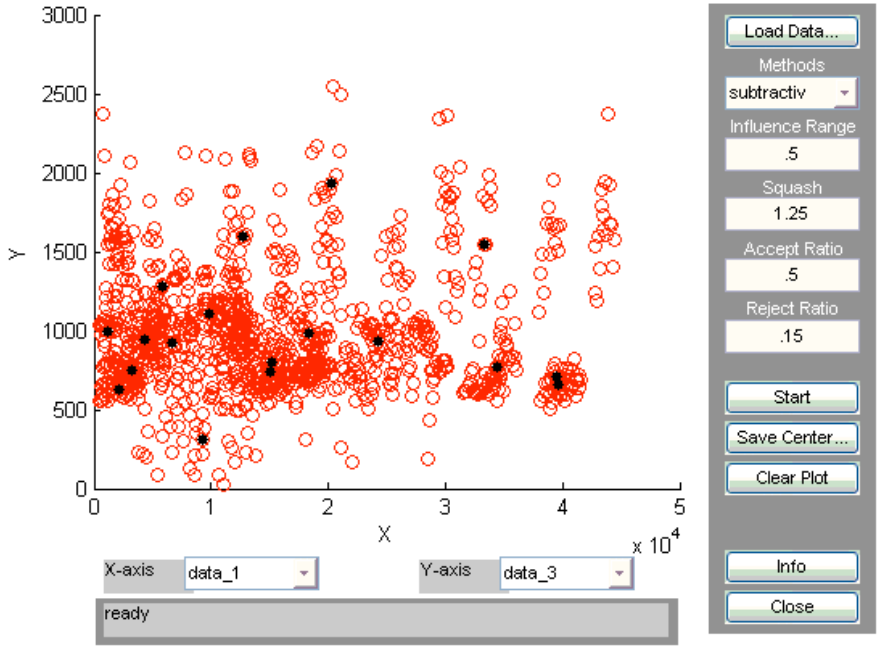


Figure 21: Subtractive Clustering with 1000-syllable data set

The results obtained predict that the number of syllables that represent the grouped data can be estimated to be from 10 to 18 different syllables or clusters. Further testing is necessary in order to determine a fixed number and to test individual birds from the Antbird family, to determine if the number of syllables matches each case. Also further research is needed in this area; unfortunately there are not many references available on the subject. In a personal communication with Dr. Martin Cody [40], and he found the results to be an accurately estimate and considered further testing vital to establish a syllable count for these species.

5.3.3 Fuzzy C-means (k-means)

Why do we mix clustering and fuzzy logic? Clustering can be a very effective technique to identify natural groupings in data from a large data set, thereby allowing concise representation of relationships embedded in the data. Fuzzy logic [8,15,37] is an effective paradigm to handle imprecision. It can be used to take fuzzy or imprecise observations for inputs and yet arrive at crisp and precise values for outputs. Also, the Fuzzy Inference System (FIS) is a simple and commonsensical way to build systems without using complex analytical equations. Clustering and fuzzy logic together provide a simple yet powerful means to model the relationship that we want to analyze.

In every iteration of the classical *k*-means algorithm, each data point is assumed to be in exactly one specific cluster. On the other hand, in fuzzy clustering each point has a degree of membership to the clusters, as in fuzzy logic, rather than just belonging to a specific cluster. The points on the edge of a cluster may be in the cluster up to a certain degree that points to that cluster center. For each point *x*, we have a coefficient giving the degree of membership of that point to the k^{th} cluster $u_k(x)$. $U_k(x)$ denotes the probability of a certain point *x* to belong to the cluster *k*, as noted in the following formula:

$$\forall x \sum_{k=1} u_k(x) = 1 \tag{16}$$

In fuzzy *c*-means the centroids of each cluster is the mean of all the points, weighted by their degree of membership to that cluster as shown below:

$$center_k = \frac{\sum_x u_k(x)x}{\sum_x u_k(x)} \tag{17}$$

where the degree of membership is calculated by:

$$u_k(x) = \frac{1}{\sum_j \left(\frac{d(center_k, x)}{d(center_j, x)} \right)^{\frac{1}{(m-1)}}} \tag{18}$$

The fuzzy *c*-means algorithm is as follows:

- Select the number of clusters to be used.
- Assign randomly to each point coefficients that will give the initial degree of membership for each cluster.
- Repeat until the algorithm converges (that is until the coefficient's change in two consecutive iterations is below a certain sensitivity threshold):
    - Calculate the centroids for each cluster.
    - For each point compute the coefficients of membership to the clusters
    - Adjust cluster centroids correspondingly to neighbors.

We used the Fuzzy Logic Toolbox in Matlab [21] in order to do the fuzzy clustering for our data. The inputs for our program were the inverse matrices obtained from SA+ that contain the data-vectors of 15 dimensions representing the bird's syllables. We decided based on testing to use 10 clusters for our analysis. The outputs are the following:

- A table of *N* vectors x *k* clusters with a degree of correspondance (%).
- Index *(x)* = that indicates which vector corresponds to each cluster.
- Center = a matrix that contains the final cluster centroids.

One problem we encountered when trying to visualize the results obtained from this algorithm was one of the questions we had in mind. How do we plot our data if we have syllable-vectors with fifteen dimensions? We have to perform a data reduction in order to plot our results. For this purpose we utilize Sammon mapping [15]. Sammon mapping is a mathematical technique that allows the data to be represented in a lower dimensional space, usually 2-D. It is an iterative method that uses the gradient of the error to minimize the actual error. It is important to note that there will always be be some type of error in the projection of the points.

Two different type of tests were performed whose results can be seen in figures 22 and 23:

- The full data set was used for the fuzzy c-means clustering. One thousand samples were used from three different species. Sammon mapping was only used to plot the results.
- The full data set was reduced using Sammon mapping before running the fuzzy *c*-means algorithm. One thousand samples were used from three different species. This was done only as a trial test in order to see if the results were comparable to those obtained in the other test. The goal of this specific test was to reduce the computational cost and see if it's viable to use this reduction with a sensor network.

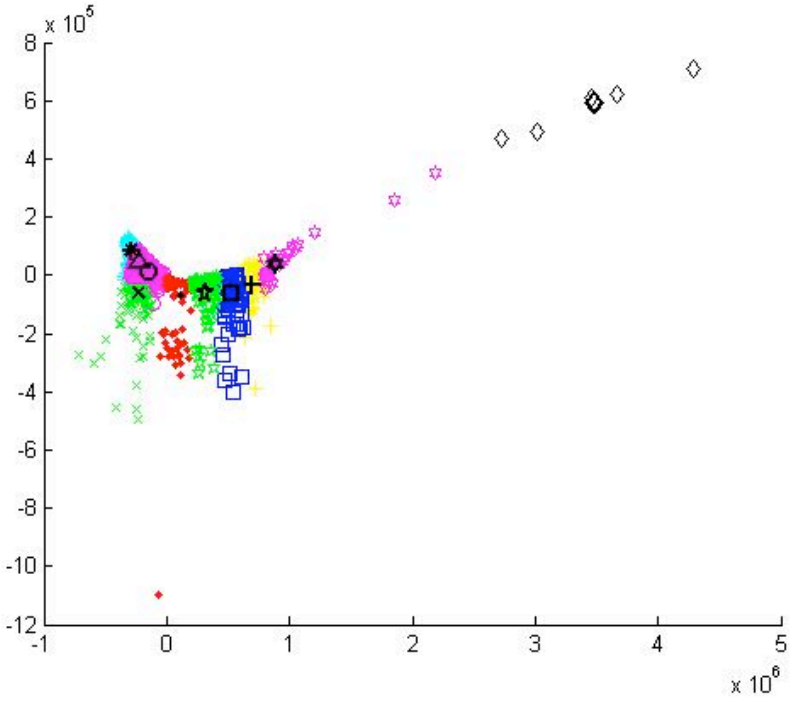


Figure 22: Fuzzy c-means using Sammon mapping only to plot

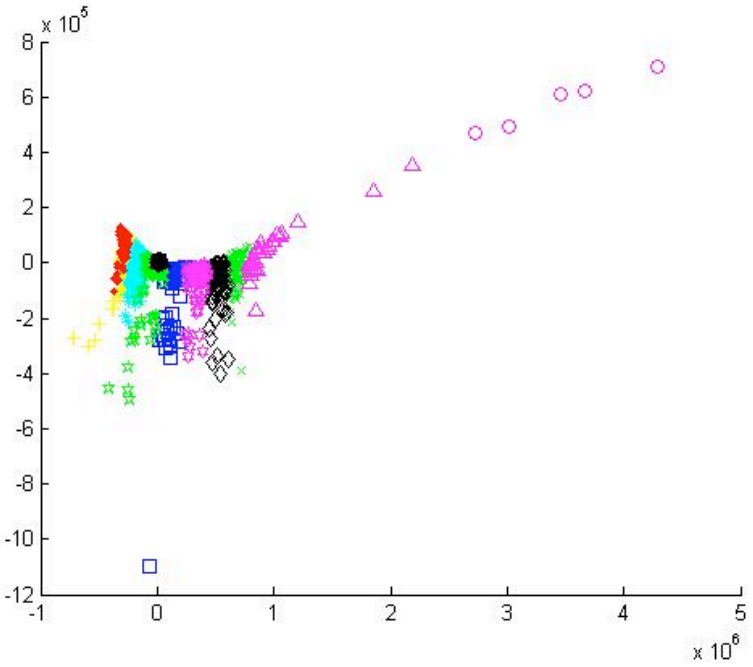


Figure 23: Full reduction using Sammon mapping prior to fuzzy c-means

In figure 22 we can see the results of processing the full data set using fuzzy c-means. Sammon mapping was only used to reduce the dimensionality of the data from 15 dimensions into a bi-dimensional plot. We can clearly observe that defined clusters are organized and the data is precisely split into ten different clusters. When performing these same tests with the 163-samples data set we obtained similar results. When doing similar tests with 8 to 17 clusters, we got similar results but with some inconsistencies. Further testing is required in order to establish a definite number of syllables for these three species when clustering their syllables all together. An important factor to consider is that we performed these clustering tests with a combined data set from the three species of Antbirds all together. We also performed individual tests for each species obtaining similar results as it can be seen in figures 24 and 25. It can clearly be seen that some syllables separate nicely into clusters while others are too tightly packed together to visualize using this clustering technique.

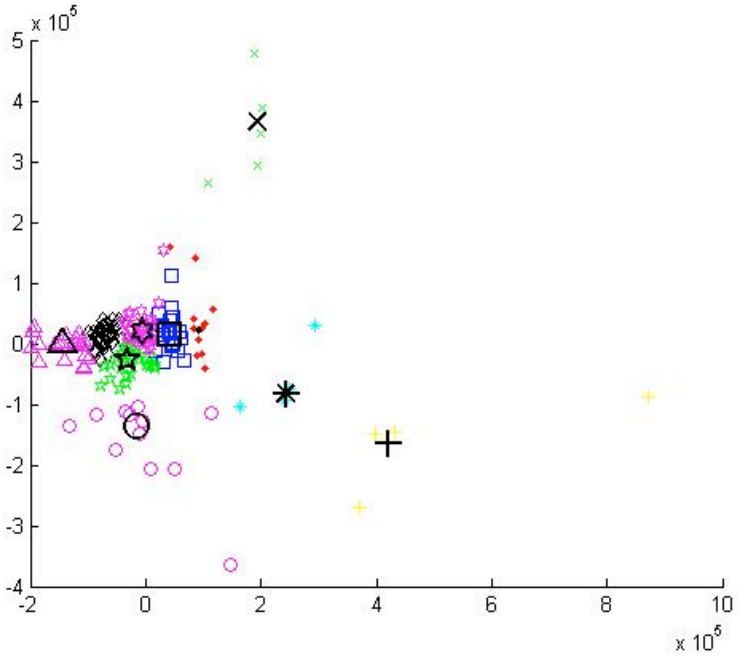


Figure 24: Fuzzy c-means, 163 samples of Great Antshrike, using Sammon mapping only to plot

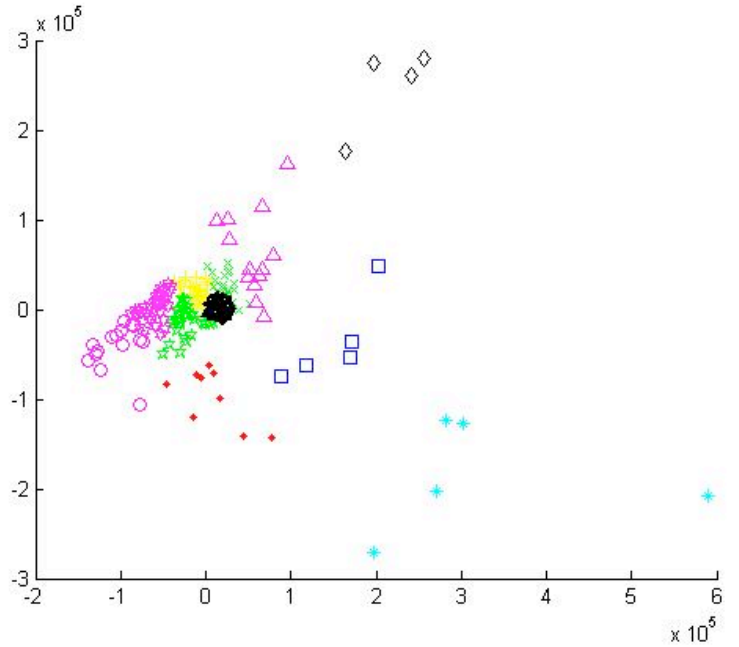


Figure 25: Full reduction using Sammon mapping with 163 samples of Great Antshrike prior to fuzzy c-means

Although we can visualize some of the natural groupings using these clustering techniques we cannot precisely determine if there are certain similarities between the syllables of the different birds used for this study. We can certainly distinguish some of the natural groupings and start making analysis for further testing. In order to analyze each species by themselves and contrast them altogether to see if there are similar syllables in each species, we decided to use another unsupervised learning technique that works extremely well with multidimensional data, the technique is called *Self-Organizing Maps.*

5.4 Self Organizing Maps

Self-organizing maps [8,15,16,34] (SOMs) are a data visualization technique invented by professor Teuvo Kohonen [16] to reduce data dimensions through the use of self-organizing neural networks. The SOM is an algorithm to visualize and interpret large high-dimensional data sets. The problem they try to solve is that humans cannot visualize high dimensional data; therefore techniques are created to help us understand the relationships that can be found in these high dimensional data sets.

A self-organizing map is trained using unsupervised learning methods. They produce low dimensional representations of high dimensional data. They permit us to observe properties and relationships of high dimensional data while preserving their original properties.

A SOM consists of neurons organized on a regular low-dimensional grid (figure 26). The number of neurons varies considerably from a couple of neurons to a few thousand. Each neuron is represented as a *x*-dimensional weight vector. The dimension of this vector is the same of the one from the input vector. Neurons are connected to adjacent neurons by a neighborhood relation (hexagonal or square fig 26), which will dictate the topology of the map. The SOM training algorithm resembles the ones used by vector quantization as they were shown in a previous section of this document. It is important to mention that this algorithm differs from the one of vector quantization in that in addition to doing the best matching weight vector (BMV), they also update the neighbors on the topological map that are close to the Best Matching Unit. As an end result, the neuron on the topological map are ordered and grouped closer to the BMU as seen in figure 27. Self Organizing Maps are usually referred as a clustering technique or a vector quantization application. They are much more than they appear but at the same time they used similar algorithms to those presented by those subjects.

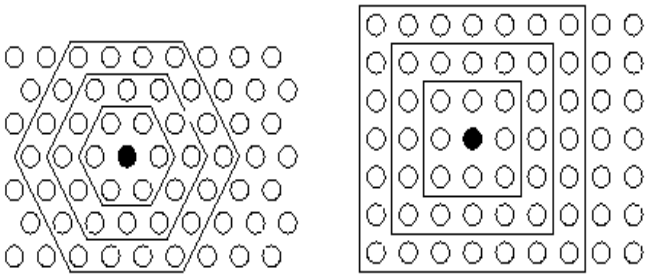


Figure 26: Neuron Neighborhoods

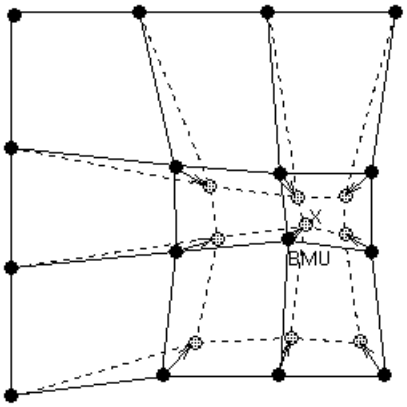


Figure 27: Neighborhood after training and BMU[44]

5.4.1 SOM Sequential Training Algorithm

The SOM is trained iteratively. In each training step, one sample vector x from the input data set is chosen randomly and the distances between this vector and all the weight vectors of the SOM are calculated using a distance measure. The neuron whose weight vector is closest to the input vector x is called the Best Matching Unit denoted by c in the following equation [44]:

$$\|x - m_c\| = \min\{\|x - m_i\|\} \tag{19}$$

where: $\|\ \|$ is the distance measure, typically the Euclidean distance. The toolbox used for this project, the SOM Toolbox [32] uses a more complicated method to calculate the distance measure in order to handle missing data from vectors. Also, each variable has an associated weighting factor, defined by a mask. This mask is primarily used to exclude certain variables from the BMU finding process. The final distance measure therefore takes the form of [44]:

$$\|x - m\|^2 = \sum_{k \in K} w_k (x_k - m_k)^2 \tag{20}$$

where: K is the set of known variables of the sample vector x, x_k and m_k are the kth components of the sample and weight vectors and w_k is the kth mask value.

Finally, after finding the BMU, the weight vectors of the SOM are updated so that the BMU is moved closer to the input vector in the input space [44].

5.4.2 SOM: Preliminary Results

When interpreting the results obtained from the SOM, we must first analyze the ones based solely on the SOM's model vectors. These results are presented in a SOM map that display not only the clustering information, but also a component vector plane that shows the influence of each single component in a vector, to the final output of the clusters as shown in figure 28. The U-Matix (Unified Distance Matrix) indicates how close are neighboring map units in the input space. It determines the different clusters that can be obtained from the input data. This is achieved by using topological relations property among neurons after the learning process. From figure 28, top left, we can observe that the light areas indicate the cluster borders. High values in the U-Matrix represent a frontier region between clusters, while low values represent a high degree of similarities among neurons on that region or cluster.

Four distinctive clusters can be observed, although there are only three different species producing the data. It is clear that the only marked cluster is the one in the top right of the figure. The other clusters are a bit harder to recognize and can represent that there is some common elements between them. This can represent that the syllables of the two species in those clusters are highly related. By comparing these cluster sections to the uncolored hexagonal diagram bottom right, we can observe that it is quite simple to distinguish the species in the top right by means of the U-matrix. The *Dusky Antbird* can be easily identified from the other two, and its syllables are also easily separated. Further tests will show this in a clearer manner. It is important to note that the values represented in figure 28 were normalized before applying the SOM but were shown de-normalized in the diagram

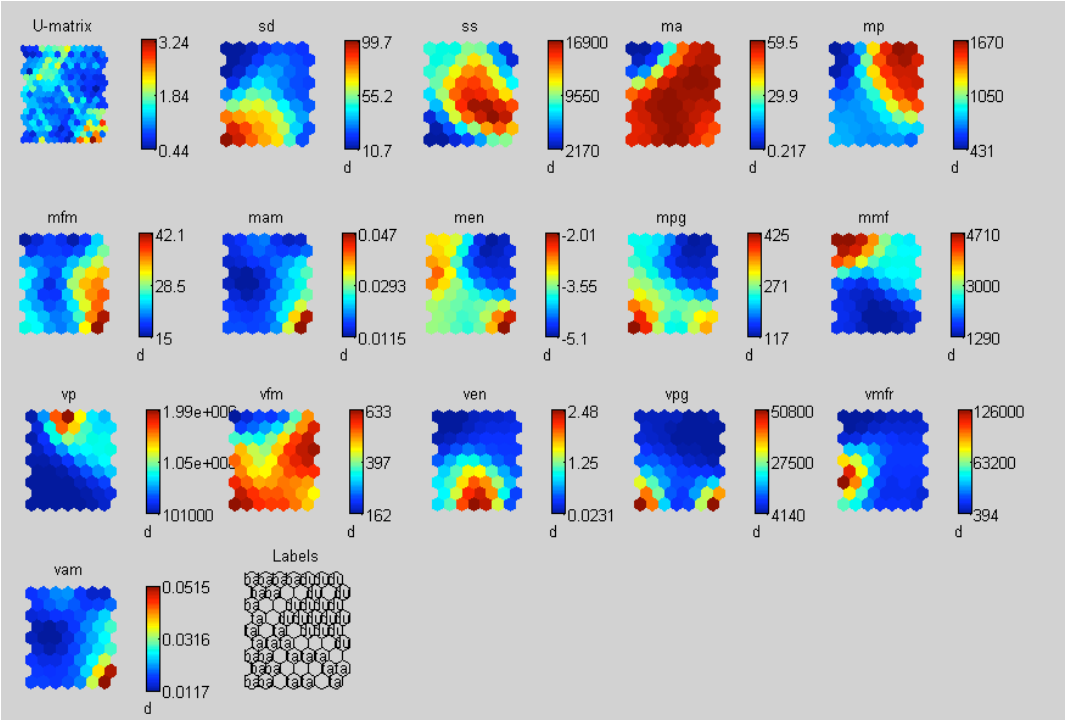


Figure 28: Self Organizing Maps Results: Component Plane and U-Matrix

The next 15 sub-figures represent the different vector components of the data set. They are called the component planes. After the learning process we can color each neuron according with each individual component value. By means of these component planes we can realize emerging patterns of data distribution on the SOM's grid. We can also detect correlations among variables and the contribution of each on to the SOM differentiation only viewing the colored patterns for each Component Plane.

From our results we can notice that some components like mean entropy and mean pitch goodness are highly correlated. This clearly indicates that there are natural relationships in the components themselves that make the SOM and that there might still be possibility for reduction. It is important to note that contrasting these results with those obtained in [34], they mentioned to have done a reduction of up to 7 components for their classification tests and with those results in mind and the results obtained from the component planes, we can do further testing reducing our own 15 dimension vectors.

Although component planes are very useful when trying to represent a lot of information at once, we also are interested in viewing fewer variables in a scattered plot to give us another idea of how the data is organized in the data set. For our case, we are interested to observe natural relationship among the syllables to try to identify not only different clusters but also how close are the different bird species in hand to each other.

Figures 29 and 30 show two scatter plots from the dataset using PCA [8,15] projection to reduce dimensionality.

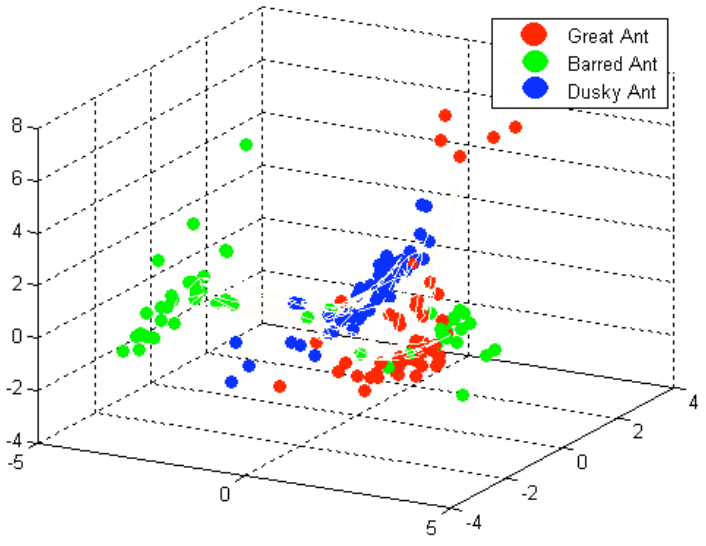


Figure 29: SOM species syllables using PCA projection 3d

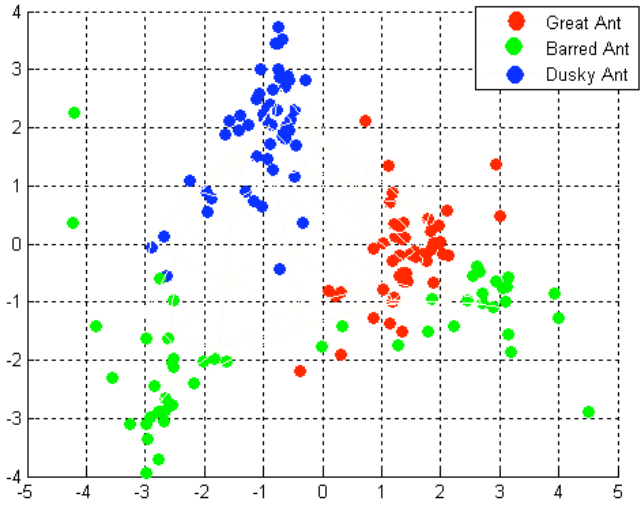


Figure 30: SOM species syllables using PCA projection 2d

By analyzing figures 29 and 30, we can observe how the syllables from the *Dusky Antbird* clearly separate from the other two species. It can be seen that there are some false positives from the *Dusky Antbird* that were identified too close to part of the group that has the syllables from the *Barred Antshrike*. It can also be observed that the *Barred Antshrike* has two blocks of syllables. One block that is clearly separated from the other two species and the other one that has shared components with the *Great Antshrike*.

It is very interesting to corroborate some of the theories we had, in which we assumed that the *Dusky Antbird* had elements in its song that could clearly separate them from the other two species. Such elements can be seen in the spectrograms presented earlier in this document. It can also be seen from these results that this particular bird has very little in common in its syllabic structure from the other two and that there might be the need to create a more generalized language from the acquired syllables. There might also be the possibility that a more detailed species-specific string construction might be necessary, but further testing is required at this stage in order to reach a final conclusion.

From these results we can conclude that further experiments are needed in order to gather enough visual information to create different syllable groups. We must further understand the relationships between these common syllables in order to start separating the data into different blocks.

To conclude, it was demonstrated that the use of both clustering techniques and Self Organizing Maps are vital in the development of this project and that the results they can generate will further our understandings on bird songs, since they are giving us insights on bird syllable organization and groupings that would otherwise be complicated to obtain due to the high dimensionality of the data.

6 Conclusions and Future Work

6.1 Conclusions

From the results obtained up to this moment, it is safe to conclude that we are on the right track towards generating an automatic bird song classifier based on an acoustic syllable model. It is important not to rush the syllable visualization stage, represented in this work by the clustering techniques and the self-organizing maps, since the initial organization of the syllables will form a key factor to the remaining stages in this project. It is also important to work in parallel in this stage to try to explore other areas for feature extraction, such as those presented by applying the wavelet transform.

The preliminary results obtained up to this moment suggest that there is some sort of relationship between two of the three selected species. It can be clearly seen from the results of the U-matrix and the PCA projection of the SOM that both *Taraba Mayor* and *Thamnophilus Doliatus* share syllables that make up their songs. It can also be observed that although they share some syllables others are easily separated as well as those from the *Cercomacra Tyrannina*. These findings can start to answer some of the questions that were formulated at the beginning of this document. We can clearly observe that there might be a number of species-specific of syllables but there are also syllables that are common to the entire family of Antbirds.

The classification results obtained in the preprocessing stage can help us focus our efforts towards a certain bird in particular and they also give us insights on the signal as well serve as a control measure once the syllable based classifier is up and running. The spectral analysis and the syllable extraction techniques are very important since they are the ones that define the actual syllables for this work. The syllables are the building blocks of this project and we must be very careful in their generation, therefore we will concentrate our efforts in perfecting this part of the project before starting the actual string generation and language construction.

Finally it is important to remember that our main goal is to be able to perform bird species classifications based on the acoustic signals they produce on a sensor network platform, and that most of our work has to be performed keeping in mind that our algorithms must be as simple as possible since they will be running in these power restricted platforms. This is why we are working on a syllable-based classifier that will incorporate the results that were obtained in the past and also run in these types of platforms. As a final remark, the final advantage our syllable-based classifier will have is that it will work efficiently and semi-automatically on all samples from the same species, since it will use the pre-acquired information from the bird song characteristics database. Our tests up to this point support this assumption, since by applying the current syllable extraction software to different samples, we did not have to re-calibrate the software more than once per species.

6.2 Future Work

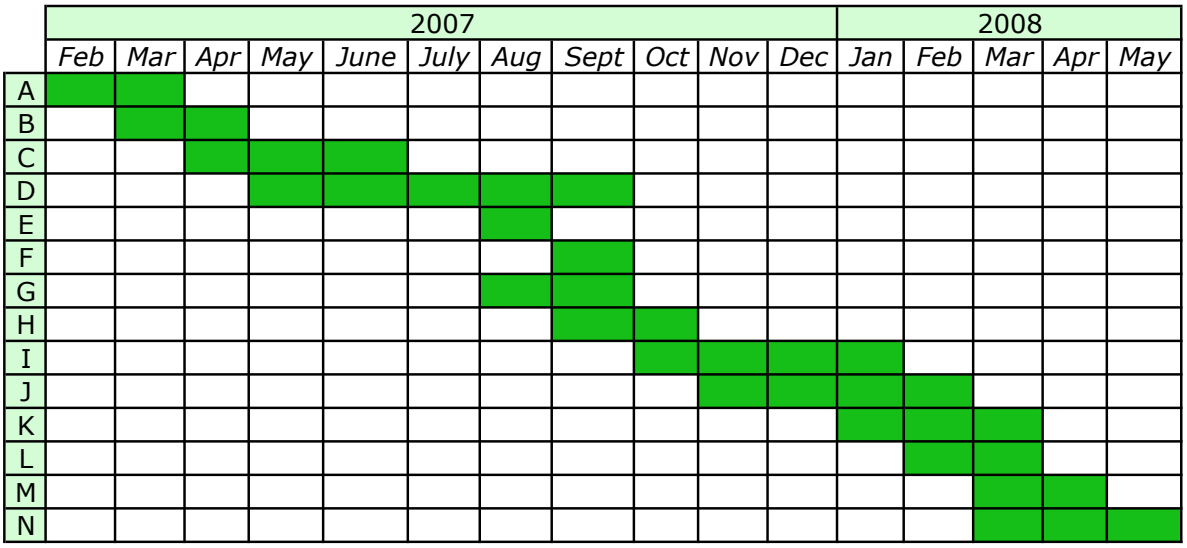
We divide our future work in two stages. The first stage will consider the following approaches to perfect the syllable acquisition techniques that we already have. We have to consider alternate approaches to do spectral analysis and also we must consider more classic approaches for classification, such as those used by speech recognition with the traditional Hidden Markov Models [25].

- ***Wavelet Analysis***: we are going to try another approach to perform feature extraction without loosing too much information.
- ***Using HMMs for classification:*** we are going to consider HMMs for our preprocessing stage.
- ***Automatic Spectral Analysis:*** we are going to collect the necessary information in order to automatize this process.
- ***Custom built syllable extraction software***: this software will use the species selection database for automatic configuration.

As a second stage we consider to gather further information from the extracted syllables and to start the construction of our syllable species database. We consider that this database is a fundamental piece of the project and will influence significantly the development of the syllable-based language. Once completed the first stage we will work in the following areas:

- ***Bird Species Characteristics Database:*** we will gather information from the preprocessing stage and the spectral analysis to construct an online database used for syllable detection.
- ***Species Selection Syllable Database Construction:*** we will use the information from the Bird Species Characteristics Database and the knowledge gained from further experimentation with SOM, to construct this database.
- ***String Generation:*** once we have the syllables in the database and the natural relationships acquired by means of SOM, we will start generating strings from the extracted syllables.
- ***Regular Language Induction:*** once we have the strings representing each individual syllable we will develop rules that will conform a small regular language from the data.
- ***Regular Language Construction:*** once the rules have been generated, we will model the bird songs as simple phrases from our regular language.
- ***String Classification using HMMs:*** once the regular language is generated, we will use Hidden Markov models in order to perform a comparison of the results obtained, with a finite state automaton for species and individuals classification.
- ***Generate strings using the regular language constructed:*** once generated we will convert the strings into a sound signal. If possible play the sound signal to the corresponding species bird and monitor its behavior.
- ***Language Interpretation Results*:** for the final stage of this project we will interpret the results from all previous sections and try to analyze if it's possible to interpret avian behavior by means of analyzing the regular language expressions.

6.3 Planned Time Table



Stage 1

A Wavelet Analysis.
B Using HMMs for classification.
C Automatic Spectral Analysis.
D Custom built syllable extraction software that will use the species selection database for automatic configuration.

Stage 2

E Bird Species Characteristics Database.
F Species Selection Database Construction.
G Use SOM to gain more knowledge from the syllables.
H String Generation.
I Regular Language Induction.
J Regular Language Construction.
K String Classification using HMMs.
L Once the regular language is generated, use hidden markov models in order to perform species and individuals classification comparisons.
M Generate strings using the regular language constructed, and convert the generated string to a sound signal. If possible play the sound signal to the corresponding species bird and monitor its behavior.
N Language Interpretation Results.

Figure 31: Gant Diagram

7 Appendix 1

7.1 Mathematical Concepts:

In this section we present the mathematical concepts related to the signal processing stage of this project. They will aid in the comprehension of the work done with the syllable generation analysis and the preprocessing stage.

Definition 1: Period. Is the number n of revolutions that a particle performs in a given time. It is the number of cycles as a result of time. It is the inverse of frequency and it is expressed in seconds.

Definition 2: Frequency. It is the measurement of the number of times that a repeated event occurs per unit of time. It is the inverse of the period (T).

$$f = \frac{1}{T} \tag{21}$$

Definition 3: Band Limit. It is the maximum frequency of a sound in a given time interval f_{bl}.

In general terms an analog signal can be represented mathematically as a function of time without loosing the precision of its content. Such representation is given by the following formula, where t is the independent variable.

$$x(t) = A\sin(2\pi ft + \phi) \tag{22}$$

where: a is the amplitude of the signal, f is the frequency and ϕ is the phase of the signal.

Definition 4: Phase: It is a measure of the relative position in time within a single period of a signal. Normally phase is used in the literature to represent a *phase shift* of a signal. A phase shift is a difference or change of the initial phase of the signal.

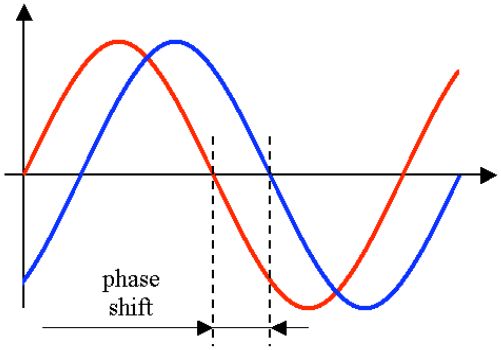


Figure 32: Phase shift of a signal [51]

Definition 5: If x(t) is a signal with a band limit of f_{bl} Hz, then the frequency f_{ny} is defined as $2f_{bl}=f_{ny}Hz$ which is the Nyquist frequency of the signal *x(t).*[45]

In order to digitze sound we must consider Nyquist's sampling theorem, which is shown below.

Theorem 1: Nyquist Sampling Theorem. Let x(t) be a signal whose Nyquist frequency is of f_{ny} Hz, then this singal can only be rebuilt from its sampling values if the sampling frequency if greater than the Nyquist frequency of the signal, satisfying that $f_{ny} < f_s$.

Sampling is the process of converting a signal into a numeric sequence. It is the process of taking samples of a signal in a time scale in order to convert it into a digital sample. From this theorem we can conclude that in order to reconstruct any sound signal, we must sample it at discrete time intervals and express it digitally.

Definition 6: Periodic signal. A signal *x(t)* is periodic if there is a constant $T_0 > 0$ such that :

$$x(t+T_0) = x(t) \text{ for every t} \tag{23}$$

By means of approximation theory, [5] any periodic signal can be represented by means of a Fourier series [18]. A Fourier series is the sum of sinusoidal signals of different frequency and amplitude. In this way any periodic signal (for example a sound signal), can be analyzed and represented by means of sinusoidals expressed mathematically as follows:

$$\begin{aligned} x(t) &= a_0 + \sum_{m=1}^{\infty} a_m \cos(mw_0 t) + \sum_{m=1}^{\infty} \sin(mw_0 t), \qquad w_0 = \frac{2\pi}{T_0} \\ a_0 &= \frac{1}{T_0} \int_{t0}^{t0+T0} x(t)dt \\ a_m &= \frac{2}{T_0} \int_{t0}^{t0+T0} x(t)\cos(mw_0 t)dt \qquad m = 1,2,3,... \\ b_m &= \frac{2}{T_0} \int_{t0}^{t0+T0} x(t)\sin(mw_0 t)dt \qquad m = 1,2,3,... \end{aligned} \tag{24}$$

where w_0 is known as the fundamental frequency of the signal. The fundamental frequency is the lowest frequency in a harmonic series and it is considered to be the part of the signal that carries the most information.

In an alternate approach, the Fourier series can be expressed as follow according to the statement above:

$$x(t) = \frac{1}{2} + \sum_{m=1}^{\infty} \cos(m\pi t + \theta_m) \qquad m = impar$$

$$donde \tag{25}$$

$$\theta_m = \begin{cases} -\pi, & m = 3,7,11,... \\ 0, & m = 1,5,9,... \end{cases}$$

where: θ_m are the phase coefficients and A_m are the amplitude coefficients.

For practical purposes, this sum is restricted to a few of its terms lowering the approximation's precision. There are different methods available that can let us obtain the coefficients and the constants in the equation above. This type of analysis is quite complex and time consuming for analog signals through time x(t). In order to represent a periodic signal by means of a Fourier analysis, we use the Fast Fourier Transform (FFT) or the Discrete Fourier Transform (DFT). Through these algorithms we can obtain a spectral analysis where we can observe the different frequencies that compose the sound signals.

Definition 7: Fourier Transform. The Fourier transform is a reversible integral transform of one function into another. The second function, called the Fourier transform, gives the coefficients of sinusoidal basis functions that can be recombined to obtain the original function, and which serves as the basic feature extraction operation for this work. The recombination of the sinusoidal basis functions is called the *inverse Fourier transform*. Both are given by the integrals below:

Fourier Transform:

$$S(f) = \int_{-\infty}^{\infty} s(t) e^{-j2\pi ft} dt \tag{26}$$

Inverse Fourier Transform:

$$s(t) = \int_{-\infty}^{\infty} S(f) e^{j2\pi ft} df \tag{27}$$

Definition 8: Discrete Fourier Transform (DFT). The discrete Fourier transform also known as the finite Fourier transform, is a technique widely used in signal processing to analyze the frequencies contained in a sample signal to solve partial differential equations and to perform operations such as convolutions.

The DFT can be practically computed by using a FFT since the complexity of the DFT is $O(N^2)$ while a simpler approach FFT has lower complexity of $O(N \log N)$. DFT or simple the Fourier transform of a discrete time sequence x[n] is a representation of the sequence in terms of the complex exponential sequence $\{e^{-jwn}\}$ where *w* is the real frequency variable. The DFT representation of a sequence, if it exists, is unique and the origin al sequence can be computed from its DFT by an inverse transform operation [22].

The discrete-time Fourier transform $X(e^{jw})$ of a sequence x[n] is defined by:

$$X(e^{jw}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \tag{28}$$

The simplest relation between a finite-length sequence x[n], defined for $0 \le n \le N-1$, and its DFT $X(e^{jw})$ is obtained by uniformly sampling $X(e^{jw})$ on the ω axis between $0 \le \omega \le 2\pi$ at $\omega_k = 2\pi k / N$, $0 \le n \le N-1$ resulting in:

$$Xk = \sum_{n=0}^{N-1} x[n]e^{-(j2\pi/N)kn},\ 0 \le n \le N-1 \tag{29}$$

The inverse discrete Fourier transform (IDFT) is used to obtain the original signal when applied to the DFT and is given by:

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]e^{(j2\pi/N)kn}, 0 \le n \le N-1 \tag{30}$$

Definition 9: Fast Fourier Transform (FFT). The basic idea behind all fast algorithms for computing the discrete Fourier transform (DFT), commonly called the *fast Fourier transform* (FFT) algorithms, is to decompose successively the N-point DFT computation into computations of smaller-size DFTs and to take advantage of the periodicity and symmetry of the complex number:

$$W_N^{kn} = e^{(-j2\pi/N)kn} \tag{31}$$

Such decomposition if properly carried out, can result in a significant savings in the computational complexity [22].

8 References:

[1] Anderson S.E., Dave A.S., Margoliash D., "Template-based automatic recognition of birdsong syllables from continuous recordings" *Journal of the Acoustical Society of America* (1996) 100:1209-1219.

[2] Avesound Project, http://www.acoustics.hut.fi/research/avesound/avesound.html

[3] Bard, S. *et. al.;* "Vocal Distinctiveness and Response to Conspecific Playback in the Spotted Antbird, a Neotropical Suboscine". *The Cooper Ornithological Society, The Condor* 104:387-394, 2002.

[4] Brainard M.S., Doupe A.J., "What songbirds teach us about learning", *Nature* Vol 417, pg 351.

[5] Burden & J.D. Faires, *Numerical Analysis*, 6th Ed., Brooks/Cole, Pacific Grove, CA, 1997.

[6] Catchpole, C.K. and Slater, P.J.B, *Bird Song - Biological themes and variations.* New edition, London: Cambridge University Press, October 30, 2003. 256p.

[7] Cornell Lab of Ornithology – Macaulay Library, http://birds.cornell.edu/MacaulayLibrary/

[8] Duda R.O., Hart, P. E., Stork, D.G., *"Pattern Classification"*, 2nd Edition, John Wiley & Sons, USA, 2001, 653 p.

[9] Escobar I., Vilches E.,Vallejo E., Taylor C., "Acoustic Bird Species Recognition: Advantages of Data Mining over Hidden Markov Models", 2006 manuscript.

[10] Fagerlund S., Härmä A., ``Parametrization of inharmonic bird sounds for automatic recognition" *13th European Signal Processing Conference (EUSIPCO 2005)*, (Antalya, Turkey), 4-8 September, 2005

[11] Gentner, T. Q., Fenn, K. M., Margoliash D., Nusbaum, H.C. (2006) "Recursive syntactic pattern learning by songbirds". *Nature,* 440:1204 --1207.

[12] George, B., Smith, M. J. T., "Speech analysis/synthesis and modification using analysis-by-synthesis/overlap-add sinusoidal model", *IEEE Trans. Speech Audio Processing, vol. 5, no. 5*, pp. 389-406, September 1997.

[13] Harma, A.; "Automatic identification of bird species bases on sinusoidal modeling of syllables". *ICASSP'03*, editor *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5444-5548. IEEE, 2003.

[14] Härmä, A., Somervuo P., *"*Classification of the Harmonic Structure in Bird Vocalization*"*, *IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP 2004)*, (Montreal,Canada), 17-21 May, 2004.

[15] Hastie T., Tibshirani R., Friedman J. "*The elements of statistical learning: Data mining, inference, and prediction"*. New York, 2001. Springer.

[16] Kohonen T.,"Self-Organized formation of topologically correct feature maps", *Biological Cybernetics, vol. 43*, pp. 59-69, 1982.

[17] Kroodsma, Donald, *"The Singing Life of Birds: The Art and Science of Listening to Birds"*, Houghton Miffling, Boston, USA, 2005

[18] Lindner, D.K., *"Introduction to Signal and Systems"*, McGraw Hill Intl., 1999.

[19] Mainwaring, A. *et al*.; "Wireless Sensor Networks for Habitat Monitoring". *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications,* September 2000

[20] Marcus, G. F., (2006) Startling starlings. *Nature, 440*:1117--1118.

[21] Matlab, version 7.0.0.19920 (R14), http://www.mathworks.com, by The Mathworks Inc.

[22] Mitra, S.; *Digital Signal Processing: A Computer Based Approach*. New York, NY: Second Edition, McGraw-Hill Irwin. 866 p.

[23] Nelson, D. "The importance of Invariant and Distinctive Features in Species Recognition of Bird Song", *The Cooper Ornithological Society,* The Condor 91:120-130, 1989.

[24] Okanoya, K.; "The Bengalese Finch, A Window on the Behavioral Neurobiology of Birdsong Syntax", *Annals of the New York Academy of Sciences*, Volume 1016 Page 724, June 2004.

[25] Rabiner, L. R., and Juang, B. H. *Fundamentals of Speech Recognition,* Prentice-Hall, Englewood Cliffs, NJ, 1993.

[26] Raven, version 1.2, http://birds.cornell.edu/brp/Raven/RavenFullVersion.html by Bioacoustics Research Program - Cornell Lab of Ornithology, (10 de mayo 2005).

[27] Russel, S.; Norvig, P.; *Artificial Intelligence: A Modern Approach*. Second Edition, Prentice Hall, Englewood Cliffs, New Yersey, 2003. 1132 p.

[28] Sasahara, K. and Ikegami, T.: "Coevolution of Birdsong Grammar without Imitation", *ECAL 2003, Advances in Artificial Life*. 7th European Conference (2003) LNAI. 2801. Vol

[29] Sasahara, K. and Ikegami, T.: "Song Grammars as Complex Sexual Displays". *Artificial Life IX. Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems*. (2004) The MIT Press.

[30] Sibley, D. A., *The Sibley Guide to Bird Life & Behaviour*, First Edition, New York, National Adubon Society, 2001, 587p.

[31] Sipser, Michael, *"Introduction to the Theory of Computation"*, PWS Publishing Company, Boston, USA, 1997.

[32] SOM Toolbox, http://www.cis.hut.fi/projects/somtoolbox/

[33] Somervuo P., Härmä A., "Bird Song Recognition *Based on* Syllable Pair Histograms", in *IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP 2004)*, (Montreal,Canada), 17-21 May, 2004

[34] Somervuo P., Härmä A., ``Analyzing bird song syllables on the self-organizing map," *Proc. of Workshop on Self-Organizing Maps (WSOM '03)*, (Hibikino, Japan), September 2003.

[35] Sound Ruler, version 0.941, http://soundruler.sourceforge.net/, by Marcos Gridi Papp

[36] Sudkamp, Thomas A., *"An Introduction to the Theory of Computer Science: Languages and Machines"*, 3rd Edition, Addison Wesley, Boston USA, 2005.

[37] Suryavanshi B.S, Shiri N., S.P. Mudur. "Incremental Relational Fuzzy Subtractive Clustering for Dynamic Web Usage Profiling", *WEBKDD Workshop on Taming Evolving, Expanding and Multi-faceted Web Clickstreams*, Chicago, Illinois, USA, August 21, 2005.

[38] Szewczyk, R. *et al*. "Habitat Monitoring with Sensor Networks", *Communications of the ACM*, Vol. 47, No. 6, p. 34-40, June 2004.

[39] Taylor, C. *et al.*; "*Sensor Arrays for Acoustic Monitoring of Bird Behavior and Diversity*", project proposal submitted to National Science Foundation., http://www.nsf.gov/awardsearch/showAward.do?AwardNumber=0410438

[40] Taylor C., *"Taylor Lab EEB, UCLA"*, http://taylor0.biology.ucla.edu

[41] Teal T.K., Taylor C.: "Effects of Compression on Language Evolution" *Artificial Life 6*: 129-143, 2000.

[42] Tchernichovski, Ofer, Mitra, Partha, P, *"Sound Analysis Pro"*, http://ofer.sci.ccny.cuny.edu/index.html

[43] Todt, D., "From birdsong to speech: a plea for comparative approaches", *Annals of the Brazilian Academy of Sciences*, Manuscript, 2004. Pg 201-208.

[44] Vesanto, J., Himberg, J., Alhoniemi, E., Parhankangas, J., "SOM Toolbox for Matlab 5", Helsinki University of Technology, Report A57, 2000.

[45] Vilches Erika, *"Application and Comparison of Artificial Intelligence and Statistics Methodologies for Acoustic Bird Species Recongition"*, Master Thesis Document, ITESM-CEM, August 2006.

[46] Vilches Erika, Escobar Ivan A., Vallejo Edgar E., Taylor Charles E, "Data Mining Applied to Acoustic Bird Species Recognition*"*, *ICPR, pp. 400-403 18th International Conference on Pattern Recognition (ICPR'06)* 2006.

[47] Virtanen, T., Klapuri A., "Separation of harmonic sound sources using sinusoidal modeling*," International Conference on Acoustics, Speech, and Signal Processing, vol. 2*. Istanbul, Turkey: IEEE ICASSP, 2000, pp. 765--768.

[48] Weka, version 3, http://www.cs.waikato.ac.nz/ml/weka/, by Ian H. Witten and Eibe Frank

[49] Wilde, M.; Menon, V.; *Bird call recognition using Hidden Markov Models. Technical report*. EECS Department Tulane University, 2003.

[50] Witten, I.; Frank, E.; *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. 1st edition, San Francisco, California, Morgan Kaufmann, October 11, 1999. 416 p.

[51] Wikipedia, http://en.wikipedia.org