HIDDEN MARKOV MODEL BASED ANIMAL ACOUSTIC CENSUSING: LEARNING FROM SPEECH PROCESSING TECHNOLOGY

by

C. Kuntoro Adi, M.A., M.S.

A Dissertation submitted to the Faculty of the
Graduate School, Marquette University,
in Partial Fulfillment of
the Requirements for
the Degree of
Doctor of Philosophy

Milwaukee, Wisconsin

May, 2008

Marquette University

This is to certify that we have examined
this copy of the
dissertation by

C. Kuntoro Adi, M.A., M.S.

and have found that it is complete
and satisfactory in all respects.

This dissertation has been approved by:

____
Michael T. Johnson, Ph.D., P.E.
Dissertation Director, Department of Electrical and Computer Engineering

____
Craig A. Struble, Ph.D.
Committee Member

____
Richard J. Povinelli, Ph.D., P.E.
Committee Member

____
Tomasz S. Osiejuk, Ph.D.
Committee Member

____
Xin Feng, Ph.D.
Committee Member

Approved on

April 28, 2008

ABSTRACT

Individually distinct acoustic features have been observed in a wide range of vocally active animal species and have been used to study animals for decades. Only a few studies, however, have attempted to examine the use of acoustic identification of individuals to assess population, either for evaluating the population structure, population abundance and density, or for assessing animal seasonal distribution and trends.

This dissertation presents an improved method to acoustically assess animal population. The integrated framework combines the advantages of supervised classification (repertoire recognition and individual animal identification), unsupervised classification (repertoire clustering and individual clustering) and the mark-recapture approach of abundance estimation, either for population structure assessment or population abundance estimate. The underlying algorithm is based on clustering of Hidden Markov Models (HMMs), commonly used in the signal processing and automatic speech recognition community for speaker identification, also referred to as voiceprinting.

A comparative study of wild and captive beluga, *Delphinapterus leucas*, repertoires shows the reliability of the approach to assess the acoustic characteristics (similarity, dissimilarity) of the established social groups. The results demonstrate the feasibility of the method to assess, to track, and to monitor the beluga whale population for potential conservation use.

For the censusing task, the method is able to estimate animal population using three possible scenarios. Scenario 1, assuming availability of training data from a specific

species with call-type labels and speaker labels, the method estimates total population. Scenario 2, with availability of training data with only call-type labels but no individual identities, the proposed method is able to perform local population estimation. Scenario 3 with availability of a few call-type examples, but no full training set on individual identities, the method is able to perform local population estimation.

The experiments performed over the Norwegian ortolan bunting, *Emberiza hortulana*, data set show the feasibility and effectiveness of the method in estimating ortolan bunting population abundance.

ACKNOWLEDGMENTS

This dissertation materialized as the result of a number of discussions, valuable direction, and support of numerous colleagues and friends. I would like to thank my advisor, Dr. Michael T. Johnson, for this wonderful opportunity, for his advice, encouragement, and support throughout my Ph.D study. Thanks for always having time for answering questions, and keeping me balanced by regular individual and research group meetings. Thank you to my committee for showing that the research is interesting and important, and giving insightful comments.

I thank all the current and former members of the Speech Research Group: Marek Trawicki, Jidong Tao, Ren Yao, Jianglin Wang, Kevin Indrebo, Patrick Clemins, and Xi Li for inspiring discussion and suggestion.

I am also grateful for the Jesuits in Marquette University community and Arrupe House community for their constant care, understanding, and valuable questions in my pursuit of knowledge. I would like to thank especially Michael Kolb, S.J. for proof reading the entire document.

Finally, I want to thank the Indonesian community in Milwaukee for their trust, love, inspiration, and friendship. Thanks to Romo Yulius Sunardi SCJ for wonderful and enriching times together.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1. Background and motivation

Studies in bird vocalizations within a single species of bird suggest that differences in the songs of individual birds are distinguishable. Birds use vocal differences to identify other members of their species nearby and to identify individual birds in their immediate vicinity. They have been shown to use vocalizations in recognizing their mates, their parent, and to differentiate between neighbors and strangers (Holschuch, 2004).

A wide variety of approaches has been used to count and monitor bird populations within a species (Peake, McGregor 2001). Most of those approaches do not require the identification of individual birds. Methods that involve the ability to identify individual animals are important for providing ecological information that can not be obtained in other ways. Such information generally falls into three categories, namely: (a) managing census error, (b) estimating demographic information such as age, morbidity, the time of migration, and (c) detecting individual behavioral differences (McGregor, Peake, 1998).

Individually distinct acoustic features have been observed in a wide range of vocally active animal species, for example: cetaceans (Janik *et al*., 1994), bats (Master *et al*., 1995), and primates (Butynski *et al*., 1992). Within birds, the presence of vocal individuality has been shown in the European Bitterns and Black-throated Divers (Gilbert *et al*., 1994), American Woodcock (Beightol and Samuel, 1973), Australian Kingfishers (Saunders and Wooller, 1988), and Tawny Owls (Galeotti and Pavan, 1991).

Techniques for identifying individual animals by the variations in vocalization fall into two broad categories: qualitative and quantitative (Mc.Gregor *et al*., 2000). The qualitative approach involves comparison of spectrograms by human observers. Much of the popularity of this approach can be traced to its conceptual simplicity and ease of operation (Wakita, 1976). However, while manual inspection and labeling of the sound spectrogram allow measurements to be made simply and with reasonable accuracy, the variables measured are too few to characterize the spectral content and patterning of a signal. So, qualitative methods are typically followed by more rigorous quantitative methods that employ a detailed measurement of the frequency and temporal parameters of the vocalizations.

Several different quantitative approaches for analyzing vocal individuality exist. Otter (1996) was able to differentiate individual birds through a series of nested ANOVA. Holschuch (2004) did the same using Discriminant Function Analysis (DFA). Current research on bird vocalizations has begun to use more advanced methods to perform identification. Kogan (1997) evaluated two methods, namely, dynamic time warping (DTW) and hidden Markov models (HMMs) for automated recognition of zebra finche and indigo bunting song-units from continuous readings. Ito *et al*. (1996) used dynamic programming (DP) matching to classify budgerigar contact calls into natural groups. Meanwhile, Harma (2003) compared simple sinusoidal representation of syllables to identify the willow warbler bird species. Somervuo (2004) investigated the possibility of common chaffinch and great tit bird species recognition based on the syllable pair histogram of the song. In marine mammals, Buck and Tyack (1993) utilized DTW to classify 15 dolphin signature whistles into 5 groups. Later Brown *et al*. (2007) used

DTW to measure the dissimilarity of killer whale calls and to classify the calls using frequency contours of their biphonic vocalizations.

Acoustics has been used to study animals for decades. For rare or elusive species that are hard to monitor or to mark visually, the possibility of recognizing individuals by their vocalizations may provide a useful census tool (e.g. Saunders and Wooller, 1998; Gilbert *et al*., 1994; Jones and Smith, 1977). Only a few researchers, however, have attempted to examine the use of vocalizations to assess populations. The term assessment is usually used to describe the process of evaluating the status of population relative to some management goal. This involves studies of the population structure, abundance and density, seasonal distribution and trends, and the evaluation of human-made noise impacts on the animals (Mellinger and Barlow, 2003).

In a few studies the feasibility of using vocal individuality (vocalizations) to monitor habitat quality has been demonstrated. Peake and McGregor (2001) employed a statistical Pearson-correlation approach to identify corncrake vocal individuality and to estimate numbers of individuals in species. Holschuh (2004) used discriminant function analysis to explore vocal individuality of the saw-whet owl to monitor its habitat quality. The use of vocal individuality in a census must be capable of discriminating between unknown groups and identifying new individuals entering a population. McGregor (2000, 2001) suspected that both features are problematic for current techniques such as discriminant function analysis (McGregor, 2000, 2001). The use of discriminant function analysis can only classify vocalizations of known individuals. It is not able to accommodate vocalizations of new individuals.

Terry and McGregor (2002) suggest a different method to monitor and census male corncrake species. They employ three different neural network models, namely, a backpropagation and probabilistic network to re-identify the members of the known population (monitoring task) and a Kohonen network to count a population of unknown size (census task). Neural networks have been used in a wide range of discrimination tasks. Terry and McGregor see neural networks as having potential in monitoring and censusing because (a) they can work with data that cannot be separated linearly, (b) the learning procedure that creates the network allows generalization to unknown data.

In studies of cetaceans, the best examples of the use of vocalizations in assessment are the studies of sperm whale population (Barlow and Taylor, 1998), the humpback whales in the Caribbean (Garrison *et al*., 2003), and harbor porpoises in the Northwest Atlantic (Palka, 2003), where combined visual and acoustic methods have significantly improved the population estimate.

The objective of this dissertation, therefore, is to develop an improved method to assess population (namely, animal population structure and animal abundance) based upon animal vocalizations. The suggested framework is based on Hidden Markov Models (HMMs) commonly used in the signal processing and automatic speech recognition community. Previous and current studies show the feasibility of the HMM – based method to automatically classify ortolan bunting call-types, to identify individual birds (Adi, Johnson, 2004, 2006; Trawicki *et al*., 2005), to classify African elephant vocalizations (Clemins, 2005) and to cluster Beluga repertoires (Clemins, 2005; Adi *et al*., 2008). This dissertation proposes an integrated method of the supervised classification task (repertoire recognition and individual animal identification), unsupervised classification task (repertoire clustering and individual animal clustering) and the mark-recapture approach of abundance estimation, either for population structure assessment or population abundance estimate.

The method uses less effort and cost, is less time consuming and is more accurate. The most compelling reason for using individually distinctive vocalizations as a census tool is that the technique causes minimal disturbance and does not require the capture and handling of the animals. Thus it will be useful for species that are secretive, sensitive to disturbance, and which cannot be readily caught or observed. These are often species of considerable conservation interest.

1.2. Contribution and significance

This research has applications in many areas. These include bioacoustics, bird and marine mammal communication, behavior and conservation, audio signal processing, and machine learning. The main contribution of this study is the development of robust models that will improve our understanding on bird and marine mammal communication and behavior. It will also afford an easier way for humans to monitor and census cetaceans and bird populations. For the field of machine learning, the research contributes to the improvement upon existing HMM-based clustering by incorporating an initialization method to build initial clusters, adding dissimilarity analysis and deltaBIC analysis to estimate the number of clusters in a data set, using a resampling dissimilarity computation to assess consistency of the clustering results.

Though the animals studied in this research are the ortolan bunting and beluga whale, the methods and approaches used here can be expanded and adapted for other animal species.

The results of this study will be disseminated to the broader research community by means of journals articles and conference papers. Due to the multi-disciplinary nature of the study, these research results will be of interest to people who work in the areas of animal behavior and conservation, speech signal processing and machine learning.

1.3. Dissertation overview

This section describes the organization of the dissertation. Chapter Two provides the necessary background knowledge in the field of speech processing, bioacoustics and machine learning.

Chapter Three discusses feature selection methods to discover the features that can be employed for beluga whale and ortolan bunting repertoire analysis and for bird abundance estimation. It addresses the question of which among the features, or combination of features, are fit for the repertoire recognition task and which are robust for the individual identification task.

Chapter Four applies the HMM-based unsupervised clustering framework to assess beluga whale population structure, in order to determine the relationship between established beluga social groups as indicated by their vocalizations.

Chapter Five applies the proposed HMM-based framework to the tasks of supervised classification and unsupervised clustering to estimate the number of birds in a population.

Chapter Six concludes the dissertation, summarizes the main contributions of the dissertation and underlines several potential future directions.

CHAPTER 2

BACKGROUND AND RELATED WORKS

2.1. Introduction

This chapter provides necessary background knowledge in the field of speech processing, machine learning, and bioacoustics.

The three tasks in speech processing that are most closely associated with this research are speech recognition, speaker identification, and speaker diarization. Section two in this chapter gives an overview of these areas, focusing on the use of hidden Markov models (HMMs), to determine word sequence from statistical measures of the similarity or dissimilarity between reference examples and test data. In addition, the use of Gaussian mixture models (GMMs) for speaker recognition is explored.

Section three presents model-based unsupervised clustering tasks and reviews two different clustering methods, namely, HMM-based *k*-model clustering and HMM-based hierarchical agglomerative clustering. The section addresses methods to estimate the number of clusters *K* and to assess the clustering results as well.

Section four focuses on feature extraction approaches such as Greenwood function cepstral coefficients, pitch tracking, delta and acceleration computation and some cepstral normalizations.

Section five gives an overview of bioacoustics by addressing the tasks associated with call-type recognition, individual animal identification, and animal abundance estimation, using approaches such as cross-correlation, dynamic time warping and self-organizing map (SOM). Section six concludes the discussion with a short summary.

2.2. Automatic speech and speaker recognition, and speaker diarization

The term automatic speech recognition refers to methods for developing and implementing algorithms on a computer in order to recognize the linguistic content of a spoken utterance. Speaker recognition refers to method for identifying the person who speaks the utterance (Lee *et al.*, 1996). Speaker diarization is the task of marking and categorizing the speakers within a spoken document (Tranter and Reynolds, 2006).

In recent decades research in the field of automatic speech processing has made significant improvement due to the advances in signal processing, algorithms, architecture, and hardware. These includes the adoption of a statistical pattern recognition paradigm to analyze the problem, the use of the hidden Markov modeling framework to characterize both temporal and spectral variations in the speech signal, and the use of dynamic programming based search methods to find the best word sequence in the lexical network that corresponds to the spoken utterance.

Speech processing systems have been developed for a wide variety of applications, ranging from small vocabulary keyword recognition over dial-up line to large vocabulary speech dictation and spontaneous speech understanding. The following is a brief description each of the task: automatic speech recognition, speaker recognition, and speaker diarization.

2.2.1. Automatic speech recognition

The speech signal is a complex signal that is not easy to process. The physical production system of speech signals differs from one person to another. The observed time-series signal is different for every utterance, even when produced by the same person and even for multiple utterances with the same sequence of words. There are ranges of plausible variants of each speech sound, some which are more likely than others. The extent and type of possible variation, both in time scale and in frequency spectrum, will be different for different sounds. In addition to the vast differences across different speakers, the speech signal is influenced by the transducer used to capture the signal, the channel used to transmit the signal, and the speaking environment that can add noise or change the way the signal is produced in a noisy environment (Rabiner *et al*., 1996).

Automatic speech recognition (ASR) by machine is difficult is due to inherent signal variabilities. There are at least four types of variability in speech signals (Zue *et al*., 1997). First, phonetic variability, in that the acoustic realizations of phonemes (the smallest sound units of which word are composed) are highly dependent on the context in which they appear. Second, acoustic variability can result from changes in the environment as well as in the position and characteristics of the transducer. Third, within-speaker variabilities can result from changes in the speaker's physical and emotional state, speaking rate, or voice quality. Finally, differences in linguistic background, dialect, vocal tract size and shape can contribute to across-speaker variabilities.

Speech recognition systems attempt to model the sources of variability described above in several ways. At the level of signal representation, researchers have developed representations that emphasize perceptually important speaker-independent features of the signal and de-emphasize speaker dependent characteristics. At the acoustic-phonetic level, speaker variability is modeled using statistical techniques applied to large amounts of data. Effects of linguistic context at the acoustic phonetic level are typically handled by training separate models for phonemes in different contexts (referred to as context-dependent acoustic modeling). Word level variability can be handled by allowing alternate pronunciations of words in representations known as pronunciations networks.

A successful approach to automatic speech recognition is to treat the speech signal as a stochastic pattern and to analyze the signal using statistical pattern recognition. Speech recognition systems treat the acoustic input as if it were a noisy version of the source sentence. In order to decode this noisy sentence one needs to consider all possible sentences and to choose one which has the highest probability of generating the sentence. Speech recognition is therefore formulated as a maximum *a posteriori* (MAP) decoding problem (Jelinek, 1998).

The noisy channel in Figure 2.1 is a model that jointly characterizes the speech production system, the speaker variability, and the speaking environment.

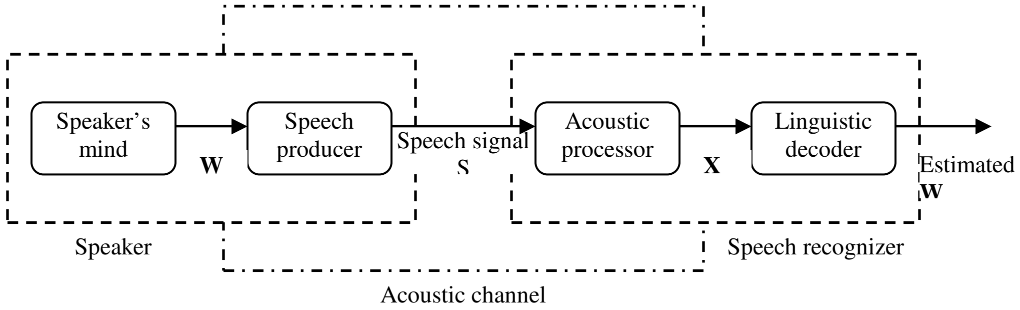

Figure 2.1. Source-channel model of speech recognition (after Jelinek, 1998)

To simplify the problem, in general the speech signal **S** is first parametrically represented as a sequence of acoustic vectors **X**. Let **W** = $w_1, w_2, \dots w_n$, where $w_i \in V$ denotes a string of n words, each belonging to a fixed and known vocabulary V.

The decoding problem of speech recognition systems are defined as

$$\hat{\mathrm{W}} = \mathrm{argmax}_{W \in V} P(\mathbf{W}|\mathbf{X}). \tag{2.1}$$

The recognizer will pick the most likely word **W** given the observed acoustic **X**. Bayes' formula allows us to write the right-hand side probability of equation (2.1) as

$$P(\mathbf{W} \mid \mathbf{X}) = \frac{P(\mathbf{X} \mid \mathbf{W})P(\mathbf{W})}{P(\mathbf{X})}, \tag{2.2}$$

where $P(\mathbf{X}|\mathbf{W})$ is the conditional probability of the acoustic vector sequence **X**, given a particular sequence of word **W**, $P(\mathbf{W})$ is the *a priori* probability of generating the sequence of word **W**; and $P(\mathbf{X})$ is the average probability that **X** will be observed. Since equation (2.2) maximizes over all possible words, one has to compute $P(\mathbf{X}|\mathbf{W})P(\mathbf{W})/P(\mathbf{X})$ for each word in the vocabulary. $P(\mathbf{X})$, however, doesn't change for each word. For each potential word one still examines the same observation **A** which have the same probability $P(\mathbf{X})$. Equation (2.1) and (2.2), therefore, can be simplified as follows

$$\hat{\mathbf{W}} = \arg\max_{\mathrm{W} \in V} \frac{P(\mathbf{X} \mid \mathbf{W})P(\mathbf{W})}{P(\mathbf{X})} = \arg\max_{W \in V} P(\mathbf{X} \mid \mathbf{W})P(\mathbf{W})\,. \tag{2.3}$$

Thus, the most probable word **W** given some observation sequence **X** can be computed by taking the product of two probabilities for each word, and choosing the word for which this product is greatest. The first term $P(\mathbf{X}|\mathbf{W})$, the observation likelihood, is often referred to as an acoustic model; and the second term $P(\mathbf{W})$, the prior probability, is known as a language model.

2.2.1.1. Principles of speech recognition

Figure 2.2 shows the basic structure of an automatic speech recognition system.

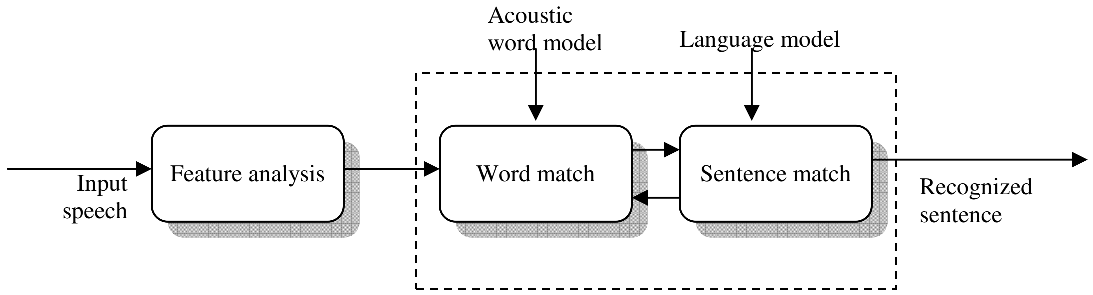

Figure 2.2. Block diagram of call-type recognition, classification (after Rabiner, 1996)

Feature analysis provides the acoustic feature vectors of the input signal. The acoustic word match component determines which words are most likely spoken by evaluating the similarity between input feature vectors and a set of acoustic word models for all words in the vocabulary. The sentence-match uses a model of syntax and semantics (language model) to determine the most likely sequence of words. Recognition is made by considering all likely word sequences and choosing the one with the best acoustic matching score.

Feature analysis

The focus of feature analysis is to parameterize the speech into a sequence of feature vectors **X** that contains the relevant information about the sounds within the utterance. For speech, the analysis is typically done over a fixed length frame of analysis window. A window of length 30 ms and overlap 10 ms might be used as the input to feature analysis. The analysis often includes short time spectral features that incorporate cepstral features along with its first and second time derivatives (Furui, 1986). Fourier analysis is still the most widely used method for extracting spectral features for speech recognition (Rabiner *et al.*, 1996). Sometimes, non-uniform frequency scales are employed in spectral analysis to provide mel-frequency or bark-scale spectral feature sets. A full discussion of feature analysis will be presented in section four.

Acoustic modeling

As shown in equation (2.3), the system needs to be able to determine the value $P(\mathbf{X}|\mathbf{W})$ – the probability of observation data **X** given a specific word sequence **W**. To compute $P(\mathbf{X}|\mathbf{W})$ one needs a statistical acoustic model of the speaker's interaction with the acoustic processor. The modeling process involves the way the speaker pronounces the word **W**, the microphone placement and the acoustic processing performed by the front end.

The usual acoustic model employed in speech recognizers, hidden Markov models, will be discussed in the next section. Other models are possible, for instance those based on artificial neural networks (Furui, 1981), or on dynamic time warping (Rabiner, Levinson, 1981). Speaker recognition systems often use a more direct statistical model of each speaker, the Gaussian mixture models.

Language modeling

Early ASR systems used only acoustic information to evaluate text hypotheses. It was quickly found that by incorporating language knowledge significantly raised ASR accuracy (O'Saughnessy, 2000). The grammar or structure of permitted word sequence increases ASR accuracy by eliminating candidate word sequences that are not legal under the grammar (Seneff, 1992).

Equation (2.3) is used to compute the probability of a complete string of words (represented either as $w_1, \ldots w_\mathrm{n}$). If one considers each word occurring in its correct location as an independent event, one might represent the probability as follows

$$P(w_1^n) = P(w_1)P(w_2 \mid w_1)P(w_3 \mid w_1^2)...P(w_n \mid w_1^{n-1})$$

$$= \prod_{k=1}^{n} P(w_k \mid w_1^{k-1}) \,. \tag{2.4}$$

Equation (2.4) can be simplified using approximation of the probability of a word given all previous words. The bigram model approximates the probability of the proceeding word by looking one word into the past $P(w_n \mid w_{n-1})$. One can generalize the bigram into the trigram (which looks two words into the past) and the *N*-gram (which looks *N*-1 words into the past) models.

2.2.1.2. Hidden Markov Models

The most widely used and the most successful modeling approach to speech recognition is the use of Hidden Markov Models (HMMs). The HMM is a statistical model that uses a finite number of states and associated state transitions to jointly model the temporal and spectral variation of signals. The time varying nature of spoken utterances is accommodated through an underlying Markov process. Statistical processes associated with the model states define output probability distributions and so encompass the variability which occurs both between and within speakers when producing equivalent speech sounds.

The following section offers an overview of HMMs and presents methods for evaluating, decoding and learning. For more detail discussion on HMMs, the reader may refer to Rabiner (1989) and Rabiner, Juang (1993).

HMM definition

Let S_1, S_2…., S_n be a sequence of random variables. Bayes' theorem results in

$$P(S_1, S_2 ... S_n) = \prod_{i=1}^{n} P(S_i \mid S_1, S_2, ..., S_{i-1}) \,. \tag{2.5}$$

The random variables form a Markov chain if $P(S_i \mid S_1, S_2, ..., S_{i-1}) = P(S_i \mid S_{i-1})$ for all value of *i*. As a consequence,

$$P(S_1, S_2, ..., S_n) = \prod_{i=1}^{n} P(S_i \mid S_{i-1}) \,. \tag{2.6}$$

The above random process thus incorporates a minimum amount of memory since the value at time *t* depends only on the value at the previous time. Furthermore, the Markov chain is time invariant or homogenous if regardless of the value of the time index *i*

$$P(S_j = s' \mid S_j = s) = p(s' \mid s) = a_{ij} \text{ for all } s',\, s \in S \tag{2.7}$$

where a_{ij} is the transition function that satisfies the condition

$$\sum_i a_{ij} = 1, \qquad a_{ij} \geq 0 \tag{2.8}$$

A Markov chain is a finite state process with transitions between states specified by the transition probabilities a_{ij}.

A Markov chain is useful for computing the probability of a sequence of observed events. In many cases, people are interested in events that are not observable. In speech recognition, for example, one sees some acoustic events and has to infer the presence of "hidden" words underlying a causal source of the events. A hidden Markov model, is an extension of a Markov chain that deals with observed events and hidden events.

An HMM represents a stochastic sequence where the states are not directly observed, but are associated with a probability density function (pdf). Each HMM state *j* has an associated observation probability distribution $b_j(x_t)$ that determines the probability of generating observation x_t at time *t*. Each pair of states *i* and *j* also has an associated transition probability a_{ij}. In speech models the entry state 1 and the exit state *N* of an *N* states HMM are non-emitting, a modification which allows for easy connection between multiple HMM sequences.

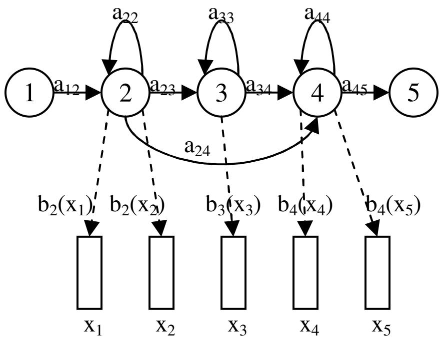

Figure 2.3. A left-to-right HMM (after Young *et al.*, 2002)

Figure 2.3 shows a left-to-right HMM where a five states model moves through the state sequence S = 1, 2, 2, 3, 4, 4, 5 in order to generate the sequence x_1 to x_5. Three of the above states are emitting states and have an output probability distribution associated with them. The transition matrix of this model has 5 rows and 5 columns, which might be as follows:

$$a_{ij} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0.3 & 0.4 & 0.3 & 0 \\ 0 & 0 & 0.3 & 0.7 & 0 \\ 0 & 0 & 0 & 0.3 & 0.7 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Each row will sum to one except for the final row that is always all zero since no transitions are allowed out of the final state.

The joint probability that *X* is generated by the HMM model *M* moving through the state sequence *S* is computed simply as the product of the transition probabilities and the output probabilities. So for the state sequence *S* in figure 2.3

$$P(X, S \mid M) = a_{12} b_2(x_1) a_{22} b_2(x_2) a_{23} b_3(x_3) ... \tag{2.9}$$

In this approach, only the observation sequence *X* is known and the underlying state sequence *S* is hidden. It is, therefore, called a Hidden Markov Model.

The sequence of states, which is the quantity of interest in speech recognition, can be observed only through the stochastic processes defined into each state. One has to know the pdfs of each state before being able to associate a likely sequence of states $S = \{s_1, \ldots, s_K\}$ to a sequence of observations $X = \{x_1, \ldots, x_K\}$.

A hidden Markov model is defined by a set of parameters (Φ):

- $\{s\}$ – a set of states that include an initial state S_I and a final state S_F
- $\{a_{ij}\}$ – the probability of taking a transition from state *i* to state *j*
- $\{b_j(k)\}$ – the probability of emitting output *k* while in state *j*.

a and *b* satisfy the following properties:

$a_{ij} \geq 0$, $b_{ij}(k) \geq 0$, $\forall i,j,k$

$$\sum_j a_{ij} = 1, \forall i$$

$$\sum_{k} b_j(k) = 1, \forall j$$

There are two assumptions in a first-order hidden Markov model. The first is the Markov assumption equation (2.6). The second assumption is the output-independence assumption. It states that the probability that a particular symbol will be emitted at time t depends only on the transition taken at that time (from state s_t to s_{t+1}), and is conditionally independent of the past.

Given the basic structure of the HMM above, there are three fundamental problems to address; the evaluation problem, the decoding problem and the learning problem (Rabiner, 1989).

1. The evaluation problem:

   Given an HMM model and a sequence of observation data, determine the likelihood that the model generates the observation.

2. The decoding problem:

   Given a model and an observation sequence, discover the most likely hidden state sequence in the model that generates the observation

3. The learning problem:

   Given a set of observations and a model, learn the HMM parameters.

The evaluation problem – the forward and backward algorithm

The evaluation problem focuses on calculation of the probability of the observation sequence $X = (X_1, X_2, \ldots X_T)$, given the HMM Φ, namely, $P(X|\Phi)$. The intuitive way to calculate the probability $P(X|\Phi)$ is to sum up the probabilities of all possible state sequence of length T

$$P(X \mid \Phi) = \sum_{allS} P(S \mid \Phi) P(X \mid S, \Phi)$$

$$= \sum_{allS} \pi_{s_1} b_{s_1}(x_1) a_{s_1 s_2} b_{s_2}(x_2) ... a_{s_{T-1} s_T} b_{s_T}(x_T) \qquad (2.10)$$

The enumeration of every possible state sequence is computationally expensive with time complexity of $O(N^T)$, where N is number of states and T is number of observation. However, one can compute $P(X|\Phi)$ in a recursive way using a dynamic programming approach, called the forward recursion method. Based on the first order Markov assumption, it is possible to compute the likelihood $P(X|\Phi)$ recursively with time complexity $O(N^2T)$.

To do this the forward algorithm defines a forward variable $\alpha_t(i)$ corresponding to

$$\alpha_t(i) = P(X_1^t, s_t = i \mid \Phi), \qquad (2.11)$$

the probability of having observed the partial sequence X_1^t (namely, x_1, x_2, …x_t) and being in state i at time t given the parameter Φ.

For an HMM with N number of states, where states 1 and N are the non-emitting initial and final states, $\alpha_t(i)$ can be computed recursively as follows:

1. Initialization

$$\alpha_1(i) = a_{1i} b_i(x_1) \qquad 1 \le i \le N$$

2. Recursion

$$\alpha_t(j) = \left[\sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} \right] b_j(x_t) \qquad 2 \le t \le N;\ 1 \le j \le N$$

3. Termination

$$P(X \mid \Phi) = \sum_{i=1}^{N} \alpha_T(i) a_{iN}$$

The forward algorithm employs a dynamic programming approach by using a table to store intermediate values as it computes the probability of the observation sequence.

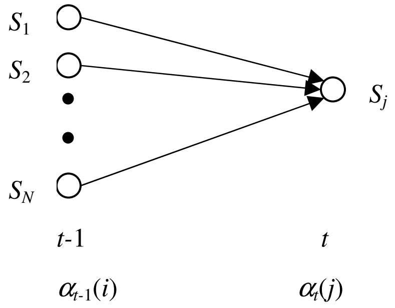

Figure 2.4. The computation of forward variable $\alpha_t(j)$

Each $\alpha_t(j)$ represents the probability of being in state *j* after seeing the first *t* observation. The value of $\alpha_t(j)$ is computed by summing over the probabilities of every path that could lead to this cell.

The backward probability is defined in a similar manner to the forward probability as

$$\beta_t(i) = P(X_{t+1}^{T} \mid s_t = i, \Phi) \tag{2.12}$$

where $\beta_t(i)$ is the probability of generating partial observation X_{t+1}^{T} (from *t*+1 to *T*) given that the HMM is in state *i* at time *t*. Similar to forward probability α, β can be computed with recursion on *t* as follows:

1. Initialization

$$\beta_T(i) = 1; \qquad 1 \le \text{i} \le N$$

2. Recursion

$$\beta_t(i) = \sum_{j=i}^{N} a_{ij} b_j(x_{t+1}) \beta_{t+1}(j); \qquad t = T\text{-}1, \ldots 1;\ 1 \le \text{i} \le N$$

3. Termination

$$P(X \mid \Phi) = \sum_{i=1}^{N} \pi_i \beta_i(1)$$

where π_i is the starting probability.

The decoding problem – the Viterbi algorithm

The forward algorithm computes the probability that an HMM generates an observation sequence by summing up the probabilities of all possible paths. However, it does not provide a state sequence. In many applications such as a speech recognition application, it is useful to associate an "optimal" sequence of states to a sequence of observations, given the parameters of a model. A reasonable optimality criterion is to choose the state sequence or path that has a maximum likelihood with respect to a given model. In other words, given an observation sequence $X = (X_1, X_2, \ldots X_T)$, one is looking for the state sequence $S = (s_1, s_2, \ldots s_T)$ that maximizes $P(S, X|\Phi)$. This sequence can be determined recursively via the Viterbi algorithm.

The Viterbi algorithm make use of two variables, $\delta_t(i)$ and $\psi_t(i)$. $\delta_t(i)$ is the highest likelihood value along a single path among all the paths ending in state i at time t:

$$\delta_t(i) = \max_{s_1,s_2,\dots,s_{t-1}} p(s_1,s_2,\dots,s_{t-1},s^t = s_i,x_1,x_2\dots x_t \mid \Phi)\,. \tag{2.13}$$

Meanwhile, $\psi_t(i)$ is a variable to keep track of the best path ending in state *i* at time *t*:

$$\psi_t(i) = \arg\max_{s_1,s_2,\dots,s_{t-1}} p(s_1,s_2,\dots,s_{t-1},s^t = s_i,x_1,x_2\dots x_t \mid \Phi)\,. \tag{2.14}$$

$\delta_t(i)$ is similar to the forward recursion $\alpha_t(i)$, only with respect to a single state sequence, and its computation is nearly identical.

For an HMM with *N* states, the procedure to find the best state sequence is as follows:

1. Initialization

$$\delta_1(i) = \pi_i b_i(x_1)\,, \qquad 2 \le i \le N\text{-}1$$

$$\psi_1(i) = 0$$

2. Recursion

$$\delta_{t+1}(j) = \max_{2\le i\le N-1}[\delta_t(i).a_{ij}].b_j(x_{t+1})\,, \qquad 1 \le t \le T\text{-}1;\ 2 \le j \le N\text{-}1$$

$$\psi_{t+1} = \arg\max_{2\le i\le N-1}[\delta_t(i).a_{ij}]\,, \qquad 1 \le t \le T\text{-}1;\ 2 \le j \le N\text{-}1$$

3. Termination

$$p^*(X \mid \Phi) = \max_{2\le i\le N-1}[\delta_T(i)]$$

$$s_T^* = \arg\max_{2\le i\le N-1}[\delta_T(i)]$$

4. Backtracking

$$S^* = \{s_1^*,\dots,s_T^*\} \quad \text{so that} \quad s_t^* = \psi_{t+1}(s_{t+1}^*)\,, \qquad t = T\text{-}1,\ T\text{-}2,\ \dots,\ 1.$$

Given an observation sequence $X = \{x_1, .. x_T\}$ and a model parameter Φ, the Viterbi algorithm delivers two useful results, namely the selection of the best path among all the possible paths in the considered model, $S^* = \{S_1^*, \ldots S_T^*\}$, and the likelihood along the best path $P(X, S^* | \Phi) = P(X | \Phi)$.

The learning problem: the Baum-Welch algorithm

The learning problem involves the optimization of the model parameters Φ = (A, B,π) to obtain the best model to represent a certain set of observations. The learning problem can be approached using an iterative procedure, the Baum-Welch algorithm.

Key aspect of estimating the model parameters is calculating state occupancy probabilities. Define $\gamma_t(i)$ - the probability of being in state S_i at time t, given the observation sequence X and the model Φ - as follows:

$$\gamma_t(i) = P(s_t = S_i | X, \Phi) \tag{2.15}$$

Equation (2.15) can be expressed in terms of the forward-backward variables as

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(X | \Phi)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i)\beta_t(i)} \tag{2.16}$$

Also define another probability function $\xi_t(i,j)$, the probability of being in state S_i at time t and going to state S_j at time t+1, given the model Φ and observation sequence X as follows:

$$\xi_t(i,j) = P(s_t = S_i, s_{t+1} = S_j \mid X, \Phi) \tag{2.17}$$

From the definition of the forward and backward variables, the above equation can be written in the form

$$\begin{aligned}\xi_t(i,j) &= \frac{\alpha_t(i)a_{ij}b_j(x_{t+1})\beta_{t+1}(j)}{P(X \mid \Phi)} \\ &= \frac{\alpha_t(i)a_{ij}b_j(x_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)a_{ij}b_j(x_{t+1})\beta_{t+1}(j)}\end{aligned} \tag{2.18}$$

The relationship between $\gamma_t(i)$ and $\xi_t(i,j)$ can be shown by summing over j, giving

$$\gamma_t(i) = \sum_{j=1}^{N}\xi_t(i,j)\,. \tag{2.19}$$

Summing $\gamma_t(i)$ over all instances and excludes the time t=T, one gets the expected number of times that state S_i is visited. Summing $\xi_t(i,j)$ over t and excluding t=T, one obtains the expected number of transitions from state S_i to state S_j. The re-estimation of the model parameters, then, proceeds as follows:

$$\hat{\pi}_i = \text{expected number of times in state } S_i \text{ at time } (t = 1) = \gamma_1(i) \tag{2.20}$$

$$\begin{aligned}\hat{a}_{ij} &= \frac{\text{expected number of transitions from state } S_i \text{ to } S_j}{\text{expected number of transitions from state } S_i} \\ &= \frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)} = \frac{\sum_{t=1}^{T-1}\alpha_t(i)a_{ij}b_j(x_{t+1})\beta_{t+1}(j)}{\sum_{t=1}^{T-1}\alpha_t(i)\beta_t(i)}\end{aligned} \tag{2.21}$$

$$\hat{b}_j(k) = \frac{\text{expected number of times in state } j \text{ and observing } x_k}{\text{expected number of times in state } j}$$

$$= \frac{\sum_{\substack{t=1 \\ x_t = x_k}}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)} = \frac{\sum_{\substack{t=1 \\ x_t = x_k}}^{T} \alpha_t(j)\beta_t(j)}{\sum_{t=1}^{T} \alpha_t(j)\beta_t(j)} \tag{2.22}$$

After the re-estimation of the model parameters, a new model $\hat{\Phi}$ which is more likely to generate observation sequence X than model Φ is obtained. This means that $P(X \mid \hat{\Phi}) > P(X \mid \Phi)$. The re-estimation process continues until it converges.

The Baum-Welch algorithm described above is an implementation of the general EM algorithm. Beginning with some initial estimate of the HMM parameters $\Phi = (A, B, \pi)$ the E (expectation) step and M (maximization) step are run alternately. In the E-step one computes the expected state occupancy count γ and the expected state transition count ξ from the earlier A and B probabilities using the forward-backward algorithm. In the M-step γ and ξ are used to recompute new A, B, and π probabilities using equations 2.20, 2.21 and 2.22.

2.2.2. Automatic speaker recognition

Speech contains many characteristics specific to each individual. These characteristics are mostly independent of the utterance and its linguistic message. Each utterance from an individual is produced by the same vocal tract. It tends to have a typical pitch range and a characteristic articulator movement associated with dialect or gender. These indicate that the speech is highly correlated with the particular individual who is speaking. Listeners, therefore, are often able to recognize speaker identity fairly quickly, even over the telephone (Gold, Morgan, 2000).

Speaker recognition by computer is the task of recognizing automatically who is speaking based upon information obtained from speech signals. Applicable speaker recognition services include voice dialing, banking over a telephone network, telephone shopping, database access services, information and reservation services, forensic application and voice login (O'Saughnessy, 2000).

Speaker recognition includes two important tasks: speaker identification and speaker verification. Speaker identification focuses on determining from which of the registered speakers a given utterance comes. Speaker verification, also known as speaker authentication is the binary classification task to accept or to reject the identity claim of a speaker. The difference between speaker identification and speaker verification is in the number of decision alternatives. In identification the number of decisions is equal to the size of population; whereas in verification there are only two decisions, accept or reject, regardless the population size.

Speaker recognition can be either text-dependent or text-independent utterances, depending on whether or not the recognition process is constrained to a predefined text or not.

This section gives an overview of the tasks and basic structure of speaker recognition systems. For more detail reviews, the reader is referred to papers by Campbell (1997) and Furui (1996).

2.2.2.1. Principles of speaker recognition

Figures 2.5 and 2.6 show the structure of a typical speaker recognition system

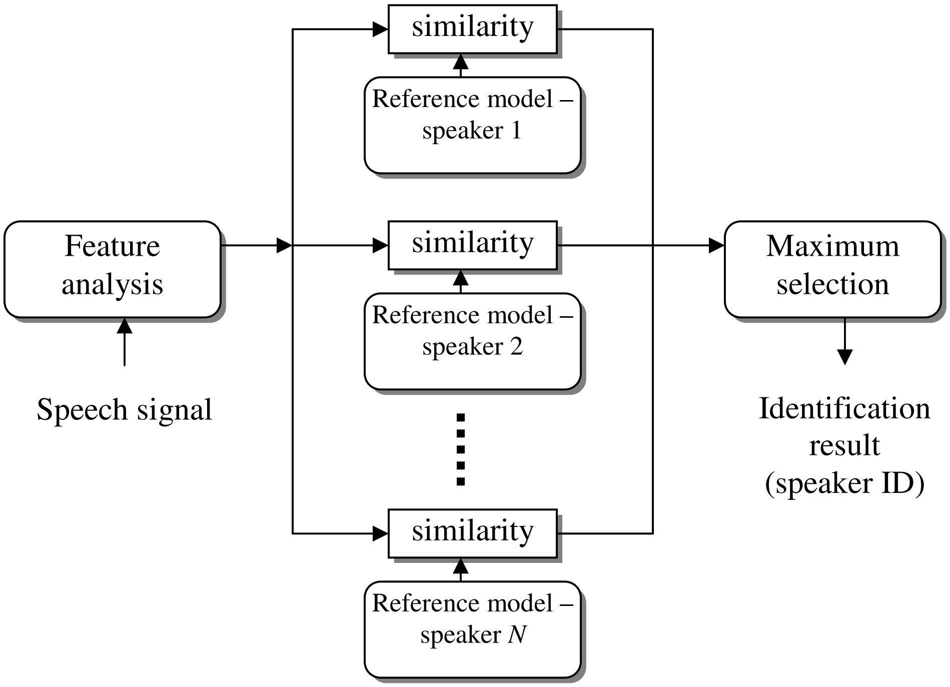
similarity
Reference model – speaker 1
Feature analysis
similarity
Reference model – speaker 2
Maximum selection
Speech signal
Identification result (speaker ID)
similarity
Reference model – speaker *N*

Figure 2.5. Basic structure of speaker identification system (after Furui, 1996)

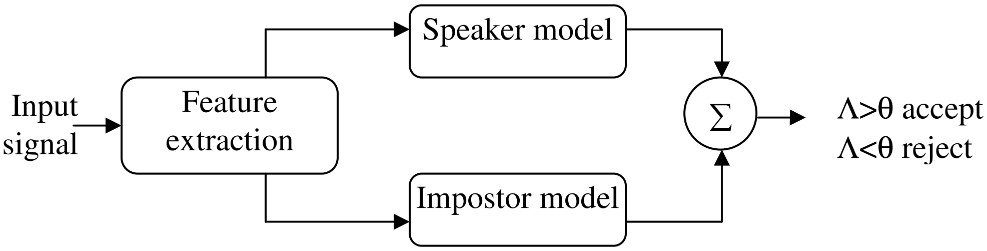

Figure 2.6. Basic structure of speaker verification system (after Reynolds, 2002)

In speaker identification, a speech signal from an unknown speaker is analyzed. The system then computes the similarity of the unknown speaker with models of known speakers. The input is identified as the speaker whose model best matches the input signal.

A speaker verification system implements a likelihood ratio test to discriminate between two hypotheses: the test speech comes from the claimed speaker or from imposter (non-claimed speaker). Features extracted from the input signal are compared to a model representing the claimed speaker obtained from a previous enrolment and to some models representing potential imposter speakers. The match score is then compared to a threshold to decide whether to accept or to reject the speaker.

The section below describes the architectural components of a speaker recognition system.

Feature extraction

In contrast with speech recognition, speaker recognition benefits from the features that are independent of the particular spoken words. Such characteristics include the average range of fundamental frequency or the overall properties of the spectral envelope (such as the average formant position over many vowels). In general, speaker recognition features are typically based on some kind of short-term spectral measure as they are in automatic speech recognition.

Speaker model

A speaker model is created during enrollment using feature vectors extracted from the input signals. There are at least three desirable characteristics of a speaker model (Reynolds, 2002), namely, (a) has a theoretical explanation so one can understand model behavior; (b) generalizable to new data so that the model can match to new data; (c) concise representation in both size and computation.

There are several modeling approach used in speaker recognition systems, including neural network, template matching, Gaussian mixture models (GMMs) and hidden Markov models (HMMs).

In the neural network approach the speaker model can have many forms such as multi-layer perceptrons or radial basis function. The models are explicitly trained to discriminate between the speaker being modeled and alternative speakers.

In HMMs, the method models the temporal evolution and statistical variation of the features. It provides the statistical representation of how a speaker produces sounds. A speaker model can be represented as GMMs. Conceptually, the GMM is similar to HMM, except that the GMM does not account for temporal ordering of feature vectors (Quatieri, 2001).

Impostor model

The use of impostor modeling is a normalization to minimize non speaker related variability (e.g., text, microphone, noise…) in the likelihood ratio score. There are two approaches to represent impostor model. The first, known as likelihood sets, cohorts or background sets, uses a collection of other speaker models to compute the impostor match score. It is usually a function of the match scores from a set of non-claimant speaker models. These non-claimant models can come from other enrolled speakers or as fixed models from a different corpus.

The second approach, referred to as general or universal background modeling, uses a single speaker-independent model trained on speech from a large number of speakers to represent speaker-independent speech. The approach represents impostors using a general speech model that is compared to a speaker-specific speaker model. The advantage of this approach is that only a single impostor model needs to be trained and scored.

Similarity measure (Bourlard, Morgan, 1998)

As mentioned in the previous section, speaker verification is a form of hypothesis test. A likelihood ratio test is used to discriminate the test speech comes from a claimed speaker or impostor. The system will verify the hypothesis that speaker S_i is indeed the presumed speaker S_c if

$$P(S_c \mid X) > P(\bar{S}_c \mid X) \tag{2.23}$$

where $P(\bar{S}_c \mid X)$ is the probability of the speaker is being anyone except S_c.

With some threshold δ, equation (2.23) can be expressed as

$$\frac{P(S_c \mid X)}{P(\bar{S}_c \mid X)} = \frac{P(X \mid S_c)}{P(X \mid \bar{S}_c)} > \delta \tag{2.24}$$

with $\delta > 1$.

Using logarithm of the likelihood ratio, the above equation becomes

$$S{=}S_c,\text{ if }\ \log\ P(X \mid S_c) - \log\ P(X \mid \overline{S}_c) > \Delta \tag{2.25}$$

where $\Delta = \log\ \delta$.

Equation (2.25) states that the identity of S_c is accepted/validated when the difference is above threshold. Otherwise, it is rejected.

For a detailed discussion on some main speaker recognition approaches such as text-dependent, text independent and text-prompted speaker recognition, we refer the reader to Gold and Morgan (2000).

2.2.2.2. Gaussian Mixture Models (GMMs)

As mentioned in the previous section, current studies in automatic speech recognition treat speech signal as stochastic patterns and analyze signal using statistical pattern recognition. Speech signal is observed as random variables that often have distribution following Gaussian distribution referred to as normal distribution. A continuous random variable *X* is said to follow Gaussian distribution if *X* has a probability density function pdf in the form

$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2} \right) \tag{2.26}$$

This normal distribution is specified by two parameters: mean μ and variance σ^2 where $\sigma > 0$ and sometimes denoted as $N(\mu,\sigma^2)$.

For the *n*-dimensional continuous random vector $\mathbf{X} = (X_1, \ldots, X_n)$ the multivariate Gaussian pdf is:

$$f(\mathbf{X}{=}\mathbf{x}|\boldsymbol{\mu}, \Sigma) = N(\mathbf{x};\, \boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{n/2}\, |\Sigma|^{1/2}} \exp\left[-\tfrac{1}{2}(\mathrm{x}-\mu)^t \Sigma^{-1} (\mathrm{x}-\mu)\right] \tag{2.27}$$

where $\boldsymbol{\mu}$ is an n-dimensional mean vector with $\boldsymbol{\mu} = E(\mathbf{x})$, Σ is the $n\times n$ covariance matrix $\Sigma = \mathrm{E}[(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^t]$, and $|\Sigma|$ is the determinant of the covariance matrix Σ**.** The covariance matrix Σ has the i-j^{th} element σ_{ij}^2 is as follows

$$\sigma_{ij}^2 = E[(x_i - \mu_i)(x_j - \mu_j)] \tag{2.28}$$

In more complex distributions, random variables can be approximated using Gaussian mixtures as follows

$$f(\mathbf{X}{=}\mathbf{x}|\boldsymbol{\mu}, \Sigma) = \sum_{m=1}^{M} w_m N_m(\mathbf{x};\boldsymbol{\mu}_m \boldsymbol{\Sigma}_m) \tag{2.29}$$

where w_m is the mixture weight associated with m-th Gaussian component. This weight has the following constraints: $w_m \geq 0$ and $\sum_{m=1}^{M} w_m = 1$.

A Gaussian mixture model (GMM), given an adequate number of mixture, has been shown to be able to model any arbitrary continuous pdf (Duda, *et al.*, 2001). This study employs GMMs for unsupervised individual animal clustering task. The parameters for the GMM, $\boldsymbol{\mu}_m$ and Σ_m and w_m, are estimated using standard Baum-Welch estimation algorithm.

2.2.3. Automatic speaker diarization

Audio diarization is a task of marking and categorizing the audio sources within a spoken document (Tranter, Reynolds, 2006). The audio sources may consist of music segments, various speakers, noise sources, and other signal source/channel characteristics. Most current audio diarization research focuses on speaker diarization to identify who is speaking and when. This is a task of speaker segmentation and clustering.

Three domains which have been used for speaker diarization include broadcast news audio (Tranter, Reynolds, 2004; Meignier *et al.*, 2006), recorded meetings (Pardo *et al.,* 2007), and telephone conversations (NIST, 2006).

Based on Tranter and Reynolds (2006), this section introduces the current speaker diarization framework and describes each of the tasks included in the system.

2.2.3.1. Principles of speaker diarization

Figure 2.7 presents a block diagram of a speaker diarization system. Typically the framework consists of tasks to perform speech detection, gender and/or bandwidth segmentation, speaker segmentation, and final boundary refinement.

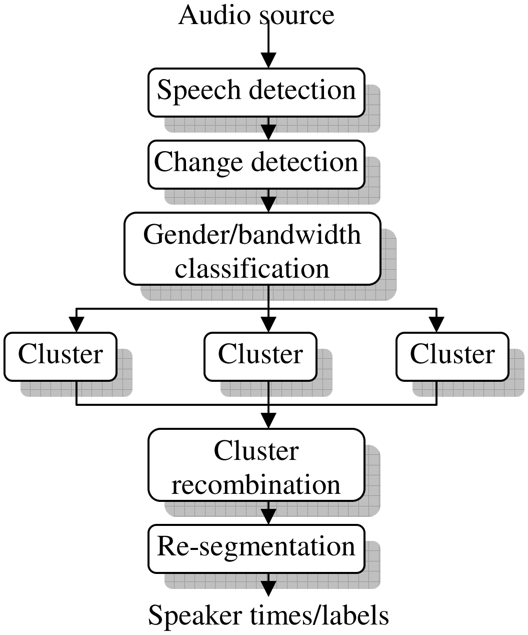

Figure 2.7. A speaker diarization system (after Tranter and Reynolds, 2006)

Speech detection

The purpose of speech detection is to find the speech regions present in the audio data. The task starts with building speech and non-speech models using Gaussian mixture models (GMMs) or hidden Markov models (HMMs) trained on labeled data. A Viterbi segmentation is employed to identify unknown speech regions. At this stage, regions that are of no interest for the final output may be automatically detected and removed.

Change detection

Change detection tries to find points which are likely to be change points between audio sources. In the un-segmented audio data, the change detection looks for change points of both speaker and speech/non-speech. The method involves processes of looking at adjacent windows of data, calculating a distance metric between the two windows, and deciding whether the windows come from a different or the same source.

Gender/bandwidth classification

In order to reduce the load of clustering, to give more parameter flexibility, and to provide more information about the speaker, the gender/bandwidth step classifies the segments into gender (male-female) segments and bandwidth segments where low-bandwidth represents narrow band/telephone and high-bandwidth represents studio data.

The classification is a supervised task utilizing maximum likelihood with GMMs or HMMs trained on labeled data.

Clustering

The goal of the clustering stage is to associate together segments of the same speakers. Ideally, the process results in one cluster for each speaker and segments from a given speaker for one cluster.

A typical clustering method utilized in speaker diarization is hierarchical agglomerative clustering with a Bayesian information criterion (BIC) (Chen, Gopalakhrisnan, 1998), widely used in statistics; or a modification of BIC (Pardo *et al.*, 2007) as a stopping criterion.

The clustering stage consists of the following steps:

1. Create cluster initializations using speech segments
2. Compute pair-wise distances between clusters
3. Merge closest clusters
4. Update distances of remaining clusters to new cluster
5. Repeat steps 2-4 until stopping criterion is met.

Cluster recombination

Cluster recombination runs clustering to create smaller clusters of the audio data. The method builds a universal background model (UBM) utilizing training data to represent general speakers. In order to form a single model for each cluster, maximum *a posteriori* (MAP) adaptation is applied on each cluster from the UBM. The method then defines the cross-likelihood ratio (CLR) (Barras *et al.*, 2004) as follows:

$$\mathrm{CLR}(c_i, c_j) = \log\left(\frac{L(x_i \mid \lambda_j)L(x_j \mid \lambda_i)}{L(x_i \mid \lambda_{UBM})L(x_j \mid \lambda_{UBM})}\right) \tag{2.30}$$

where L(x_i|λ_j) is the average likelihood per frame of data x_i given model λ_j. A new model is created from the merged of the pair clusters with the highest CLR. The process is repeated until the CLR reaches below the predefined threshold.

Resegmentation

The goal of this final step is to refine the original segment boundaries and to fill in short segments that have been removed during the clustering steps. This post-processing step is done utilizing Viterbi decoding, using final cluster models and non-speech models.

2.3. Unsupervised clustering with HMMs

This section presents model-based clustering using likelihood feature space. It addresses the problem of clustering using HMMs and reviews two different clustering methods: HMM-based *k*-model clustering and HMM-based hierarchical agglomerative clustering. The approach extends those methods by adding techniques to estimate the number of clusters *K*, and to assess the clustering results.

Unsupervised clustering deals with the task of learning without supervision a natural or appropriate way of dividing a data set into groups. The goal is to assign labels to a data set in a way that maximizes the similarity among items in the same cluster and minimizes the similarity between items in different clusters.

It is known that unsupervised clustering is a more challenging task than supervised classification. In supervised classification the system can be trained adequately using classes or labels that are already identified.

The unsupervised clustering of acoustic waveforms based on their similarity is becoming important in animal vocalization analysis and in human speech tasks. Recent work in animal vocalization analysis includes the use of a self organizing map (SOM) to analyze the syllables of bird song (Somervou, Harma, 2003) and to associate the specific information content of the prairie dog vocalizations (Placer *et al*., 2006), the use of HMMs for beluga whale and elephant vocalizations clustering (Clemins, 2005). In human speech tasks, meanwhile, the work has included the effort in speaker indexing based on utterance segmentation and speaker clustering (Nishida, Kawahara, 2003), labeling speaker turns by segmenting and clustering a continuous audio stream (Johnson, 1999), speaker segmentation and clustering in meetings (Qin Jin *et al.*, 2004) and speaker diarization (Pardo *et al.*, 2007; Tranter, Reynolds, 2006).

Clustering sequential data such as acoustic waveforms is a difficult subject. Often there is no natural distance function between sequential data. These cannot be expressed efficiently as points in a finite dimensional vector space. The structure of an underlying process at work within data is often difficult to infer, and typically one has to deal with sequences of different length (Bicego *et al.*, 2003).

Sequential data clustering is generally classified into discriminative (proximity-based) and generative (model-based) approaches (Bicego *et al.*, 2003; Ghosh 2003). A discriminative approach assumes that the data exist in a space where any pair of data has a well-defined distance or similarity measure. For numeric sequences such as time series, for example, there are possible ways of measuring similarity such as normalization transformation (Goldin, Kannelakis, 1995), dynamic time warping (Berndt and Clifford, 1996), and the longest common subsequence similarity (LCSS) measure. For a more detailed discussion on similarity measure in time series, the reader may refer to Das and Gunopulos (2003).

Generative (model-based) approaches, on the other hand, assume that the sequential data belonging to the various clusters have been generated by a probabilistic pattern generation process. A model-based clustering tries to make the best estimate of the parameters and then obtains the data cluster models using these estimates. Parameter estimation can be done using Maximum Likelihood (ML), Maximum Aposteriori (MAP) or Mean Posterior (MP) computation (Ghosh, 2003).

The type of models in the generative method is often specified *a priori*, such as Gaussian mixture models (GMM) or Hidden Markov Models (HMMs). The model structure (e.g, number of Gaussians in a mixture of a Gaussian model, or number of states in an HMM) can be determined using model selection approaches (Li, Biswas 2000; Biernacki *et al*., 2000) and the parameters are estimated using an EM algorithm that optimizes a likelihood criterion. Given the complexity of constructing feature vectors for sequential data such as acoustic data, generative model-based clustering that involve Hidden Markov Models are naturally fit (Smyth, 1997; Cadez *et al*., 2000; Pannucio *et al*., 2002; Zhong, 2003).

2.3.1. HMM-based *k*-model clustering

As discussed in the previous section, HMMs employed to represent each of the different clusters are statistical models that use a finite number of states and the associated state transition to jointly model the temporal and spectral variations of repertoires. They have been used extensively to model fundamental speech units in speech recognition because they can adequately characterize both the temporal and spectral varying nature of the speech signals (Rabiner, Juang, 1993; Rabiner, 1989).

Smyth and Cadez (Smyth, 1997; Cadez *et al*., 2000) suggested the use of HMMs in clustering by utilizing two steps, first defining pairwise distances using the log-likelihood values to cluster the sequences into *K* groups, and then fitting *K* HMMs one to each group. Following this, the method refines the initial estimate using a Baum-Welch procedure. The best number of clusters *K* is then selected using a Monte-Carlo cross-validation approach.

The assumption underlying an HMM-based method of clustering is that all of the data that belong to a cluster is generated by the same HMM, and as such, have high probability under this HMM. If a vocalization has a high probability under an HMM model, it is considered to be generated or accepted by the model (Oates *et al*., 1999).

The algorithm used for HMM-based *k*-models clustering is "hard clustering", meaning that on each iteration every vocalization data is assigned to a single cluster represented by an HMM. The HMM parameter updates are influenced only by data items currently in the associated clusters. The benefits of this approach are twofold (Butler, 2003): each training iteration involves processing of each vocalization only once. The method performs only *N* instances of the Baum-Welch training procedure per cycle. Moreover, the expected number of cycles is smaller because cluster membership tends to change by large jumps and then to settle to a static configuration once parameter changes become sufficiently small and leading to a rapid convergence.

The standard *k*-means clustering approach is straightforward. How many clusters are being sought, the *k* parameter, is specified in advance. Then *k* points are chosen at random as cluster centers. Data instances are assigned to their closest cluster center according to some distance metric. The centroids of all instances in each cluster are calculated. These centroids are assigned to be the new center values for their respective clusters. Finally the whole process is repeated with the new cluster centers, until a convergence criterion is reached.

The HMM-based *k*-models algorithm is a generalization of the above standard *k*-means, with the cluster centroid vectors being replaced by probabilistic models. The criterion to re-assign data to clusters is maximization of the likelihood of the data points.

The re-assignment of the data employs a Viterbi algorithm. The computation of clusters is done by re-estimation of the model parameters using the Baum-Welch re-estimation algorithm (Knab *et al.*, 2003).

Given a set of data **D** = {X_1, X_2, … X_n} and a fixed integer $K <<n$, HMM-based *k*-models algorithm computes a partition **C** = {C_1, C_2, … C_K} of **D** and finds HMMs λ_1, λ_2, … λ_K as to maximize the objective function

$$f(C) = \prod_{k=1}^{K} \prod_{X_i \in C_k} L(X_i \mid \lambda_k) \qquad (2.31)$$

In equation (2.30) $L(X_i|\lambda_k)$ denotes the likelihood function, namely, the probability density of the generating repertoire X_i by model λ_k. Therefore,

$$L(X_i \mid \lambda_k) = p(X_i \mid \lambda_k) \qquad (2.32)$$

The problem of computing a *k*-model clustering, then, can be formulated as a joint likelihood maximization problem (Knab *et al.*, 2003; Cadez *et al.*, 2000).

Assume that the number of estimated cluster *K* is known. Given *K* initial HMMs λ_1^0, λ_2^0, . . ., λ_K^0, the *k*-models clustering algorithm can be summarized as follows:

Algorithm : the *k*-models clustering algorithm

Input : estimated cluster *K*, *K* initial HMMs

Output : *K* HMMs clusters

Steps :

1. Iteration t∈ 1, 2, …

- Data assignment:

for each repertoire data X_i, assign data to the model of maximum likelihood, namely $L(X_i \mid \lambda_k^{t-1})$ is maximal

- Model estimation:

  calculate new parameters of λ_1^t, λ_2^t, . . ., λ_K^t using data assigned to the models and using previous parameters λ_1^{t-1}, λ_2^{t-1}, . . ., λ_K^{t-1}

2. Termination:

   Terminate if no labels have changed or a given iteration number is reached.

2.3.2. HMM-based hierarchical agglomerative clustering

In the above partitional clustering approach, the number of clusters *k* is specified in advanced. This number, however, is often unknown. Additionally, it is sometimes preferable to have a method that returns a hierarchical structure of the data objects. Hierarchical clustering methods provide such a structure, creating a hierarchical decomposition of the given data objects and displaying the result via tree diagrams or dendrograms as shown in Figure 2.7. At the leaves of the dendrogram, each object is a cluster by itself. The height where two clusters are merged signifies the distance between these two clusters.

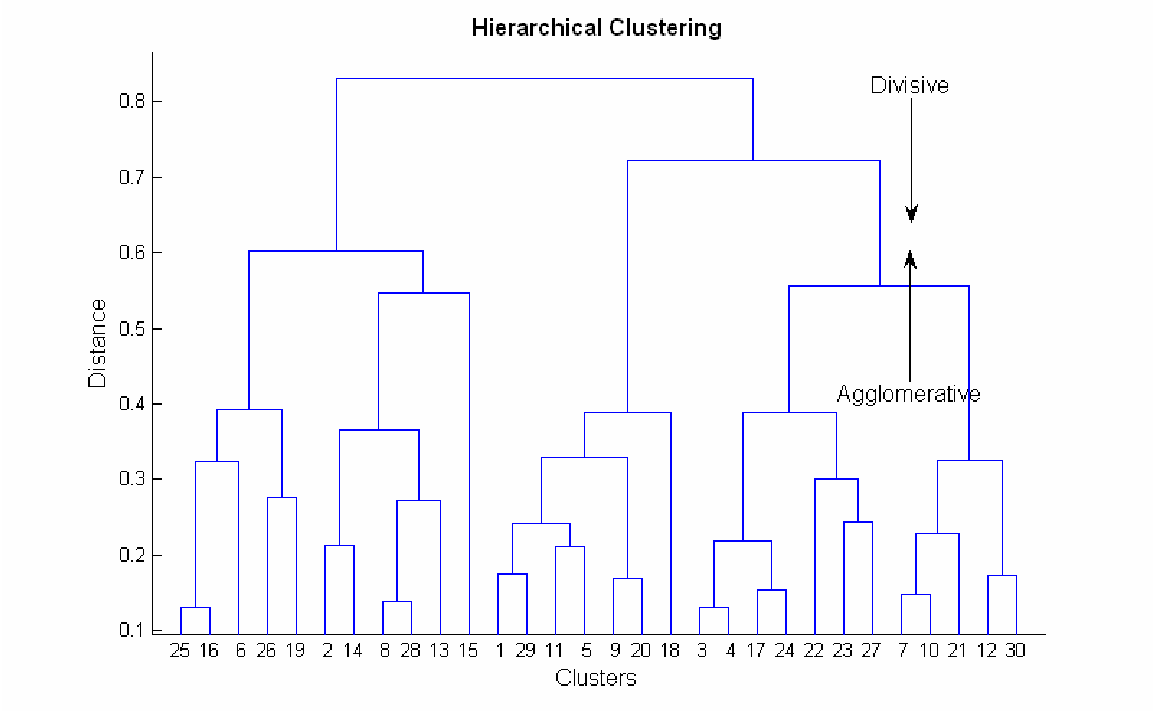

Figure 2.8. Hierarchical clustering

The dendrogram can be bottom-up or top-down. The bottom-up approach, called the agglomerative method, starts with each data object as a distinct cluster. The clusters, that are closest according to a specific distance measure are successively merged until a termination condition is satisfied. Most agglomerative clustering approaches use variants of single-link, complete-link and average-link distances, which differ in the way to characterize the similarity or distance between a pair of clusters. In single-link the distance between two clusters is the minimum of the distances between all pairs of the data drawn from two clusters. In complete-link it is the maximum of all pair-wise distance between data objects in the clusters. Meanwhile, average-link uses an average among all pairs as its distance metric (Jain *et al.*, 1999).

The top-down approach, called the divisive method, begins with all data in a single cluster then performs splitting into smaller clusters according to some measures until a stopping criterion is met.

Due to high computational complexity, some researchers, Ajmera *et al.*, (2002), Ajmera, Wooters (2003) for instance, have tried to increase efficiency by building model-based hierarchical clustering from the results of partitional clustering. Data objects are first clustered into K_0 groups, greater than the expected final K, using a partitional method such as k-models algorithm. The hierarchical agglomerative clustering starts from K_0 and iteratively merges the two closest clusters until all data objects are in one cluster. The method returns a series of nested structure that can be further analyzed using some optimization criteria.

To design model-based hierarchical clustering algorithms, it is necessary to define a distance measure between clusters (i.e., models) and then iteratively merge the closest pair of clusters. Usually clusters are chosen such that merging them results in the largest log-likelihood log $P(X|\Lambda)$. The distance is defined as

$$D^W(\lambda_k, \lambda_j) = \log P(X|\Lambda_{before}) - \log P(X|\Lambda_{after}) \qquad (2.33)$$

where Λ_{before} and Λ_{after} are the set of parameters before and after merging two models (λ_k and λ_j), respectively. The distance computation in equation (2.33) is not efficient since to find the closest pair one needs to train a merged model for every pair of clusters and then evaluate the resulting log-likelihood.

In practice, the Kullback-Leibler (KL) distance measure has been commonly used (Ziad Rached *et al*., 2004). An empirical KL divergence between two models λ_k and λ_j , is defined as

$$D^K(\lambda_k, \lambda_j) = \frac{1}{|X_k|}\sum_{x\in X_k}(\log p(x|\lambda_k) - \log p(x|\lambda_j)) \tag{2.34}$$

where X_k is the set of data objects being grouped into cluster k. This distance can be made symmetric by defining (Juang and Rabiner, 1985)

$$D_s^K(\lambda_k, \lambda_j) = \frac{D^K(\lambda_k,\lambda_j) + D^K(\lambda_j,\lambda_k)}{2} \tag{2.35}$$

Zhong (2003) proposed some adaptations related to single-link, complete-link and boundary density as follows. Corresponding to single-link, a *minKL* distance is defined as

$$D^m(\lambda_k, \lambda_j) = \min_{x\in Xk}(\log p(x|\lambda_k) - \log p(x|\lambda_j)) \tag{2.36}$$

and corresponding to complete-link, *maxKL* distance is defined as

$$D^M(\lambda_k, \lambda_j) = \max_{x\in Xk}(\log p(x|\lambda_k) - \log p(x|\lambda_j)) \tag{2.37}$$

To characterize high boundary density between two clusters for building complex-shape clusters, Zhong suggested a *boundaryKL* distance measure

$$D^B(\lambda k, \lambda j) = \frac{1}{|B_k|}\sum_{x\in B_k}(\log p(x|\lambda_k) - \log p(x|\lambda_j)) \tag{2.38}$$

where B_k is η fraction of X_k (one can set η around 10%) that have smallest $\log p(x|\lambda_k) - \log p(x|\lambda_j)$ values. A value of 0 for $\log p(x|\lambda_k) - \log p(x|\lambda_j)$ defines the “boundary” between cluster k and j.

The algorithm for HMM-based hierarchical agglomerative clustering is as follows:

Algorithm : HMM-based hierarchical agglomerative clustering

Input : A set of N data objects $X = \{x_1, \ldots x_N\}$, model structure λ and the depth of hierarchy K_0

Output : An K_0 level cluster (model) hierarchy and hierarchical partition of the data objects, with k clusters at the k-th level.

Steps:

1. Partitional clustering: partition data objects into K_0 clusters using HMM-based partitional clustering.
2. Distance calculation: compute pairwise inter-cluster distances using an appropriate measure (equation 2.34-2.37)
3. Cluster merging: merge two closest clusters and re-estimate a model from the merged data objects $X_k = X_k \cup X_j$, i.e.,

   $$\lambda_k = \arg\max_{\lambda} \sum_{x \in X_k} \log p(x \mid \lambda)$$

4. Stop if all data objects have been merged into one cluster, otherwise go back to step 2.

2.3.3. Estimating number of clusters in the data set

A major challenge in cluster analysis is to estimate the optimal number of clusters in the data set. A comprehensive survey of approaches to estimate the number of clusters is given by Milligan and Cooper (1985). This section addresses two method of estimating the number of clusters, namely, dissimilarity analysis and deltaBIC analysis. The following are brief descriptions of each method.

Dissimilarity analysis

This section introduces the clustering distance method of Lange (Lange *et al.*, 2004) that leads to a metric of dissimilarity. The number of clusters is then estimated from the cross-data dissimilarity analysis.

Let a data set **D** consist of *n* observations $\mathbf{D} = \{ X_1, \ldots, X_n\}$. Xi = $\{\mathbf{x}_{i1}, \ldots \mathbf{x}_{it}\}$ is an observation of length *t* composed of potentially multivariate feature vectors **x**. The problem of clustering is to find a partition of the data set into *k* disjoint clusters. A clustering algorithm A_k builds a solution $\mathbf{L}= A_k(\mathbf{D})$, where $\mathbf{L} = \{L_1, \ldots L_n\}$ is a vector of labels, and $L_i \in \{1, \ldots k\}$ denotes the cluster label. Note that the algorithm A_k is not a classifier itself, but rather a software tool to establish a matching between a specific finite data set and associated cluster labels.

Consider a comparison of solutions computed on two different data sets. Let $\mathbf{L}_1= A_k(\mathbf{D}_1)$ be defined with regard to a data set $\mathbf{D}_1$ and $\mathbf{L}_2= A_k(\mathbf{D}_2)$ for data set $\mathbf{D}_2$. The goal would be to compare two solutions $\mathbf{L}_1$ and $\mathbf{L}_2$ and to assess their similarity or dissimilarity. They are, however, are not directly comparable since they come from different data sets. To assess the distance of clustering solutions, Lange *et al.* devise a predicted label or classifier that renders the solutions comparable.

In general, supervised classification generates a classifier function *C* that assigns an arbitrary observation from a designated feature space to one of *k* classes based on a labeled input data. A dataset $\mathbf{D}_1$ together with clustering solution $\mathbf{L}_1$ can be considered as a training data set used to construct a generalized classifier function. This classifier C trained from ($\mathbf{D}_1$, $\mathbf{L}_1$) predicts a new label $\mathbf{L}_3$= $C(\mathbf{D}_2)$ for data set $\mathbf{D}_2$. These labels $\mathbf{L}_3$= $C(\mathbf{D}_2)$, then, can be compared to those generated by the clustering algorithms, that is, with $\mathbf{L}_2$= $A_k(\mathbf{D}_2)$.

Lange *et al.* define a measure of the distance of $\mathbf{L}_3$ and $\mathbf{L}_2$ using a normalized Hamming distance measure as follows:

$$d(\mathbf{L}_3, \mathbf{L}_2) := \frac{1}{n} \sum_{i=1}^{n} 1\{L_{3_i} \neq L_{2_i}\} \tag{2.39}$$

where $\mathbf{1}\{L_{3i} \neq L_{2i}\}=1$, if $L_{3i} \neq L_{2i}$ and zero otherwise. Equation (2.39), compares two sets of labels that are not necessarily in natural correspondence. This measure quantifies the average distance of two clustering solutions. This can be seen as a misclassification risk with respect to class labels produced by a clustering algorithm.

One significant problem with this approach is the non-uniqueness of label representations. Two partitionings of a data set $\mathbf{D}_2$ might be structurally equivalent although the labelings $\mathbf{L}_3$ and $\mathbf{L}_2$ are differently represented. For instance, a cluster labeled 2 in the first solution might correspond to the one labeled 1 in the second solution, and vice versa. This ambiguity poses a problem for validation.

To overcome the non-uniqueness of representation, the label indices in one solution need to be optimally permuted so as to maximize the agreement between the two solutions under comparison. The distance value, then, is modified as follows:

$$d_{Pk}(\mathbf{L}_3, \mathbf{L}_2) := \min_{\pi \in P_k} \frac{1}{n} \sum_{i=1}^{n} 1\{\pi(L_{3i} \neq L_{2i})\} \tag{2.40}$$

where P_k is the set of all permutations of the label elements. Equation (2.40) quantifies the fraction of differently labeled points and can be regarded as the empirical misclassification risk of classifier C with respect to the clustering algorithm A_k.

To use this concept of solution distance in a way that indicates overall dissimilarity value, we denote $Dis(A_k)$ as the average of $d_{Pk}(\mathrm{L}_3, \mathrm{L}_2)$ obtained for r times of the split over the data **D**:

$$Dis(A_k) = \frac{1}{r}\sum_{i=1}^{r} d_{Pki}(\mathbf{L}_3, \mathbf{L}_2) \tag{2.41}$$

Lange *et al*. refer to this metric as “stability”; however, since its value increases rather than decreases with distance between solutions, we here refer to it as a “dissimilarity” index of clustering solutions with regard to the distribution of the data.

Applying this dissimilarity value to estimate the number of clusters in the data, equation (2.41) is normalized using the misclassification rate of a random labeling $Dis(R_k)$ that assigns an observation to cluster v with probability $1/k$:

$$\bar{D}is(A_k) = \frac{Dis(A_k)}{Dis(R_k)} \tag{2.42}$$

The smaller the value of $\bar{D}is(A_k) \in [0,1]$, the more similar are the solutions being compared.

Using this approach to estimate the cluster number we have the following algorithm:

Algorithm : Dissimilarity analysis to estimate number of clusters in a data set

Input : Data **D**, clustering algorithm A_k, number of split r, number of sample s

Output : Estimated number of clusters $\hat{k}$

For each number of clusters $k \in \{k_{\min}, \ldots, k_{\max}\}$ perform the following steps

1. Estimate $\hat{D}is(A_k)$ by averaging r splits of the data:
    a. Split the given data set into two halves $\mathbf{D}_1$, $\mathbf{D}_2$ and apply a clustering algorithm A_k to both
    b. Construct classifier C using $\mathbf{D}_1$ and its cluster labels $\mathbf{L}_1 = A_k(\mathbf{D}_1)$; then compute $\mathbf{L}_3 = C(\mathbf{D}_2)$
    c. Use equation (2.40) to calculate the distance of the two solutions $\mathbf{L}_3 = C(\mathbf{D}_2)$ and $\mathbf{L}_2 = A_k(\mathbf{D}_2)$
2. Sample s random k-labels, compare pairs of these, and compute the empirical average of the dissimilarities to estimate $\hat{D}is(R_k)$
3. Normalize each $\hat{D}is(A_k)$ with $\hat{D}is(R_k)$ to get an estimate for $\overline{D}is(A_k)$ using equation (2.42)

Return the estimated number of clusters $\hat{k} = \arg\min_k \overline{D}is(A_k)$.

Delta BIC analysis

In the context of automatic speaker clustering, Ajmera *et al.* (2002) present a similarity measure between two probability density functions estimated by Gaussian mixture models (GMMs). Starting from over clustering – clustering the data into a number of clusters larger than the expected number of clusters – the method converges to a final clustering using an iterative merging and retraining process. The process consists of training a GMM for each cluster, selecting the closest pair of clusters for merging, and retraining the GMM of the merged cluster. The similarity measure, referred to as *deltaBIC* is employed without the need for any threshold or penalty term. The merging and retraining are repeated until no possible pair of clusters for merging is left.

Let $\{D_1\}$ and $\{D_2\}$ be two data sets and θ_1 and θ_2 be the maximum likelihood estimates of the parameters of the PDF of $\{D_1\}$ and $\{D_2\}$ respectively. In the case when the PDF is modeled by a Gaussian Mixture Model (GMM), let θ_1 and θ_2 are parameters of GMMs having components M_1 and M_2 GMMs. A similarity or distance measure is required to decide if two clusters representing data sets $\{D_1\}$ and $\{D_2\}$ should be merged or not. Letting θ to be a maximum likelihood estimate of the parameters of the PDF of data $\{D\} = \{D_1\} \cup \{D_2\}$. Let θ be the parameter of a GMM having $M = M_1 + M_2$ component, the distance measure or *deltaBIC* is calculated as

$$deltaBIC = \sum_{X \in D} \log p(X \mid \theta) - \sum_{X \in D_1} \log p(X \mid \theta_1) - \sum_{X \in D_2} \log p(X \mid \theta_2) . \tag{2.43}$$

If *deltaBIC* is greater than 0 for a pair of clusters, those two clusters are believed to be similar enough to merge. The method finds and merges the cluster pair that gives the largest *deltaBIC* value.

The algorithm to estimate number of clusters in a data set using *deltaBIC* as a stopping criterion is as follows.

Algorithm : DeltaBIC analysis to estimate number of clusters in a data set

Input : Initial clusters I

Output : Number of clusters K

Steps :

1. Start by over-clustering *I* using some initialization approaches. An additional loop of segmentation and training could be made before proceed to the clustering module.
2. Cluster comparison and merging:

- Search for all possible candidate pairs satisfying *deltaBIC* > 0, and select the best pair.
- Merge the pair
- Retrain the GMM for the new merged cluster with the data assigned to it, and the number of parameters (mixtures) of the merged model is the sum of the number of mixtures of the component models.

3. The cluster comparison and merging are repeated until a stopping criterion (*deltaBIC* < 0) is reached.

2.3.4. Cluster evaluation

In general, there are three basic approaches to investigate the validity of a cluster, based on assessment using external criteria, internal criteria, and relative criteria (Jain, Dubes, 1998; Halkidi, 2001). An external assessment criterion evaluates the clustering result using an *a priori* known structure, i.e. it is a supervised approach to validation. Internal criteria determine if the structure is intrinsically appropriate for the data using only data comparisons, while relative criteria compare two structures resulting from the algorithm to find out which is more stable or more appropriate for the data. Thus internal criteria are data-based and relative criteria are algorithm-based. Both internal and relative validation criteria are unsupervised approaches. Jain and Dubes (1998) discussed in more detail validity indices to evaluate clusters. This section underlines two validity criteria used for cluster evaluation, one supervised and one unsupervised.

Supervised evaluation criteria: average purity

The supervised clustering evaluation employs the average cluster purity (ACP) and average speaker purity (ASP) metrics explained in Solomonoff (Solomonoff *et al.*, 1998) and Ajmera (Ajmera *et al.*, 2002). ASP provides a measure of how well an individual speaker is focused on only one cluster, while the ACP evaluates how well a cluster is focused on only one speaker.

The purity of a cluster p_{i*} is defined as

$$p_{i*} = \sum_{j=1}^{Ns} \frac{n_{ij}^2}{n_{i*}^2} \tag{2.44}$$

where n_{ij} is the total number of data in cluster i spoken by speaker j, N_s is the total number of speakers, N_c is the total number of clusters, N is the total number of data, n_{*j} is the total number of vocalizations spoken by speaker j; and n_{i*} is the total number of data in cluster i. The average cluster purity (ACP) is computed using

$$ACP = \frac{1}{N} \sum_{i=1}^{Nc} p_{i*}.n_{i*} \tag{2.45}$$

The speaker purity p_{*j} and its average speaker purity (ASP) can be calculated in similar ways as

$$p_{*j} = \sum_{i=1}^{Nc} \frac{n_{ij}^2}{n_{*j}^2} \tag{2.46}$$

$$ASP = \frac{1}{N} \sum_{j=1}^{Ns} p_{*j}.n_{*j} \tag{2.47}$$

An overall evaluation criteria to compare between systems is derived from the ACP and AIP to obtain

$$\mathrm{K} = \sqrt{ACP * ASP} \tag{2.48}$$

Values of the ASP and ACP are each 1 when a clustering algorithm results in exactly one cluster for each individual.

Unsupervised evaluation criteria: dissimilarity index

The dissimilarity value employed in this work to assess the consistency of clustering results is a generalization of the cross-data cluster dissimilarity computation mentioned in the previous section. To implement this, the clustering algorithm is run *t* times on the same data set using different initial conditions (or different parameter settings if parameter variation is of interest) and computes the average dissimilarity value of the labeling results as follows:

$$d_{Pk}(\mathbf{L}_i,\mathbf{L}_j) = \min_{\pi \in P_k} \tfrac{1}{n}\sum_{m=1}^{n} 1\{\pi(L_i \neq L_j)\}\,, \quad i, j = 1, \ldots, t \tag{2.49}$$

where P_k is the permutation of all label elements, $\mathbf{1}\{L_i \neq L_j\}=1$ if $L_i \neq L_j$ and zero otherwise. The smaller the multi-run dissimilarity value $\in [0,1]$, the more consistent is the clustering algorithm across this dataset. To incorporate the impact of data inclusion as well as the initial conditions, this idea can easily be extended to use random subsets for each run in a resampling-with-replacement fashion.

2.4. Feature extraction

The overall purpose of feature extraction is to parameterize a vocalization into a sequence of feature vectors that contain concise and relevant information about the sequence of sounds within the vocalization. The features extracted are expected to be able to discriminate similar vocalizations, able to create models without the need for an excessive training data; and have statistical properties which are invariant across vocalizations and over a wide range of environment (Rabiner *et al.*, 1999).

Feature extraction in speech recognition generally consists of three processes (Reynolds, 2002). First, some form of speech detection is used to remove non-speech portions from the signals; second, the features are extracted to convey or represent information, and third, some channel compensation is applied. It is known that different input devices will impose different spectral characteristics on the vocalizations. Channel compensation such as cepstral mean subtraction and variance normalization are often used to remove the channel effects to get robust features (Furui, 1981; Viikki, Laurila, 1998).

In speech or speaker recognition Fourier analysis is still the most widely used approach for extracting spectral features. Sometimes, non-uniform scales are employed to provide a better perceptual representation, such as the mel-frequency spectral feature set. This is to mimic the human auditory system that processes the spectral information on a non-uniform frequency scale (Davis and Mermelstein, 1980).

This section focuses on feature extraction approaches that include Greenwood function cepstral coefficients (GFCCs), pitch tracking, delta and acceleration computation, cepstral mean normalization and cepstral variance normalization.

2.4.1. Greenwood function cepstral coefficients (GFCCs)

Frequency-domain features such as Mel-frequency cepstral coefficients (MFCCs) (Davis and Mermelstein,1980) are commonly used in most speech recognition systems. These take into account the perceptual model of the human auditory system by warping the linear frequency axis to match the Mel-scale cochlear frequency map.

This section briefly reviews the Greenwod function cepstral coefficients (GFCCs), that is discussed in more detail in (Clemins *et al.*, 2006). This is a generalization of MFCC where the frequency warping component is adjusted according to the perceptual model of the species – in our case, ortolan bunting bird species.

Greenwood (Greenwood, 1961, 1990) shows that many land and aquatic mammals perceive frequency on a logarithmic scale along the cochlea. This can be modeled as

$$f = A(10^{ax}\text{-}b) \tag{2.50}$$

where f is frequency (Hz), A, a and b are species specific constants, and x is the cochlea position.

For a real frequency f a frequency warping is defined as

$$F_p(f) = \tfrac{1}{a}\log_{10}(\tfrac{f}{A} + b) \tag{2.51}$$

Given the approximate hearing range ($f_{\min} - f_{\max}$) of the species under study, and using approximation of $b = 0.88$ (LePage, 2003), constants A and a can be derived as follows

$$A = \frac{f_{\min}}{1-b} \tag{2.52}$$

$$a = \log_{10}(\frac{f_{\max}}{A} + b) \tag{2.53}$$

Thus, one may construct a frequency warping function using a species specific value for $f_{\min}$ and $f_{\max}$ and assumed value of b = 0.88.

Figure 2.9 shows a block diagram to compute the Greenwood function cepstral coefficients.

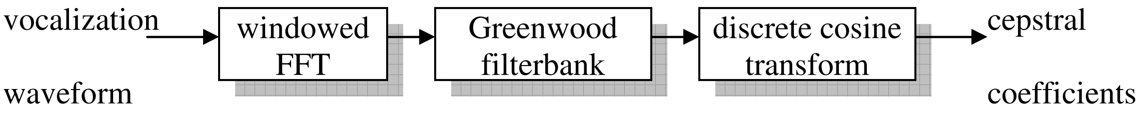

Figure 2.9. Steps in the computation of GFCC coefficients

The vocalization signal is segmented into frames, and each frame is windowed. Windows of vocalization data are transformed using a fast Fourier transform

$$X(k) = \sum_{n=0}^{N-1} w(n)x(n)\exp(-j2\pi kn/N) \qquad (2.54)$$

where $x(n)$ is the discrete-time signal with length N; k=0,1, .. N-1, and k corresponds to the frequency $f(k) = kf_s/N$, f_s is the sampling frequency in Hertz and $w(n)$ is a time-window. Often the Hamming window is used, given by $w(n)$ = 0.54 - 0.46cos($\pi n/N$).

Window sizes of 30ms are typical for human speech, based on tradeoffs between frequency resolution and signal stationarity. Since ortolan bunting bird vocalizations have a fundamental frequency range that much higher than human speech, the window size is adjusted to 3ms - 6ms. In all experiments the frame rate is one-half the window size so that consecutive windows overlap by fifty percent. This overlapping allows improved temporal resolution for time alignment while still maintaining sufficient frequency resolution for spectral analysis.

The magnitude coefficients $|X(k)|$ are then binned by correlating them with each triangular filter in the Greenwood filterbank $H(k,m)$. Binning means that each fast Fourier transform magnitude coefficient is multiplied by the corresponding filter-gain; and the results are accumulated, giving

$$X'(m) = \ln\left(\sum_{k=0}^{N-1} | X(k) | H(k,m)\right) \tag{2.55}$$

for $m = 1, 2, \ldots, M$, where M is the number of filter banks and $M \ll N$. The Greenwood filterbank is a collection of triangular filters defined by the center frequencies.

The filterbank center frequencies are computed using the Greenwood scale of equation (2.51). The triangular filters are spread over the whole frequency range from zero up to the Nyquist frequency. However, band-limiting using lower and upper frequency cut-offs is often useful to reject unwanted frequencies or to avoid allocating filters for frequency regions in which there is no useful signal energy. For ortolan bunting vocalizations, the Greenwood filterbank (Figure 2.10) is adjusted to the range 400 to 7400 Hz to fit the $f_{\min}$ and $f_{\max}$ of the song-bird species (Edwards, 1943).

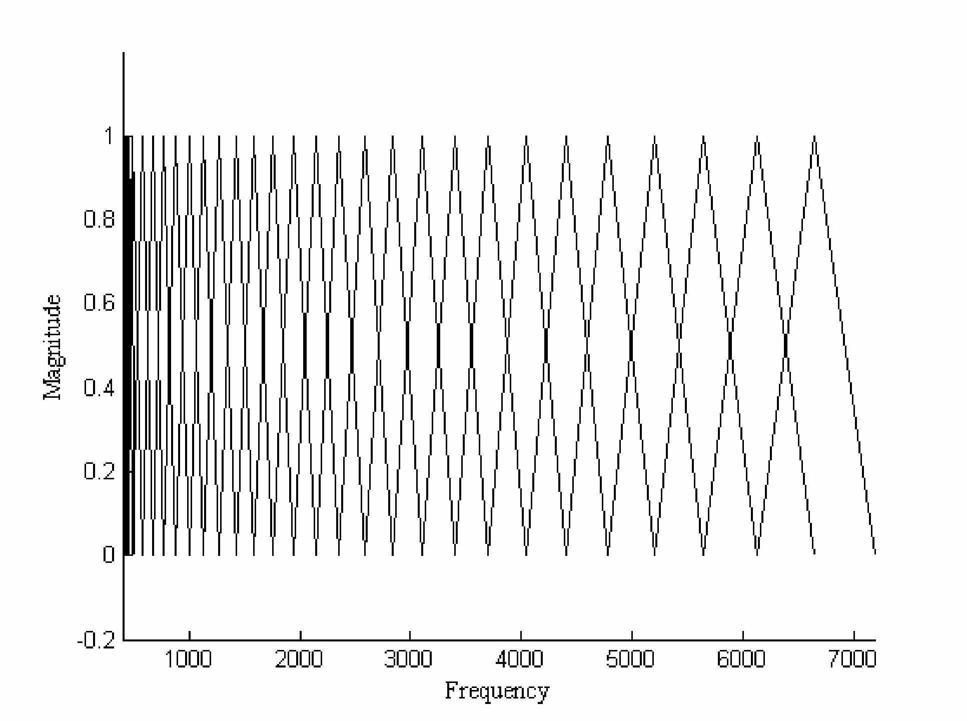

Figure 2.10. Greenwood filterbank

Filterbank amplitudes, however, are typically correlated and the use of a cepstral transformation provides a better representation for pattern recognition. The discrete cosine transform is used to calculate the cepstral coefficients from the log filterbanlk amplitudes *X'*(*m*) as follows

$$c(i) = \sum_{m=1}^{M} X'(m)\cos(i\frac{\pi}{N}(m-0.5)) \qquad (2.56)$$

for i = 1, 2, … *M*, where *c*(*i*) is the *i*th GFCC.

2.4.2. Pitch tracking

Fundamental frequency estimation, sometimes referred to as pitch detection, has been a popular research topic for many years. This feature is important for speech analysis, speaker recognition, foreign language training, and automatic speech recognition in tonal languages such as Mandarin etc. Although many pitch detection algorithms have been developed using a variety of approaches with varying degree of accuracy (see Rabiner *et al*., 1976; Mousset *et al*., 1996) robust methods are still problematic.

Existing pitch extraction methods are generally classified into time domain, auto-correlation, and frequency-based methods (Nobuyuki, 2000). Time domain methods employ a number of zero crossings or peak clippings. These work well in real time and are simple in operation, but are often not reliable in noisy environments. The correlation-based approach is comparatively known to be robust against noise. This has the merit of simplicity but often misses the pitch frequency due to the low frequency components of the formants and periodic noise. The frequency-based method or cepstrum approach sees the spectrum of a voiced speech signal mainly consists of periodic higher harmonics or the fundamental frequency. The cepstrum method is little influenced by the formant frequency but easily influenced by noise.

This section reviews a method of extracting pitch based on auto-correlation approach and their variation and modification such as the autocorrelation function on the log spectrum.

Pitch extraction using autocorrelation function of the log spectrum

Human speech production is generally being modeled as an excitation that is input to a system of resonators. The convolution of the excitation with the impulse response of the resonator components produces the approximation of the speech signal model. It is, therefore, natural to analyze a signal as a separation of the source (excitation) and filter (resonators). Cepstral analysis performs deconvolution of the source and filter.

Nobuyuki Kumida *et al*. (2000) proposed a method of pitch extraction using a combination of the cepstrum approach and the autocorrelation method. They suggested a pitch extraction by using an autocorrelation function on the log spectrum. The autocorrelation function of the log spectrum *S*(*i*) is given by

$$r(j) = \tfrac{1}{N}\sum_{i=0}^{N-1} S(i)S(i+j); \quad j = 0,1,...M \tag{2.57}$$

where *N* is the upper limit of the product sum of the frequency range for calculation of the autocorrelation. Due to the fact that the harmonic properties of the signals are disturbed in the higher frequency range, the range of *N*, therefore, selected to be in the lower range (between 0 and 2.5 kHz) where the harmonic structure is regular. *M*, meanwhile, is the upper limit of *j* for the calculation of *r*(*j*) and is selected corresponding to the existing range of pitch frequency (for human being, it is between 50 and 400 Hz).

The autocorrelation is calculated after the formants are eliminated. The pitch frequency is selected by detection of the minimum *j* at which *r*(*j*) has a peak.

Figure 2.11 shows block diagram of the pitch extraction.

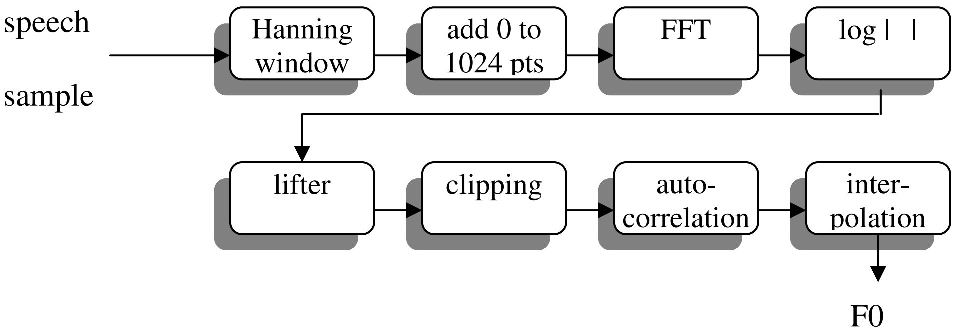

Figure 2.11. Pitch extraction using autocorrelation function of the log spectrum

The method of pitch extraction using autocorrelation function of the log spectrum is as follows:

1. Segment the speech waveform using Hanning window (512 sampling points)
2. Take the fast Fourier transform (FFT) of the 1024 point segment
3. Take the log of the spectrum
4. Lifter and flatten the log spectrum
5. Clip the lower level to reduce the noise effect
6. Calculate the autocorrelation function defined by equation (2.57).
7. The frequency of the true peak of the correlation is estimated by interpolation of the data around the peak.

2.4.3. Delta and delta-delta

Feature vectors computed from the Greenwood function provide a good estimate of local spectra. However, an important characteristic of vocalization data is its dynamic behavior. The performance of a speech system can be greatly enhanced by adding time derivatives to the basic static parameters. Many researchers, therefore, have made use of estimates of the local time derivatives. The delta cepstrum (Furui, 1986) is one of the common forms of this measure. It is typically implemented as a least square approximation to the local slope, or as first order regression coefficients. The time derivative is expressed as follows

$$d_t = \frac{\sum_{k=1}^{N} k(c_{t+k} - c_{t-k})}{2\sum_{k=1}^{N} k^2} \tag{2.58}$$

where d_t is a delta coefficient at time t computed in term of the static coefficients c_{t-k} to c_{t+k}.

The second derivative, referred to as the delta-delta cepstrum or acceleration coefficient, corresponds to similar correlation applied to the delta coefficient.

In practice, most systems that incorporate delta and delta-delta features use them as an add on to a static measure such as MFCCs or GFCCs.

2.4.4. Cepstrum normalization

Robust features are desired to provide acceptable performance under various noisy conditions. For cepstral-based recognition, some methods such as Cepstrum Mean Normalization (CMN) (Furui, 1981) and Cepstrum Variance Normalization (CVN) (Viikki *et al.*, 1998) have been commonly used. CMN works to remove the channel distortion in the cepstral domain, and avoid the further amplification of low frequency noise (Chang-wen Hsu and Lin-shan Lee, 2004). The CVN, meanwhile, reduces the difference in probability density function between the clean and noisy speech signals. Some researchers have also suggested further normalization, such as third order normalization (Yong Ho Suk *et al.*, 1999) and higher order normalization (Chang-wen Hsu and Lin-shan Lee, 2004).

The CMN and CVN are derived as follows. For a given cepstrum vector sequence $\mathbf{X} = \{\mathbf{x}(1), \mathbf{x}(2), \ldots, \mathbf{x}(N)\}$, the mean normalization of a vector $\mathbf{x}(n)$ is

$$\mathbf{x}_{\mathrm{CMN}}(n) = \mathbf{x}(n) - \mu_x \qquad 1 \leq n \leq N \tag{2.59}$$

and the variance normalization of $\mathbf{x}(n)$ is

$$\mathbf{x}_{CVN}(n) = \frac{\mathbf{x}_{CMN}(n)}{\sqrt{\mathbf{v}_x}}, \qquad 1 \le n \le N \qquad (2.60)$$

with a mean vector $\mu_x = (1/N)\sum_{n=1}^{N} \mathbf{x}(n)$, and a covariance matrix $\mathbf{v}_x = (1/N)\sum_{n=1}^{N} ((\mathbf{x}(n) - \mu_x)(\mathbf{x}(n) - \mu_x)^T)$. The cepstrum vector in equation (2.60) has a zero mean vector and an identity covariance matrix so that the elements of $\mathbf{x}_{\mathrm{CVN}}(n)$ are uncorrelated with each other.

2.5. Bioacoustics

The purpose of this section is to offer an overview of common approaches conducting bioacoustics research – the study of sound in non-human animals - in diverse sub-disciplines. It addresses the tasks associated with call-type recognition, individual animal identification and animal censusing using several different approaches such as cross-correlation, dynamic time warping (DTW), and self organizing map (SOM). The following are brief description of each method. The current methods on estimating animal abundance will be discussed in a more detail.

2.5.1. Animal vocalization recognition and individual identification

Cross-correlation approach

Humans and animals have the temporal and spectral structure in their vocalizations. The temporal attributes of a vocalization include duration, repetition, sequences of the vocalization element. The temporal characteristics can be measured through the amplitude-time waveform. The spectral structure that can be derived from the power spectrum consists of frequency, bandwidth, and harmonic structures (Beeman, 1998).

In bioacoustics the inspection of sound spectrogram has become a standard method to compare and to find out the similarity or dissimilarity of the animal vocalizations. Clark *et al*. (1987) suggested a way to measure the similarity between bird songs utilizing sound-spectrogram cross-correlation. The spectrogram correlation between two syllables is examined by sliding one syllable on top of the other, and the correlation peak is computed. In doing so, Clark *et al*. employ two methods: sound comparative and sound averaging.

The sound comparative method underlines the idea that a full representation of a vocalization in its frequency-time structure better serves in a vocalization analysis than a few of its acoustic features.

Vocalizations to be compared are transformed into digital spectrograms to give a set of discrete frequency spectra. The resultant spectrogram is a matrix with time and frequency as rows and columns in the matrix. The similarity between two vocalizations is computed by correlating of their frequency-time matrices.

In the computation, Clark *et al*. observed the similarity as the peak value of the correlation function computed by cross-correlating the two frequency matrices. This value is achieved by time-shifting one matrix with respect to the other and calculating the correlation coefficients between the two matrices at each offset. This results in a sequence of correlation coefficients.

The peak value of the correlation function represents the similarity of the two vocalizations and is used as a quantitative measure of similarity. Both matrices to compare are normalized to give a similarity value between ± 1.00.

When vocalizations are classified to be in the same group, Clark *et al*. saw the importance of generating an average vocalization type. The averaging process starts with the alignment of the set of frequency time matrices using sound comparative method, then summing the matrices and dividing the resultant summed matrix by the number of sounds in the set. The result of the above steps is an average sound matrix represented the average sound-type for the set. This can be displayed in a spectrogram similar to that used for displaying individual vocalizations.

The methods of sound-comparison and averaging are applied to analyze the syllables or notes in one song-type of an individual male swamp sparrow. The song consists of repetitions of a two note syllable composed of note called type II and IV. The results show the usefulness of the methods and reveal new details in the description of the swamp sparrow's set of note types and in the decrease of note variability during the developmental transitions from sub-song, to song crystallization.

Melinger and Clark (2000) extended the spectrogram correlation method to address the recognition problem of the end notes of bowhead whale songs. The notes are portions of a bowhead's song that occur one or more times in succession at the end of each song repetitions. They are distinctly different from the preceding portion of the song. This end notes are chosen because they are relatively loud and typically occur several times per song (Clark, 1991).

The method operates on spectrograms computed from the time-series waveform of a sound. Examples of the sound-type (in this case bowhead song end note) are used to construct a correlation kernel for the vocalization. A kernel is a two dimensional image, consists of several segments. Each segment represents each frequency sweep of the desired vocalizations.

To recognize the vocalization of interest in a recording, one makes a spectrogram of the recording and cross-correlates with the kernel representing the signal of interest. The result of this cross-correlation is a recognition function. It is a time series of recognition values that represents the closeness of the match between the kernel and the recording at each time increment of the spectrogram. Larger value in the time-varying recognition function represents higher likelihood that a bowhead song end note presents in the recording.

Mellinger and Clark tested the method by comparing the performance with a matched filter and an HMM approaches. Match filtering (van Trees, 1968) is a method for detecting a signal in a white Gaussian noise. The match filter kernel is constructed from signals of several high-quality vocalizations. All time series are placed in a matrix. The method calculates a sample covariance matrix, and computes the eigenvector corresponding to the maximum eigenvalue. In order to produce output function, the resulting kernel is cross-correlated with the test vocalizations. The correlation score is derived from the maximum correlation value of a given vocalizations.

The comparative results show that matched filter work poorly in recognizing bowhead song end notes; the HMM works fairly well, and the spectrogram correlation offers the best results.

Measures based on the full spectrogram suffer from a fundamental problem of high dimensionality of the basic features (Tchernichovski *et al*., 2000). Cross-correlation between vocalizations can be useful if vocalization is first partitioned into its notes or syllables and if the notes compared are simple.

The partition of vocalization into syllables or notes, however, in itself can be a problem. Partitioning a song into syllables is relatively straightforward in a song where syllables are always preceded and followed by a silent interval, such as the canary *Servinus canania* (Nottebohm and Nottebohm, 1978). It is more difficult in the zebra finch, where song includes many changes in frequency modulation and in which diverse sounds often follow each other without intervening silent intervals.

Some studies address the above problems by reducing complex sounds to an array of simple features and by implementing algorithms that do not require partition of the vocalization into its syllable component.

Dynamic time warping (DTW)

As mentioned in the previous section, Clark *et al* (1987, 2002) suggest a quantitative measure of vocalization similarity using the entire spectrogram of the signal. The magnitudes of the spectrograms of the two vocalizations are cross-correlated temporally and the peak value of the correlation gives a relative value of similarity. The method, however, is not appropriate for some animal vocalization because it does not account for the possibility of time dilation of the vocalization (Buck, Tyack, 1993).

The animal vocalizations are not always produced with temporal consistency. The length of a specific vocalization may vary between occurrences. A dynamic time warping (DTW) method, widely used in the early of speech recognition, allows limited compression and expansion of the time axis to align the vocalizations and provides a quantitative distance measure between vocalizations.

In marine mammals, DTW was first used by Buck and Tyack (1993) to classify 15 dolphin signature whistles into 5 groups. Later Brown *et al* (2007) use DTW to measure the dissimilarity of killer whale calls and to classify the calls using frequency contours of their biphonic vocalizations. Killer whale pulsed calls contain two overlapping but independently modulated contours or voices. This biphonation feature is common in birds but has been described for few marine mammal sounds (Tyson, 2006). The challenge in analyzing this complex sound is to determine the fundamental frequency of these two components from the same sound. Brown *et al.* separate low and high frequency contours for analysis, and then employ DTW for automatic classification of curves of different lengths. They employ four different cost matrices in DTW, namely, Ellis method, Sakoe-Chiba method, Itakura approach and Chai Vercoe method (Brown, 2007). For each method the distances given by the dissimilarity matrices are transformed into a Euclidean-like space using multi-dimensional scaling. They are then clustered using a *k*-means algorithm into seven call-types to compare to the perceptual classification.

Self Organizing Map (SOM)

The self organizing map (SOM) of Kohonen (1990) is a clustering and visualization tool that enables the organization of data in an unsupervised manner. It projects high-dimensional data into a set of models located at the nodes of a low-dimensional grid. The similar data patterns in the input space, therefore, will be assigned to the same map unit (node) or nearby units on the trained map.

The SOM has widely applied in image processing (Dong, Xie, 2005), process monitoring and control (Kasslin *et al.*, 1992), speech recognition (Guterman *et al.*, 2002), and flaw detection in machinery (Vapola *et al.*, 1994). Recently, applications in other fields have emerged including information retrieval (Kohonen *et al.*, 2000), medical diagnosis (Chen *et al.*, 2000), time-series prediction (Bareto, 2004) and bioacoustics (Placer *et al.*, 2006).

The construction of a SOM is based on competitive learning and the use of neighborhood when adapting the models. The training of a SOM involves two steps, namely, determining the best-matching unit (BMU) and updating the BMU and its neighbors.

Determining the best-matching unit

Let an input data be a high-dimensional vector of real number $X = \{x_1, x_2, \ldots x_n\}$ in the Euclidean space where x_i is the value of ith component. Each unit in an associated SOM for a n-dimensional training data set is also a n-dimensional real vector $m_i = \{ m_{i1}, m_{i2}, \ldots m_{in}\}$ where m_{ik} is the value of k-th component.

Generally, a distance function is employed to measure the similarity. The BMU of an input x is defined as $m_v = \arg\min_i \{d(x,m_i)\}$ where m_i is a unit on the map. A method to compute the distance $d(x,m_i)$ is typically using the Euclidean distance function

$$d(x,m_i) = \|x - m_i\| = \left(\sum_{k=1}^{n} (x_k - m_{ik})^2\right)^{1/2} \qquad (2.61)$$

Updating BMU and its neighbors

The BMU and its neighbors are updated to reduce the difference with the input pattern, once the BMU is identified. The updating is centered at the BMU, and the adjustment amount decreases with the increasing distance to the BMU. Similarly, the update neighborhood also decreases with the increasing training epochs. The update rule for a neighborhood unit m_i is as follows:

$$m_i(t+1) = m_i(t) + \alpha(t)\, h_{j,i}(t)\{x(t)\text{-}\, m_i(t)\} \qquad (2.62)$$

where $0 < \alpha(t) < i$ is the learning-rate function and $h_{j,i}(t)$ is the neighborhood function. Both $\alpha(t)$ and $h_{j,i}(t)$ decrease gradually with the increasing step t.

Placer *et al.* (2006) employ a SOM to identify acoustic units of Gunnison's prairie dog alarm calls in the presence of three different predator species. The approach allows individual calls to be classified by its predator species. A SOM is trained to identify clusters of acoustic units in Gunnison's prairie dog alarm calls where each cluster contains sounds with similar acoustic properties. Individual sounds belonging to specific clusters as well as combination of these sounds are found to be associated exclusively with alarm calls vocalized in the presence of a specific predator species.

Sumervou and Harma (2000), meanwhile, present a method to organize and visualize the syllables of 5 species of Phylloscopus birds. Each syllable is represented as a sequence of two-dimensional feature vectors. One component represents the instantaneous frequency and the other represents the amplitude. The organization of syllables utilizes DTW to compute the pair-wise distance between data sequences, and SOM is employed to visualize the data. They use eigenvector decomposition to project the high-dimensional feature vectors into lower dimensional space before training the SOM with fixed-dimensional model vectors.

2.5.2. Animal population estimation

Some studies of animal populations require estimates of population density *D* or size *N,* or, the rate of population change $\lambda_t = D_{t+1}/\mathrm{D}_t = \mathrm{N}_{t+1}/\mathrm{N}_t$. These estimates vary in time and over space as well as by species, sex and age.

This section presents two main approaches for estimating animal population, the mark-recapture method and the distance sampling method (line transect and point transect). The method of estimates described in this section assumes the population to be ‘closed’. That is, there are no gains (births or immigration) or losses (deaths or emigration) during the course of the study. To minimize the chances of losing or gaining individuals, the study should be over a short period and at a time of the year when births, deaths and movements are likely to be few. The discussion is mostly based on Buckland *et al*. (2004) and Borchers *et al*. (2004). The first part of this section presents the maximum likelihood framework in estimating abundance, followed by the distance sampling method, and the mark-recapture approach.

2.5.2.1. Using likelihood for estimating animal population

Buckland *et al*. and Borchers *et al*. address the method of estimating animal population using the maximum likelihood estimation (MLE) framework. To construct the likelihood function one needs to formulate the estimation problem in statistical terms. The whole region to estimate abundance is called the survey region. The number of animals in the survey region is *N*, and the number of animals detected is *n*. Assume that the probability of detecting any animal in the survey region is *p*, and that the detections are independent events.

A survey can be seen as consisting of *N* independent trials, where each animal is a trial. When the animal is detected, it is a “successful” outcome; otherwise, it is a “failure”. These are the conditions underlying the derivation of the binomial distribution. In this case *n* is a binomial random variable with parameters *N* and *p*. The probability density function (pdf) that gives the probability of getting exactly *n* successes on the survey is

$$f(n;N,p)=\binom{N}{n}p^{n}(1-p)^{N-n} \tag{2.63}$$

where $\begin{pmatrix} N \\ n \end{pmatrix}$ is the number of ways for choosing the n animals to appear in the sample from the N animals in the population. When the survey is done, one knows the number of n. Assume that the expected proportion of the detected animal is $p = 0.5$; then the only unknown quantity is N.

Equation (2.63) may be expressed as the likelihood function of N as follows

$$L(N \mid n, p) = \begin{pmatrix} N \\ n \end{pmatrix} p^n (1-p)^{N-n} \qquad (2.64)$$

The difference between equation (2.63) and (2.64) is that N is treated as a fixed constant in equation (2.62). One needs to plot and evaluate the likelihood as a function of N in equation (2.64). The N where the likelihood function of equation (2.64) achieves its maximum value is the maximum likelihood estimator (MLE) of N.

2.5.2.2. Simple mark-recapture

The basic idea of the mark-recapture method is quite simple. It involves capturing animals, marking them and releasing them back into the population. A second sample is taken some time later. The population estimate is calculated from the ratio of marked to unmarked animals in the second sample.

Suppose n_1 is the number of animals first marked and released, n_2 is the size of the second sample, m_2 is the number of marked animals in the second sample. If N is the total population size, the mark-recapture method estimates that $m_2/n_2 = n_1/N$. Since n_1, n_2, m_2 are known, it is obvious that N can be easily estimated. Most mark-recapture methods rest on that idea, though the animals may be caught or marked on several occasions.

It is interesting to note that any estimate derived from this approach is only representative of that fraction of the population that can be caught. Members of population that can not be sampled by a particular method do not take part in the estimate.

Animals can be captured and marked in a variety ways. Capture might be physical capture or simply detection (by eye, satellite, radio). Marking might involve physically attaching a mark to the animal, using acoustically or naturally occurring markers based on variation in phenotypes of genotypes, or notional marking of the animal by its location at a given time. McGregor and Peake (1998) list details on marking approaches that render animals individually distinctive to an observer.

A two-sample mark-recapture involves one session of catching and marking, and one session of recapturing. One might assume that all animals in the population are at risk of being caught, and capture probability (p) does not depend on individual animal characteristics. The likelihood, then, consists only of an observation (capture) model. Buckland *et al*. underline two main assumptions, namely, that (a) all animals are equally catch-able on any one survey occasion. (b) Detections of animals are independent events, both within a survey and between surveys.

This section now derives a likelihood for population estimate N, the probability of catching animal p_1 and p_2, given the number of marked animals in the population.

First capture occasion

On the first survey, animals are assumed to be captured independently with equal probability p_1. The probability of catching $n_1 = u_1$ unmarked animals, given that there are $U_1 = N$ animals in the population is

$$P_1 = \binom{U_1}{u_1} p_1^{u_1} (1 - p_1)^{U_1 - u_1} \tag{2.65}$$

The likelihood function for the first capture is a function of $N = U_1$ and p_1.

Second capture occasion

In the first survey one catches animals in the population, marks them and releases them. By the time of the second survey the population is split into two types: marked and unmarked. The number of marked animals by the start of the second occasion is the number of animals captured on the first occasion, namely, $M_2 = n_1 = u_1$. The outcome of the second survey, therefore, is not just the number captured on the survey (n_2), but both the known M_2 marked animals that were captured (m_2) and how many unknown $U_2 = (N\text{-}M_2)$ unmarked animals were captured (u_2). The probability of observing $n_2 = (u_2, m_2)^T$ is therefore a product of two binomial likelihoods. The first is the probability of catching marked animals m_2:

$$P_{2m} = \binom{M_2}{m_2} p_2^{m_2} (1 - p_2)^{M_2 - m_2} \tag{2.66}$$

Second, the probability of unmarked animals which is as follows:

$$P_{2u} = \binom{U_2}{m_2} p_2^{u_2} (1 - p_2)^{U_2 - u_2} \tag{2.67}$$

The combination of equation (2.65) and (2.66) to estimate p_2 and N is:

$$L_2 = P_{2u} \times P_{2m}$$

$$= \binom{U_2}{m_2} p_2^{u_2} (1-p_2)^{U_2-u_2} \binom{M_2}{m_2} p_2^{m_2} (1-p_2)^{M_2-m_2} \tag{2.68}$$

Putting the two capture occasions together

The estimate of N, p_1, p_2 simultaneously from the full likelihood for both capture occasions, given the observed data u_1, m_2 and u_2 can be written as

$$L = \prod_{s=1}^{2} P_{su} \times P_{sm}$$

$$= \prod_{s=1}^{2} \binom{U_s}{u_s} p_s^{u_s} [1-p_s]^{U_s-u_s} \binom{M_s}{m_s} p_s^{m_s} [1-p_s]^{M_s-m_s} \tag{2.69}$$

Interval estimation

Most uncertainty in simple mark-recapture estimates arises from the observation process. Variance and confidence interval estimates are based on the observation model alone. Profile likelihood confidence intervals can be constructed using the probability density function. For the case in which the capture probability is the same on both capture occasions, the profile likelihood is obtained by evaluating equation (2.69) with $p_1 = p_2 = p$ at

$$p_1(N) = p_2(N) = p(N) = \frac{n_1 + n_2}{2N} \tag{2.70}$$

Alternatively, one can use the following approximately unbiased estimator of the variance of Chapman's modified estimator, due to Seber (1970) and Wittes (1972)

$$\text{Var}\,[\,\hat{N}\,] = \frac{(n_1+1)(n_2+1)(n_1-m_2)(n_2-m_2)}{(m_2+1)^2(m_2+2)} \tag{2.71}$$

A likelihood for multiple capture occasions

The likelihood of equation (2.69) may be extended to more than two occasions simply by taking the product over all occasions. In general, for *S* occasions

$$L_S \quad = \prod_{s=1}^{S} L_S \; = \; \prod_{s=1}^{S} P_{su} \times P_{sm}$$

$$= \prod_{s=1}^{S} \binom{U_s}{u_s} p_s^{u_s} (1-p_s)^{U_s-u_s} \binom{M_s}{m_s} p_s^{m_s} (1-p_s)^{M_s-m_s} \tag{2.72}$$

Assessment of the method

More information can be collected from a study population when individual animals can be identified. The mark-recapture method fulfills this task, namely that detailed information can be recorded on each captured animal. Survival rates may be estimated; and recaptures provide information on animal movement.

The method is also preferable for populations of animals that may not be sufficiently detectable in a sighting survey, perhaps because they are small, are hidden amid vegetation, move away from the observer before they can be detected or identified, or spend much time in the ground.

The mark-recapture method provides invaluable information, but may be inappropriate for a variety of reasons. Marks, for example, may modify the normal behavior and physiology of animals and affect their survival. Marking needs catching. Any capture method has the potential to produce biased data because it will preferentially catch particular animals. Capture may impose costs on the animal, generally in the form of direct physical injury from the catching and holding equipment. Some species may be difficult to catch or difficult to track post-release. It may be desirable to avoid any disturbance associated with the capture.

2.5.2.3. Distance sampling

Distance sampling covers several related methods that involve measuring the distances of detected animals from a line or point. The methods estimate animal abundance in two steps, namely:

1. estimate the number of animals in the covered region $\hat{N}_c$

   $$\hat{N}_c = \frac{n}{\hat{p}}$$

   where *n* is the number of detected animals, and $\hat{p}$ is the detection probability (to be estimated)

2. estimate the animal abundance in the survey area $\hat{N}$ using the number of animal in the covered region $\hat{N}_c$

   $$\hat{N} = \frac{\hat{N}_c}{\pi_c}$$

   where π_c is the known coverage probability.

Putting the two steps together result in

$$\hat{N} = \frac{\hat{N}_c}{\pi_c} = \frac{n}{\pi_c \hat{p}} \qquad (2.73)$$

The key idea in the distance sampling is to estimate the probability of detection *p* by modeling the decline in detection frequency with distance.

There are two main methods of distance sampling, line transect sampling and point transect sampling. A survey using line transect sampling searches *J* strip areas of width 2*w*. Let the strip *j* has length l_j with $\sum_{j=1}^{J} l_j = L$. The size of the covered region in this study is $a = 2wL$.

The observer in line transect sampling travels along a centerline, and records each detected animal with the perpendicular distance *x* from the line. All animals on or near the centerline should be detected, and a proportion of animals within distance *w* of the line may be missed. Probability of detection, therefore, may decrease with distance from the line, out to some distance *w*. This method is called a distance sampling because it samples distances of animals from a line.

Point transect sampling, meanwhile, can be considered as a line transect of zero length (i.e., a point). Assume a series of *k* points randomly positioned in the survey area. An observer measures the distance *x* of each animal from a point. Upon completion of the survey there are *n* distance measurements of the animals. In point transect sampling, area closed to the point are censused. A proportion of animals away from the point but within the sampled area may remain undetected.

The important element in distance sampling is the detection function $g(x)$. This function is the probability of detecting an animal that is at a perpendicular distance *x* from the center line (for line transect method) or at a radial distance *x* from a point (for point transect method). The function $g(0) = 1$ when all animals on the line or at the point close to the observer are detected. The expected proportion of animals detected within a strip in line transect is:

$$p = \frac{\int_0^w g(x)dx}{w} \tag{2.74}$$

and the abundance estimation is

$$D = \frac{n/p}{2wL}. \tag{2.75}$$

For point transect method the expected proportion of animals detected is

$$p = \frac{\int_0^w xg(x)dx}{w^2} \tag{2.76}$$

and the abundance estimation for *k* points in the point transect is

$$D = \frac{n/p}{k\pi w^2} \tag{2.77}$$

Assessment of the methods

A line transect sampling method is mostly used for estimating the abundance of cetaceans and large terrestrial mammals; line and point transects for birds.

The line transect method is useful for a population in which animals are readily detectable if they are close to the observer, and useful for sparsely distributed populations. The method is also particularly suited for large survey regions. A high proportion of time in the field is typically spent on “effort” (namely, searching for animals from the line), so the method makes efficient use of resources. Further, overall precision is largely determined by the number of animals detected, not by the size of the population.

Point transect method, meanwhile, is used mostly for songbirds, although it has been applied to spotlight counts of hares and foxes and to estimate number of rare species of trees in tropical rain forests. For songbirds, there are several advantages to point transect sampling over line transect approach. In difficult terrain, it is easier to stand at a point and record birds rather than walk along a line. In places where accesses are difficult, it is advisable to locate and get to a random point rather than navigate along a random line. The disadvantages of this method are: random movement of birds generate greater bias for point transects, more time is spent "off-effort" – traveling from one point to the next. A larger area sometimes is needed because the covered area close to the observer is smaller.

2.6. Summary

The study presented in this dissertation borrows methods and concepts from human speech processing. The two supervised tasks in speech processing that are most closely associated with this research - speech recognition and speaker identification - along with the use of HMMs and GMMs have been introduced.

The idea of an unsupervised task of clustering using HMMs has been discussed as well. The discussion addresses two clustering methods: HMM-based k-model clustering and HMM-based hierarchical agglomerative clustering. The approach incorporates the dissimilarity analysis and delta-BIC computation to estimate number of clusters in a data set, as well as supervised and unsupervised methods to assess the validity of clustering results.

The next chapter will address the question of which among the features or combination of features employed in the above tasks are fit for the vocalization recognition and which are better suited for the individual identification.

In the following chapters the HMM-based supervised recognition – automatic speech recognition and speaker recognition – along with HMM-based unsupervised clustering approach will be integrated as a framework and applied to the task of animal population structure assessment and animal abundance estimation.

CHAPTER 3

FEATURE ANALYSIS

3.1. Introduction

As mentioned in the previous chapter, the overall purpose of feature extraction is to parameterize a vocalization into a sequence of feature vectors that contain concise and relevant information about the sequence of sounds within the vocalization. The features extracted are expected to be able to discriminate similar vocalizations, able to create models without the need for an excessive training data; and have statistical properties which are invariant across vocalizations and over a wide range of environment (Rabiner *et al*., 1996).

This chapter describes experimental comparison of features for song-type recognition and individual identification animal vocalizations. The overall goal is to find out which features are most appropriate for the song-type recognition and which features are significant for the individual identification task.

The organization of the chapter is as follows. Section two briefly presents some characteristics of the study population, ortolan bunting. Section three overviews a HMM-based method of vocalization recognition and individual identification, focusing on the training of the models and the use of the models for recognition or classification. Section four presents the results and discussion. Section five concludes with a short summary.

3.2. Ortolan bunting data

The subject for this feature study is the ortolan bunting. Their vocalization data was collected from County Hedmark, Norway in May of 2001 and 2002 (Osiejuk *et al*., 2003). The most frequent song-types within the studied population were chosen, i.e., *ab*, *cd, gb*, *eb*, *huf, h, jufb, guf* and *ef*. All recordings were transferred to a PC using 48 kHz/16 bit sampling. For more detail description of the ortolan bunting data set, the reader is referred to Chapter 5.

3.3. Methods

This section deals with supervised recognition tasks. A supervised recognition is a task in which a set of data has been labeled with the correct classification. The labeled data is split into training data used to train the system, and a test set for evaluation. Two recognition tasks are investigated, including determination of song-type given a known repertoire, as well as identification of individual bird. As mentioned in the previous section in the context of this feature study, the objective is to find out the best feature for call-type recognition and the most appropriate feature for the individual identification.

3.3.1. Model training and call-type/individual recognition

This study uses syllables as unit models for recognition. Training of unit models consists of estimating the model parameters from a training set of vocalizations in which all of the relevant syllables are known to occur sufficiently often. The way in which training is performed greatly affects the overall recognition system performance, and having a large training set is a key factor.

The Hidden Markov Models (HMMs) are used to model each of the different call-types. As mentioned in Chapter 2, a HMM is a statistical model that uses a finite number of states and the associated state transition to jointly model the temporal and spectral variations of signals. For bird vocalizations, states represent the time sequence of a syllable as shown in Figure 3.1.

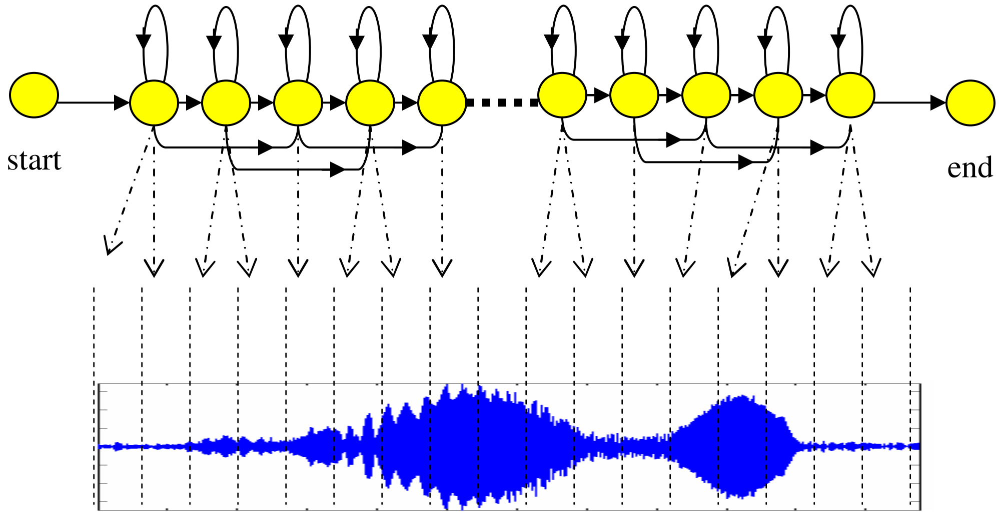

Figure 3.1. Markov model for ortolan bunting song

The task of a HMM is essentially to map a sequence of observations onto a sequence of states, and determine the likelihood that the observation could have been generated by that model.

A grammar model is constructed to model the variants of each call-type. The complete grammar can be depicted as a network as shown in Figure 3.2.

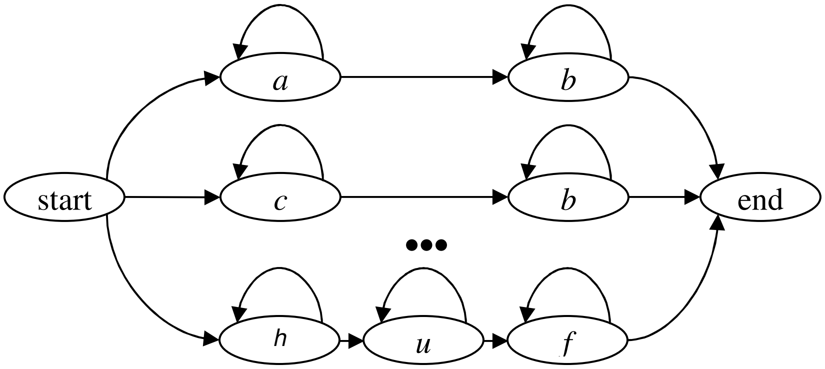

Figure 3.2. Language model for call-type in ortolan bunting

The above figure shows the topology of grammar constraints that allow only recognition of valid call variants. The ability of the HMM to incorporate such a constraint offers significant benefit for performance.

Figures 3.3 and 3.4 summarize the use of HMM for call-type and individual recognition tasks. First, a HMM is trained for each call-type or individual bird using a number of examples of that call-type or bird. This step estimates the model parameters (A, B, π) that optimize the likelihood of the training set for call-types or individual birds in the vocabulary.

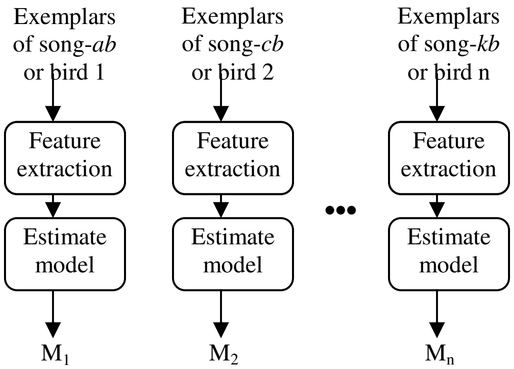

Figure 3.3. The use of HMM for model training

To recognize an unknown vocalization, the likelihood of each model generating that call-type or bird is calculated; and the most likely model identifies the call-type or bird.

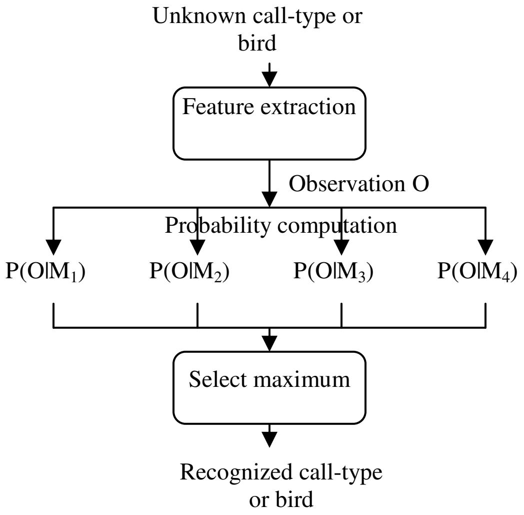

Figure 3.4. The use of HMM for recognition

To explore the application of HMMs to bird song, the programming toolkit Hidden Markov Model Toolkit (HTK) developed by Young *et al.* (2002) has been employed. HTK provides sets of tools that include the Baum-Welch re-estimation algorithm to compute the maximum-likelihood estimates of unknown parameters for training and the Viterbi algorithm to classify new vocalizations for the recognition.

Though HTK is primarily designed for speech recognition, in this research its tools will be adapted and used for analysis and recognition for bird vocalizations.

3.3.2. Assessment of the methods

The accuracy of the method is evaluated by comparing the test vocalizations that match the recognizer output with the correct reference transcription. For the analysis of call-type recognition and individual identification output, the comparison is performed using dynamic programming to align two transcription and then count substitution (*S*), deletion (*D*) and insertion (*I*) errors. The percentage number of labels correctly recognized is given by

$$\text{Percent Correct} = H/N \text{ x } 100\%$$

and the accuracy is computed by

$$\text{Accuracy} = (H\text{-}I)/N \text{ x } 100\%$$

where *H* is the number of correct labels and *N* is the total number of labels in the defining transcription files.

3.4. Experimental results and discussion

Features for the experiments consist of all features discussed in Chapter 2. These include the Greenwood function cepstral coefficients (GFCCs), delta (D) and acceleration (A), energy (E), cepstral mean normalization (MN), cepstral variance normalization (VN), and pitch (P). The repertoires are Hamming windowed with frame sizes varying from 2 to 6 ms and step sizes varying from 1 to 3 ms. The Greenwood frequency warping constants are calculated using an appropriate ortolan bunting hearing range of f_{min} 400 Hz to f_{max} 7400 Hz as determined by Edwards (1943). Classification and individual identification models are 15-state left-to-right HMMs with each state containing a single diagonal-covariance Gaussian. Silence models are added at the beginning and end of all vocalizations. The Baum-Welch expectation maximization algorithm is used to estimate the model parameters and the Viterbi algorithm is employed for recognition. The recognition is performed a number of time using various features and parameters to show the effect of each feature on recognition accuracy.

Song-type recognition

Speaker independent song-type classification experiments are performed across the eight most common song-types selected from data 2001 and data 2002. Each call-type contains multiple song-variants. The experiments employ 2,997 vocalizations of data 2002 as a training set for recognition of 1,190 vocalizations of data 2001.

Initially song-type recognition experiments are performed using GFCC features for window-size selection to find out the best window-size for recognition in terms of overall accuracy. Table 3.1 shows the accuracy of song-type recognition results on five different window-sizes. Of the different window-sizes, GFCC with window-size 3 ms and overlap 1.5 ms leads to the highest accuracy. This window-size is then selected for the rest of the song-type and individual bird recognition procedures.

No	Window size (ms)	Overlap (ms)	Accuracy (%)
1	2	1	75.55
2	**3**	**1.5**	**76.47**
3	4	2	75.55
4	5	2.5	71.51
5	6	3	71.59

Table 3.1. Song-type recognition accuracy for various window-sizes and frame step-sizes

Table 3.2 shows the accuracy results of song-type recognition with different features.

No	Features	Accuracy (%)
1	GFCC	76.47
2	GFCC_D	80.84
3	GFCC_D_A	82.60
4	GFCC_E	77.47
5	GFCC_E_D	82.26
6	GFCC_E_D_A	83.86
7	GFCC_E_D_A_MN	90.00
8	**GFCC_E_D_A_VN**	**91.59**
9	Pitch	72.94
10	Pitch_D	77.39
11	Pitch_D_A	78.82
12	GFCC_E_Pitch_D_A	86.22
13	GFCC_E_Pitch_D_A_MN	77.14
14	GFCC_E_Pitch_D_A_VN	82.60

Table 3.2. Song-type recognition accuracy results for various features

Results range from 72.94% with pitch feature to 91.59% for GFCC_E_D_A_VN feature. It can be observed that most cepstral features outperformed pitch feature sets. GFCC along with energy, delta, acceleration and variance normalization coefficients give HMMs the highest discriminant power, correctly classifying 91.59% of the test samples.

As discussed in the previous section, cepstral normalization is a method used to normalize the effect of convolutional noise in a system. By subtracting the mean cepstrum from the cepstral coefficients, convolutional noise can be removed assuming the noise is stationary throughout the vocalization. Cepstral variance normalization, meanwhile, has been argued effectively reduce the difference in probability density function between the clean and noisy speech signals. The HMM with pitch feature, meanwhile, can only correctly recognize 72.94% of the test song-type samples.

Table 3.3 presents the confusion matrix for GFCC_E_D_A_VN feature vector.

	ab	*cb*	*cd*	*eb*	*gb*	*guf*	*h*	*huf*
ab	200	0	0	0	0	0	0	0
cb	1	121	10	0	65	0	3	0
cd	0	1	199	0	0	0	0	0
eb	0	1	1	198	0	0	0	0
gb	0	12	2	0	134	0	1	1
guf	0	0	0	0	0	59	0	1
h	0	0	0	0	0	0	100	0
huf	0	0	1	0	0	0	0	79

Table 3.3. Song-type recognition with GFCC_E_D_A_VN feature

Each row of the above matrix represents the labeled class of the song-type while the columns represent the classification given by the system to each song-type. The numbers on the diagonal are the number of correctly classified vocalizations for each song-type. Song-type *cb* and *gb* seem the most difficult song-type to classify. The above result indicates the similar characteristics of call-type *cb* and *gb*. It can be expected that this acoustic similarity may create inconsistency in the song-type recognition.

Individual bird recognition

Song-type dependent individual bird identification experiments are performed using 100 exemplars of the most frequent song-type *ab* for each 9 individual birds. As indicated in the previous section, most of the features are extracted using a 3 ms Hamming window with 1.5 ms frame step size. The goal of this task is to show that individual birds can be identified from their acoustic features, and to find out which feature is best for individual bird recognition.

Five fold cross validation is used. The data is split into five approximately equal partitions and each in turn is used for testing and the reminder is for training, so that every vocalization has been used exactly once for testing.

Table 3.4 shows the results for each of the feature sets.

No	Features	Accuracy(%)
1	GFCC	77.44
2	GFCC_D	79.22
3	GFCC_D_A	80.00
4	GFCC_E	81.00
5	GFCC_E_D	81.44
6	**GFCC_E_D_A**	**82.00**
7	GFCC_E_D_A_MN	79.33
8	GFCC_E_D_A_VN	79.22
9	Pitch	59.11
10	Pitch_D	64.00
11	Pitch_D_A	66.89
12	GFCC_E_Pitch_D_A	74.33
13	GFCC_E_Pitch_D_A_MN	73.00
14	GFCC_E_Pitch_D_A_VN	71.56

Table 3.4. Individual bird recognition accuracy results for various features

The accuracy results vary from 59.11% to 82.00%. The HMMs with pitch feature shows poor performance, correctly recognizing only 59.11% of the test samples. The variance normalization that increases accuracy for song-type recognition experiments doesn't give significant discriminant ability to the system. The HMMs with GFCC coefficients together with energy and their dynamic coefficients delta and acceleration is able to correctly identify 82% of the individual birds in the test set, the best those presented.

Table 3.5 presents the confusion matrix for individual bird recognition using GFCC_E_D_A as its feature.

	Bird1	Bird2	Bird3	Bird4	Bird5	Bird6	Bird7	Bird8	Bird9
Bird1	97	0	0	0	1	1	0	0	1
Bird2	0	87	12	0	0	1	0	0	0
Bird3	0	22	78	0	0	0	0	0	0
Bird4	0	39	0	36	3	0	0	22	0
Bird5	9	0	0	0	80	8	0	2	1
Bird6	6	0	0	0	2	91	0	0	1
Bird7	0	0	0	0	0	0	96	2	2
Bird8	6	0	0	4	1	0	0	86	3
Bird9	9	1	0	0	1	0	2	0	87

Table 3.5. Individual bird recognition with GFCC_E_D_A feature

The system does a better task of identifying Bird1 and Bird7, and poorer effort of recognizing Bird4. Most samples of Bird4 are classified either as Bird2 or Bird8. It can be expected that the acoustic similarity in these three birds may create inconsistency in individual bird recognition.

Since most useful features vary by tasks, and since it is preferable to limit the number of feature used, this study implements feature selection for vocalization analysis. The experiments through the song-type recognition and individual bird identification show that the selection of features determines the separability of call-types and individual birds. The selection of the features has a large influence on the classification step and should be carefully considered in the system design, since the classifier must be tuned to the given feature space (Eriksson *et al.*, 2005).

3.5. Summary

This chapter addresses the question of which among the features or combination of features are fit for the repertoire recognition task and which are robust for an individual classification task. The study examines some feature extraction approaches such as the Greenwood function cepstral coefficients (GFCCs), pitch tracking, delta and acceleration computation, cepstral mean and cepstral variance normalization.

Some validations through song-type recognition and individual identification of ortolan bunting show the features that combine GFCC, energy (E), delta (D), acceleration (A) and variance normalization (VN) are best for call-type recognition, and therefore best for call-type clustering; and likewise, GFCC along with energy, delta and acceleration features give better discriminant power for individual bird recognition and for individual bird clustering.

Unfortunately, none of the features evaluated here had the property of being well-suited for one of these tasks and not the other, making the target task of unsupervised speaker clustering across multiple call-types much more challenging.

Feature extraction steps and feature selection processes discussed in this chapter, will be incorporated in the HMM-based supervised and unsupervised recognition as a framework applied to the task of population assessment discussed in the following chapters.

CHAPTER 4

AN ACOUSTIC ASSESSMENT OF THE BELUGA WHALE POPULATION STRUCTURE

4.1. Introduction

Free-ranging cetaceans which are visible above waters for only short periods of times are challenging research subjects. The lack of precision in most whale and dolphin studies is best exemplified by population estimates; it is currently not possible to count most populations with any degree of confidence. It is not easy to observe the changes in population characteristics such as birth and death rates in response to the changing of the environmental conditions.

Acoustics has been used to study marine mammals for decades. Only a few researchers, however, have attempted to examine the use of their vocalizations to assess populations. The term assessment is usually used to describe the process of evaluating the status of population relative to some management goal. This involves studies of the population structure, abundance and density, seasonal distribution and trends, and the evaluation of human-made noise impacts on the animals (Mellinger and Barlow, 2003).

Studies of cetaceans' population structure have mainly focused on the use of genetics, tagging and photo-identification (Barlow, 2003). The use of acoustic animal vocalizations observation can complement visual observation to provide more accurate estimates of the population. The best examples of the use of animal vocalizations in assessment are: the studies of sperm whale population (Barlow and Taylor, 1998), the humpback whales in the Caribbean (Garrison *et al*., 2003), and harbor porpoises in the Northwest Atlantic (Palka, 2003), where combined visual and acoustic methods have significantly improve the population estimate.

Visual and acoustic methods have different strengths and weaknesses. Visual techniques observe animals during periods of surfacing, whereas acoustic methods monitor submerged animals. Acoustic approaches can include data that are difficult to obtain with visual methods by providing continuous temporal coverage that are relatively independent of daylight and weather. It is, therefore, able to offer information on seasonal animal presence. Acoustic methods are often to be best applied to small areas. Visual surveys, on the other hand, are typically designed to cover a broad region as to provide a synoptic assessment of the total population (Hildebrand, 2003).

The objective of this chapter is to explore how the HMM-based framework discussed in the previous chapters might be utilized for marine mammal population structure assessment. This chapter is an effort to determine the relationship between established beluga social groups as indicated by their vocalizations. If acoustic differences between populations of marine mammals – in this case beluga whales - are closely connected to genetic differences, then their vocalizations' analysis would offer a relatively fast and inexpensive method to assess their social groups or their population structure (Mellinger and Barlow, 2003).

The chapter is organized as follows. Section two briefly presents some characteristics of the study population, beluga whales. Section three overviews a method to analyze repertoire similarity among established beluga social groups. The method integrates feature analysis to extract feature vectors of beluga vocalizations, dissimilarity analysis to estimate the number of different repertoires in the data sets, maximum variance initialization to initialize cluster models, HMM-based *k*-model clustering to group similar repertoires, and dissimilarity value computation to assess consistency of the clustering results. Section four presents the results and compares the repertoire grouping especially among wild and captive belugas. Section five concludes with a discussion and a short summary.

4.2. Beluga whales, *Delphinapterus leucas*

Cetaceans live in an environment in which vision is not the most important sense. They rely upon sound as their means of communication and assessment of their surroundings. They employ their vocalizations for echolocation, navigation and communication (Parson, Dolman, 2004).

Echolocation is the ability of animals to determine the physical features of the surroundings by producing mid or high-frequency vocalizations and detecting the echoes of sound that are reflected by distant objects. Cetaceans use echolocation to detect and catch prey and to observe the environment around them. Bottlenose dolphins make echolocation clicks of 50 to 130 kHz (Au, 1993), whereas porpoises produce echolocation clicks in frequencies of 110 up to 150 kHz (Kamminga and Wiersma, 1981).

Many cetaceans produce low frequency calls that theoretically could travel great distances. It has been suggested that mysticete whales, for example, use low frequency calls to navigate and orientate in a way similar to echolocation (Norris, 1969; Payne and

Webb, 1971). Such a form of direction-finding would seem essential for navigation on their long migrations.

Cetaceans communicate using acoustic signals within and between species. Their vocalizations have a variety of functions such as intra-sexual selection (maintain social orders within the sexes such as hierarchies of dominance and territory maintenance), inter-sexual selection (vocal calls to demonstrate fitness), mother-calf cohesion (communication to maintain social bonds between mother and calf), group cohesion (vocalization to co-operate, co-ordinate group members for foraging), individual recognition (calls that allowing individuals to identify relatives from alliances, and aid to coordinated behaviors), and danger avoidance.

Beluga whales live in Artic and sub-Artic waters. Some migrate south to warmer water during summer. They are a highly social species and congregate in pods (social group) of 2 – 25 whales, with an average pod size of 10 whales, consisting of both males and females or mothers and calves.

Beluga whales are one of the most vocal of the toothed whales (odontocentes). They produce a wide range of variable underwater calls that have been shown to vary according to their behavioral context (Sjare and Smith, 1986b; Bel'kovich and Sh'ekotov, 1992). Sjare and Smith (1986a, 1986b) divide wild beluga vocalizations into tonal calls (whistles), pulsed calls (clicks and pulsed tones) and noisy calls. Whistles or tonal calls are classified further by their frequency modulation and sequence of call components. They are sub-divided further into 7 contour types. The rate of vocalizations is influenced by changes in behavioral activities.

It is believed that beluga population is especially susceptible to the effect of human-made noise due to the large amount of shipping traffic, aircraft flight and sonar, drilling, construction and recreational boating (Sonstrom, 2007).

The vocalizations being examined in this research were made by a population of beluga whales residing in the St. Lawrence River estuary, Quebec, Canada. The data was selected from four study sites, namely, Baie St. Marquerite (BSM), Saguenay (SAG), Allouette (AL) and Channel Head (CH). They were recorded in July and August over the course of six years.

Repertoire recordings were taken from three different social groups. One set of vocalizations were from female beluga, another were from male beluga, both of which inhabit the Saguenay (SAG) area; and one vocalization set were from unknown social group residing at Channel Head (CH) location.

Table 4.1 shows seven data sets of beluga and total number of available vocalizations.

No	Year	Data name	Social group	Total number of vocalizations
1	1996	Data96	N/A	497
2	1996	Data96a	Wild adult male	659
3	2003	Data03	Wild adult female and young (BSM)	764
4	2005	Data05	Wild adult female and young (SAG)	60
5	2006	Data06	Wild adult and young (SAG)	130
6	2007	DataVM	Captive male Vancouver aquarium	137
7	2007	DataVF	Captive female Vancouver aquarium	58

Table 4.1. Beluga data sets used for the experiments

The acoustic data of wild beluga was collected with an omni-directional hydrophone and recorded on a Sony TCD-D8 digital audio tape (DAT) with 48 kHz sampling frequency and 16-bit quantization. For more detailed information about this beluga data, readers may refer to Sonstrom (2007).

The HMM-based clustering framework discussed here was used in Sonstrom (2007) to group similar vocalizations and to assess the social group identifications and breeding patterns of beluga whales in the St. Lawrence River estuary. To better understand the framework and clustering methods used, a block diagram to cluster beluga repertoire data will be given before the results of the experiments.

4.3. Methods

Figure 4.1 shows block diagram to cluster beluga vocalizations.

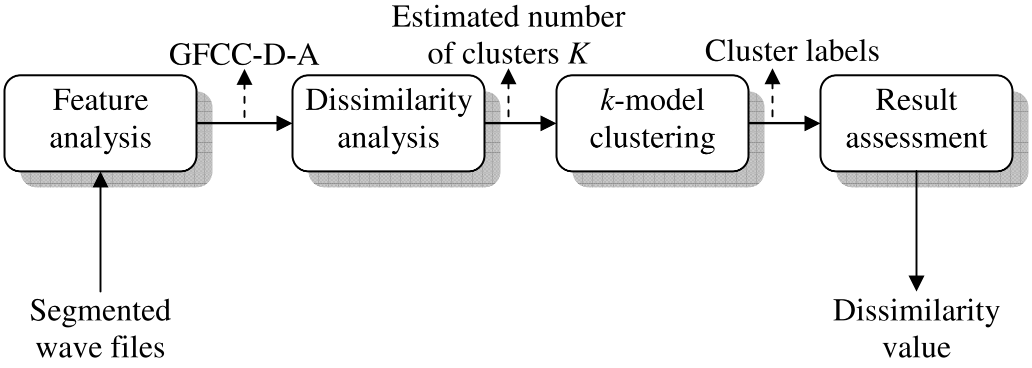

Figure 4.1. System block diagram

The feature analysis module provides the acoustic feature vectors used to characterize the spectral properties of the repertoire data. The dissimilarity index analysis element estimates the number of clusters *K* (in this case number of call-types) in the data; hidden

Markov model (HMM)-based *k*-model clustering groups data into *K* clusters, and dissimilarity computation evaluates the clustering results.

4.3.1. Feature analysis

Features for beluga repertoire clustering consist of the Greenwood function cepstral coefficient (GFCC) features discussed in Chapter 2. The vocalizations are segmented to construct quasi-stationary frames for accurate spectral estimation. A windowing function is applied to each frame to reduce the artifact that would arise from performing spectral analysis on a non-windowed frame. For the beluga repertoire, vocalizations are Hamming-windowed with frame size 30 ms and step-size 15 ms.

The appropriate Greenwood frequency warping constants are calculated using an approximate beluga hearing range of 100 Hz to 150 kHz as determined by Scheifele (2003). The GFCCs are normalized using cepstral mean normalization (CMN) and variance normalization (CVN) to remove channel distortion in the cepstral domain and to avoid the further amplification of low-frequency noise. Energy is not used due to recording variations. Thirty-six element feature vectors are extracted. They consist of cepstral coefficients along with delta and acceleration coefficients.

4.3.2. Dissimilarity analysis

The number of clusters in the beluga data sets is estimated from the cross-data dissimilarity analysis discussed in Chapter 2. The method varies the number of clusters *K* from 1 to 15. For each number of cluster $K \in \{1, 2, \ldots 15\}$, the following steps are performed:

1. estimate the label distance of equation (2.40) by averaging across a 20-way split of the data (refer to steps on dissimilarity analysis from Chapter 2).
2. normalize the label distance with its respected random labeling distance using equation (2.42)

The estimated number of clusters k = argmin$_k$ (normalized label distance).

4.3.3. HMM-based *k*-model clustering

A hidden Markov model based *k*-model clustering discussed in Chapter 2 is employed for beluga repertoire clustering. The assumption underlying an HMM-based method of clustering is that all repertoires that belong to a cluster are generated by the same HMM. This clustering algorithm is a hard clustering; each iteration every beluga repertoire is assigned to a single cluster represented by an HMM. The HMM parameter updates are influenced only by data items currently in the associated clusters.

Assume that the number of estimated cluster K is known from the previous step, namely the dissimilarity analysis process. Given K initial HMMs λ_1^0, λ_2^0, . . ., λ_K^0 the clustering algorithm proceeds as follows.

Iteration $t \in$ 1, 2, …

1. Data assignment

   for each beluga repertoire X_i, assign data to the model of maximum likelihood where $L(X_i \mid \lambda_k^{t-1})$ is maximal.

2. Model estimation

calculate new parameters of λ_1^t, λ_2^t, . . ., λ_K^t using data assigned to the models and using previous parameters λ_1^{t-1}, λ_2^{t-1}, . . ., λ_K^{t-1}

3. Termination

   Terminate the process if no label has changed.

In the above process, the re-assignment of the repertoires employs the Viterbi algorithm, and the re-estimation of model parameters utilizes a Baum-Welch re-estimation algorithm.

4.3.4. Cluster initialization

A suitable model topology, including the number of states and the allowed transitions in the HMM as well as the number of initial clustering models, should be motivated by the application. Generally, the topology remains unchanged during the training process (Knab *et al*., 2003).

Since the clustering algorithm will converge only to a local maximum, the choice of the model's initial parameters will have an impact on the maximum computed. The simplest approach is to set the initial models based on the global mean and variance of each element of the feature vectors and then use *K* copies of that model after adding small random perturbations to the parameters of the *K* copies individually. This approach, however, can easily lead to uneven cluster memberships, as some random model might have near zero probabilities of generating any repertoires in the data. This research, therefore, employs maximum variance initialization (Al-Daoud, 2005) with modification for HMM-based clustering. The algorithm for model initialization is as follows.

Algorithm : Cluster initialization

Input : random number *R*, number of clusters *K*

Output : *K* initial clusters

Steps :

1. Random initialization

   Set a random number of clusters *R*. Initialize *R* HMMs based on global mean and variance of the data samples

2. Create loglikelihood matrix of data over *R* HMMs
3. Compute the variance in each model
4. Find the column (model) with maximum variance, sort in any order
5. Divide the data-set of maximum variance into the desired number of clusters *K*
6. Train an HMM λ for each subset.

The *k*-model clustering in this research utilizes the maximum variance initialization discussed above. The experiments vary the random number of initial clusters from 5 to 25, and then choose the best for the maximum variance initialization approach.

Cluster models for the experiment are 15-state left-to-right HMMs with each state containing a single diagonal-covariance Gaussian. The silence model before and after the vocalizations is built using the same topology. The Hidden Markov Model Toolkit

(HTK) version 3.2.1 from Cambridge University (2002) is used with modification, to implement the HMM functionality.

4.3.5. Result assessment

The notion of dissimilarity can be used to assess the consistency of clustering results. The dissimilarity index computation to assess the consistency of clustering results is a generalization of the cross-data cluster dissimilarity analysis to estimate the number of clusters in a data set. To implement this, the clustering algorithm is run 10 times on the same beluga data set using different initial conditions. The average dissimilarity value is computed using equation (2.49). The smaller the multi-run dissimilarity value $\in$ [0, 1] the more consistent is the clustering algorithm across this data set.

4.4. Results and discussion

4.4.1. Initial parameters

Initially the dissimilarity metric is used for feature selection to find the best features for Beluga repertoire clustering in terms of overall dissimilarity. Figure 4.2 shows dissimilarity results on three different features: the cepstral coefficients GFCCDA, the mean-normalization cepstral coefficients (GFCCDA-MN), and the variance-normalization cepstral coefficients (GFCCDA-VN) across different number of clusters for beluga data06 (wild adult and young social group) data set. Of the three features, GFCCDA leads to the best performance. This feature is then selected for the rest of clustering procedure for beluga data sets.

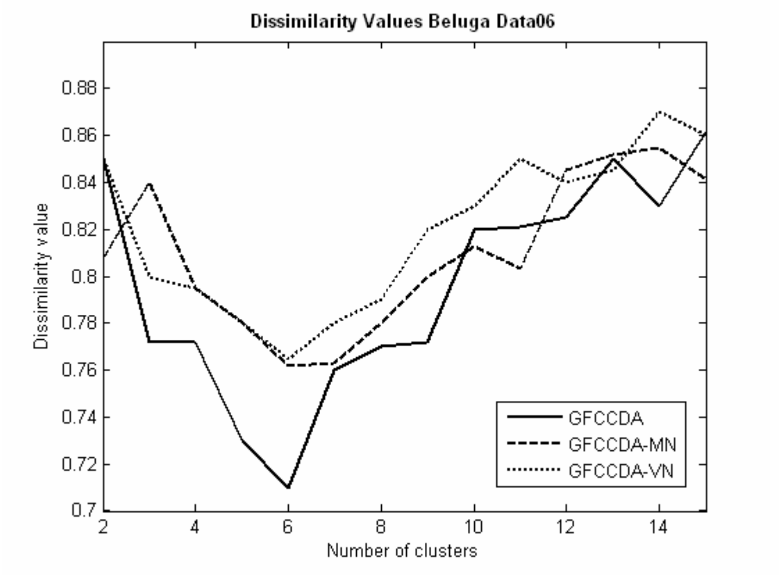

Figure 4.2. The dissimilarity index values of the beluga data06
from three different cepstral coefficient features

Additionally, the dissimilarity index computation is employed for feature selection to find out the best initial number of random value *R* for variance initialization of the *k*-model clustering, as discussed in the previous section. Table 4.2 presents the dissimilarity index values for differing number of initial *R* used for variance initialization over beluga data06 data set. Variance initialization with *R* = 10 yields the best dissimilarity index value. The *k*-model clusterings for beluga data then use *R* = 10 for their variance initialization value.

No	Variance Initialization (initial number of *R*)	Dissimilarity value
1	5	0.2387
2	10	**0.2111**
3	15	0.2165
4	20	0.2521
5	25	0.2354

Table 4.2. The dissimilarity values for different variance initialization over beluga data06

4.4.2. The estimated number of clusters

Figure 4.3 shows the use of the cross-data dissimilarity method to estimate the number of clusters *k* from four different beluga data sets. A total of two repertoire clusters are assigned to the data96a, three clusters to the data96, five clusters to the data03, and six clusters to data05 and data06 (wild adult and young beluga repertoires).

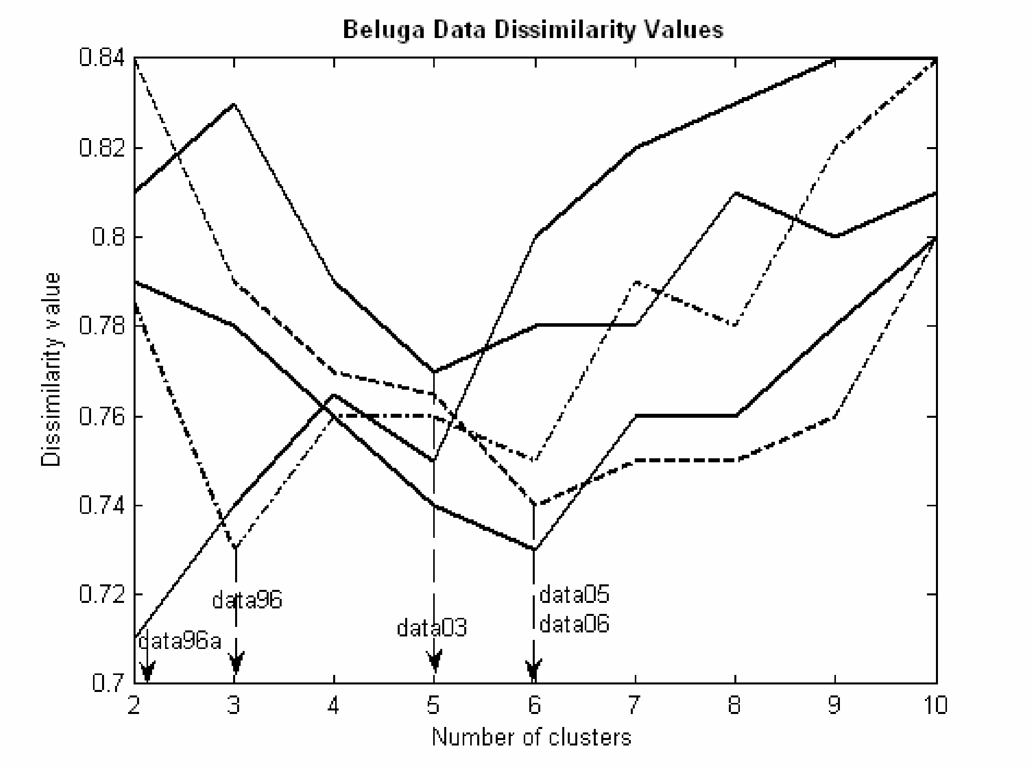

Figure 4.3. Cluster estimates for four different Beluga data sets.

To illustrate the consistency of clustering results, Table 4.3 presents the dissimilarity values and their respected standard deviations from 10 separate clustering runs from 5 different beluga data sets.

	Data	Number of clusters	Dissimilarity value
1	data96a	2	0.026±0.003
2	data96	3	0.130±0.012
3	data03	5	0.473±0.046
4	data05	6	0.359±0.049
5	data06	6	0.211±0.019

Table 4.3. The estimated number of clusters and dissimilarity values of 5 different beluga data sets

Results indicate that the dissimilarity value has a significant range for the different data sets. It gives an almost perfect match (0.026) for data96a (wild adult social group), and a relatively high dissimilarity value (0.473) for data03 (female and young social group of BSM). A dissimilarity value 0.026 means that for different runs 2.6% of the data are clustered inconsistently and 97.4% of the vocalizations are always assigned into the same clusters or groups. A dissimilarity value 0.473 for data03 shows that 47.3% of the data are grouped inconsistently while 52.7% are clustered to the same groups for different runs.

Inconsistent clustering runs as shown by a high dissimilarity value may indicate that there are a relatively large range of vocalization types in the data set with only a few example of each, so that data limitation prevents accurate grouping.

4.4.3. The similarity of repertoire among different social groups

As mentioned in the previous section, the objective of the study is to explore how the HMM-based framework might be utilized for beluga whale population structure assessment. The focus of this section is, therefore, to determine the relationship between established beluga social groups as indicated by their vocalizations. The study compares wild and captive beluga vocalizations to assess the characteristic of their social groups, and computes the loglikelihood distance to measure how similar their vocalizations are. The loglikelihood distance is calculated using Viterbi algorithm discussed in section 2.2.1.2. A total of 984 vocalizations are used. They consist of 130 vocalizations from wild adult and young beluga (data06), 659 from wild male (data96), 137 from captive females (dataVF), and 58 from captive male belugas (dataVM) of Vancouver aquarium.

Tables 4.4 – 4.10 represent the vocalization log-likelihood distances of wild and captive belugas. The log-likelihood tables show the similarity of vocalization groups expressed as distance metric values (per frame log-likelihood distance) from one cluster model (the row) to the vocalizations in another cluster (column).

Table 4.4 indicates that vocalizations in cluster 1 of wild adult beluga (data96) are more similar to vocalizations in cluster 5 of wild adult-young (data06) than any other clusters; and vocalizations in cluster 2 are closer to vocalizations in cluster 3 of data06.

		Wild adult-young belugas					
		C1	C2	C3	C4	C5	C6
Wild adult belugas	C1	-97.857	-99.582	-96.495	-100.65	**-95.908**	-98.466
	C2	-98.601	-100.16	**-96.872**	-101.03	-97.810	-99.176

Table 4.4. Log-likelihood distance between repertoires of wild adult and wild adult-young belugas

All wild adult-young beluga repertoires, meanwhile, fall within cluster 1 of wild adult male vocalizations (Table 4.5). This may imply that this group of adult beluga has a vocalization cluster (C1) that has characteristics unique to this group. Group specific vocal repertoires are common to several different Cetacean species (Sonstrom, 2007).

		Wild adult belugas	
		C1	C2
Wild adult young belugas	C1	**-102.037**	-131.284
	C2	**-116.042**	-149.075
	C3	**-110.204**	-143.572
	C4	**-112.668**	-143.897
	C5	**-118.543**	-147.695
	C6	**-108.871**	-139.212

Table 4.5. Log-likelihood distance between repertoires of wild adult-young and wild adult belugas

Tables 4.6 – 4.7 represent the similarity between wild adult-young and captive female repertoires. Most vocalizations of the captive female are similar to the vocalizations in cluster 3 and 5 of wild adult-young (Table 4.6). A large number of wild adult-young repertoires are closer to cluster 2 of the captive female repertoires (Table 4.7).

		Wild adult young belugas					
		C1	C2	C3	C4	C5	C6
Captive female belugas	C1	-84.989	-85.285	**-83.668**	-85.966	-84.595	-85.323
	C2	-83.423	-83.982	-82.654	-84.676	**-82.386**	-83.914
	C3	-84.432	-84.862	**-83.885**	-85.271	-84.189	-84.785
	C4	-83.765	-84.018	-83.870	-84.175	**-83.437**	-83.885
	C5	-82.319	-82.513	-81.182	-82.893	**-79.575**	-82.468

Table 4.6. Log-likelihood distance between repertoires of captive female and wild adult-young belugas

		Captive female belugas				
		C1	C2	C3	C4	C5
Wild adult young belugas	C1	**-89.435**	-89.940	-91.932	-92.658	-96.788
	C2	-90.332	**-88.949**	-91.651	-91.567	-105.783
	C3	-92.362	**-91.224**	-94.376	-94.561	-101.822
	C4	-92.777	**-91.587**	-94.604	-94.775	-106.272
	C5	-95.763	**-94.409**	-97.121	-97.227	-114.314
	C6	-90.718	**-89.111**	-92.272	-92.257	-100.246

Table 4.7. Log-likelihood distance between repertoires of wild adult-young and captive female belugas

Tables 4.8 – 4.9 show the distance between captive male and female repertoires. Table 4.8 indicates that repertoires in cluster 1 of captive males are closer to cluster 5 of captive females, cluster 2 to cluster 3, and cluster 3 to cluster 2. All repertoires of the captive females, meanwhile, fall into cluster 2 of male belugas (Table 4.9).

		Captive female belugas			
		C1	C2	C3	C5
Captive male belugas	C1	-60.896	-61.663	-60.891	**-59.646**
	C2	-69.559	-69.470	**-69.456**	-69.876
	C3	-79.406	**-78.591**	-79.935	-81.792

Table 4.8. Log-likelihood distance between repertoires of captive male and female belugas

		Captive male belugas		
		C1	C2	C3
Captive female belugas	C1	-68.207	**-67.724**	-68.465
	C2	-67.245	**-66.335**	-67.169
	C3	-69.276	**-68.857**	-69.513
	C5	-57.228	**-54.340**	-59.080

Table 4.9. Log-likelihood distance between repertoires of captive female and male belugas

The likelihood distances shown in the previous tables measure how close group vocalizations are among each other. The study is not only measure how close those vocalizations are, but also how separable those vocalizations are. The following section, therefore, investigates a supervised group repertoire recognition utilizing labels assigned to the vocalizations during clustering processes. The labeled vocalizations are split into training data used to train group vocalization models and test sets for evaluation.

Tables 4.10 – 4.12 show the confusion matrices from the classification experiments of wild and captive belugas. The confusion matrix gives the results of a classification experiment where the cluster models (the rows) are used to classify individual test vocalizations (the columns). The numbers on the diagonal are the number of correctly classified vocalizations for each repertoire cluster. Five fold cross validation is used. The data is split into five approximately equal partitions and each in turn is used for testing and the reminder is for training. That is, the experiment uses four fives for training and one five for testing, then repeats the procedure five times so that in the end every vocalization has been used exactly once for testing.

		Wild adult-young						Wild adult	
		C1	C2	C3	C4	C5	C6	C1	C2
Wild adult young	C1	15	1	0	0	1	1	0	0
	C2	1	21	0	0	5	2	0	0
	C3	1	0	8	4	2	1	0	0
	C4	1	2	2	4	6	1	0	0
	C5	2	7	2	3	14	4	0	0
	C6	0	3	0	1	6	5	0	0
Wild adult	C1	0	0	0	0	0	0	259	15
	C2	0	0	0	0	0	0	14	370

Table 4.10. Repertoire classification results of wild adult-young and wild beluga data (Accuracy 88.78%)

		Wild adult young						Captive female				
		C1	C2	C3	C4	C5	C6	C1	C2	C3	C4	C5
Wild adult young	C1	13	1	1	0	2	1	0	0	0	0	0
	C2	2	14	0	1	9	3	0	0	0	0	0
	C3	1	2	16	5	0	1	0	0	0	0	0
	C4	1	3	3	3	5	1	0	0	0	0	0
	C5	1	7	0	7	15	2	0	0	0	0	0
	C6	0	2	0	4	5	4	0	0	0	0	0
Captive female	C1	0	0	0	0	0	0	32	1	2	0	0
	C2	0	0	0	0	0	0	0	24	1	0	0
	C3	0	0	0	0	0	0	3	0	24	0	0
	C4	0	0	0	0	0	0	0	0	0	20	0
	C5	0	0	0	0	0	0	0	0	0	1	30

Table 4.11. Repertoire classification results of wild adult-young and captive female beluga data (Accuracy 70.45%)

		Captive female				Captive male		
		C1	C2	C3	C5	C1	C2	C3
Captive female	C1	32	1	2	0	0	0	0
	C2	2	20	3	0	0	0	0
	C3	2	0	25	0	0	0	0
	C5	0	0	0	30	1	0	0
Captive male	C1	0	1	0	2	5	0	0
	C2	0	0	2	1	0	27	0
	C3	0	0	0	0	0	0	9

Table 4.12. Repertoire classification results of captive female and male beluga data (Accuracy 89.70%)

The above matrices illustrate that each social group is different from one another and show that their vocalizations are separable. The vocalizations of the captive belugas are selected from known animals under a specific behavioral context. In the wild environment it is expected to observe more complex repertoires due to the beluga's ecological constraints and survival needs such as vocalization to maintain pod cohesion, navigation, find prey and avoid predator. As shown in the above results, the vocalization taken from a captive environment, therefore, are analyzed with more efficiency and clustered with higher accuracy.

4.5. Summary

This chapter shows how the HMM-based framework can be used to perform an unsupervised classification task to differentiate different vocal repertoires of different beluga whale social groups. The approach consists of the GFCC feature analysis to extract feature vectors of beluga whale vocalizations, the dissimilarity analysis to estimate the number of different repertoires in the data sets, HMM-based *k*-model clustering to group similar repertoires and dissimilarity value computation to assess consistency of the clustering results.

The results show the reliability of the method to acoustically identify the established social structure of the beluga whale population in the St. Lawrence River Estuary. The results also demonstrate the feasibility of the approach to assess, track and monitor social groups of the beluga whale population for potential conservation use.

CHAPTER 5

HMM-BASED ACOUSTIC CENSUSING OF THE ORTOLAN BUNTING POPULATION

5.1. Introduction

Individually distinct acoustic features have been observed in a wide range of vocally active animal species, for example: cetaceans (Janik *et al.*, 1994), bats (Master *et al.*, 1995), and primates (Butynski *et al.*, 1992). Within birds, the presence of vocal individuality has been shown in the European Bitterns and Black-throated Divers (Gilbert *et al.*, 1994), American Woodcock (Beightol and Samuel, 1973), Australian Kingfishers (Saunders and Wooller, 1988), and Tawny Owls (Galeotti and Pavan, 1991).

The feasibility of using vocal individuality (vocalizations) to monitor habitat quality has been demonstrated. Peake and McGregor (2001) employed a statistical Pearson-correlation approach to identify corncrake vocal individuality and to estimate numbers of individuals in species. Holschuh (2004) used discriminant function analysis to explore vocal individuality of the saw-whet owl to monitor its habitat quality. Terry and McGregor (2002) suggest a different method to monitor and census male corncrake species. They employ three different neural network models, namely, a backpropagation and probabilistic network to re-identify the members of the known population (monitoring task) and a Kohonen network to count a population of unknown size (census task).

The research presented here employs a well established automatic human speech recognition framework for the potential censusing and monitoring of animals. Previous and current studies show the feasibility of a hidden Markov models (HMMs) – based method to automatically classify ortolan bunting song-types, to identify individual birds (Adi, Michael Johnson, 2004, 2006; Trawicki, 2005), and to cluster Beluga repertoires (Adi *et al.*, 2008). The objective of this chapter, therefore, is to develop a method that integrates tasks of vocalization recognition, individual identification, and vocalization and individual clustering – discussed in the previous chapters – to estimate animal abundance. The suggested framework is based on Hidden Markov Models (HMMs) commonly used in the signal processing and automatic speech recognition community.

This chapter is organized as follows. Section two gives an overview of some characteristics of the study population, the ortolan bunting *Emberiza hortulana*. Section three introduces three possible scenarios for estimating the number of animals in a population, and discusses a method that combines the tasks of supervised classification and unsupervised clustering. Section four presents the experiment of each scenario, the results and some discussion. Section five concludes the chapter.

5.2. Ortolan bunting, *Emberiza hortulana*

The ortolan bunting is a migratory passerine bird distributed from Western Europe to Mongolia (Cramp, Perrins, 1994). They winter in Africa. The species inhabits open agricultural areas, raised peat bogs, clear-cut forest on poor sand, and cleared farmland and forest burn (Dale, Hagen, 1997; Dale, 2001b).

Ortolan buntings are monogamous birds. The females lay 3-5 eggs. They are ground breeders and feeders. The nests are usually placed on the ground. The diet consists of seeds obtained from farmland, some insects and other small invertebrates.

Ortolan buntings are classified as endangered species (Tucker, Heath, 1994; Storkersen, 1999). This species has shown a major population decline both in individual numbers and in their distribution. In Finland, Vepsalainen *et al*. (2005) studied their population density changes and environment associations in years 1984 – 2002. They observed a population crash between 1990 and 1993 resulting in a 54% reduction in population density. The decline continued steadily, giving a total density reduction of 72% between years of 1984 to 2002. The primary causes have been related to agricultural changes such as reductions in environmental heterogeneity and crop diversity and the consequent loss of breeding habitat. Vepsalainen *et al*. speculated that the European-wide decrease in the ortolan bunting is probably also due to changes in migration and wintering areas.

The Norwegian ortolan bunting, meanwhile, currently numbers approximately 150 singing males and has shown decline over the past fifty years as well. In years 1996 – 2000 the decline rate was 8% per year (Dale, 2001). The basic demographic parameters such as juvenile and adult survival, variation in habitat, mortality during migration, breeding success, nest losses, were within the normal range (Steifetten, 2006; Dale, 2001). Instead, the decline is most likely related to female-biased dispersal away from the population which results in many unpaired males and low population productivity (Dale 2001 a, b; Steifetten, 2006).

The ortolan bunting vocalizations being examined for this study were collected from County Hedmark, Norway in May of 2001 and 2002 (Osiejuk *et al*., 2003). The male vocalizations were recorded on 11 out of 25 sites within an area of about 500 km^2. The total number of males in the covered area of the years 2001 and 2002 was about 150.

The ortolan bunting has a relatively simple song and small repertoire size of typically 2 – 3 song-types for each individual. Song frequencies are in a range between 1.9 kHz and 6.7 kHz. Songs of ortolan buntings are described in terms of their syllable, song-type and song variant. In total, there are 63 different song types and 234 different song variants, composed of 20 different syllables.

A syllable is a minimal unit of song production. A song is described by using letter notation, e.g. *aaaabb* or *hhhhuff*, where letters denote particular syllables. Although syllables of the same category might differ in length and frequency between individuals, they have the same shape on sonograms.

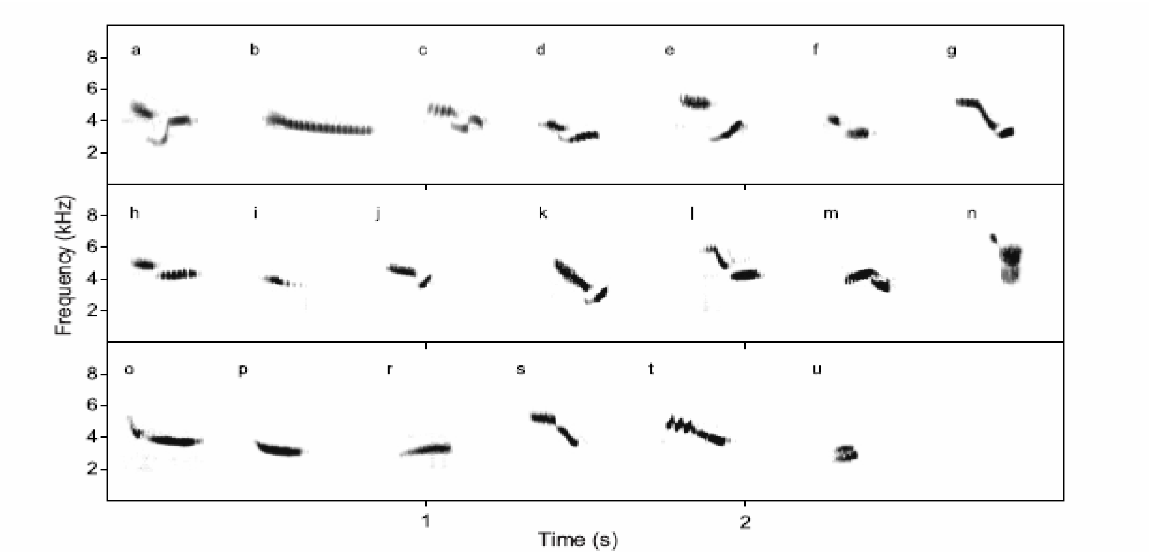

Figure 5.1. Ortolan bunting syllables (after Osiejuk, 2003)

A song-type is a group of songs that consist of the same syllables arranged in the same order. For example: *ab*-type (*aaabb*), *kb*-type (*kkkkkbb*).

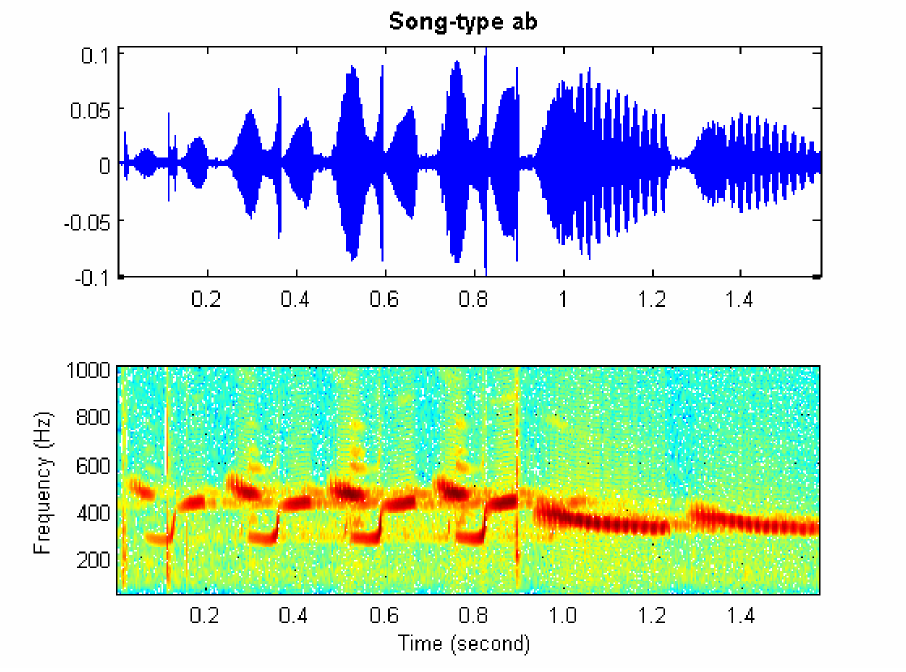

Figure 5.2. Time series and spectrogram of song-type *ab*

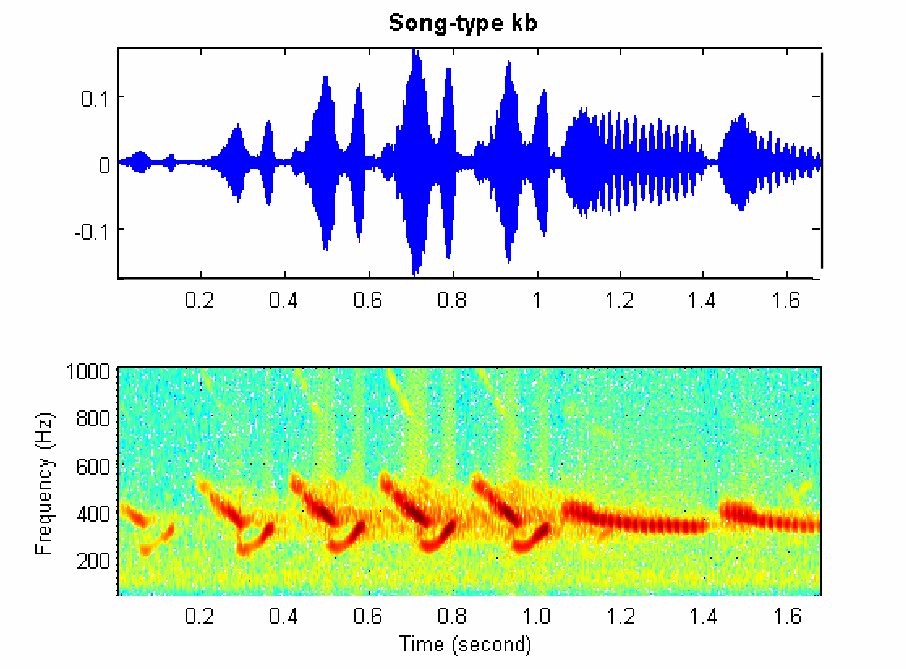

Figure 5.3. Time series and spectrogram of song-type *kb*

Songs of the same type, which differ only in the number of syllables within the songs, are termed song variant. For example, within song-type *gb*, many song variants might exists, e.g. *gggb*, *ggbbbb*, *gggbb*. The initial and final syllables may differ slightly in amplitude and frequency due to sound production mechanisms

As described by Osiejuk (2003), these ortolan vocalizations were recorded between 04:00 and 11:00 am by using Telinga V Pro and Sennheiser ME 67 recorder. All recordings were transferred to a PC using 48 kHz/16 bit sampling.

5.3. Methods

5.3.1. Three scenarios in estimating bird population

This research proposes three possible scenarios to estimate the number of animals in a population. Scenario 1 assumes data of known species, including some training data with song-type labels and speaker labels. Scenario 1 then estimates the animal abundance with integrated methods of song-type recognition, individual animal classification and individual animal clustering. Scenario 2 assumes that one has data sets of known species and available training data with repertoire labels, but without known labels of individual animals. In this scenario the problem of a population estimate is addressed using joint repertoire classification and individual animal clustering. Scenario 3 assumes that one has a known species data set without repertoire labels. In this case automatic animal censusing is approached using joint repertoire clustering and individual animal clustering methods.

Figure 5.4 presents a block diagram of estimating bird abundance utilizing Scenario 1. Scenario 1 assumes the availability of the data of known species, some training data with song-type labels and another data set with bird labels.

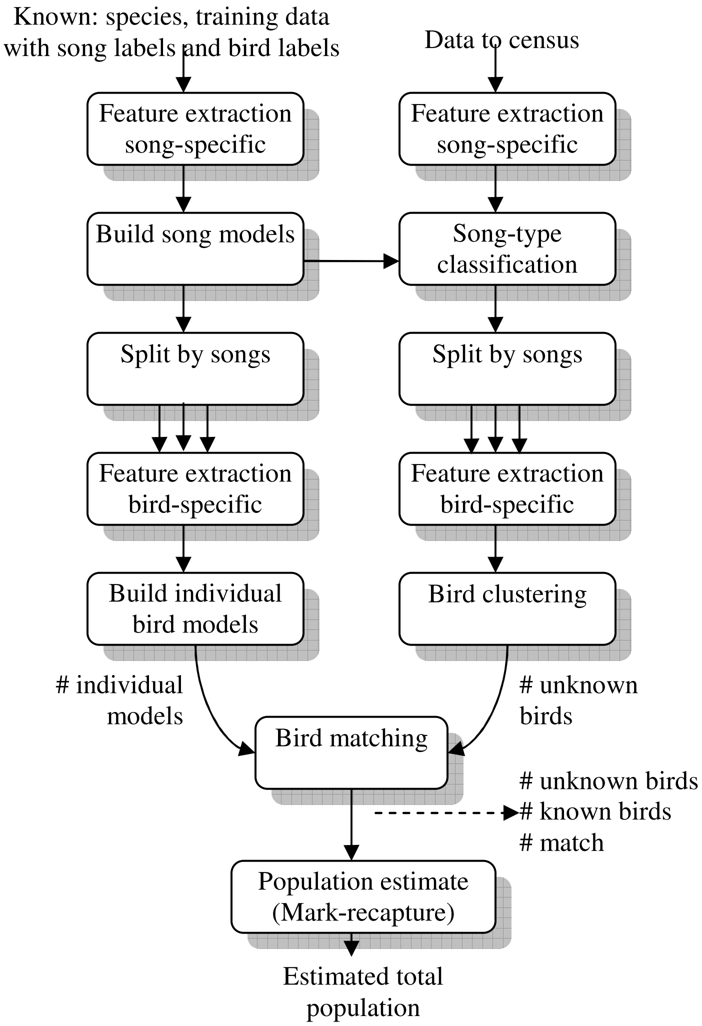
Known: species, training data with song labels and bird labels
Data to census
Feature extraction song-specific
Feature extraction song-specific
Build song models
Song-type classification
Split by songs
Split by songs
Feature extraction bird-specific
Feature extraction bird-specific
Build individual bird models
Bird clustering
# individual models
# unknown birds
Bird matching
# unknown birds
# known birds
# match
Population estimate (Mark-recapture)
Estimated total population

Figure 5.4. Block diagram of estimating bird abundance using Scenario 1

There are five tasks involved: song-type classification, bird recognition, bird clustering, bird matching and population size estimation. Song-type classification trains repertoire models using a labeled song-type data set and classifies data into groups of song-types. In each song-type data set, there are exemplars with bird labels and some exemplars without labels. For labeled data, the study proceeds with bird recognition experiments to classify birds in the data, and bird clustering experiments to estimate the number of birds in the unlabeled data set. Bird matching identifies birds present in both data sets. Using the information of birds "marked" in the first data set, the number of birds "recaptured" in the second data set, and the number of birds present in both, the population size estimate computes bird abundance using a simple mark-recapture method.

Figure 5.5 shows block diagram of estimating animal abundance using Scenario 2. In this scenario an HMM-based repertoire (or song-type) classification module trains song-type models using available labeled data sets and classifies unknown data into groups of song-types. HMM-based clustering module estimates the number of animals in each song-type. A population size estimate then computes the overall population.

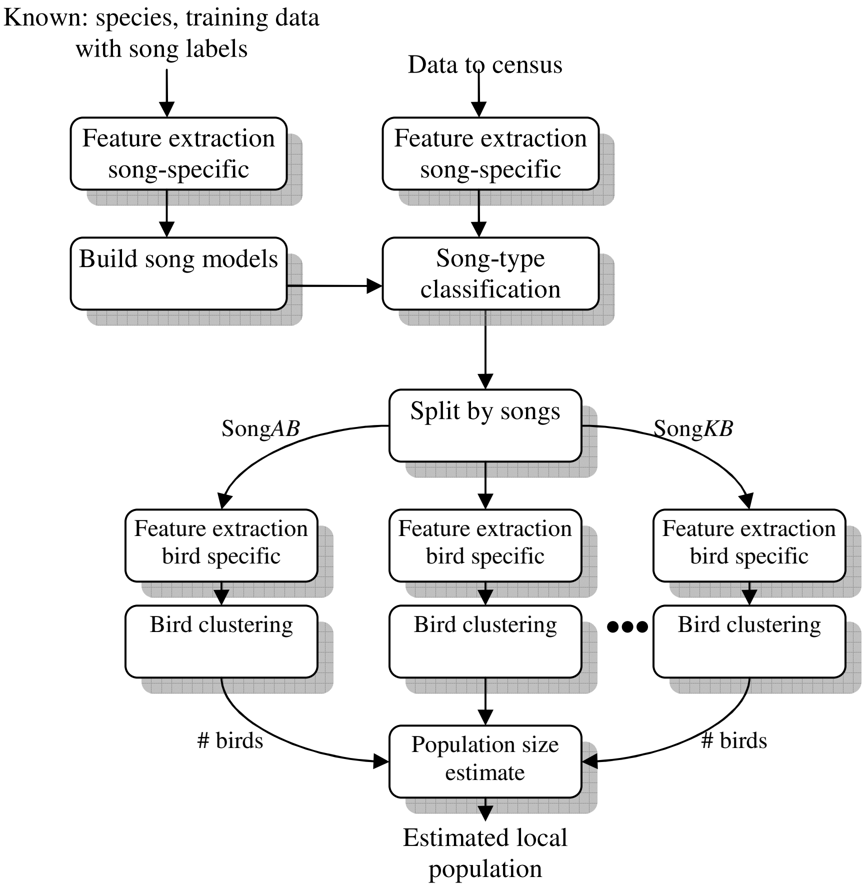
Known: species, training data with song labels
Data to census
Feature extraction song-specific
Feature extraction song-specific
Build song models
Song-type classification
Split by songs
SongAB
SongKB
Feature extraction bird specific
Feature extraction bird specific
Feature extraction bird specific
Bird clustering
Bird clustering
Bird clustering
●●●
# birds
# birds
Population size estimate
Estimated local population

Figure 5.5. Block diagram for bird population estimate using Scenario 2

Figure 5.6 presents a block diagram of animal abundance estimate utilizing Scenario 3. This scenario involves two processes: song-type clustering and individual animal clustering. Due to unknown training data, the song-type clustering starts with creating initial song-type models using some presumed small data as seeds. Using these initial HMM song-type models, the module groups data into *K* song-type clusters. An individual animal clustering component then estimates the number of animals in each song-type cluster. A population size estimate computes the animal abundance as a final result.

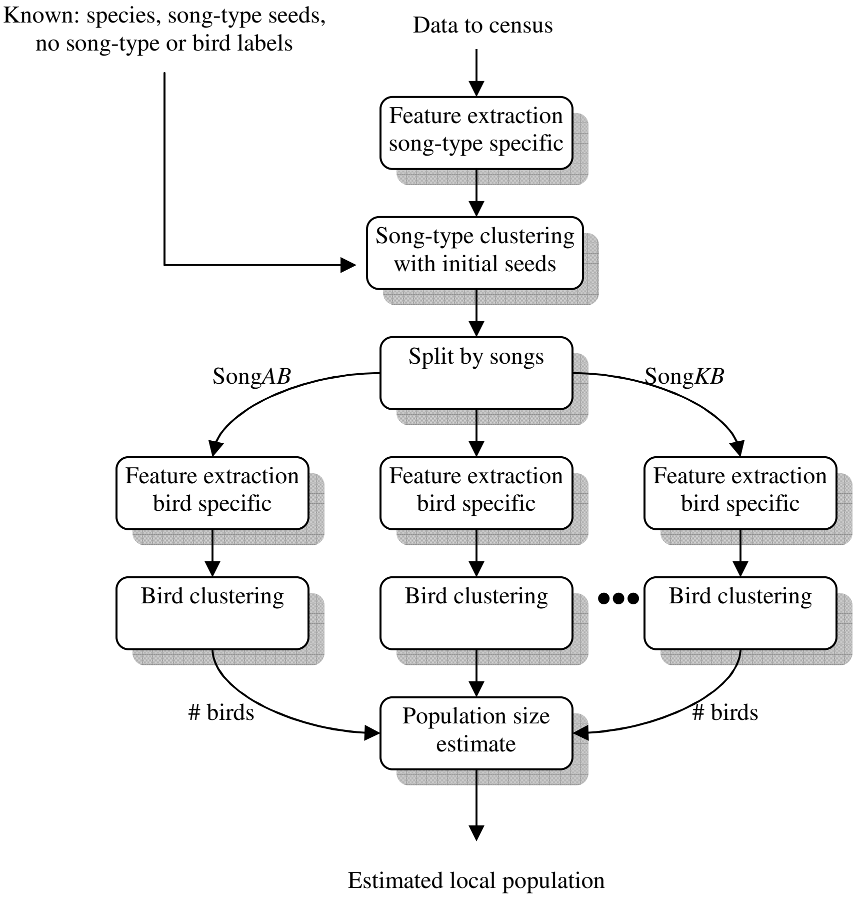
Known: species, song-type seeds,
no song-type or bird labels
Data to census
Feature extraction song-type specific
Song-type clustering with initial seeds
Split by songs
SongAB
SongKB
Feature extraction bird specific
Feature extraction bird specific
Feature extraction bird specific
Bird clustering
Bird clustering
Bird clustering
# birds
# birds
Population size estimate
Estimated local population

Figure 5.6. Bird estimate abundance using Scenario 3

The following section discusses in more detail the tasks involved in the above scenarios before the results of the experiments. These tasks include song-type classification and individual identification (supervised tasks), song-type clustering and individual bird clustering (unsupervised tasks), and population size estimation.

5.3.2. Supervised tasks: song-type classification and individual bird recognition

As discussed in Chapter 3, song-type classification and individual bird recognition consist of a feature extraction process, model training for each song-type or each individual bird, and recognition of unknown song-types or individual birds. The hidden Markov models (HMMs) are utilized to model each of the different song-types and each of the individual birds.

Figures 3.3 and 3.4 of Chapter 3 summarize the use of HMM for song-type and individual recognition tasks. First, a HMM is trained for each song-type or individual bird using a number of examples of that song-type or bird. This step estimates the model parameters (A, B, π) that optimize the likelihood of the training set for song-types or individual birds in the vocabulary.

To recognize an unknown vocalization, the likelihood of each model generating that song-type or bird is calculated, and the most likely model identifies the song-type or bird.

The features for song-type recognition consists of a feature vector that unites GFCC, energy, delta, acceleration and variance normalization as discussed in Chapter 3. For individual bird recognition, the study uses combined features of GFCC, energy, delta and acceleration. The repertoires are Hamming windowed with frame-size 3 ms and overlap 1.5 ms. The Greenwood frequency warping constants are calculated using an appropriate ortolan bunting hearing range of f_{min} 400 Hz to f_{max} 7400 Hz as determined by Edwards (1943). Twenty-six filter banks are spaced across that range. For song-type recognition the GFCCs along with energy, delta and acceleration are normalized using variance normalization. Thirty-nine element feature vectors are selected.

The HMM for song-type and individual bird recognition are 15-state left-to-right HMMs with each state containing a single diagonal-covariance Gaussian. The Baum-

Welch expectation maximization algorithm is utilized to estimate the parameters and the Viterbi algorithm is employed for song-type and individual recognition.

5.3.3. Unsupervised tasks: HMM-based song-type clustering and individual bird clustering

HMM-based song-type clustering

The song-type clustering approach employs an HMM-based *k*-model clustering as discussed in Chapter 2. The method builds initial song-type models using some known small data as seeds. Given *K* initial HMMs of song-type models λ_1^0, λ_2^0, . . ., λ_K^0, the clustering proceeds as follows.

Iteration t∈ 1, 2, …

- Data assignment:

  for each song-type data X_i, assign data to the model of maximum likelihood, namely $L(X_i \mid \lambda_k^{t-1})$ is maximal

- Model estimation:

  calculate new parameters of λ_1^t, λ_2^t, . . ., λ_K^t using song-type data assigned to the models and using previous parameters λ_1^{t-1}, λ_2^{t-1}, . . ., λ_K^{t-1}

- Termination:

  Terminate if no labels have changed.

The data assignment of the above process employs the Viterbi algorithm, and the model estimation utilizes a Baum-Welch re-estimation algorithm.

Features for song-type clustering consist of features that combine GFCCs, energy, delta and variance normalization. The song-type data are Hamming windowed with frame size 3 ms and overlap 1.5 ms. The ortolan bunting hearing range of 400 Hz to 7400 Hz is used to calculate the Greenwood frequency warping constants. Thirty-nine element feature vectors are extracted. The HMM for song-type models are 15-state left-to-right HMM with each state containing a single diagonal-covariance Gaussian.

HMM-based individual bird clustering

The goal of this process is to estimate the number of birds in each song-type data set. The clustering method employs two methods of estimating the number of birds in each song-type, dissimilarity analysis and deltaBIC analysis.

Dissimilarity analysis

In the dissimilarity analysis approach, the number of birds is estimated by using the cross-data dissimilarity analysis discussed in Chapter 2. The method varies the number of clusters *K* from 1 to a prespecified maximum. For each number of cluster $K \in \{1, 2, \ldots\}$ it performs the following steps:

1. Estimate the label distance of equation (2.40) by averaging 20 times split of the song-type data (cf. dissimilarity analysis in Chapter 2)
2. Normalize the label distance with its respected random labeling distance value using equation (2.42)

The estimated number of birds in a song-type data set is extracted from the value of K that results in the smallest value of the dissimilarity indices.

DeltaBIC analysis

As mentioned in Chapter 2, this approach employs a similarity measure between two probability density functions estimated by Gaussian mixture models.

DeltaBIC analysis starts with over-clustering of the data sets and iteratively merges clusters and retrains a new cluster until no possible pair of clusters is left. The new merged cluster is represented by a GMM that has a number of mixtures equal to the sum of mixtures of the individual clusters. The distance measure, referred to as deltaBIC is formulated using equation (2.43). The approach finds and merges a cluster pair that gives the largest deltaBIC value.

The procedure of the deltaBIC analysis is as follows.

1. Over cluster the data

   Initially cluster song-type data into classes greater than the expected number of birds in the data set. This experiment employs uniform segmentation of the data set, and train initial GMMs using data assigned to each initial clusters.

2. Cluster comparison and merging

   Search for all possible candidate pairs satisfying deltaBIC > 0, and select the best pair. Merge pair, and train GMM of new merged cluster using samples assigned to that cluster.

3. The cluster comparison and merging are repeated until no possible pair of cluster is left.

Feature extraction

Features for individual bird clustering consist of features that combine GFCCs, energy, delta and acceleration discussed in Chapter 3. The repertoires are Hamming windowed with frame-size 3 ms and overlap 1.5 ms. The Greenwood frequency warping constants are calculated similar to the experiments for song-type classification using an ortolan bunting hearing range of 400 Hz to 7400 Hz.

The initial individual bird models are 15 mixture GMMs. Thirty-nine element feature vectors are employed as the main features. The implementation is done in HTK, using a single-state HMM with a GMM observation model.

5.3.4. Population size estimate

Bird matching

In Scenario 1, it is necessary to learn individual bird models on one data set and then perform speaker identification and speaker clustering on a second data set. This study implements this bird matching using the speaker verification approach used in the field of human speaker recognition. Figure 5.7 shows the scheme for bird matching or bird verification.

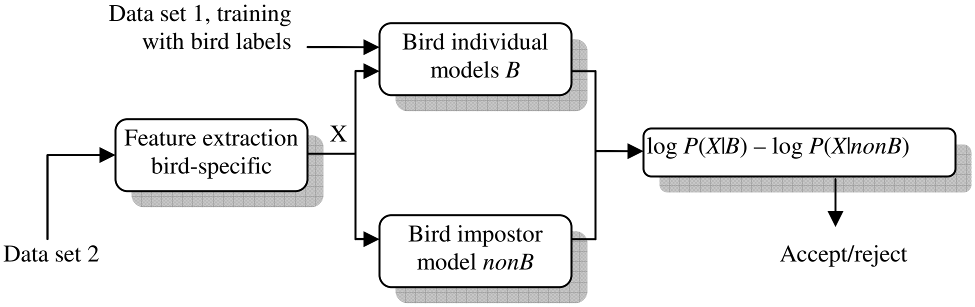
Data set 1, training with bird labels
Bird individual models *B*
Feature extraction bird-specific
X
log *P*(*X*|*B*) – log *P*(*X*|*nonB*)
Data set 2
Bird impostor model *nonB*
Accept/reject

Figure 5.7. Individual bird verification system

Bird models *B* are created from the training data set, using known individual labels. An HMM is trained for each individual bird using the number of examples assigned to them. Using the universal background modeling approach, the method also builds an impostor model trained over all samples in the data set.

As mentioned in Chapter 2, a speaker verification system implements a likelihood ratio test to discriminate whether the vocalization comes from the claimed bird or from an impostor (non claimed bird). Bird matching verifies the hypothesis that new bird *Bn* is the presumed bird *B* if $P(B|X) > P(nonB|X)$. By using equation (2.25) of Chapter 2 this expression becomes: $Bn = B$ if

$$\log P(X|B) - \log P(X|nonB) > \Delta \qquad (5.1)$$

The above equation states that the identity of bird *Bc* is accepted or validated when the difference is above the threshold. In this study the method selects the threshold as a positive value above zero.

Population size estimate – mark recapture model

Scenario 1 addresses the bird abundance estimation problem using the maximum likelihood estimation (MLE) framework of a mark-recapture model. A two sample mark-recapture involves one session of catching and marking, and another session of recapturing. In the context of this study, catching means recording bird vocalizations, and marking means labeling the vocalizations.

The process of labeling and recapture or re-labeling involves tasks of supervised recognition, unsupervised clustering and bird matching discussed in the previous sections. The previous steps, therefore, provide the number of birds (u_1) in one data set, the number of birds (u_2) in the second data set, and the number of birds present in both data sets (m_2).

Given the observed data u_1, m_2 and u_2, the likelihood of population estimate is computed using equation (2.72) discussed in Chapter 2 as follows:

$$L(N,p)=\prod_{s=1}^{2}\binom{U_s}{u_s}p^{u_s}(1-p)^{U_s-u_s}\binom{M_s}{m_s}p^{m_s}(1-p)^{M_s-m_s} \tag{5.2}$$

In the above equation one needs to plot and evaluate the likelihood as a function of N and p (the probability to capture, record vocalizations). The N and p where the likelihood function achieves its maximum value is the maximum likelihood estimation of N.

Population size estimate – using reference bird distribution

The method in Scenario 2 estimates the number of birds in the population based on a known repertoire distribution of known birds for each song-type. From the available known data sets, the approach lists all individual birds and computes the total number of birds in the data sets. For each song-type the number of birds that make that song is estimated using clustering methods, and, using the known percentage of birds that make that song, the total number of birds in the population is calculated as follows:

Total number of birds = (number of birds estimated/known percentage of bird)×100 %.

Population size estimate – upper and lower bound

When the known distribution of birds singing each song-type is available, Scenario 3 estimates the total number of birds in the population based on that reference distribution, as described in the pervious paragraph. Otherwise, the lower bound of the population estimate is computed from the maximum number of birds in the data sets, and the upper bound is the total number of birds estimated from the song-type data sets.

5.4. Experimental results and discussion

5.4.1. Estimating bird population using Scenario 1

Scenario 1 estimates bird abundance by utilizing the known information of the species, some training data with song-type labels and another data set with bird labels. As mentioned in the previous section, there are five tasks involved, namely, song-type classification, bird recognition, bird clustering, bird matching and population size estimation. Song-type classification trains repertoire models using a labeled song-type data set and classifies data into groups of song-types. In each song-type data set, there are exemplars with bird labels (in this case data 2001) and some exemplars without labels (data 2002). The study proceeds with bird recognition experiments to classify birds in the labeled data, and bird clustering experiments to estimate the number of birds in the unlabeled data set. Bird matching identifies birds present in both data 2001 and data 2002. Using the information of birds "marked" in data 2001, the number of birds "recaptured" in data 2002, and the number of birds present in both, the population size estimate computes bird abundance using a simple mark-recapture method.

Song-type recognition

Speaker independent song-type classification experiments are performed across the fourteen song-types selected from data 2001 and data 2002. Each song-type contains multiple song-variants. The experiment employs data 2001 as a training set for data 2002 and vice versa.

Table 5.1 shows the confusion matrix for song-type classification of data 2001 using GFCC, energy E, delta D, acceleration A and variance normalization features. Each row represents the labeled class of song-type while the column represents the classification given by the system to each song-type. The numbers of the diagonal are the number of song-types correctly classified.

	ab	*c*	*cb*	*cd*	*eb*	*ef*	*gb*	*guf*	*h*	*hb*	*hd*	*huf*	*jd*	*kb*
ab	1467	1	157	17	0	0	0	0	1	10	1	0	0	135
c	0	25	0	10	0	0	0	0	0	0	1	0	0	0
cb	21	2	550	31	0	0	144	0	2	7	3	1	0	2
cd	0	1	11	422	0	0	0	0	0	0	10	0	0	3
eb	0	0	11	0	202	2	0	0	0	0	0	0	0	0
ef	0	0	0	1	62	117	0	0	0	0	0	0	0	0
gb	0	0	29	0	0	0	477	0	0	3	0	1	0	1
guf	0	0	0	0	0	0	1	109	0	0	0	1	0	0
h	0	0	0	0	0	0	0	0	128	0	1	0	0	0
hb	0	0	1	0	0	0	0	0	0	18	1	5	0	0
hd	0	0	0	0	0	0	0	0	0	4	55	3	0	0
huf	0	0	0	0	0	0	0	0	0	0	0	87	0	0
jd	0	0	0	0	0	0	0	0	0	0	0	0	124	0
kb	0	0	0	0	0	0	0	0	0	0	0	0	0	137

Table 5.1. Song-type recognition result of data 2001 (Accuracy 84.86%)

84.86 % of the data are correctly classified. The matrix indicates that song *ab* and *cd* in data 2001 are difficult to recognize. The syllable characteristics of song *ab* are easily misclassified as *cb* or *kb* song-types. In the *cb* song-type, meanwhile, many samples are recognized as song-type *gb*.

Table 5.2 shows the confusion matrix for classification of data 2002. The method is able to recognize correctly 89.59 % of the data set.

	ab	*c*	*cb*	*cd*	*eb*	*ef*	*gb*	*guf*	*h*	*hb*	*hd*	*huf*	*jd*	*kb*
ab	1561	3	24	1	0	0	1	0	0	5	0	0	0	9
c	0	53	1	11	0	0	1	0	2	0	0	7	0	0
cb	1	2	706	11	0	0	85	1	0	2	0	2	1	1
cd	0	0	9	434	0	0	0	0	0	0	0	0	5	0
eb	1	1	4	0	384	11	1	0	0	0	0	0	0	0
ef	0	0	0	0	0	57	0	0	0	0	0	0	0	1
gb	3	1	19	1	0	0	320	5	0	3	0	0	0	37
guf	0	0	2	0	0	0	1	130	0	0	0	6	0	0
h	0	32	44	0	0	0	3	0	138	27	0	0	0	17
hb	0	0	0	0	0	0	0	0	0	32	0	0	0	1
hd	0	0	0	0	0	0	0	0	0	5	8	3	0	0
huf	0	2	7	41	0	0	0	1	0	2	22	285	0	1
jd	0	0	1	2	0	0	0	2	0	0	0	0	47	0
kb	0	1	0	0	0	0	0	0	0	0	0	0	0	87

Table 5.2. Song-type recognition result of data 2002 (Accuracy 89.59%)

The classification result indicates a similar trend to data 2001. Some samples of song *ab* are misclassified as *cb*, and song *cb* has similar acoustic characteristics with song-type *gb*. In addition, the matrix shows the similar acoustic characteristic of song-type *h* with song-types *c*, *cb* and *hb*. It is worth noting that perfect song-type separation is not necessarily needed to do the larger task of counting individuals.

Bird clustering

Here we examine the two most common song-types of the ortolan bunting, namely, song-type *ab* and *cb* to estimate bird abundance from their data sets. The study splits data *ab* and *cb* into data of years 2001 and 2002 and uses data 2001 for building bird models.

The number of birds in the 2002 song-type data set is estimated using the deltaBIC analysis discussed in the previous chapter. In song-type *ab* the number of initial clusters is 49 and deltaBIC value reaches its peak at iteration 14, where each iteration involves reducing the number of clusters by one. The estimated number of birds in that song-type is, therefore, (49-14) or 35 birds.

Figure 5.8 shows the accumulative deltaBIC value as a function of the number of birds in data *ab* and *cb* year 2002. The estimated numbers of birds in song *ab* and *cb* are 35 and 22 respectively.

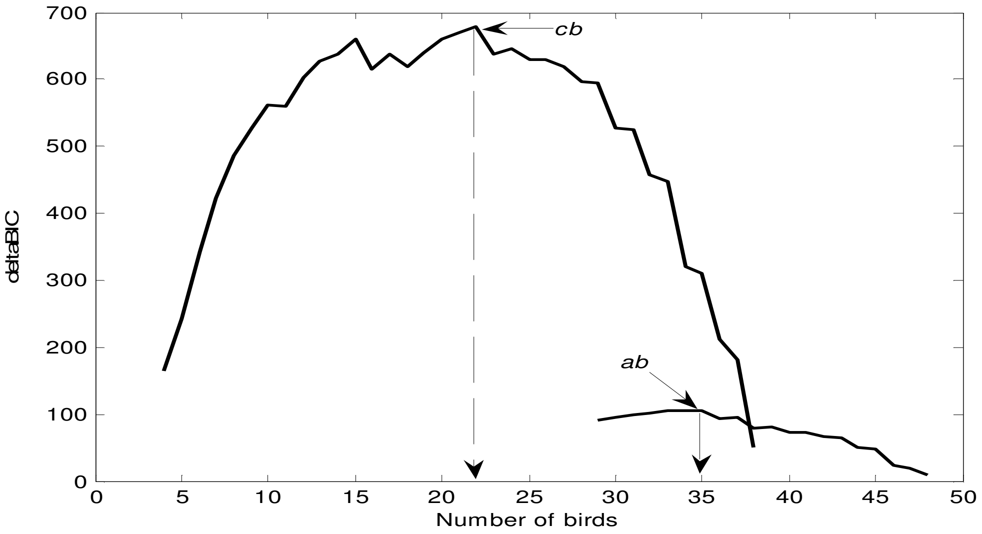

Figure 5.8. DeltaBIC analysis of song-type *ab* and *cb* of birds 2002

Bird matching

The next step is to identify birds present both in data 2001 and data 2002. The bird matching method discussed in section 5.3.3 is used to implement a likelihood ratio test to discriminate whether the vocalization comes from a known bird or from an unknown one. Using equation (5.1) the identity of a bird is accepted if the difference between log $P(X|B)$ and log $P(X|nonB)$ is above a threshold of some positive value.

Tables 5.3 and 5.4 present the acceptance value of birds in *cb* and *ab* data sets. For song *cb* 8 birds in data 2002 (birds 1, 2, 10, 13, 14, 15, 16, and 22) are verified as bird 3 in data 2001. Bird 20 of data 2002 is identified as bird 5 in data 2001, bird 5 is verified as bird 6, bird 7 as bird 7, birds 3, 4, 9 and 19 are verified as bird 11 of data 2001, and bird 18 of data 2002 is identified as bird 12 in data 2001. Thus there are six birds present in both data 2001 and 2002 of song-type *cb*.

		Birds in song-type *cb* 2001												
		1	2	3	4	5	6	7	8	9	10	11	12	13
Birds in song *cb* 2002	1	-6.1911	-4.8384	**0.2598**	-4.8071	-4.8664	-1.7287	-1.4306	-2.2258	-7.4660	-6.6986	-2.0243	-3.5749	-4.4040
	2	-3.1360	-2.2224	**1.0759**	-3.5349	-3.7813	-2.4249	-1.1016	-2.9164	-5.1645	-3.5034	0.0294	-4.0007	-4.9570
	3	-1.2931	-4.0089	-0.9428	-7.4533	0.5253	-5.3972	-5.5260	-2.3992	-7.3673	-6.3985	**0.7298**	-2.5238	-7.1605
	4	-1.7191	-4.5681	0.1981	-7.0703	0.3497	-5.6525	-5.4115	-3.0178	-6.9223	-5.9974	**1.1603**	-3.5598	-7.2065
	5	-3.7740	-3.0045	-0.9646	-3.2053	-2.8519	**0.1312**	-1.1781	-1.4876	-4.4815	-4.4405	-2.4751	-3.5888	-1.6194
	6	-4.1879	-3.8325	-2.2324	-2.5864	-1.4738	-2.3203	-1.7527	-1.4110	-4.8890	-5.3197	-1.9983	-2.9655	-2.9405
	7	-4.0524	-3.7983	-0.1179	-1.9972	-3.6972	-3.7014	**0.0847**	-1.5080	-5.6273	-6.7853	-1.0934	-3.2686	-4.7191
	8	-1.6285	-1.0683	-2.5166	-3.7007	-1.7817	-2.0835	-2.3154	-1.9211	-4.3459	-4.5966	-0.9438	-3.1081	-2.8231
	9	0.4203	-3.9631	-3.2250	-6.2160	-0.4092	-4.9076	-3.0624	-3.7383	-5.5757	-4.7141	**0.8304**	-4.6439	-6.6938
	10	-6.1358	-4.7771	**0.5048**	-2.9390	-5.8793	-0.0643	-0.9827	-2.1916	-7.4385	-6.1976	-2.9593	-5.2991	-4.8353
	11	-6.0416	-5.6546	-4.8061	-1.5755	-1.3735	-3.8146	-1.4443	-1.3929	-6.0052	-7.2179	-4.3271	-1.7333	-4.4404
	12	-3.1280	-3.3624	-0.9467	-3.5546	-3.8155	-2.6841	-1.1312	-3.4759	-1.5937	-4.5810	-1.0143	-4.8189	-4.4083
	13	-5.5911	-4.8649	**0.2842**	-6.4990	-4.7271	-4.2375	-2.6389	-1.0592	-7.7188	-6.7524	-0.0592	-3.8018	-6.1107
	14	-6.4177	-4.8428	**0.1138**	-5.4141	-4.8679	-3.0886	-2.9369	-2.7645	-6.2715	-7.2595	-0.6921	-4.7532	-5.7999
	15	-5.5897	-6.7359	**0.1789**	-3.8895	-4.4871	-1.3617	-2.4674	-4.0671	-6.9669	-1.7588	-3.1082	-6.2582	-5.5221
	16	-3.1855	-5.3260	**0.6257**	-4.6915	-3.3572	-2.0442	-1.1992	-3.6259	-5.3410	-1.8490	-2.0575	-5.4783	-5.9606
	17	-5.0415	-6.5233	-2.2900	-1.9306	-4.1976	-4.9813	-1.2144	-3.6470	-4.0607	-7.0770	-2.6779	-6.1275	-5.9435
	18	-5.8263	-6.6272	-4.4404	-2.7344	-0.0812	-2.7804	-2.1755	-0.2400	-7.9769	-7.7126	-4.6130	**0.1282**	-4.1389
	19	-1.8148	-1.8384	-1.4369	-3.8289	-1.8039	-3.3647	-1.9384	-3.0285	-4.5124	-3.7390	**1.1541**	-3.6040	-4.5305
	20	-0.9310	-3.3875	-4.1278	-3.6522	**0.4725**	-4.5987	-1.8132	-1.5103	-4.6590	-5.4152	-1.8644	-3.8411	-4.7288
	21	-3.6516	-4.8461	-0.7648	-2.9481	-3.1399	-3.7093	-0.2985	-1.6854	-4.7744	-6.5691	-1.6969	-3.7253	-4.4330
	22	-4.6355	-4.2748	**0.4865**	-2.9744	-3.8810	-3.0470	-0.1036	-1.6721	-6.2508	-7.0564	-1.7838	-3.6584	-5.1683

Table 5.3. Birds present both in song *cb* 2001 and 2002

Using similar computation of song *cb*, for song *ab* overall there are ten birds present both in data *ab* 2001 and 2002.

		Birds in song-type *ab* 2001															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Birds in song-type *ab* 2002	1	**3.3287**	-4.5346	-2.3496	-0.5471	-2.0251	-0.4868	-1.2572	-6.0190	-3.4233	1.2589	0.6085	-4.2168	-2.2072	-1.9133	-3.5531	0.5817
	2	-4.9521	-0.4120	**3.6250**	-0.6869	-7.7675	-6.1197	-6.6364	-7.9153	-7.2320	-5.7446	-4.2239	-7.8955	-6.5088	-8.9694	0.5437	-3.9514
	3	0.0094	-4.9794	-3.0445	-2.0347	-2.8274	**1.2987**	-0.6014	-6.1907	-4.2756	-0.6311	-2.0286	-2.2806	-1.6244	-5.5008	-3.5246	-1.1717
	4	-3.5455	-5.2913	-5.0752	-4.5992	-4.2910	0.0136	-0.7237	-5.4962	-3.5415	-2.7120	-4.5323	-3.3531	-4.8986	-5.3700	-4.2718	-4.5367
	5	-1.9018	-6.1351	-5.4354	-4.4011	-3.4983	**0.6552**	-0.4045	-6.4261	-3.8286	-1.7155	-2.6767	-1.7614	-4.5711	-6.3331	-3.8717	-3.7448
	6	-1.2089	-6.6477	-4.8173	-4.2969	-4.2304	**2.8198**	1.0148	-7.9238	-4.5753	-2.9112	-1.3878	-1.6900	-1.7795	-7.4345	-5.2890	-3.1953
	7	1.0211	-3.6294	-0.7787	1.3923	-1.9047	-0.1691	-1.9307	-6.1673	-3.5546	0.2653	-1.0236	-3.7799	-1.0390	-5.3754	-2.8322	**2.1469**
	8	**1.7924**	-4.8827	-3.0877	-1.1272	-3.3325	-0.0846	-1.6708	-5.9100	-4.3454	-0.6681	0.8958	-4.4677	0.2829	-1.8564	-5.0636	-1.3392
	9	-3.1862	-3.2293	-6.0335	-3.8716	-4.7666	-1.6365	-0.5956	-3.5215	-4.8782	-2.0621	-0.9785	-2.7782	-5.5273	0.5998	-3.9299	-4.4922
	10	-1.0110	-2.0601	-4.1320	-1.8446	-3.5936	0.2727	-0.6778	-5.1737	-4.9468	-1.1128	0.5216	-3.7430	-1.1770	**0.7507**	-4.4769	-2.5578
	11	0.0344	-3.1517	-2.4113	0.2726	-0.2110	-0.1096	-1.2683	-5.5885	-1.7013	0.1989	-1.1160	-2.2014	0.5235	-0.9032	-3.2479	**0.4988**
	12	-0.0240	-3.6585	-2.2180	-0.2363	-0.5949	1.6129	-1.1999	-5.7723	-2.1734	0.4137	0.1851	-2.3263	**3.0963**	-5.4559	-3.4986	-0.4444
	13	**0.5747**	-0.6984	-1.5701	-0.8911	-0.7810	-0.1473	-0.9784	-3.5688	-1.3635	-0.2817	-0.3451	-2.7524	-1.6754	0.1466	-2.1240	-0.5024
	14	0.2603	-4.3335	-4.2428	-2.0385	-0.3846	-1.2120	-1.3088	-2.3805	-2.5932	0.2046	-0.1786	-0.4489	-4.3443	-0.7042	-1.4955	**0.4613**
	15	-3.7994	-3.8400	-5.6874	-3.9816	-6.3213	-6.6651	-3.7751	-0.0810	-6.4072	-4.8282	-3.4646	-5.0304	-8.2907	-0.1729	-1.3669	-4.5992
	16	-3.1322	-4.3712	-4.3078	-3.1597	-5.1229	-5.7386	-2.4056	-3.1253	-4.3944	-5.2688	-3.2176	-4.2462	-6.1723	-0.2337	-2.3725	-3.2701
	17	-1.5812	0.6358	**1.4078**	0.6350	-3.1697	-3.0178	-3.8149	-5.4931	-4.5359	-2.7827	-1.7833	-4.9819	-3.0653	-4.5525	-1.3843	-1.7438
	18	-1.8981	-0.5290	-4.2339	-1.5999	-2.7318	-3.0345	-1.7407	-1.4410	-3.7258	-1.9089	0.3039	-4.4479	-5.4692	**2.2361**	-1.5102	-2.6380
	19	-4.3723	**2.5562**	-0.5004	-1.4817	-5.5748	-5.4254	-5.3073	-1.6533	-5.9814	-4.6300	-1.9894	-6.4667	-6.4564	-3.4310	0.8972	-4.5663
	20	-3.1586	-4.4308	-4.7372	-3.5264	-3.7864	-3.9308	-3.2444	**1.6347**	-4.9425	-3.1223	-2.0139	-3.1887	-6.8852	-3.8172	0.6785	-2.9252
	21	-0.6836	-5.3612	-5.1553	-2.3066	-0.8731	-1.9086	-1.2863	-5.5397	-4.6642	-1.8416	-2.2393	-1.3082	-4.0091	-4.6863	-3.4346	-1.6701
	22	-1.9576	-0.3462	**0.4983**	-0.8328	-5.7671	-4.5545	-4.2411	-4.2355	-5.0266	-3.0427	-1.9594	-6.6876	-6.1513	-5.9615	0.0556	-3.2228
	23	-1.0173	-3.1648	-0.9525	-0.0742	-1.7215	-2.1823	-2.8727	-4.0014	-3.4120	-3.3416	-2.2934	-3.4086	-2.6438	-3.5942	-2.4093	-1.5531
	24	-5.5998	-0.6814	**1.5259**	-1.9446	-6.5365	-5.8078	-6.0025	-4.8316	-6.8483	-5.8448	-4.5502	-6.9106	-6.3754	-7.4118	1.2762	-4.3942
	25	-0.9918	-3.8087	-2.4317	-0.2722	-3.2170	**2.2611**	1.7144	-8.0419	-0.2204	1.3616	0.2856	-3.1186	0.0957	-6.8707	-3.3246	-1.1432
	26	-0.4822	-4.4073	-7.1259	-4.0923	-3.3212	-1.9584	-1.6182	-0.5671	-4.7150	-0.8172	-0.4958	-2.3677	-6.6990	-0.3718	-1.9094	-2.2977
	27	-6.1037	-9.4225	-9.9198	-7.4261	-9.6247	-8.8753	-3.7258	-3.5603	-6.8016	-8.5687	-7.3220	-4.5637	-9.9283	0.2971	-5.4058	-8.7279
	28	-2.2932	-2.3024	**1.7206**	0.0829	-3.7148	-4.4141	-4.8811	-5.0697	-4.8533	-2.2376	-2.6415	-7.3811	-6.7101	-10.0142	0.9841	0.6487
	29	0.2639	-3.3431	-0.7971	0.1313	-1.8326	-2.4744	-3.0561	-3.4950	-4.1534	-0.7443	-0.4675	-4.6209	-4.5840	-4.9819	-0.3142	**0.6251**
	30	-1.0945	-3.7550	-6.1084	-2.9030	-3.2488	-4.0070	-1.1507	0.3563	-4.3200	-1.0520	1.4887	-5.2123	-7.0671	**1.7815**	-1.2872	-2.8376
	31	-3.9101	0.3551	-2.9815	-2.7914	-4.9732	-5.3486	-4.3406	**2.6597**	-4.9551	-3.3794	0.1134	-6.2999	-7.3985	-1.3566	2.1058	-4.0645
	32	-4.0881	-5.7673	-5.0985	-4.7395	-3.7045	0.8830	**1.1993**	-6.0702	-3.1682	-3.1623	-3.7159	-0.4160	-3.6112	-6.0371	-4.2771	-4.6447
	33	-1.1413	-1.4539	-1.3813	-1.2428	-1.8464	-1.3091	-1.2654	-2.8176	-1.7521	-1.3891	-1.4810	-1.9419	-2.8377	-2.8126	-0.9178	-1.5366
	34	-1.2853	0.4026	**0.9408**	0.6735	-2.3300	-2.9114	-3.3500	-3.1851	-3.6146	-1.8583	-0.5685	-4.2055	-3.3303	-3.1504	-0.4176	-0.4902
	35	-3.7747	-4.0904	**0.0030**	-2.4619	-5.8853	-4.5694	-4.0613	-6.2907	-5.6708	-5.1204	-4.2093	-5.4901	-6.3505	-6.3590	-0.3940	-2.8681

		Birds in song-type *ab* 2001												
		17	18	19	20	21	22	23	24	25	26	27	28	29
Birds in song-type *ab* 2002	1	-2.7923	-1.6403	-3.0591	-6.7815	-4.7261	-7.8531	-2.0805	-3.5705	-3.5040	-2.6241	-5.4734	-7.0121	-5.2552
	2	-2.0376	-2.2067	-5.2913	-7.4954	-2.8914	-9.7565	-9.0457	-8.1580	-3.7504	-7.6467	-5.0827	-9.4171	-9.1575
	3	-5.2247	-3.6440	-4.0929	-5.9322	-4.1864	-5.2039	-5.2712	-5.4196	-3.2280	-2.6958	-5.8841	-4.0888	-5.4006
	4	-5.1233	-3.6771	-4.8497	-5.2574	-3.2070	-3.3678	-5.5137	-5.4987	-0.8412	-0.597	-5.3498	-0.8700	-5.1403
	5	-6.1144	-3.1654	-4.9712	-6.3216	-3.8467	-4.0983	-6.1435	-6.1479	-1.5502	-0.0347	-6.3072	-3.8663	-6.1023
	6	-7.1146	-5.3112	-5.6374	-7.8321	-5.8273	-7.1518	-7.3920	-7.1876	-4.5096	-2.2655	-7.7117	-6.9752	-7.1363
	7	-3.6899	-2.7908	-2.3053	-6.0012	-3.8403	-6.3174	-4.8673	-4.5437	-3.1478	-3.4209	-5.8994	-5.7616	-5.3782
	8	-3.2411	-2.5536	-3.9053	-5.7310	-5.1282	-6.4710	-4.6349	-5.1753	-4.5549	-4.2908	-5.2379	-5.9918	-5.5970
	9	-1.8455	-2.4912	-1.1234	-5.8704	-3.4359	-6.9937	-2.5286	-4.9308	-5.7443	-5.3948	-0.6914	-6.5628	**1.1755**
	10	-1.1357	-2.7229	-0.9435	-6.2785	-4.5890	-6.8538	-3.0701	-5.1090	-5.7102	-5.8126	-2.2724	-6.6518	-0.5380
	11	-1.6483	-0.8298	-2.0244	-4.9952	-3.2050	-6.4724	-0.4568	-3.3489	-4.2988	-3.2528	-3.3355	-5.8304	-2.3114
	12	-4.1665	-1.8507	-3.1598	-5.7330	-3.5536	-6.0135	-4.4469	-4.3100	-3.0425	-2.6221	-5.2460	-4.9792	-5.4295
	13	-0.7789	-1.3345	-0.9350	-3.0500	-2.5263	-3.9892	-1.2855	-2.3181	-3.4960	-3.1223	-1.9809	-3.6692	-1.8721
	14	-1.9173	-0.7236	-2.7449	-3.2978	-0.8224	-5.8155	0.1969	-1.8944	-3.2122	-1.3496	-1.9194	-4.8046	0.2969
	15	-2.6471	-3.7014	-4.0344	-1.8701	0.1249	-6.2848	-6.5507	-4.3598	-5.6859	-6.1894	-0.7238	-7.8444	**0.9041**
	16	-3.1631	-3.7504	-4.4927	-2.3630	-0.8995	-5.1289	-5.9696	-5.5419	-6.0347	-6.0715	-2.4831	-7.3097	-1.1101
	17	0.0172	-0.0858	-1.7049	-4.4772	-1.8815	-6.5960	-4.6791	-4.1328	-2.6057	-4.3442	-2.4746	-6.0908	-4.8242
	18	1.2393	0.1704	-0.0908	-0.8270	-1.6084	-5.9776	-2.1948	-0.5287	-4.3880	-4.5979	0.8254	-5.5180	1.3457
	19	2.2713	0.9027	-1.0090	-2.4853	-0.7433	-7.7882	-5.9723	-3.6718	-4.6530	-6.4565	1.3518	-7.4744	-3.5349
	20	-2.4818	-0.9923	-4.0028	-2.6969	1.2838	-5.9721	-4.3791	-1.9087	-1.5665	-1.3814	-0.3351	-4.1094	-0.5097
	21	-3.8795	-3.0501	-3.8656	-5.1885	-1.9652	-1.2854	-4.2751	-5.5415	-3.9522	-2.9016	-5.0994	-3.8953	-3.4664
	22	-0.9801	-1.4752	-2.9202	-6.1894	-3.0200	-8.6173	-6.3088	-3.2036	-3.1979	-5.6024	-3.3872	-8.2076	-6.1127
	23	-2.1282	-0.7388	-2.8500	-3.7747	-1.9827	-4.2796	-3.9759	-3.8650	-0.6947	-2.3845	-3.7211	-3.1303	-3.8731
	24	-1.3732	-0.9504	-4.4278	-5.2030	-0.6779	-8.5000	-7.6258	-6.3328	-2.9226	-6.4605	-2.3652	-7.9844	-6.3930
	25	-5.8946	-4.2496	-3.7794	-8.0400	-5.1079	-8.3196	-5.8430	-6.1686	-5.1369	-4.6430	-7.2151	-7.9335	-6.8711
	26	-1.8006	-2.0307	-2.5506	-5.5144	-1.7825	-7.7498	-1.9315	-2.1270	-5.2776	-2.7167	-0.4983	-6.5843	**0.8166**
	27	-9.3605	-9.3018	-9.3371	-1.5090	-0.5475	-2.6629	-10.1466	-10.2727	-9.1358	-8.6949	-5.7373	-9.2473	-0.9739
	28	-3.4477	-3.7094	-3.6824	-8.8101	-2.2805	-10.2468	-7.7272	-5.5513	-2.1884	-4.4090	-5.9137	-9.2576	-8.9195
	29	-1.4567	-0.3237	-2.6499	-5.9330	-1.8140	-7.7726	-3.5277	-2.4592	-0.3214	-1.4580	-4.0040	-5.7919	-4.8829
	30	0.4683	1.0680	-3.0696	0.3391	-2.0658	-7.0695	-1.2576	1.0670	-4.9866	-3.1603	1.0873	-6.3450	1.0096
	31	2.1293	1.4649	-1.6046	-1.2768	-0.3556	-8.1088	-5.1827	0.4916	-3.8388	-5.0710	2.2317	-7.5654	-1.3952
	32	-5.8955	-5.2001	-5.3518	-6.0562	-4.2765	-2.6640	-6.0660	-5.9176	-3.2138	-1.5243	-6.0198	-1.8921	-5.6420
	33	-1.9976	-1.5408	-1.8371	-2.9278	-1.5009	-3.0641	-2.3635	-1.6233	-1.5231	-1.7482	-2.4367	-2.4563	-2.3022
	34	0.4596	0.6811	-1.3543	-3.5668	-1.1483	-6.0677	-3.2567	-1.9102	-1.9724	-3.4532	-0.9946	-5.2808	-3.4540
	35	-4.0666	-3.5656	-5.5329	-6.2798	-3.9933	-7.0349	-6.7912	-5.8372	-4.0981	-5.8541	-5.8285	-6.7684	-6.2757

Table 5.4. Birds present both in song *ab* 2001 and 2002

Population size estimate

Given the observed data of 29 birds in song *ab* 2001, 35 birds in song *ab* 2002 and 10 birds "captured" both in *ab* 2001 and 2002; for the song *ab* data, the likelihood of the population estimate is computed using equation (5.2).

The likelihood of bird estimation as a function of N (number of birds) and p (capture probability) is shown in figures 5.9 and 5.10. The likelihood reaches its maximum value when p = 0.32 and N = 100. This indicates that the estimate of bird abundance $\hat{N}$ in song-type *ab* is 100 birds.

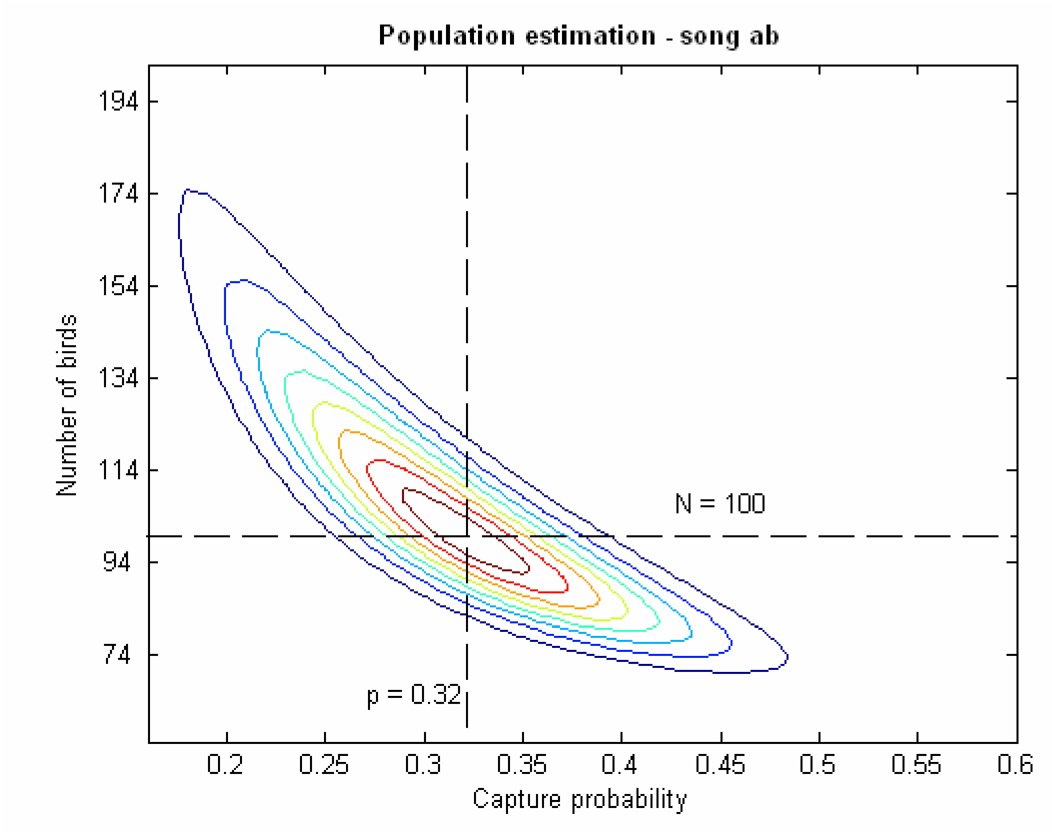

Figure 5.9. The likelihood function of birds in song *ab* – contour plot. The function reaches its maximum for p = 0.32 and N = 100.

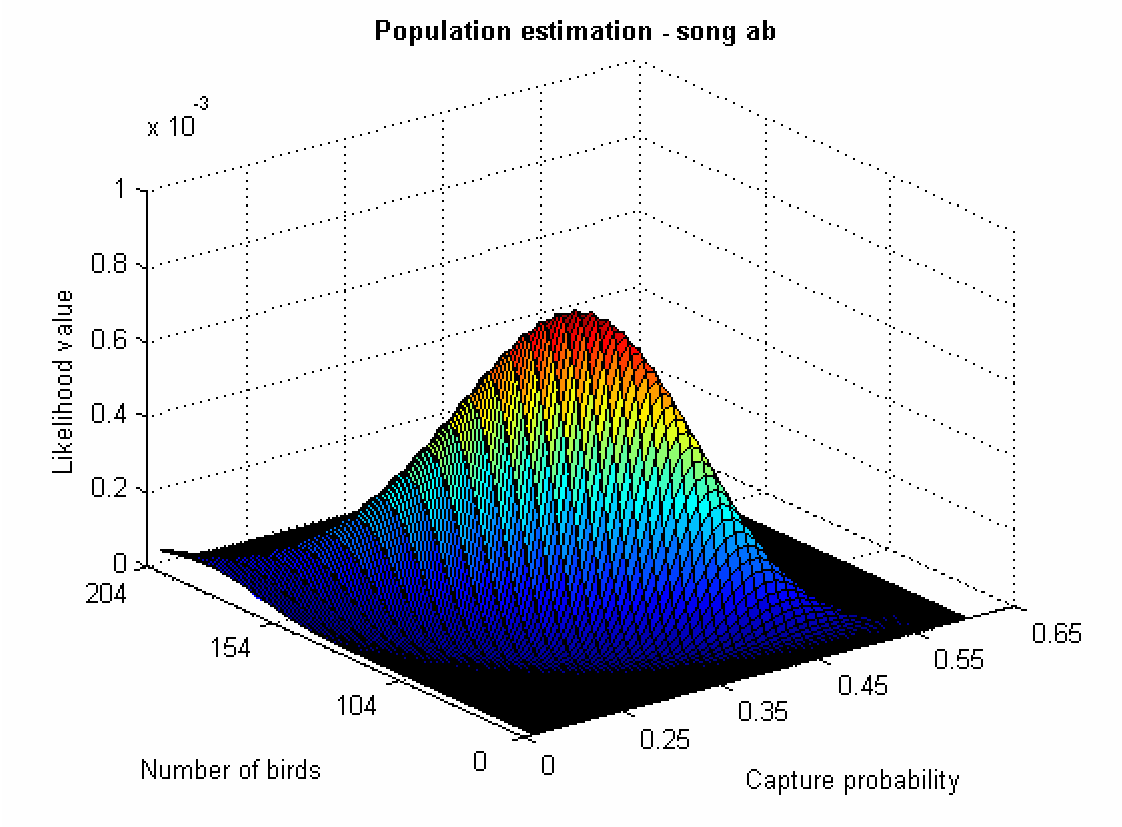

Figure 5.10. The likelihood function of birds in song *ab* – perspective plot

The profile of the likelihood confidence interval is constructed using a variance of Chapman’s modified estimator (Seber, 1970) as follows:

$$\operatorname{var}(\hat{N}) = \frac{(n_1+1)(n_2+1)(n_1-m_2)(n_2-m_2)}{(m_2+1)^2(m_2+2)} \qquad (5.3)$$

where n_1 = number of birds captured in session 1, n_2 = number of birds captured in session 2, and m_2 = number of birds present in session 1 and 2.

The 95 % confidence interval for bird estimation abundance is = $\hat{N}$ ± $1.96\sqrt{\operatorname{var}(\hat{N})}$. For song *ab* data set, given that $n_1 = 29$, $n_2 = 35$, $m_2 = 10$ and $\hat{N} = 100$;

with a 95% confidence interval, the estimate of bird abundance is equal to (100 ± 36) birds.

Figures 5.11 and 5.12 show the likelihood of bird estimation as a function of *N* (number of birds) and *p* (capture probability) in song *cb*.

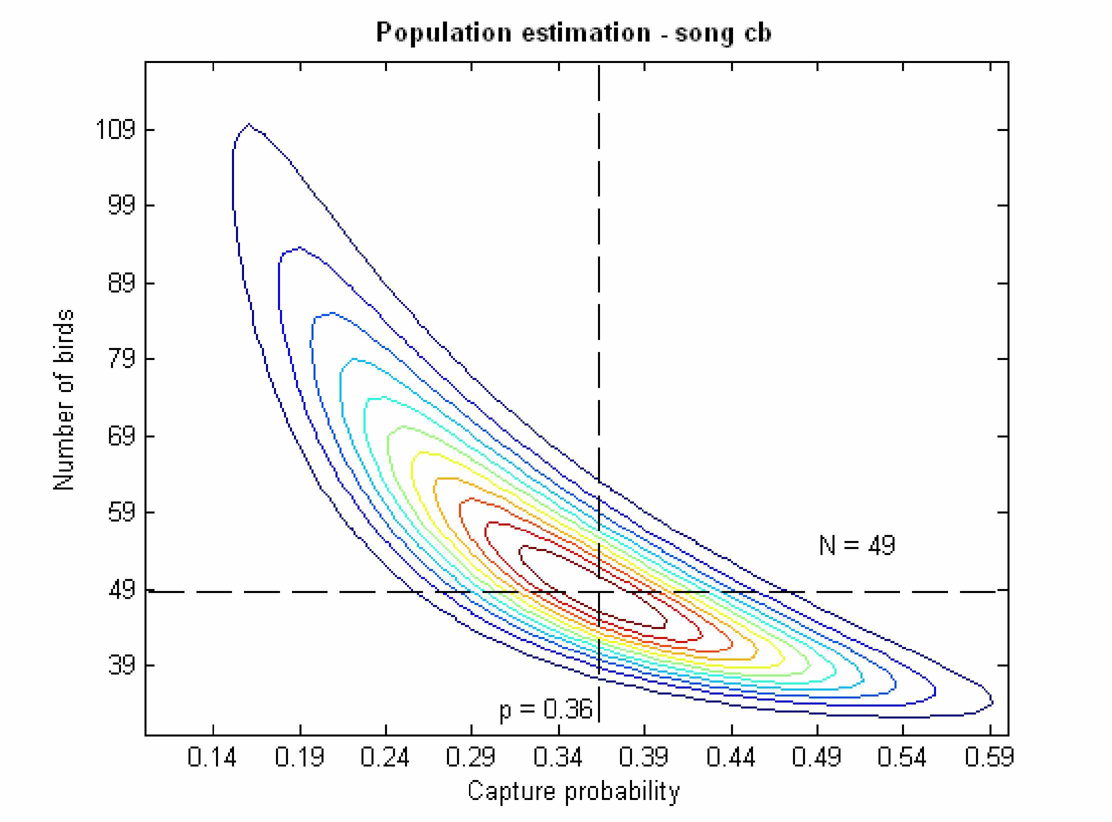

Figure 5.11. The likelihood function of birds in song *cb* – contour plot

The function reaches its maximum for *p* = 0.36 and *N* = 49

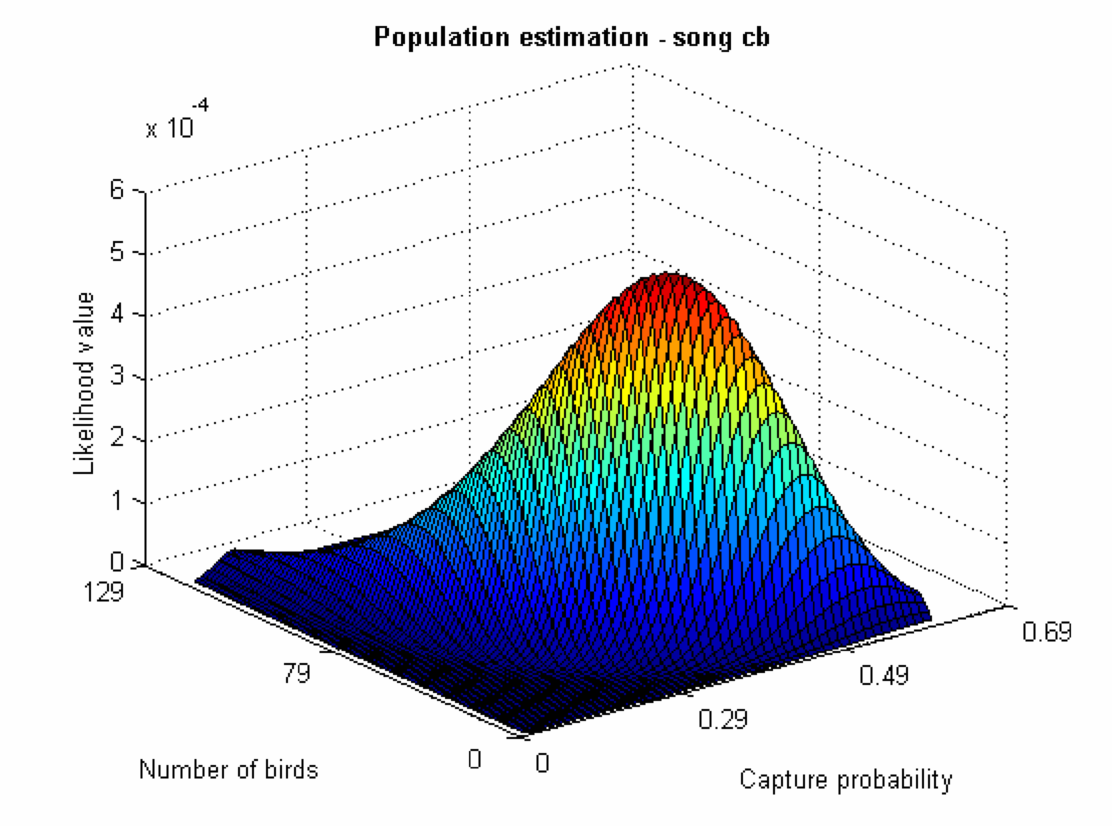

Figure 5.12. The likelihood function of birds in song *cb* – perspective plot

For song *cb*, there are 13 birds present in data 2001 (= n_1), 22 birds "captured" in data 2002 (= n_2) and 6 birds overlapped in data 2001 and 2002 (= m_2). The estimate of bird abundance for this data set with 95% confidence interval is equal to (49 ± 18) birds.

A 95% confidence interval for $N = 100$ (song *ab*) and $N = 49$ (song *cb*) is a set of estimates of N which would include the true N 95% of the time if the survey is repeated an infinite number of times. This study has only one survey and one confidence interval, and does not know whether it is one that actually includes N or not. But the study knows that it includes N with 95% probability.

In song *cb*, the likelihood function reaches its maximum for $p = 0.36$. The confidence intervals are narrower than those in song *ab* because there is less uncertainty about the number of population from a survey with $p = 0.36$ (in which on average 36% of the population is detected or recorded on one survey) from a survey with $p = 0.32$ in song *ab* (in which 32% of the population is recorded on one survey on average).

The mark-recapture method generally gives a reliable estimation of the size of a closed population when during the course of the study there are no gains or losses. The study is, therefore, conducted over a short period of time when births, deaths, and movements are few. It is worth noting that the above computation is an example of incorrect methodology implementation that leads to erroneous population estimation results.

5.4.2. Estimating bird population using Scenario 2

Utilizing data sets of known species and available training data with repertoire labels but unknown labels of individual animals, Scenario 2 explains the problem of population estimation using joint repertoire classification and individual animal clustering. There are three tasks involved, song-type classification, individual bird clustering and population size estimation.

Song-type recognition

The song-type recognition experiment in Scenario 2 is precisely the same as the song-type classification of Scenario 1. The reader may refer to the results of the previous tables (Tables 5.1 and 5.2).

Bird clustering

Because in this second scenario we do not assume or use individual labels on the first data set, we need instead to do unsupervised individual clustering on each song-type. The method utilizes two approaches of estimation, namely, dissimilarity analysis – as discussed in Chapter 2 and implemented for beluga clustering in Chapter 4 – and deltaBIC analysis, discussed in Chapter 2. The dissimilarity analysis is employed for some song-type data sets of data 2001. Due to the inability of the dissimilarity analysis to estimate correctly the number of clusters or birds in big data sets, some data sets of data 2001 (specifically data sets for song *ab*, *cb*, *cd* and *gb*) and most of data 2002, meanwhile, use deltaBIC analysis for bird estimate.

DeltaBIC analysis starts with the over-clustered initial grouping using linear uniform segmentation. The initial number of clusters is adjusted to the size of the data set. The experiments try to maintain a minimum initial cluster member of 4 samples in order to have initial stable models.

Figure 5.13 shows the accumulative deltaBIC values of data 2002 as a function of the number of birds in the data set. The number of birds in a song-type data set is estimated from the initial cluster and its peak iteration (the highest value of deltaBIC in the iterations). In song-type *cb*, for example, the number of initial clusters is 39 and deltaBIC value reaches its peak at iteration 17. The estimated number of birds in song-type *cb*, therefore, is equal to (39 – 17) or 22 birds.

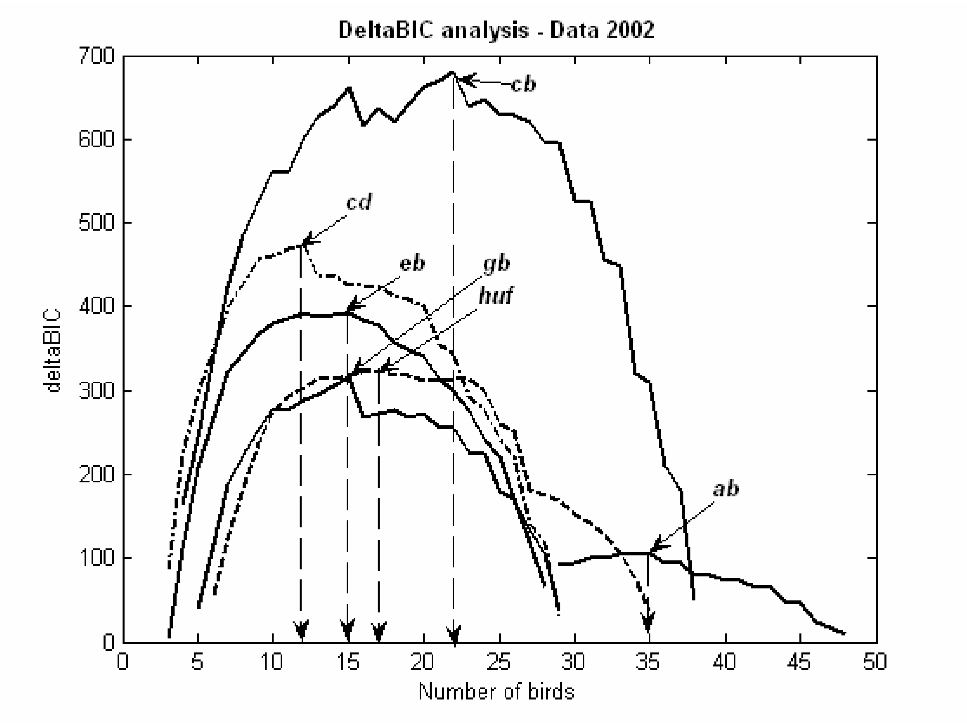

Figure 5.13. DeltaBIC analysis of data sets from data 2002

Tables 5.5 and 5.6 show the analysis results of data 2001 and 2002.

No	Song-type	Estimated number of birds	Known number of birds
1	*ab*	34	29
2	*cb*	18	13
3	*cd*	7	11
4	*eb*	6	6
5	*ef*	3	3
6	*gb*	12	11
7	*guf*	2	3
8	*h*	2	5
9	*hd*	2	2
10	*huf*	4	3
11	*kb*	2	3

Table 5.5. The estimated number of birds in some song-types of data 2001

No	Song-type	Estimated number of birds	Known number of birds
1	*ab*	35	34
2	*cb*	22	20
3	*cd*	12	9
4	*eb*	15	12
5	*ef*	3	2
6	*gb*	15	13
7	*guf*	6	4
8	*hb*	3	2
9	*huf*	17	20
10	*jd*	3	2
11	*kb*	9	7

Table 5.6. The estimated number of birds in some song-types of data 2002

The estimated number of birds in song-type *eb*, *ef* and *hd* of year 2001 are equal to the known number of birds in those song-types. For some song-type data sets the analysis gives results slightly above the known number of birds in the data sets. For other data sets, the results gives population estimates that are below the known number of birds in the respective data.

As mentioned in the previous section, deltaBIC analysis starts with the over-clustered initial grouping using linear uniform segmentation. The linear segmentation, however, does not guarantee that the initial clusters consist of vocalizations from the same bird for each cluster. Some clusters might have samples from different birds. This makes it difficult for the system to build initial GMM models, and might prevent the system to cluster accurately, as seen from the above results.

Population estimate

As mentioned in section 5.3.3, the approach in Scenario 2 computes the estimate of the current local population from the available known bird distribution. Tables 5.7 and 5.8 list the number of birds that make songs in 11 common song-types in the data 2001 and 2002.

No.	Song-type	Number of birds	Bird distribution (%)
1	*ab*	29	49.15
2	*cb*	13	22.03
3	*cd*	11	18.64
4	*eb*	6	10.17
5	*ef*	3	5.08
6	*gb*	11	18.64
7	*guf*	3	5.08
8	*h*	5	8.47
9	*hd*	2	3.39
10	*huf*	3	5.08
11	*kb*	3	5.08

Table 5.7. Bird distribution in data 2001.

No.	Song-type	Number of birds	Bird distribution (%)
1	*ab*	34	41.98
2	*cb*	20	24.69
3	*cd*	9	11.11
4	*eb*	12	14.81
5	*ef*	2	2.47
6	*gb*	13	16.05
7	*guf*	4	4.94
8	*h*	2	2.47
9	*hd*	20	24.69
10	*huf*	2	2.47
11	*kb*	7	8.64

Table 5.8. Bird distribution in data 2002.

The total number of birds in data 2001 is 59, and for data 2002 is 81. Bird distribution is calculated using (number of birds in certain song-type/total number of birds) × 100%.

Utilizing the above distribution, Tables 5.9 and 5.10 give the bird local population estimate as follows.

No	Song-type	Bird distribution (%)	Bird estimate	Population estimate
1	*ab*	49.15	34	69
2	*cb*	22.03	18	82
3	*cd*	18.64	7	38
4	*eb*	10.17	6	59
5	*ef*	5.08	3	59
6	*gb*	18.64	12	64
7	*guf*	5.08	2	39
8	*h*	8.47	2	24
9	*hd*	3.39	2	59
10	*huf*	5.08	4	79
11	*kb*	5.08	2	39
	Average			56

Table 5.9. Bird population estimate in data 2001.

No	Song-type	Bird distribution (%)	Bird estimate	Population estimate
1	*ab*	41.98	35	83
2	*cb*	24.69	22	89
3	*cd*	11.11	12	108
4	*eb*	14.81	15	101
5	*ef*	2.47	3	122
6	*gb*	16.05	15	93
7	*guf*	4.94	6	122
8	*h*	2.47	3	122
9	*hd*	24.69	17	69
10	*huf*	2.47	3	122
11	*kb*	8.64	9	104
	Average			103

Table 5.10. Bird population estimate in data 2002.

Using their respected standard deviations, the correct populations for these groups are (56 ± 18) birds for data 2001, and (103 ± 18) birds for data 2002.

As mentioned above, the local population estimations of data 2001 and data 2002 are based upon the availability of known bird distribution on those data sets. The estimates, therefore, do not reflect the animal abundance of the whole population.

Scenario 2, however, is easily extended to estimate the total population by detecting birds present both in data 2001 and 2002, and implementing a mark-recapture approach employed in Scenario 1.

As mentioned above, the mark-recapture method is a reliable means of estimating the size of a closed population when there are no gains or losses during the course of the study. The study is, therefore, assumed to be over a short period of time when births, deaths, and movements are few.

The study may also implement distance sampling – discussed in Chapter 2 – for abundance estimation. The important factor in distance sampling is the need to estimate detection distances. The estimation of the detection distance can be accomplished empirically or theoretically (Clapham, 2002). The empirical method involves measuring the actual location of vocalizing birds. The theoretical approach consists of modeling the detection distance utilizing knowledge of source levels, propagation conditions and ambient noise.

Whenever the bird detection distances are available, Scenario 2 may proceed to estimate bird total population using the above distance sampling – a method that is commonly used in the biology community.

5.4.3. Estimating bird population using Scenario 3

The previous scenarios assume data of known species including some training data with song-type labels and individual bird labels (Scenario 1), and data of known species with song-type labels (Scenario 2). Scenario 3 assumes that one has a known species data set only, without song-type or individual bird labels. In this case automatic bird censusing is approached using joint repertoire clustering and individual bird clustering methods. Due to unknown training data, the song-type clustering starts with creating initial song-type models using some presumed small data as seeds. Using these initial HMM song-type models, the module groups data into *K* song-type clusters. An individual animal clustering then estimates the number of animals in each song-type cluster. A population size estimate computes the animal abundance as a final result.

Song-type clustering

As mentioned in the previous section, the clustering experiment builds initial song-type models using some small known exemplars referred to as seeds. Dissimilarity analysis is then employed to assess the consistency of the results.

To implement this, HMM-based *k*-model clustering is run 10 times on the same data set. The equation (2.48) calculates the average dissimilarity value. The smaller the multi-run dissimilarity value $\in$ [0, 1] the more consistent is the clustering method across this song-type data set.

Table 5.11 presents the dissimilarity values and their respected standard deviation from 10 separate clustering runs of two different ortolan bunting data sets, data 2001 and data 2002.

No	Data	Dissimilarity value	Consistency (%)
1	Data 2001	0.2085 ± 0.0705	79.15
2	Data 2002	0.1664 ± 0.0863	83.36

Table 5.11. The dissimilarity values and consistencies of data 2001 and 2002

Results indicate different dissimilarity and consistency values of data 2001 and 2002. It gives 0.2085 ± 0.0705 dissimilarity value for data 2001. This means that for different clustering runs 20.85 % of the song-type data are clustered inconsistently and 79.15 % of the vocalizations are always assigned into the same group. A dissimilarity value of 0.1664 ± 0.0863 for data set 2002 indicates that 16.64 % of the data set are grouped inconsistently, while 83.36 % are clustered to the same group for different run.

The previous results (Scenario 2) on song-type classification of the same data sets indicate the similar acoustic characteristics of song-type *ab* and *kb*, song-type *cb* and *gb*. It can be expected that this acoustic similarity may create inconsistency in the clustering experiments.

Bird clustering

The experiments to estimate the number of birds in song-type data sets utilize deltaBIC computation as a “stopping criterion.” DeltaBIC analysis builds initial bird models by over-clustering data with linear uniform segmentation. Similar to the approach in Scenario 2, the initial number of clusters is adjusted to the size of the data set. To create more initial stable models, the approach maintains a minimum initial cluster member of each cluster to 4 exemplars.

Figures 5.14 and 5.15 show the accumulative deltaBIC values of data 2001 as the function of the number of birds.

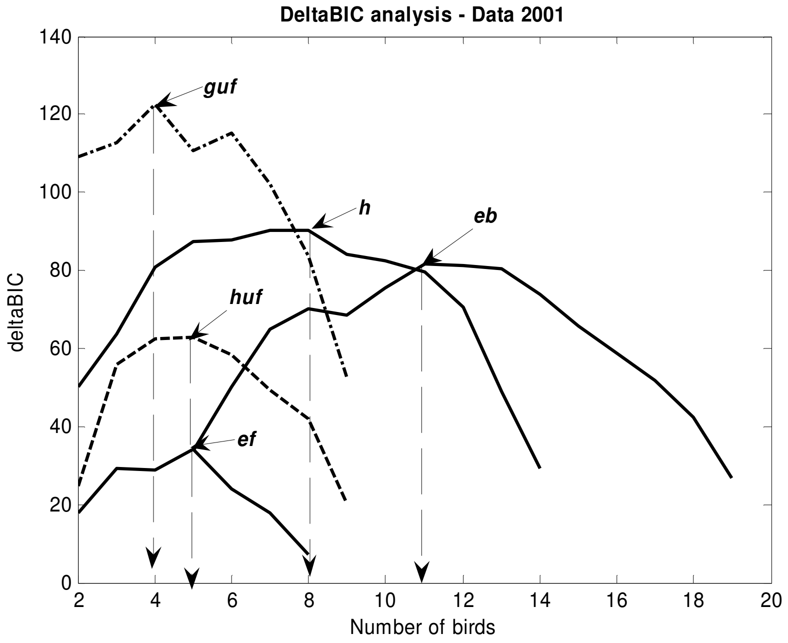

Figure 5.14. DeltaBIC analysis of data 2001 for song *eb*, *ef*, *guf*, *h*, and *huf* data sets

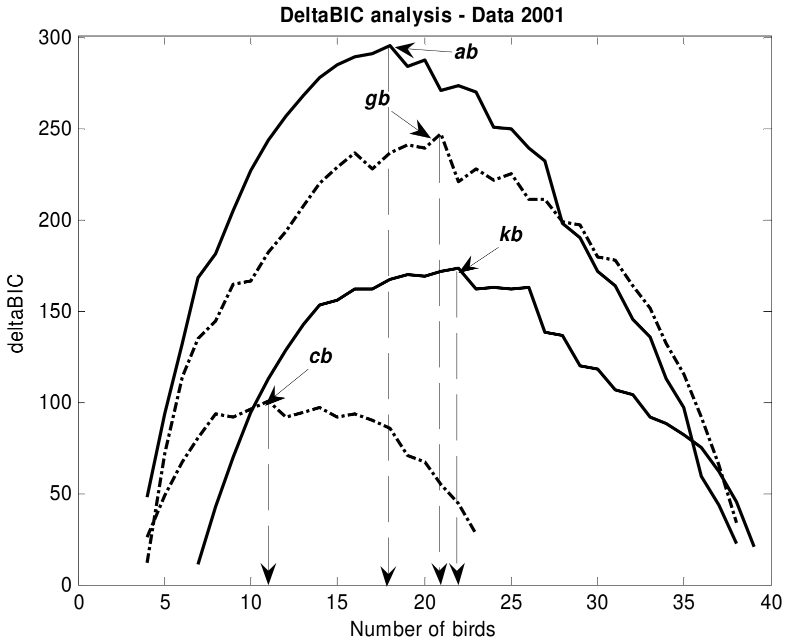

Figure 5.15. DeltaBIC analysis of data 2001 for song *ab*, *cb*, *gb*, and *kb* data sets

Table 5.12 summarizes deltaBIC analysis results of estimating number of birds in data 2001.

No	Song-type	Estimated number of birds	Correct number of birds
1	*ab*	18	23
2	*cb*	11	13
3	*eb*	11	7
4	*ef*	5	3
5	*gb*	21	21
6	*guf*	4	3
7	*huf*	5	3
8	*h*	8	5
9	*kb*	22	20

Table 5.12. DeltaBIC analysis results of data 2001

The results show a different trend than the similar data clustered using Scenario 2 (Table 5.5). The major difference is specifically in the estimation results of song-type *ab*, *gb* and *kb*. Here our method estimates 22 birds in song-type *kb*, and 21 birds in *gb*, both higher than the birds in song *ab*. The similar acoustic characteristics of *ab*, *kb* and *gb* as observed previously may have resulted in many song-type *ab* exemplars that are clustered in song-type *gb* and *kb* and caused the higher number of birds.

Figures 5.16 and 5.17 show the accumulative deltaBIC values of data 2002 for each song-type data. From these deltaBIC analyses the estimated number of birds in each song-type is summarized in Table 5.13.

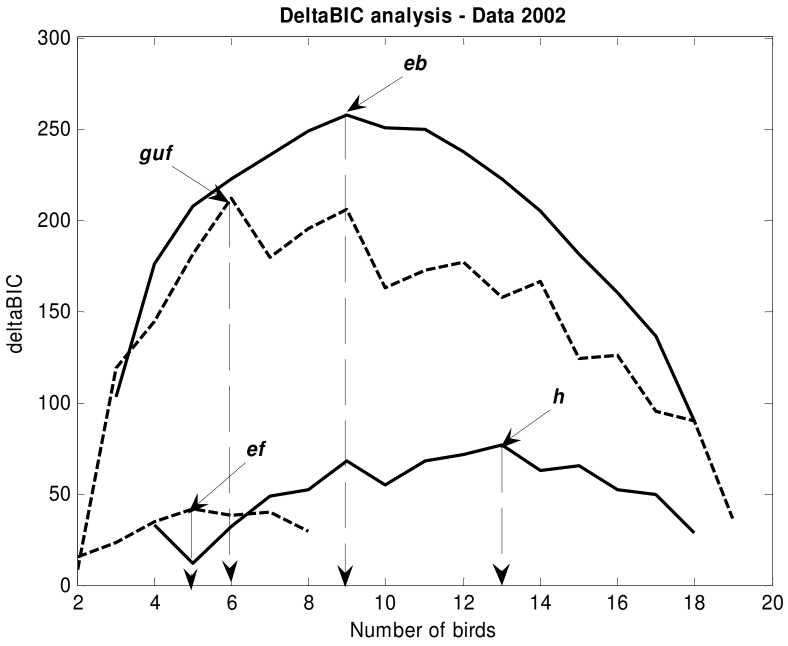

Figure 5.16. DeltaBIC analysis of data 2002 for *eb*, *ef*, *guf*, and *h* data sets

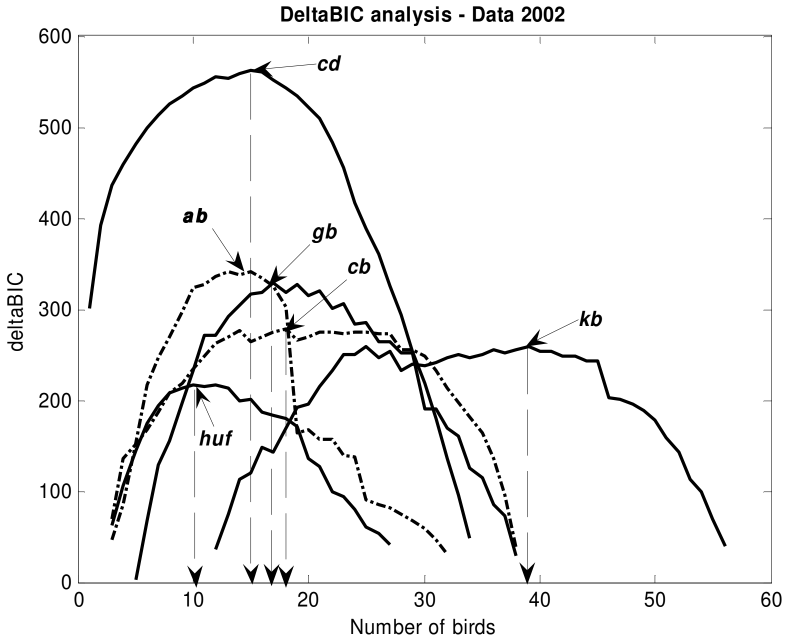

Figure 5.17. DeltaBIC analysis of data 2002 for *ab*, *cb*, *cd*, *gb*, and *kb* data sets

No	Song-type	Estimated number of birds	Correct number of birds
1	*ab*	15	18
2	*cb*	18	20
3	*cd*	15	32
4	*eb*	9	13
5	*ef*	5	2
6	*gb*	17	23
7	*guf*	6	11
8	*huf*	10	15
9	*h*	13	9
10	*kb*	39	50

Table 5.13. DeltaBIC analysis results of data 2002

The results indicate a similar trend to that shown in data 2001. The method estimates a high number of birds in the song *kb* data set. The previous song-type clustering step most probably allocates many numbers of data from different song-types to *kb* due their similar acoustic characteristics. For the song *kb* data set there are individual birds coming from song *ab*, *cb*, *cd* and *gb* data sets. This results in a high number of individual birds grouped together in that data set. Furthermore, there are some individual with only a few samples each. *Bird*166, for instance, has 1 sample vocalization, *bird*171 has 2 samples, and *bird*241 has 1 sample. This data limitation prevents the system to create stable GMM models and as a result of an un-accurate clustering.

Population estimate

When the known distribution of birds in each song-type is available, the method estimates the number of birds in the population based on that reference. Assume that this study has some known distributions of birds as shown in Table 5.9 (for data 2001) and

Table 5.10 (for data 2002). Using these distributions, Tables 5.14 and 5.15 present the bird local population estimation as follows.

No	Song-type	Bird distribution(%)	Bird estimate	Population estimate
1	*ab*	**49.15**	18	**37**
2	*cb*	**22.03**	11	**50**
3	*eb*	**10.17**	11	**108**
4	*ef*	5.08	5	98
5	*gb*	**18.64**	21	**113**
6	*guf*	5.08	4	79
7	*huf*	5.08	5	236
8	*h*	8.47	8	98
9	*kb*	5.08	22	433
	Average (top 4 distribution)			77

Table 5.14. Bird population estimate in data 2001

No	Song-type	Bird distribution(%)	Bird estimate	Population estimate
1	*ab*	**41.98**	15	**36**
2	*cb*	**24.69**	18	**73**
3	*cd*	11.11	15	135
4	*eb*	**14.81**	9	**61**
5	*ef*	2.47	5	203
6	*gb*	**16.05**	17	**106**
7	*guf*	4.94	6	122
8	*huf*	2.47	10	41
9	*h*	2.47	13	81
10	*kb*	8.64	39	451
	Average (top 4 distribution)			69

Table 5.15. Bird population estimate in data 2002

For the above case, however, it is not advisable to implement the reference to the clustering results. For data sets 2001 and 2002, song *kb* consists not only birds from *kb* itself but also birds from *ab*, *cb*, *cd* and *gb* data sets. Birds in song *huf* of data set 2002 cluster together with some birds from song *cd* data. Utilizing known distribution of birds in each song-type might create an over-estimation of the population.

Assuming that this "combined" distribution is unknown, the research selects the lower bound of the population estimate from the maximum number of birds estimated in the song-type data sets. The upper bound, meanwhile, is the total sum of birds estimated in each song-type. Table 5.16 presents the population estimate for data 2001 and data 2002.

Year	Lower bound	Upper bound	Actual number of birds
2001	22	105	50
2002	39	147	71

Table 5.16. Population estimate for data 2001 and 2002

For data 2001 the bird population estimate, then, has a lower bound of 22 birds and the upper bound of 105 birds. For data 2002 the population estimate is between 39 (lower bound) and 147 (upper bound) birds. The estimates are, therefore, (22; 105) birds for data 2001 and (39; 147) birds for data 2002.

5.5. Summary

This chapter presents a new automated method to estimate animal abundance in a population, performed on a Norwegian ortolan bunting data set. The method integrates supervised tasks of repertoire recognition and individual classification, unsupervised tasks of song-type clustering and individual animal clustering, dissimilarity analysis and deltaBIC analysis to estimate the number of birds in a data set. The suggested framework is based upon hidden Markov models commonly used in the signal processing and speech recognition community.

The study discusses three scenarios. Scenario 1 computes the bird population utilizing an approach that starts with song-type recognition over the data set, followed by bird recognition and bird clustering of the song-type data. Bird matching step and population estimate procedure give final result of the population abundance. Scenario 2 explains the abundance estimate using joint repertoire classification and individual bird clustering approach. Scenario 3, meanwhile, employs repertoire clustering and individual bird clustering methods for population estimation. The experiments presented in this chapter show the applicability and effectiveness of the approach in estimating ortolan bunting abundance.

CHAPTER 6

CONCLUSION AND FUTURE WORK

Individually distinct acoustic features have been observed in a wide range of vocally active animal species. The possibility of identifying individuals by their vocalizations may provide a useful tool to assess populations, including studying the population structure, animal abundance and density, seasonal distribution and trends.

This dissertation has explored the problem of population assessment in a new way, by employing an automatic human speech recognition framework to assess marine mammal (beluga whale) population structure and to estimate animal abundance (ortolan bunting).

The method has advantages over physical and visual marking techniques, being non-invasive, using less effort and cost, and being relatively fast and simple to apply. It provides minimal disturbance and does not require the capture and handling of the animals. This would be useful for species that are secretive, sensitive to disturbance and which cannot be readily caught physically or observed visually.

This chapter summarizes main contributions of this dissertation (Section 6.1) and underlines several potential future directions in Section 6.2.

6.1. Summary of contribution and significance

The contributions of this dissertation are as follows.

1. *Feature separation for animal repertoire analysis and for individual animal identification.*

The dissertation discusses feature extraction approaches employed for animal repertoire analysis and for individual animal identification. It addresses the question of which among the features or combination of features are fit for the repertoire recognition task and which one is robust for an individual classification task.

The study examines some feature extraction approaches such as the Greenwood function cepstral coefficients (GFCCs), pitch tracking, delta and acceleration computation, cepstral mean and cepstral variance normalization.

Some validations through song-type recognition and individual identification of ortolan bunting show the features that combine GFCC, energy (E), delta (D), acceleration (A) and variance normalization (VN) are best for call-type recognition; and likewise, GFCC along with energy, delta and acceleration features give better discriminant power for individual bird recognition. For beluga whale repertoire data, meanwhile, GFCC, delta and acceleration features lead to the best performance for repertoire clustering.

2. *An integrated framework for the unsupervised task of model-based clustering.*

This dissertation has presented an integrated framework for model-based clustering using the likelihood feature space. It addresses the problem of clustering of animal vocalizations using HMMs.

The framework incorporates a cluster dissimilarity evaluation and deltaBIC computation to estimate the number of clusters in the data, and maximum variance initialization to initialize cluster models; and suggests dissimilarity index analysis as an unsupervised way of evaluating clustering results.

3. *An empirical study on acoustic assessment of the beluga whale population structure.*

This case study investigates the suitability and usefulness of the framework for marine mammal population structure assessment. It explores the relationship between established beluga social groups as indicated by their vocalizations.

The study addresses the GFCC feature analysis to extract feature vectors of beluga whale repertoires, dissimilarity analysis to estimate the number of different repertoires in the data set, HMM-based *k*-model clustering to group similar repertoires, and dissimilarity value computation to assess consistency of the clustering results.

A comparative study of wild and captive beluga repertoires shows the reliability of the approach to assess the acoustic characteristics (similarity, dissimilarity) of the established social groups. The results demonstrate the feasibility of the method to assess, to track and to monitor the beluga whale population for potential conservation use.

4. *An integrated framework for supervised classification and unsupervised clustering in population estimates.*

The dissertation has proposed an integrated framework that combines the advantages of supervised classification and unsupervised clustering approaches to estimate the number of animals in a population. The method is able to estimate animal abundance using three possible scenarios:

(a). Assuming availability of training data from a specific species with call-type labels and speaker labels, the method estimates abundance utilizing supervised repertoire classification and individual recognition, unsupervised individual animal clustering, and mark-recapture computation (Scenario 1).

(b). With availability of training data with only call-type labels (no individual identities), the proposed method is able to perform population estimation by implementing joint supervised repertoire classification and unsupervised individual clustering (Scenario 2).

(c). With availability of a few call-type examples, but no full training set, the method is able to perform population estimation using joint unsupervised repertoire clustering and individual animal clustering (Scenario 3).

The experiments performed over the Norwegian ortolan bunting data set show the feasibility and effectiveness of the method in estimating ortolan bunting population.

6.2. Future work

There are several potential future directions for this work.

1. Generalize-able method: species assessment and population estimation.

The dissertation suggests three possible scenarios to assess animal populations, whether for population structure assessment or for population abundance estimate. The framework is easily expandable and generalized to other species. The modification and adjustment would be related to the features being used, the HMM parameters being employed, and some species-specific characteristics being examined. Species-specific frequency warping functions that characterize the frequency range of the species vocalizations can be incorporated into the GFCC features. The window size and its step size can be adjusted to accommodate the rate of change of the species vocalizations. The HMM and GMM mixtures might be adjusted to account for different degree of the vocalization complexity. A language model, especially those species which use a syllabic structure, can be applied to model the grammar of the repertoire.

For estimating abundance of animals comprised of different species, whenever species specific data sets are available, the study may be expanded to incorporate initial species separation and classification. This starts with an initial species recognition task, followed by the method used in the Scenario 3 that utilizing joint repertoire and individual animal clustering.

The HMM-based framework proposed in this dissertation would be a valuable survey tool to investigate and to improve current methods (Terry and McGregor, 2002; Holschuh, 2004; Hartwig, 2005) and even further to estimate their abundance.

2. Call-independent HMM-based abundance estimate.

The current approaches of the individual animal identification and clustering employed in this dissertation are based upon the similarity of the same call-type or repertoire. The study splits the data set into data of the same calls and identifies the individuals based upon the data of those calls.

This call-dependent technique is arguably difficult to implement under the following conditions (Fox, 2008): (a) individuals temporarily change their vocalizations due to social context, body condition, time of year, emotional state and temperature; (b) individuals permanently change their calls and have new syllables or entirely new vocalizations because of the natural progression, (c) individual animals in a species have limited call sharing, (d) individuals have extensive repertoire variations.

To carry out individual identification regardless of call-types or repertoires, future study sees the importance of addressing failure to identify such features so far, investigating features that are specific to the individual's vocalization, stable in spite of the particular repertoire produced; constant over a long period, robust to noise and robust against the variation of voice quality and speaking manner.

3. Combining acoustic-based and visual-based method of population assessment

In most species studies there has been greater emphasis on visual methods to estimate density or abundance. Few studies, however, have initialized an integration of visual and acoustic methods for population assessment which has the potential to greatly improve the abundance estimate. The greatest benefit, therefore, might be obtained by using the best attributes of each method in an integrated survey.

Acoustic methods can provide improvement by extending search range, by allowing survey at night, by detecting animals that are not visible, by estimating the fraction of animals missed by visual method (Clapham, 2002). The purely acoustic estimation of the number of individuals in a large group with many overlapping vocalizations remains challenging. For bird population, only active calling or singing birds (mostly males) can be detected. Females and immature males may be missed because they vocalize less frequently. Visual methods, therefore, are still important.

BIBLIOGRAPHY

Adi Kuntoro, Osiejuk, T.S., Johnson, M.T., (2004), “Automatic song-type classification and individual identification of the ortolan bunting (Emberiza hortulana L) bird vocalizations,” *The Journal of the Acoustical Society of America*, vol. 116:2639.

Adi Kuntoro, Johnson, M.T., (2006), “Feature normalization for robust individual identification of the Ortolan bunting (*emberiza hortulana L*),” *The Journal of the Acoustical Society of America*, vol. 119, no. 5.

Adi Kuntoro, Sonstrom, K., Scheifele, P., Johnson, M.T., (2008), “Unsupervised validity measures for vocalization clustering”, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Las Vegas, Nevada, March 30 – April 4, 2008

Ajmera, J., Bourlard, H., Lapidot, I., McCowan, I., (2002), “Unknown-multiple speaker clustering using HMM,” *IDIAP Research Report* 02-07.

Alain de Cheveigne, Hideki Kawahara, (2002), YIN, a fundamental frequency estimator for speech and music, *The Journal of the Acoustical Society of America*. Vol. 111, no. 4, pp. 1917-1930.

Au, W.W.L., (1993), *The sonar of dolphins*, New York: Springer-Verlag.

Barlow, J., “Management aspects of population structure,” in Mellinger, D., and J. Barlow (2003), *Future directions for acoustic marine mammal surveys: Stock assessment and habitat use,* Report of a workshop held in La Jolla, CA, 20-22 November 2002, NOAA OAR Special Report, NOAA/PMEL Contribution No. 2557.

Barlow, J., and B.L. Taylor (1998), “Preliminary abundance of sperm whales in the northeastern temperate Pacific estimated from a combined visual and acoustic survey,” *Int. Whal. Comm. Working Paper SC*/50/*CAWS*20, 19 pp.

Barras, C., Zhu, X., Meignier, S., Gauvain, J. L., (2004), “Improving speaker diarization,” *Proc. DARPA 2004 Rich Transcription Workshop* (*RT* ’04), November 2004.

Barreto, G.A., Araujo, A.F.R., (2004), “Identification and control of dynamic systems using self-organizing map,” *IEEE Trans. Neural Networks*, Vol. 15, No. 5, pp.1244-1259.

Beeman, K., (1998), “Digital signal analysis, editing, and synthesis,” in Hopp, S.L., Owren, M.J., Evans, C.S., (eds.), *Animal Acoustic Communication: Sound Analysis and Research Methods*, Berlin: Springer.

Berndt, D. J., Clifford, J., (1996), “Finding patterns in time series: a dynamic programming approach,” *Advances in Knowledge Discovery and Data Mining*, Menlo Park: AAAI Press.

Beightol, D.R., Samuel, D.E., (1973), “Sonographic analysis of the American woodcock’s peent call,” *J. Wildl. Mgmt*, 37:470-475.

Bel’kovich, V.M., and Sh’ekotov, M.N., (1992), ‘‘Individual signals of belugas associated with hunting behavior in the white sea,’’ in Thomas, J., Kastelein, R.A., and Supin, A.Y., (eds.), *Marine Mammal Sensory Systems*, edited by, New York: Plenum, pp. 439–449.

Bicego, M., Murino, V,. Figueiredo, M.A.T., (2003), “Similarity-based clustering of sequences using Hidden Markov Models,” *Machine Learning and Data Mining (MLDM03)*, LNAI 2734, P. Perner and A. Rosenfeld Eds., Springer, pp. 86-95.

Biernacki, C., Celeux, G., Govaert, G., (2000), “Assessing a mixture model for clustering with the integrated completed likelihood,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22 (7), pp. 719-725.

Borchers, D.L., Buckland, S.T., Zucchini, W., (2004), *Estimating Animal Abundance: Closed Populations*, London: Springer.

Bourlard, H., Morgan, N., (1998), “Speaker verification, a quick overview,” *IDIAP Research Report*, IDIAP-RR 98-12.

Brown, J.C., Miller, P.J.O., (2007), “Automatic classification of killer whale vocalizations using dynamic time warping,” *Journal of the Acoustical Society of America*, 122(2):1201-1207.

Buck John R., Tyack Peter L., (1993), “A quantitative measure of similarity for *tursiops truncates* signature whistles,” *Journal of the Acoustical Society of America*, 94(5):2497-2506.

Buckland, S.T., Anderson, D.R., Burnham, K.P., Laake, J.L., Borchers, D.L., Thomas, L., (2004), *Introduction to Distance Sampling: Estimating abundance of biological populations*, New York: Oxford University Press.

Butler Matthew, (2003), *Hidden Markov Model Clustering of Acoustic Data*, Thesis, School of Informatics, University of Edinburg.

Butynski, T.M., Chapman, C.A., Chapman, L.J., Weary, D.M., (1992), “Use of male blue monkey ‘pyow’ calls for long-term individual identification,” *Am. J. Primatol*, 28:183-189.

Cadez, I. V., Gaffney, S., Smyth, P., (2000), "A general probabilistic framework for clustering individuals and objects," *Proc. 6th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pp. 140-149.

Campbell, J.P., (1997), "Speaker recognition: a tutorial," *Proceedings of the IEEE*, Vol. 85, No.9.

Chen, S. S., Gopalakrishnan, P. S., (1998), "Speaker environment and channel change detection and clustering via the Bayesian Information Criterion," *Proc. DARPA Broadcast News Transcription and Understanding Workshop*.

Chen, D.R., Chang, R.F., Huang, Y.L., "Breast cancer diagnosis using self-organizing map for sonography," *Ultrasound Med. Biol.*, Vol. 1, No. 26, pp. 405-411.

Clark, C.W., Marler, P., Beeman, K., (1987), "Quantitative analysis of animal vocal phonology: an application to swamp sparrow song," *Ethology*, 76:101-115.

Clark, C.W., Bower, J.B., Ellison, W.T., (1991), "Acoustic tracks of migrating bowhead whales, Balaena mysticetus, off Point Barrow, Alaska based on vocal characteristics," *Rep. Intl. Whal. Comm.*, 46:541-554.

Clemins, P.J., (2005), *Automatic classification of animal vocalizations*, Ph.D. Dissertation, Marquette University, Milwaukee, WI.

Clemins, P.J., Trawicki, M.B., Adi, K., Jidong Tao, Johnson, M.T., (2006), "Generalized perceptual features for vocalization analysis across multiple species," *IEE International Conference on Acoustics, Speech and Signal Processing*, ICASSP Proceedings, vol. 1: 14-19.

Cramp, S., Perrins, C.M., (1994), *The birds of the western palearctic*. Volume IX. Buntings and new world warblers, Oxford: Oxford University Press.

Dale, S., (2000), "The importance of farmland for ortolan buntings nesting on raised peat bogs," *Ornis Fennica* 77:17-25.

Dale, S., (2001a), "Causes of population decline of the ortolan bunting in Norway," in Tryjanowski, P., Osiejuk, T.S., Kupczyk, M., (eds.), *Bunting studies in Europe*, Poznan: Bogucki Wyd. Nauk, pp. 33-41.

Dale, S., (2001b), "Female-biased dispersal, low female recruitment, unpaired males, and the extinction of small and isolated bird populations," *Oikos* 92:344-356.

Dale, S., Hagen, O., (1997), "Population size, distribution and habitat choice of the ortolan bunting *Emberiza hortulana* in Norway," *Fauna norv. Ser. C, Cinclus* 20:93-103.

Das Gautam, Gunopulos Dimitrios, “Time series similarity and indexing”, in Nong Ye (ed), 2003, *The handbook of data mining*, New Jersey: Lawrence Erlbaum Associates Publisher, pp. 279-304.

Davis, S. B., Mermelstein, P., (1980), “Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28:357-366.

Deller, J.R., Hansen, J.H.L., Proakis, J.G., (1993), *Discrete-time processing of speech signals*, New York: A John Wiley & Sons, Inc.

Doddington, G.R., “Speaker recognition – identifying people by their voices,” *Proceedings of the IEEE*, Vol. 73, No. 11.

Dong, G., Xie, M., (2005), “Color clustering and learning for image segmentation based on neural networks,” *IEEE Trans. Neural Net.*, Vol. 16, No.4, pp. 925-936.

Duda, O.R., Hart, P.E., Stork, D.G., (2001), *Pattern Classification*, New York: John Wiley & Sons, Inc.

Edwards, P.P., “Hearing ranges of four species of birds,” *The Auk*, Vol. 60:139-241.

Eriksson Thomas, Kim Samuel, Kang Hong-Goo, Lee Chungyong, (2005), “An information-theoretic perspective on feature selection in speaker recognition,” *IEEE Signal Processing Letters*, Vol. 12, No. 7, July 2005.

Fox, E.J.S., (2008), “A new perspective on acoustic individual recognition in animals with limited call sharing or changing repertoire,” *Animal Behaviour*, doi:10.1016/j.anbehav.2007.11.003.

Furui, S., (1981), “Cepstral analysis technique for automatic speaker verification,” *IEEE Trans. Acoust. Speech Signal Process*.

Furui, S., (1986), “Speaker-independent isolated word recognition using dynamic features of speech spectrum,” *IEEE Transaction on Acoustics, Speech and Signal Processing*, Vol. ASSP-34, No. 1.

Galeotti, P., Pavan, G., (1991), “Individual recognition of male tawny owls (*Strix aluco*) using spectrograms of their territorial calls,” *Ethol. Ecol. Evol.*, 3:113-126.

Garrison Lance, Swartz Stephen, Martinez Anthony, Stamates Jack, Proni John, “Passive hydro-acoustic detection of marine mammals at SEFSC,” in Mellinger, D., and J. Barlow (2003), *Future directions for acoustic marine mammal surveys: Stock assessment and habitat use*, Report of a workshop held in La Jolla, CA, 20-22 November 2002, NOAA OAR Special Report, NOAA/PMEL Contribution No. 2557.

Ghosh Joydeep, (2003), “Scalable clustering”, in Nong Ye (ed), *The handbook of data mining*, New Jersey: Lawrence Erlbaum Associates Publisher, pp. 247-277.

Gilbert, G., McGregor, P.K., Tyler, G., (1994), “Vocal individuality as a census tool: practical considerations illustrated by a study of two rare species,” *Journal Field Ornithology*, vol. 65(3), pp. 335-348.

Gold, B., Morgan, N., (2000), *Speech and audio signal processing: processing and perception of speech and music*, New York: John Wiley & Sons, Inc.

Goldin, D. Q., Kanellakis, P. C., (1995), On similarity queries for time-series data: constraint specification and implementation, in *Proceeding of the 1995 International Conference on the Principles and Practice of Constraint Programming*, Berlin: Springer, pp. 137-153.

Greenwood, D.D., (1961), “Critical bandwidth and the frequency coordinates of the basilar membrance,” *The Journal of the Acoustical Society of America* 33(10):1344-1356.

Greenwood, D.D., (1990), “A cochlear frequency-position function for several species – 29 years later,” *The Journal of the Acoustical Society of America* 87(6):2592-2605.

Guterman, H., Cohen, A., Lapidot, I., (2002), “Unsupervised speaker recognition based on competition between self-organizing maps,” *IEEE Trans. Neural Net.*, Vol. 13, No. 4, pp. 877-887.

Halkidi Maria, Batistakis Yannis, Vazirgiannis Michalis, (2001), “On clustering validation techniques,” *Journal of Intelligent Information Systems*, The Netherlands, Kluwer Academic Publishers, 17:2/3, pp. 107-145.

Harma, A., (2003), “Automatic identification of bird species based on sinusoidal modeling of syllables,” *IEEE Int. Conf. Acoust. Speech and Signal Processing* (*ICASSP’2003*), Hongkong.

Hartwig Simone, (2005), “Individual acoustic identification as a non-invasive conservation tool: an approach to the conservation of the African wild dog *Lycaon pictus* (Temmnick, 1820),” *Bioacoustics: The International Journal of Animal Sound and its Recording*, Vol. 15:35-50.

Hildebrand, J., “Acoustic assessment of marine mammal populations,” in Mellinger, D., and J. Barlow (2003), *Future directions for acoustic marine mammal surveys: Stock assessment and habitat use*, Report of a workshop held in La Jolla, CA, 20-22 November 2002, NOAA OAR Special Report, NOAA/PMEL Contribution No. 2557.

Holschuh, C. (2004), "Monitoring habitat quality and condition of Queen Charlotte saw-whet owls (*Aegolius Acadicus Brooksi*) using vocal individuality," web.unbc.ca/~otterk/publications/Holscuh%202004.pdf

Huang, X., Acero, A., Hon, H.W., (2001), *Spoken language processing: a guide to theory, algorithm, and system development*, New Jersey: Prentice Hall PTR.

Ito, K., Mori K., Iwasaki, S., (1996), "Application of dynamic programming matching to classification of budgerigar contact calls," *The Journal of the Acoustical Society of America*, vol. 100, pp. 3947-3956.

Jain, A. K., Dubes, R. C., (1988), *Algorithms for clustering data*, Prentice Hall.

Janik, V.M., Dehnhardt, G., Todt, D., (1994), "Signature whistle variation in a bottlenosed dolphin, *Tursios truncates*," *Behavioral Ecology and Sociobiology*, 35:243-248.

Jelinek, F., (1997), *Statistical methods for speech recognition*, Cambridge, Massachussetts: MIT Press.

Jin Qin, Laskowski Kornel, Schultz Tanja, Waibel Alex, (2004), "Speaker segmentation and clustering in meetings," *Proceeding of the International Conference of Spoken Language Processing*, South Korea.

Johnson, S. E., (1999), "Who spoke when? – Automatic segmentation and clustering for determining speaker turns," *Proc. Eurospeech*.

Jones, D. N., Smith, G. C., (1977), "Vocalizations of the Marbled Frogmouth II: an assessment of vocal individuality as a potential census technique," *Emu* 97:296-304.

Kamminga, C. and Wiersma, H., (1981), Acoustical similarities and differences in odontocete sonar signals. *Aquatic Mammals* 8: 41-62.

Kasslin, M., Kangas, J., Simula, O., (1992), "Process state monitoring using self-organizing maps," in Aleksander, I., Taylor, J., (Eds.), *Artificial Neural Networks*, Vol. 2, pp. 1532-1534.

Knab Bernhard, Schliep Alexander, Steckemetz Barthel, Wichern Bernd, (2003), "Model-based clustering with Hidden Markov Models and its application to financial times-series data," www.zaik.uni-koeln.de/~ftp/paper/zaik2002-429.ps.gz

Kogan, J.A., Margoliash, D., (1998), "Automated recognition of bird song elements from continuous recording using dynamic time warping and hidden Markov models: a comparative study," *The Journal of the Acoustical Society of America*, vol. 103(4), pp. 2185-2196.

Kohonen, T., (1990), “The self organizing map,” *Proceeding IEEE*, Vol. 78, No. 9., pp. 1464-1480.

Kohonen, T., Kaski, S., Lagus, K., Salojarvi, J., Honkela, J., Paatero, V., Saarela, A., (2000), “Self-organizing of a massive document collection,” *IEEE Trans. Neural Networks*, Vol. 11, No. 3, pp. 574-585.

Lange Tilman, Roth Volker, Braun Mikio L., Buhmann Joachim M., (2004), “Stability-based validation of clustering solutions,” *Neural Computation* 16:1299-1323.

Lee, C.H., Soong, F.K., Paliwal, K.K., (1996), *Automatic speech and speaker recognition: advanced topics*, Boston: Kluwer Academic Publishers.

Levenshtein, “Binary codes capable of correcting deletions, insertions, and reversals,” *Soviet Physics Doklady* 10 (1966):707–710.

Li, C., Biswas, G., (2000a), A Bayesian approach to temporal data clustering using hidden Markov models, *International Conference on Machine Learning*, pp. 543-550.

Li, C., Biswas, G., (2000b), Improving clustering with hidden Markov models using Bayesian model selection, *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 1, pp. 194-199.

Masafumi Nishida, Tatsuya Kawahara, (2003), “Unsupervised speaker indexing using speaker model selection based on Bayesian information criterion,” *IEEE International Conference on Acoustics, Speech, and Signal Processing*.

Masters, W.M., Raver, K.A., Kazial, K.A., (1995), “Sonar signals of big brown bats, *Eptesicus fuscus*, contain information about individual identity, age and family affiliation,” *Animal Behaviour*, 50:1243-1260.

McGregor, P.K., Peake, T.M., Gilbert, G., (2000), “Communication behaviour and conservation,” in Gosling, L.M., Sutherland, W.J. (eds), *Behaviour and Conservation, Cambridge: Cambridge University Press*, pp. 261-280.

McGregor, P.K., Peake, T.M., (1998), “The role of individual identification in conservation biology,” in Tim Caro (ed), *Behavioral Ecology and Conservation Biology*, New York: Oxford University Press, pp. 31-49.

Mellinger, D.K., and J. Barlow (2003), *Future directions for acoustic marine mammal surveys: Stock assessment and habitat use*, Report of a workshop held in La Jolla, CA, 20-22 November 2002, NOAA OAR Special Report, NOAA/PMEL Contribution No. 2557.

Mellinger, D.K., Clark, C.W., (2000), “Recognizing transient low-frequency whale sounds by spectrogram correlation,” *The Journal of the Acoustical Society of America* 107(6):3518-3529.

Meignier, S., Moraru, D., Fredouillea, C., Bonastre, J. L., Besacier, L., (2006), “Step-by-step and integrated approaches in broadcast news speaker diarization,” *Computer Speech and Language,* Vol. 20, pp. 303-330.

Moth’d Belal Al-Daoud (2005), “A new algorithm for cluster initialization,” *Transactions on Engineering, Computing and Technology*, vol. 4, pp. 74-76.

Mousset, E., Ainsworth, William, A., Fonollosa, José A. R. (1996): "A comparison of several recent methods of fundamental frequency and voicing decision estimation", In *ICSLP-1996*, 1273-1276.

NIST Spring 2006 (RT06s) Rich Transcription Meeting Recognition, http://www.nist.gov/speech/tests/rt/rt2006/spring/

Nobuyuki Kunieda, Tetsuya Shimamura, and Jouji Suzuki, (2000), Pitch extraction by using autocorrelation function on the log spectrum, *Electronic and Communication in Japan*, part 3, vo. 83, no. 1, pp. 90-98.

Noll A. Michael, (1967), Cepstrum pitch determination, *The Journal of the Acoustical Society of America*, vol. 41, no. 2, pp. 293-309.

Norris, K.S., (1969), “The echolocation of marine mammals,” in Andersen, H.J., (ed), *The Biology of Marine Mammals*, New York: Academic Press, pp. 391-423.

Nottebohm, F., Nottebohm, M., (1978), “Relationship between song repertoire and age in the canary, *Serinus canarius*,” *Zeitschrift fur Tierpsychologie*, 46:298-305.

Oates Tim, Laura Firoiu and Paul R. Cohen, (1999), “Clustering Time Series with Hidden Markov Models and Dynamic Time Warping,” *IJCAI*-99 *Workshop on Sequence Learning.*

Osiejuk, T. S., Ratynska, K., Cygan, J. P., Dale, S., (2003), “Song structure and repertoire variation in ortolan bunting (*Emberiza hortulana L*) from isolated Norwegian population,” *Ann. Zool. Fennici*, vol. 40, pp. 3-16.

Osiejuk, T. S., Ratynska, K., Cygan, J. P., Dale, S., (2005), “Frequency shift in homologue syllables of the Ortolan Bunting, *Emberiza hortulana*,” *Behavioural Processes* 68:69-83.

O’Shaughnessy, D., (2000), *Speech communications: human and machine*, NJ: IEEE Press.

Palka Debra, “Using passive acoustics during cetacean abundance surveys in the Northwest Atlantic,” in Mellinger, D., and J. Barlow (2003), *Future directions for acoustic marine mammal surveys: Stock assessment and habitat use*, Report of a workshop held in La Jolla, CA, 20-22 November 2002, NOAA OAR Special Report, NOAA/PMEL Contribution No. 2557.

Panuccio, A., Bicego, M., Murino, V., (2002), "A Hidden Markov Model-based approach to sequential data clustering", in T. Caelli, A. Amin, R.P.W. Duin, M. Kamel, and D. de Ridder Eds., *Structural, Syntactic and Statistical Pattern Recognition (SSPR02)*, LNCS 2396, Springer, pp. 734-742.

Pardo Jose, M., Anguera Xavier, Wooters Charles, (2007), “Speaker diarization for multiple-distance-microphone meetings using several sources of information,” *IEEE Transactions on Computers*, Vol. 56, No. 9.

Parsons Chris, Dolman Sarah, “The use of sounds by cetaceans,” in Simmonds Mark, Dolman Sarah, Weilgart Lindy (Eds) (2004), *Oceans of noise 2004*: *A WDCS Science Report*, Wiltshire: WDCS the whale and dolphin conservation society.

Payne, R.S. and Webb, D., (1971), “Orientation by means of long range acoustic signaling in baleen whales,” *Annals of the New York Academy of Science* 188: 110-141.

Peake, T.M., Mcgregor, P.K., (2001), “Corncrake Crex crex census estimates: a conservation application of vocal individuality,” *Animal Biodiversity and Conservation*, vol. 24(1), pp. 81-90.

Placer J., Slobodchikoff C. N., Burns Jason, Placer Jeffrey, Middleton Ryan, (2006), “Using self-organizing maps to recognize acoustic units associated with information content in animal vocalizations, *The Journal of the Acoustical Society of America*, vol. 119, no. 5.

Quatieri, T.F., (2002), *Discrete-time speech signal processing: principles and practice*, New Jersey: Prentice Hall PTR.

Rabiner, L.R. (1989), “A tutorial on hidden Markov models and selected applications in speech recognition,” *Proc. IEEE*, vol. 77, pp. 257-286.

Rabiner, L. R., Cheng, M. J., Resenberg, A. E., McGonegal, C. A., (1976), A comparative performance study of several pitch detection algorithms, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-24, no. 5, pp. 399-417.

Rabiner, L.R., Juang, B.H., (1993), *Fundamentals of Speech Recognition*, Prentice Hall.

Rabiner, L.R., Juang, B.H., Lee, C.H., (1996), "An overview of automatic speech recognition," in Lee C.H. (ed.), *Automatic Speech and Speaker Recognition: Advanced Topics*, Boston: Kluwer Academic Publishers, pp. 1-30.

Reynolds, D.A., (2002), "An overview of automatic speaker recognition technology," IEEE *International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP*'02), Vol. 4.

Rose, R.C., (1995), "Keyword detection in conventional speech utterances using hidden Markov model based continuous speech recognition," *Computer Speech and Language*, vol. 9, pp. 309-333.

Saunders, D.A., Wooler, R.D., (1978), "Consistent individuality of voice in birds as management tool," *Emu*, 88:25-32.

Scheifele, P. M., (2003), *Investigation into the response of the auditory and acoustic communication systems in the beluga whale (Delphinapterus leucas) of the St. Lawrence River estuary to noise, using vocal classification*, Ph. D. Dissertation, University of Connecticut, Hartford, CT.

Seneff, S., (1992), "TINA: a natural language system for spoken language applications," *Computational Linguistics*, 18(1):61-86.

Sjare, B. L., and Smith, T. G., (1986a), ''The relationship between behavioral activity and underwater sounds of the white whale, *Delphinapterus leucas*,'' *Canadian Journal of Zooology* 64, 2824–2831.

Sjare, B. L., and Smith, T. G., (1986b), ''The vocal repertoire of white whales, *Delphinapterus leucas*, summering in Cunningham Inlet, Northwest Territories,'' *Canadian Journal of Zooology* 64, 407–415.

Smyth, P., (1997), "Clustering Sequences with Hidden Markov Models," in Michael Mozer, Michael I. Jordan, Thomas Petsche (Eds.): *Advances in Neural Information Processing Systems* 9, MIT Press 1997, pp. 648-654.

Solomonoff, A., Mielke, A., Schmidt, M., Gish, H., (1998), "Clustering speakers by their voices," *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 757-760.

Sonstrom Kristine E., (2007), *The classification of vocalizations to identify social groups of beluga whales in the St. Lawrence River Estuary using Hidden Markov Model*, Master Thesis, University of Connecticut, Hartford, CT.

Storkersen, O.R., (1999), "Ny nasjonal rodliste over truete fuglearter," *Var Fuglefauna* 22:149-155, [In Norwegian with English summary].

Sumervuo, P., Harma, A., (2004), "Bird song recognition based on syllable pair histograms," *IEEE Int. Conf. Acoust. Speech and Signal Processing* (*ICASSP'2004*), Canada.

Tchernichovski, O., Nottebohm, F., Ho, C.E., Pesaran, B., Mitra P.P., (2000), "A procedure for an automated measurement of song similarity," *Animal Behaviour*, 59:1167-1176.

Terry, A.M., McGregor, P.K., (2002), "Census and monitoring based on individually identifiable vocalizations: the role of neural networks," *Animal Conservation*, vol. 5, pp. 103-111.

Terry, A.M., Peake, T.M., McGregor, P.K., (2005), "The role of vocal individuality in conservation," *Frontiers in Zoology*,2:10.

Tetsuya Shimmamura, Hajime Kobayashi, (2001), Weighted autocorrelation for pitch extraction of noisy speech, *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 7, pp. 727-730.

Tranter Sue, E., Reynolds Douglas, A., (2006), "An overview of automatic speaker diarization systems," *IEEE Transaction on Audio, Speech, and Language Processing*, Vol. 14, No. 5.

Trawicki, M.B., Johnson, M.T., Osiejuk, T.S., (2005), "Automatic song-type classification and speaker identification of Norwegian ortolan bunting (*Emberiza hortulana*) vocalizations," *IEEE Workshop on Machine Learning for Signal Processing*, September 2005, pp. 277-282.

Tucker, G.M., Heath, M.F., (1994), *Birds in Europe: their conservation status*. Birdlife International, Cambridge, United Kingdom.

Vapola, M., Simula, O., Kohonen, T., Merilainen, P., "Representation and identification of fault conditions of an aesthesia system by means of the self-organizing maps," *Proc. Int. Conf. Artificial Neural Networks* (ICANN'94), Vol. 1, pp. 246-249.

Vepsalainen, V., Pakkala, T., Piha, M., Tiainen, J., (2005), "Population crash of the ortolan bunting Emberiza hortulana in agricultural landscapes of Southern Finland," *Ann. Zool. Fennici* 42:91-107.

Viikki, O., Laurila, K., (1998), "Cepstral domain segmental feature vector normalization for noise robust speech recognition," *Speech Communication*, vol. 25, pp. 133-147.

Wakita, H., (1976), "Instrumentation for the study of speech acoustics," in Lass, N.J., (ed.), *Contemporary issues in experimental phonetics*, New York: Academic Press.

Young Steve, Evermann Gunnar, Kershaw Dan, Moore Gareth, Odell Julian, Ollason Dave, Povey Dan, Valtchev Valtcho, Woodland Phil, (2002), *The HTK Book*, Cambridge University Engineering Department.

Zhong Shi, Ghosh Joydeep, (2003), "A unified framework for model-based clustering," *Journal of Machine Learning Research* 4, pp. 1001-1037.

Zue Victor, Cole Ron, Ward Wayne, (1997), "Speech recognition," in Cole *et al*., *Survey of the state of the art in human language technology,* Cambridge University Press, pp. 3-15.