COMPUTER-BASED CLASSIFICATION OF DOLPHIN WHISTLES

Gao Rui

BEng(Hons), NUS

A THESIS SUBMITTED
FOR THE DEGREE OF MASTER OF ENGINEERING
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
NATIONAL UNIVERSITY OF SINGAPORE
2011

Acknowledgements

I would like to express my very great appreciation to Dr. Mandar Chitre for his valuable and constructive suggestions during the planning and review of this research work. His willingness to give his time so generously has been very much appreciated. I also wish to acknowledge the help provided by Prof. Ong Sim Heng, Dr. Elizabeth Taylor and Dr. Paul Seekings, for their useful critique and patient guidance. My grateful thanks are extended to people in Marine Mammal Research Laboratory, for their help in offering and organizing the experiment data.

Contents

Summary

Over many years, underwater vocalizations of dolphins have been recorded and studied for a variety of purposes such as dolphin behavioral and contextual association, communications, species identification, dolphin localization and census surveys. Most studies focus on dolphin whistles, which are believed to convey information about dolphin identity, relative position and even emotional state [8]. Hence automatic extraction and classification of dolphin whistles from underwater recordings are essential for dolphin researchers when there is a large amount of dolphin whistles in the recording. This thesis works on the analysis and classification of dolphin whistles, which are extracted from a de-noised spectrogram of the underwater recordings.

Two types of dolphin whistle classification are the subject of this thesis. The first one is whistle matching, which measures the level of similarity that the dolphin whistle responds to the template whistles sent by trainers. The second one is clustering, where dolphin whistles are classified with or without training whistles (whose types are labeled by researchers in advance).

This thesis firstly reviewed the past work on dolphin whistle classification and divided the general work into three steps: feature vector, similarity measurement and classification method. Currently the most common feature used to characterize dolphin whistles is the time-frequency representation (TFR) from the whistle spectrogram. The feature space constructed by this feature vector and corresponding whistle similarities were explored. Techniques of image processing and computer vision such as shape context were also applied to dolphin whistles.

Various classification methods were substantially analyzed accordingly. It turned out that these descriptors all have some deficiency in describing whistle similarity compared with human perception.

Dynamic time warping (DTW) was found to be a suitable similarity measure for whistle matching, in that it is very close to the way human copes with different whistling speeds. DTW was tested with TFR, with modifications for specific situation such as noisy or erroneous whistle traces. New feature vectors were then proposed progressively when the problem become complicated in natural clustering. A fast marching method (FMM) was adopted for dynamic warping with advantages over DTW. In all, the new feature vector and similarity measure proposed in this thesis treat whistles as image curves, and hence are named as *the image-based method*. This method was implemented to naturally cluster whistles to explore their patterns. Several experiments with different features, similarity measures and classification methods were compared. It showed that the classification from our image-based method substantially agrees with human categorization of dolphin whistles.

The experimental data was collected from the underwater recordings of the Indo-Pacific humpback dolphins (*Sousa chinensis*) in Sentosa Singapore. A subset of this collection was randomly picked and tested. Their types were labeled by experienced dolphin researchers as the benchmark.

Together with dolphin whistle detection and extraction, dolphin whistle classification will be automated. It will eliminate the tedious visual work of detecting,

extracting and classifying many dolphin whistles. It will also assist researchers in recognizing and analyzing dolphin whistles.

Abbreviations

BMU	**B**est **M**atching **U**nit
DA	**D**ifferentiation **A**bility
CDP	**C**umulative **D**istribution **P**robability
DLDA	**D**iag-**L**inear **D**iscrmininant **A**nalysis
DQDA	**D**iag-**Q**uadratic **D**iscrmininant **A**nalysis
DTW	**D**ynamic **T**ime **W**arping
FDA	**F**isher's **D**iscriminant **A**nalysis
FM	**F**requency **M**odulated
FMM	**F**ast **M**arching **M**ethod
ISPD	**I**ntegrated **S**quared **P**erpendicular **D**istance
KNN	**K** **N**earest **N**eighbors
LDF	**L**inear **D**iscriminant **F**unction
LSDTW	**L**ine **S**egment **D**ynamic **T**ime **W**arping
MDS	**M**ulti-**D**imensional **S**caling
MSE	**M**ean **S**qaured **E**rror
PCA	**P**rincipal **C**omponent **A**nalysis
PC	**P**rincipal **C**omponent

PDF	**P**robabilistic **D**ensity **F**unction
PNN	**P**robabilistic **N**eural **N**etwork
RBF	**R**adius **B**asis **F**unction
QDA	**Q**uadratic **D**iscriminant **A**nalysis
RMSE	**R**oot **M**ean **S**quared **E**rror
SOM	**S**elf-**O**rganized **M**ap
SSE	**S**um-of-**S**quared **E**rror
STFT	**S**hort-**T**ime **F**ourier **T**ransform
TFR	**T**ime-**F**requency **R**epresentation

Symbols

N	number of sampling points along whistle contour
N_S, N_R	number of whistles in Class S or Class R
$d(\mathbf{x}_m, \mathbf{x}_n), d(i,j)$	pairwise distance between whistles
$\mathbf{D}$	difference matrix between two whistle sequences in DTW
$f(i,j), F_{x,y}$	local feature difference from two whistles
$\mathbf{C}$	cost matrix in DTW
$\mathbf{C}_{SC}$	shape context cost matrix
$\mathbf{C}_{shape}$	shape difference
$\mathbf{C}_{\theta}$	shape gradient difference
w_θ, w_i, w_{i0}	weight factor
k	number of clusters defined in k-means
J_e	Sum-of-Squared Error in k-means classification
$\mathbf{w}$	a weighting neuron in competitive learning and SOM
$\mathbf{x}, \tilde{X}$	feature vector of one whistle
K_s	number of segments in contour segmentation
Q_l	left point of query segment
Q_r	right point of query segment

d_l, d_r	signed perpendicular distance
t_l, t_r	time of the end point Q_l or Q_r on query segment
t	time
k	segment curvature
λ	smoothing factor in fast marching method
$\mathbf{T}$	cost matrix by fast marching method
L	segment length
m, n	feature length of whistle T and whistle Q
$\|C_p\|$	length of matching path C_p
θ	orientation of whistle contour
W_d	weight for whistle dissimilarity
W_θ	weight for whistle orientation difference

List of Tables

List of Figures

Chapter 1

Introduction

This thesis presents a systematic review, analysis and design on recognition and classification of dolphin whistles. Due to the difficulty in visually spotting dolphins underwater, dolphin whistle recordings are essential in the recognition and study of dolphins. The classification of dolphin whistles is the first step in those dolphin studies. Hence a robust analysis tool that automatically extracts whistle information from recordings and classifies them into groups is necessary, especially when there are large amounts of whistle data.

1.1 Background and Motivation

There are many difficulties in working with or studying dolphins. Current human-dolphin interaction and training rely on hand gestures and rewarding. This only works with captive dolphins that have been trained and is limited to a very simple set of instructions. When it comes to the study of a wild dolphin, underwater

visual observation is almost impossible due to the poor propagation of light in water. Alternatively, since acoustic signals propagate well in water, underwater recording of dolphin whistles is the most direct and convenient way to detect and study dolphins. It is also possible that acoustic communications can be realized between dolphins and trainer.

The recordings of dolphin vocalizations are studied for dolphin detection, behavioral and contextual association. It has been found that dolphin vocalizations are highly correlated with their behavioral activities and social interaction. For example, echolocation of dolphins clicks is used in foraging and navigation [1]. Infant dolphins echolocate on bubbles to learn the ring play from their mothers [36]. Signature whistles appear to be used as an identity broadcaster to inform other dolphins of an individual's presence [9].

There are mainly three types of dolphin vocalizations [21]:

- Broadband short-duration sonar *clicks*
- Broadband short-duration pulsed sounds called *burst pulse*
- Narrowband frequency-modulated (FM) *whistles*

The series of clicks (called *click trains*) emitted by dolphins are thought to be exclusively used for echolocation. These clicks of different frequencies and types help dolphins examine an object or scan the environment. The burst pulse sounds are a general class containing emotional sounds such as barks, mews, chips and pops [48]. In [4], a burst pulse is found to be more correlated with aggressive encounters. Whistles are believed to be mostly associated with dolphin interactions. Each

dolphin has distinctive signature whistles, parts of which alter with changing circumstances [10]. In a project by Marine Mammal Research Laboratory (MMRL) at the Tropical Marine Science Institute (TMSI), National University of Singapore (NUS), the dolphin whistles are to be extracted, classified and analyzed. The aim is to provide a technique that may be used to study dolphin behavior and the ethology.

The whistles used in this project were extracted from underwater recordings of Indo-Pacific humpback dolphins (*Sousa chinensis*) at the Dolphin Lagoon Sentosa, Singapore. Indo-Pacific humpback dolphins (*Sousa chinensis*) are dark grey in color at birth but gradually lighter through patchy grey on pink to completely pink as they mature. The fatty hump on the back around the dorsal fin becomes more prominent compared with other types of dolphins (for example, bottlenose dolphins (*Tursips truncatus*)). The dorsal fin is small and triangular and positioned near the center of the ventral surface. The humpback dolphins are frequently seen in coastal waters in Singapore.

In a cognitive research project planned by MMRL, the dolphins were trained to pair whistles with objects or actions. These dolphins were also supposed to respond and mimic the template dolphin-like whistles synthesized by dolphin trainers. An acoustically mediated two-way exchange of information between human and dolphins will hopefully be established in long term research. The level of similarity between the template whistles and the responding dolphin whistles needs to be measured. In the meantime, during the course of the research, over 1000 whistles

were collected in underwater recordings. They are the experimental data tested in this thesis to test various methodologies.

In any experiment on dolphin whistles, classification evaluates the acoustic similarity among whistles. It has been suggested that whistle structures can be inspected to identify the dolphin species [39]. Hence classification is important for dolphin recognition and categorization. A computer-based classification is designed to be analogous to the approach of human observation by ear and eye. Optimal classification requires detailed knowledge of the criteria for whistle categorization. This could be achieved with associated dolphin behaviors and used for further dolphin studies.

1.2 Problem Statement and Thesis Goal

Whistle recordings are degraded by many kinds of background noise. For example, snapping shrimps in the habitat produce loud snapping sounds [22]. There is also mechanical noise from boats, pumps, etc. Dolphin clicks and burst pulses appear together with dolphin whistles from time to time; they are not the focus of this project and hence regarded as background noise as well. For dolphin whistles, the harmonics are similar in shape to the fundamental frequency in spectrograms. Most information about identity and behavior are believed to exist in the 'whistle shape' of fundamental frequency and hence the harmonics can be removed.

The cognitive research project by MMRL focused on the 'whistle shape' of the fundamental frequency on whistle spectrogram by the short-time Fourier transform (STFT). A time-frequency representation (TFR) of the whistles is a series of sampled points along the spectral curves of identical or maximum intensity. The number of traces along whistles depends on the time bin defined by STFT. In the first half of this research, Malawaarachchi *et al.* [33] used image processing techniques to remove unwanted noise, suppress harmonics, and trace whistles. With proper parameters, whistles can be successfully extracted. Most of the previous work [35] [28] [37] in whistle classification uses TFR and assumes whistle traces are in high quality.

The work described here is the second half of this dolphin research - classification. In template matching, the synthesized whistles are called *template whistles*, and the whistles to be matched are called *query whistles*. In natural clustering, whistles need to be clustered with little or no prior knowledge. The known prior knowledge on clustering comes from *training whistles*, whose types are pre-labeled by researchers. Correspondingly, other whistles to be classified are called *test whistles*. When there is no prior knowledge on clustering, all whistles are to be *naturally clustered* or *categorized* into different *types* (or *classes*, *groups* in equivalent meaning).

A quantitative measurement is needed to describe whistles, called as *descriptor* or *feature vector*. A *similarity measure* compares these feature vectors, numerically expresses how close the two whistles are (hence called as *similarity*) or how far in opposite (hence called as *dissimilarity* or *distance*).

Conventional descriptors are usually either the physical properties or the time-frequency representations (TFRs). Physical properties include the whistle duration, bandwidth, mean/maximum/minimum frequencies and so on. Whistle shape can be categorized as a constant frequency sweep, loops, etc. For instance, the majority of bottlenose dolphin whistles were found to have zero or one turning point, which was defined as the peak or valley in frequency [38]. Up to now, the most popular descriptor is a vector of frequencies evenly sampled along the whistle curve in the TFR. McCowan [35] presented N-point sampling where $N = 20$. Cross-correlation [28] and k-means [37] on these samples were used to measure the similarity between whistles. In k-means clustering on a small amount of whistles [37], the 20-point feature outperforms coefficients and slopes of polynomial fit. However it only demonstrated with a few dolphin whistles; it will be later shown that this 20-point feature vector does not work well when dealing with large amounts of whistles.

Whistle matching by human visual inspection typically focuses on the general structure of whistle curve rather than specific frequencies. The frequency variation of whistles may be different in time, but that does not affect the overall structure. In natural clustering, the degree of grouping depends on the variety of the entire set and the associated dolphin behaviors. The latter factor is not always available though. In this project, the associated information such as behaviors and contexts is not available.

Classification of dolphin whistles by human observers is usually done by listening to the recording (after shifting the frequency down to the audible range)

or observing the spectrogram. However, it introduces subjectivity in feature measurement and ambiguity in class boundaries. It is also a long and arduous job for researchers to go through whistles one by one in long underwater recordings. The need for an automated tool for whistle detection, tracing and classification is outlined in [39] for measurement standardization and workload reduction.

The three main steps of dolphin whistle classification are:

1. Feature selection
2. Measurements of similarity between feature vectors
3. Classification

Past methods will be reviewed in the above steps with discussion on their importance and interdependence. The first two steps - feature selection and similarity measures - form the key contribution. Several points are listed as the initial guidelines in whistle characterization:

- Features and the matching method should be robust to the imperfections in whistle extraction
- Descriptors should be simple and compact in terms of data size
- Computer-based characterization of whistles should be consistent with the recognition of human inspection
- Similarity measures should tolerate intra-class variations

- Inter-class difference should be distinguishable for a large number of dolphin whistles

With the above considerations and exploration, this thesis aims at a systematic approach characterizing and comparing whistles in a way closer to human perception of dolphin whistles. The categorization by experienced dolphin researchers is initially used as benchmark to verify performance of various methods.

1.3 Contribution

To address the issues highlighted in Section 1.2, this thesis reviews the past methods on dolphin whistle classification and presents the following:

- summarized the key steps in dolphin whistle classification
- applied dynamic time warping (DTW) in dolphin whistle matching with proper modifications
- proposed new features description
- proposed an image-based method describing and comparing dolphin whistles, which exerts the nonlinear mapping with a fast marching method (FMM)

Together with the first step for dolphin whistle detection and de-nosing, the classification proposed in this master thesis can be used to establish an automated dolphin whistle analysis tool.

1.4 Thesis Organization

Chapter 1 gives the general overview and introduction to the thesis, defines the scope and introduces the major achievements.

Chapter 2 introduces the outline of the whole project and spectrogram denoising and whistle extraction. General classification and data collection are also included.

Chapter 3 and Chapter 4 review previous methods for selecting feature vectors, measuring similarity and classification methodology. With the real whistle data, some popular feature vectors, similarity measure and classification algorithms are tested followed by a discussion of the results.

Chapter 5 introduces dynamic time warping (DTW) for template matching with some modifications. Recognizing the problem using DTW on whistle sample points, a structure-focused feature vector is initially proposed. Further improvements are presented in Chapter 6. Segment curvature is proposed to characterize whistles and recognize frequency variation in a set of unknown whistles. The optimal matching between two whistles is constructed in a more robust way by the fast marching method (FMM). Comparative tests are presented in Chapter 7.

The conclusions and future work are given in Chapter 8.

1.5 List of Publications

R. Gao, M. Chitre, S. H. Ong, and E. Taylor, "Template matching for classification of dolphin vocalizations," in Proceedings of MTS/IEEE Oceans'08, Kobe, Japan, 2008.

Chapter 2

Background and Literature Review

This chapter introduces the outline of the project for cognitive dolphin whistles research project launched by MMRL. The previous stage of work - whistle denosing and tracing - is introduced in Section 2.3. Classification, which is the second part of this project, is discussed in general.

2.1 Project Outline

It is believed that humpback dolphins (*Sousa chinensis*) might produce individually identifiable signature whistles when isolated [50]. A study of Pacific humpback dolphins off eastern Australia suggested that whistles might be used as contact calls [51]. In a cognitive dolphin whistles research project launched by MMRL, the Indo-Pacific humpback dolphins kept by Underwater World Singapore Pte. Ltd.

at Sentosa were studied. The project is to study the dolphin whistles with the aim of investigating the associated meaning of dolphin whistles and exploring the possibility of training dolphins by their whistles.

Whistles are often best visualized and described by their time-frequency characteristics in the spectrogram [23]. Rather than extracting a feature vector from the sound wave in the time domain, whistles are *extracted* or *traced* from the spectrogram after whistle detection and de-nosing. After that, whistles are classified by various methods for different applications.

Figure 2.1 shows the two stages of this project. In the first stage (the blue box), dolphin whistles are located from recordings, and de-noised and extracted. The work in the first stage has been done in [33]. The output of the first stage are the whistle traces, which is a sequence of time-frequency representation (TFR) points from the whistle spectrogram. The second stage (the orange box) outlines the main structure of this thesis. Features are selected from whistle traces (mostly) or the segmented spectrogram from the first stage. Figure 2.2 shows the type of classifications and accordingly the commonly used methods.

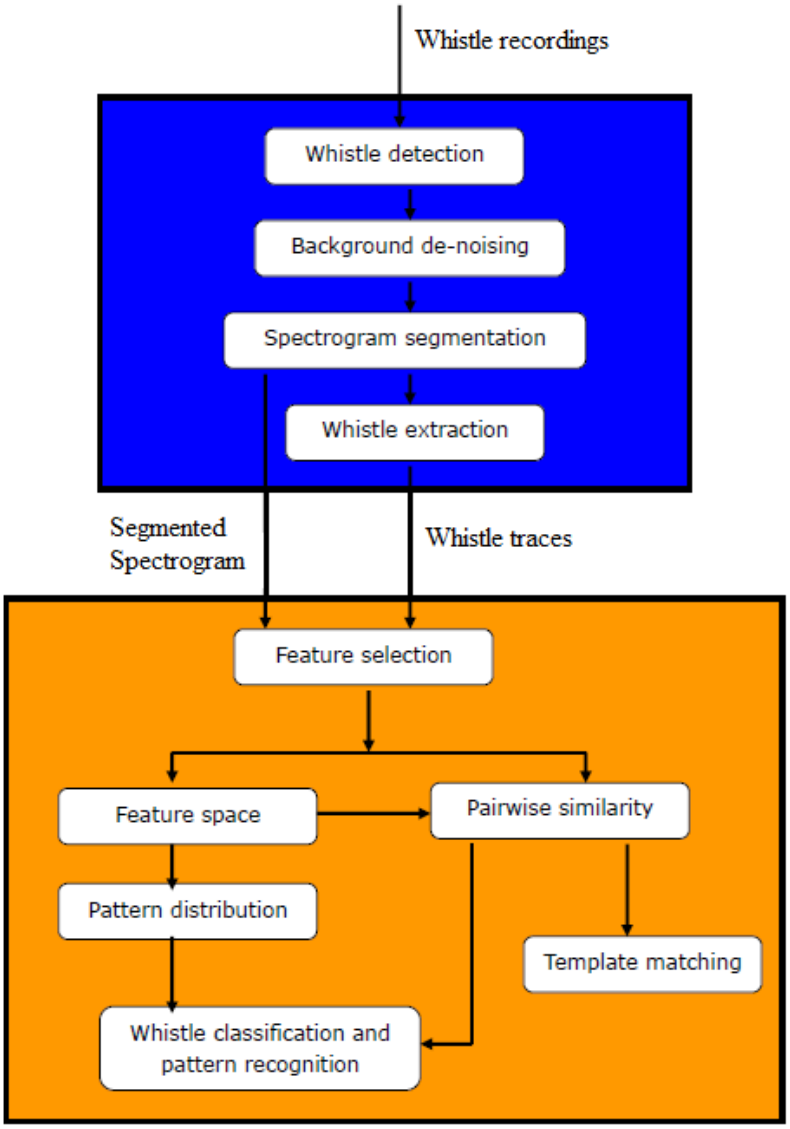


FIGURE 2.1: Block diagram of whistle detection and classification

2.2 Data Collection

The dolphin whistles used in this thesis were recorded from a group of Indo-Pacific humpback dolphins (*Sousa chinensis*) kept by Underwater World Singapore Pte. Ltd. in their facility called the 'Dolphin Lagoon'. Those dolphins are of different

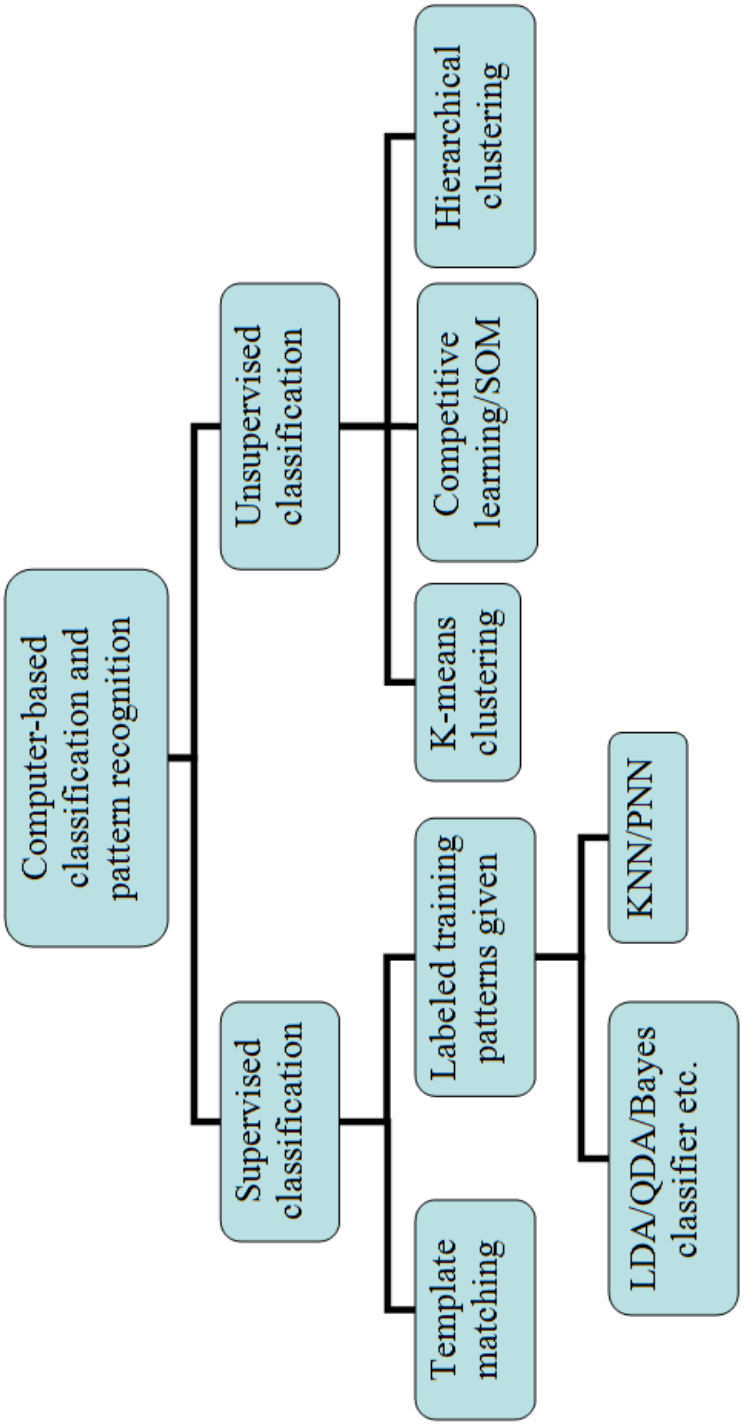


FIGURE 2.2: Overall map of whistle classification and pattern recognition

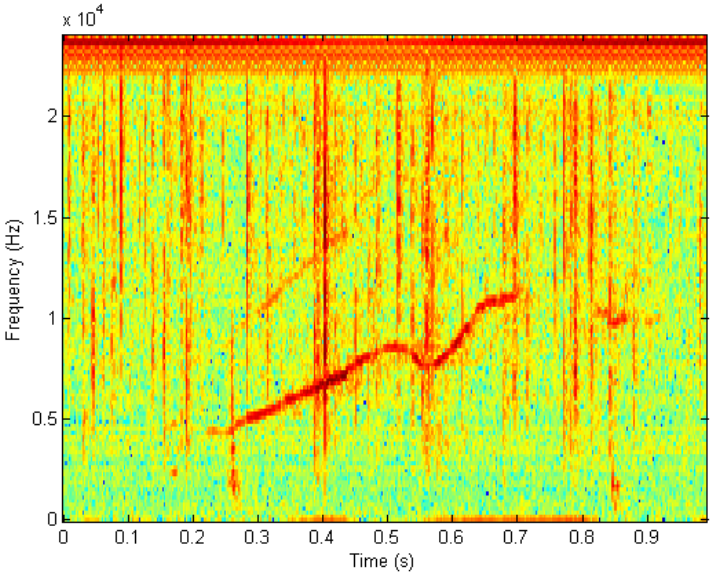
ages: a four year old juvenile male, two female young adults of approximately 14 years old, and 3 mature adults (two males and one female). The dolphins were kept in a semi-natural environment - a large man-made, sand-based, seawater lagoon divided into separate but connected enclosures that were not acoustically isolated. The snapping shrimp noise found in many tropical coastal waters tended to dominate the acoustic environment. Noise from boat passed-by was also present sometimes.

Recordings were made during the experiment sessions for the dolphin research on communications and cognition. A hydrophone was positioned in the water throughout the sessions. It is possible that whistles from dolphins which are not directly engaged in the experiments could also be recorded, with a lower amplitude due to the distance. Dolphin clicks and burst pulse might be also present. The audio sampling rate is 48 kHz.

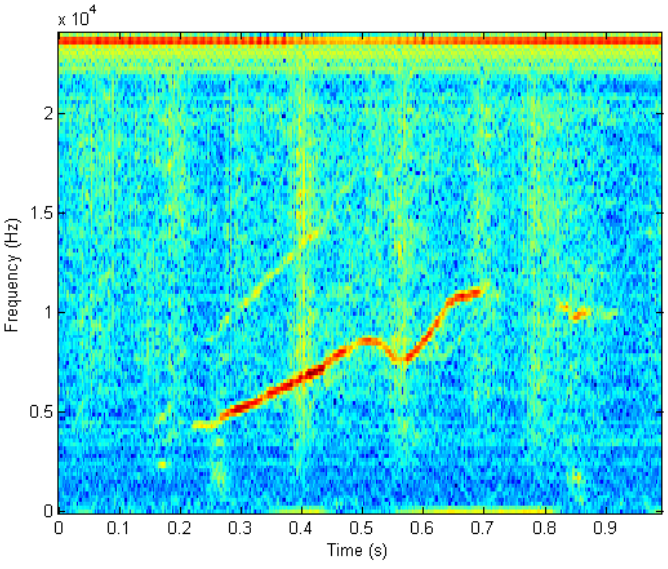
2.3 Whistle de-noising and tracing

Since the recordings were made in a seawater lagoon, the whistle recordings are degraded by a significant amount of transient broadband noise caused by snapping shrimp. Snapping shrimp noise is caused by the snap of a shrimp's claw, which is quite common and forms the ambient noise in tropical warm shallow waters [22]. It appears as vertical lines in the spectrogram (Figure 2.3(a)). A high amplitude snap of a shrimp's claw near the hydrophone could cause the whistle tracing to be

broken or mistaken. Dolphin clicks with similar patterns could also overlap with dolphin whistles.



(a) Original spectrogram of dolphin whistles with snapping shrimp noise



(b) After de-noising by TSF: the snapping shrimp noise is reduced

FIGURE 2.3: Transient suppression filter (TSF) reducing snapping shrimp noise [32]

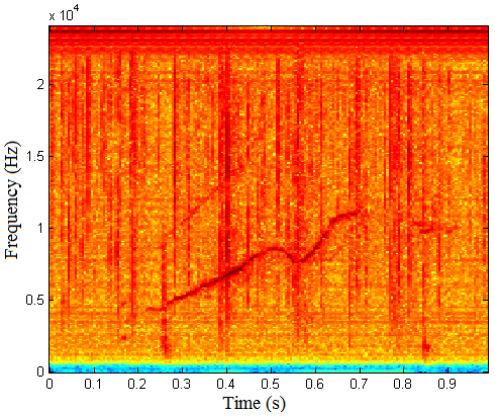
An image processing technique was desired to de-noise the whistle recording

and extract dolphin whistles. This has been implemented successfully in [32]. For example, a transient suppression filter (TSF) is used to detect and attenuate the snapping shrimp noise (Figure 2.3(b)).

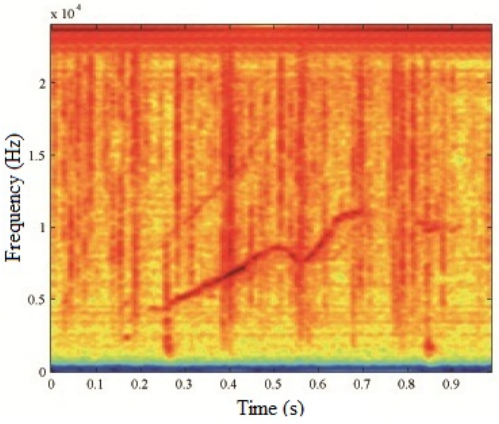
For non-impulsive noise, a bilateral filter is used to preserve edges and smooth the local pixels (Figure 2.4(b)). The harmonics are then suppressed (Figure 2.4(c)). Before tracing, this de-noised spectrogram is segmented from the background based on their intensities (Figures 2.4(d) and 2.4(e). Whistles are traced from the intensity ridge by the Euclidean distance transform, since a one-pixel thick trace is desired. Finally, whistle traces are smoothed by application of Kalman filter (Figure 2.4(f)).

This whistle de-noising and tracing is outlined in the blue box of Figure 2.1 (Section 2.1). The details and parameter settings are available in [32].

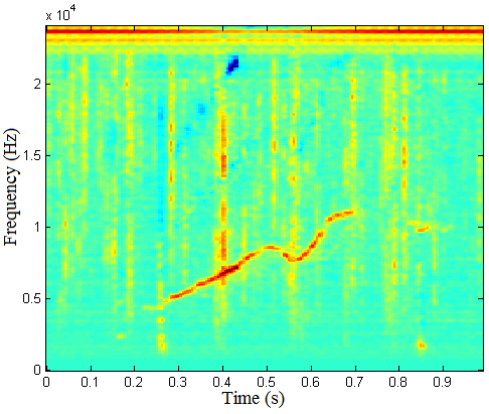
However, it should be noted that the de-nosing and tracing only work well if the parameters are tuned properly. The performance cannot be guaranteed with a large number of dolphin whistles, where we do not have enough or detailed information on the background and intensity of every individual whistle. It will be shown later that with one set of parameter settings there could be outliers (unwanted noise in traces). The pre-assumption about the tracing quality is needed for the automatic classification.



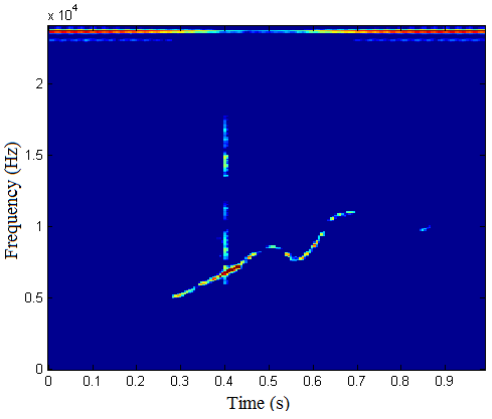
(a) Original spectrogram after high-pass filter



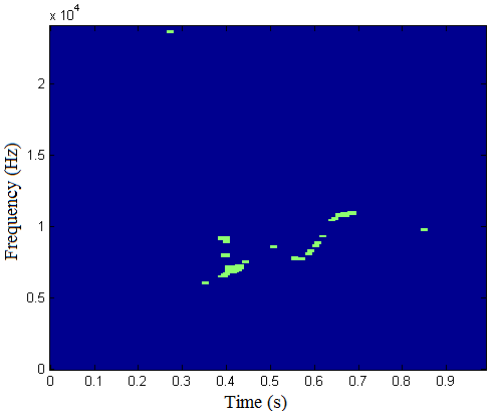
(b) Bilateral filter suppressing non-impulsive background noise



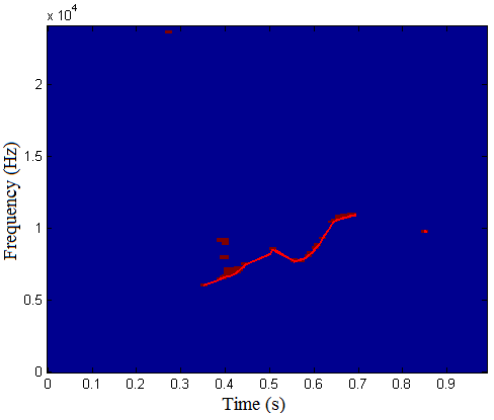
(c) Harmonics suppression



(d) Segmentation performed by regional growing



(e) Local multistage thresholding



(f) Curve tracing with 1st order Kalman filter

FIGURE 2.4: Whistle de-noising and tracing [32]

2.4 Subjective Classification

From all the recordings, over 1000 whistles were extracted and traced and were manually checked for consistency and accuracy against the original spectrograms. They were classified into mainly 7 types by experienced researchers; this classification is called as *subjective classification*. Whistles of poor quality (weak intensity, ambiguous in tracing etc.) are discarded. Whistles with high intensity and obvious tracing are selected from each type. In all, there are 151 whistles selected for the experiment of whistle pattern exploration.

The spectrograms of those 151 whistles are shown in the left column of Appendix A, while their traces (the time-frequency representation (TFR)) are shown in the right column correspondingly. The whistle types A to F are labeled behind the identification number (Whistle 1 to 151). The typical whistle shapes classified for each type are shown in Figure 2.5. The whistles in Appendix A show other variation of the same types.

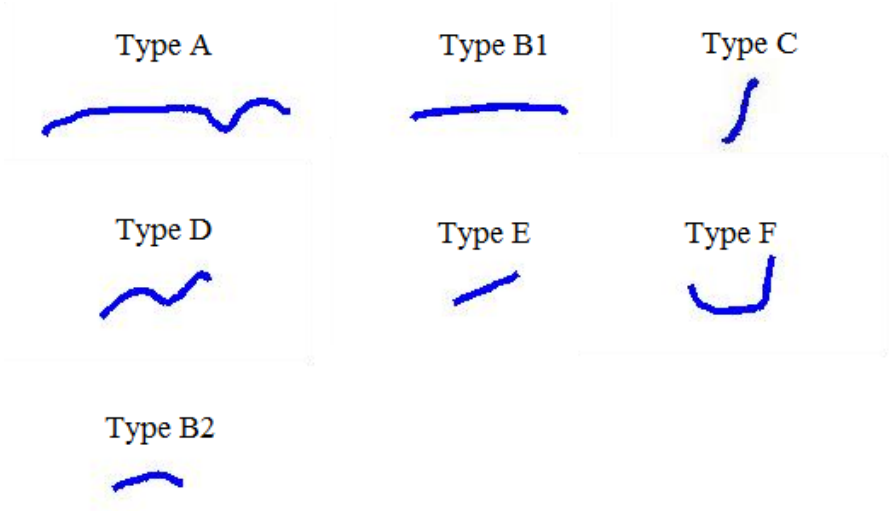


FIGURE 2.5: Typical whistle shapes for 7 types

It can be seen that Type B1 and B2 are similar with their almost constant tone. However, the frequency curve of B1 is flat throughout the duration while that of Type B2 shows a slight increase in frequency during the initial half of the whistle.

This subjective classification is used as the ground truth to verify computer-based classification methods. However it is possible that some whistles are applicable for more than one class, or are classified into a wrong class due to the subjectivity. The classification also depends on the criteria of grouping and the degree of clustering. It is also possible to discover a new class when we explore whistle classification. Only when whistles are correlated with associated dolphin behaviors and environment, can the final classes be defined.

2.5 Related Work on Dolphin Classification

As the first step of computer-based classification, a feature vector (or descriptor) describes dolphin whistles in a numerical way. Information about dolphin whistle characteristics is extracted from the input data, which, most of the time, is a sequence of time-frequency points extracted from the whistle spectrogram. The features selected should characterize whistles of the same type and distinguish those from different types.

As introduced in Chapter 1, a feature vector consisting of the physical properties is most intuitive. In the acoustic identification of nine *Delphinidae* species

[39], 12 physical features were measured for statistical analysis. Multivariate discriminant function analysis and tree-structured non-parametric data analysis were applied. These two methods gave a classification rate of 41.1% and 51.4% respectively, which is relatively low. Besides, this feature vector firstly requires high accuracy in whistle extraction. For example, in noisy environments, an outlier high in frequency compared with the correct traces due to background noise will lead to incorrect bandwidth determination. Another problem in using these features is normalization. Some features are real-valued (for example, the frequency values) while some are integer-valued (for example, the number of inflection points defined as a change in the signs of the frequency slope), and some features might even be categorical (for example, whistle shape described as a constant frequency sweep or loops - a repetition of a single whistle pattern). The features of different types have to be normalized first. Binary or categorical features need to be coded. The normalization and weighting among features probably come from empirical experience, or parameter estimation from a complete training set.

Another feature vector of dolphin whistles samples N points equally along the whistle curve traced from the spectrogram. It was shown that $N = 20$ frequency measures are enough to represent the time-frequency transients of a dolphin whistle [35]. Similarly, N-slope and N-coefficient were proposed for a polynomial fit of whistle traces [37]. These feature vectors can be normalized, square root or log transformed for pre-processing. Whistles are usually classified based on the distribution of these feature vectors in the feature space. For example, probabilistic classification such as the probabilistic neural network (PNN) and Bayesian

classifier uses training whistles to estimate the whistle distribution.

Similarity measurement aims to gain maximum similarity between whistles of the same type and at the same time maximum dissimilarity (or distance) between whistles from different types. In clustering where there are more than one whistles in a class, a representation of the class or the class distance is needed. Let $\mathbf{x}_n$ and $\mathbf{x}_m$ be the feature vectors of the nth and mth whistles in group S and group R, respectively. The feature vector is of length N and hence $\mathbf{x}_m = [x_{m,1}, x_{m,2}, ..., x_{m,N}]^T$ and $\mathbf{x}_n = [x_{n,1}, x_{n,2}, ..., x_{n,N}]^T$. The numbers of members in group S and R are N_S and N_R, respectively. When groups S and R are different, the inter-class distance can be defined as the average distance between all pairs of whistles from these two groups [49]:

$$\rho(R, S) = \frac{1}{N_R N_S} \sum_{n=1}^{N_S} \sum_{m=1}^{N_R} d(\mathbf{x}_m, \mathbf{x}_n) \tag{2.1}$$

where $d(\mathbf{x}_m, \mathbf{x}_n)$ denotes the pairwise distance between two whistles. The larger the $d(\mathbf{x}_m, \mathbf{x}_n)$ is, the less similar the two whistles are. There are other ways to represent the inter-class distance: the maximum or minimum of all the pairwise distances, distance between centroids or centers of two classes, etc. Similarly, the average intra-class distance can be defined as

$$\rho(S) = \frac{1}{N_S^2} \sum_{n=1}^{N_S} \sum_{m=1}^{N_S} d(\mathbf{x}_n, \mathbf{x}_m) \tag{2.2}$$

where feature vectors x_n and x_m come from the same group S of size N_S. To evaluate the clustering performance, a small value of $\rho(S)$ and large values of $\rho(R, S), S \neq R$ are required.

A sum-of-squared error (SSE) criterion [17] is simpler and more commonly used to evaluate the clustering. It is defined by the total squared errors in representing a given set of data by the set of cluster means (or centroids) $\{\mathbf{m}_1, ..., \mathbf{m}_k\}$, where k is the number of classes and the i^{th} class is of size N_i and has a mean

$$\mathbf{m}_i = \frac{1}{N_i} \sum_{j=1}^{N_i} \mathbf{x}_j. \tag{2.3}$$

The SSE J_e is formulated as

$$J_e = \sum_{i=1}^{k} \sum_{\mathbf{x} \in H_i} d(\mathbf{x}, \mathbf{m}_i) \tag{2.4}$$

where H_i is the ith class. An optimal clustering will minimize J_e, which is the best in SSE sense. A normalized J_e was proposed in [37] to compare data sets with different number of features and different dimensions. It is formulated as

$$\hat{J}_e = \frac{1}{d \sum_i N_i} \sum_{i=1}^{k} \sum_{\mathbf{x} \in H_i} d(\mathbf{x}, \mathbf{m}_i) \tag{2.5}$$

where d is the dimension of the feature vector and $\sum_i N_i$ gives the total number of feature vectors in the data set.

Pairwise similarity (or pairwise distance) is the basis for grouping. The similarity of two whistles is based on the qualitative features selected. These two are both crucial in pattern recognition. Examples of similarity measures between features are the cross-correlation, Euclidean distance (2-norm), and averaged absolute difference. In natural clustering without training data, Janik [23] compared

the performance of three similarity measures: McCowan's method [35], cross-correlation coefficients and average difference in frequency. Their limitations were discussed with respect to human observer's classification. Those similarities are all based on the TFR of whistles.

On the other hand, Datta et al. [13] split whistles up into sections, each indicating a 'rising', 'flat', or 'falling' frequency with time, or 'blank' indicating a break in the whistle curves. They encoded whistle curves using quadratic parameters when fitting sections with second order polynomials. This feature vector compactly describes the whistle curve, but this partitioning of whistle curves requires manual work and verification.

It can be seen that intra-class whistles have nonlinear variation in the time domain. The idea of dynamic time warping (DTW) has been very popular in speech recognition [42] [41], acoustic classification [6] [25] and other time series data [27]. It correlates two sequences and simultaneously allows nonlinear warping in time. When two sequences of frequency points are compared by DTW, non-uniform time dilation [7] aligns the whistle curves and recognizes whistles of the same type with slightly local variations. This has been applied to suggest that dolphin calves may model their signature whistles on those of the members of their community [19].

It is indeed very difficult to build up a fully automated system for satisfactory performance from whistle detection, extraction to classification. For example, parameters vary for different signal-to-noise ratio of recordings. Manual validation on whistle tracing is required before the extraction of the whistle features. The

work discussed above and done in this dissertation assume traces of good quality unless otherwise stated.

Chapter 3

Feature Vector and Similarity Measurement

A feature vector consists of information characterizing dolphin whistles in a numerical way. In the automated whistle classification of this thesis, the whistle features are derived from the whistle traces. A conventional feature vector is N-point sampling along the whistle traces where $N = 20$. It is reviewed in Section 3.1 with feature reduction in Section 3.2. This feature vector forms a feature space for similarity analysis. Some common pairwise similarities in the feature space are simply introduced in Section 3.3. On the other hand, the series of whistle traces itself can be used as a feature vector. With different vector length and local variation, dynamic time warping (DTW) and shape context (Section 3.4) are studied. The DTW is introduced in Chapter 5, together with further modifications and classification work.

3.1 Time-Frequency Representation (TFR)

A time-frequency representation (TFR) is a series of sample points along the whistle curve in the spectrogram. Besides the dolphin whistles, the spectrogram also contains the acoustic intensity of background noise. After whistle detection, de-noising and segmentation, the TFR that provides a visualization of the whistle frequency variation over time is traced out. In [35], it was shown that $N = 20$ sample points evenly along the whistle traces are enough to represent the whistles for classification. It is a simplified version of the TFR with a reduction in data sampling in time. In [37], a high-order polynomial was first used to fit the whistle traces. It was found that the 20-point feature outperforms the other two feature vectors, namely, the slopes at the 20 sample points and the coefficients of the high-order polynomial fit on the whistle traces. However, a robust polynomial fit requires shifting and scaling of time and frequency [37]. This scaling causes some difficulties. Firstly, local small frequency variations could be exaggerated when scaled by a narrow whistle bandwidth. Secondly, frequency modulation loses its bandwidth information if scaling is based on the whistles' own bandwidth. This is illustrated in Figure 3.1. After scaling, the polynomial fit of whistle curves is plotted in groups by subjective classification. The frequency range is shifted by the mean of its starting and ending frequencies and scaled by its bandwidth. Time is substituted by the sampling index. As we can see, the sampling points assume that whistles are of the same duration and only record scaled frequencies. For example, some whistles from Type B2 have similar variations as whistles from Type C. Whistles in Type D look quite different due to the different frequency

rising time.

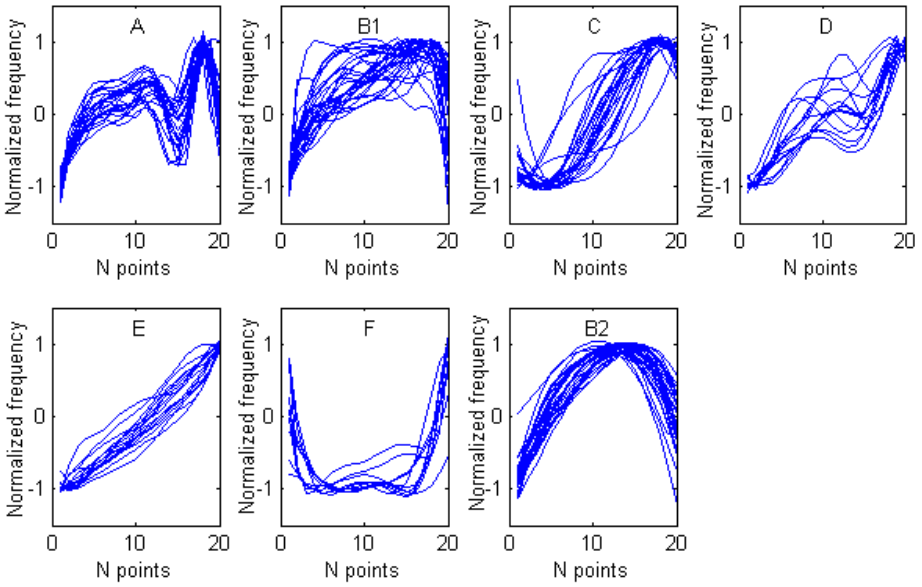


FIGURE 3.1: Group plot of 20-point feature after polynomial fit, frequency is shifted by the mean of the beginning and ending frequencies and scaled by its bandwidth.

On the other hand, human visual inspection typically focuses on the general structure of the whistle curve. Whistles may exhibit slight variations locally such as the variation of speech speed without affecting the overall shape features. Sampling points with equal distribution along different whistle contours may not form the best match when they are paired up by their indices (that is, linear mapping of an N-point feature vector).

Another version of TFR uses *cent* - a relative frequency measure. The cent is expressed with a reference frequency f_{ref}:

$$f_{cents} = 1200 \log_2 \frac{f}{f_{ref}}. \tag{3.1}$$

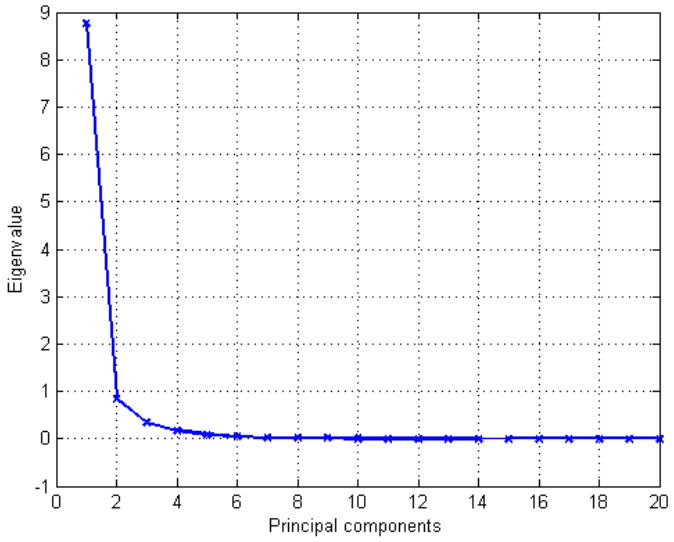
It compares ratios of frequencies rather than absolute differences. For example, a difference of 100 Hz and 200 Hz will be the same as the difference between 400 Hz and 800 Hz. This is identical to human perception of pitch and would be only helpful if we compare the frequencies without scaling. In [6], the reference frequency for orca vocalizations is chosen as 440 Hz, which serves as the standard tone for musical pitch.

3.2 Principal Component Analysis (PCA)

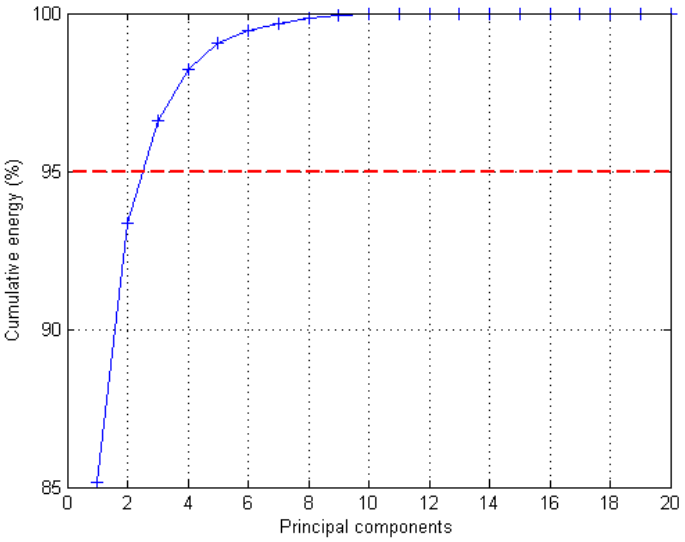
PCA transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called *principal components (PCs)*. These PCs are the dominant variables distinguishing different groups. PCA also reduces the dimension and hence the size of the data of interest. It has been shown that the PCs are the continuous solutions to the discrete cluster membership indicators for k-means clustering [16].

A covariance method is used to compute PCA. When n is the number of whistles and $N = 20$ is the number of sampling points after scaling and polynomial fit, we have an $n \times 20$ data matrix. In Appendix A, there are $n = 151$ whistles. The covariance matrix of this feature vector is a symmetrical matrix where the diagonal elements are the variances for each feature point and the off-diagonal entries are the cross-covariance between features. Among the eigenvectors and eigenvalues found for the covariance matrix, the first principal component (PC) is the data projection on the eigenvector with the largest eigenvalue. The second

PC is then found by projecting data to the eigenvector with the second largest eigenvalue. The subsequent PCs follow the same concept. The eigenvalues and eigenvectors of the covariance matrix are re-arranged in order of decreasing eigenvalues (Figure 3.2(a)). The eigenvalues can be viewed as the energy of corresponding eigenvectors and give the significance of the components. Larger energy indicates a larger variance of the data projection. The accumulated energy for the mth eigenvector is the sum of energy from the first to the mth eigenvalue. A threshold of 95% of the cumulated energy is preserved by keeping the first three PCs (Figure 3.2(b)). The corresponding eigenvectors are kept as the new major basis onto which the data is projected. From Figure 3.2(a), it can be seen that the eigenvalues of the first three PCs are 8.8, 0.84 and 0.33; from the fifth onwards, the eigenvalues are below 0.1 and approach zero. The choice of threshold depends on how much variation information is kept; the effect of the dimension reduction will be tested in the classification shown in Appendix B.



(a) Eigenvalues for principal components

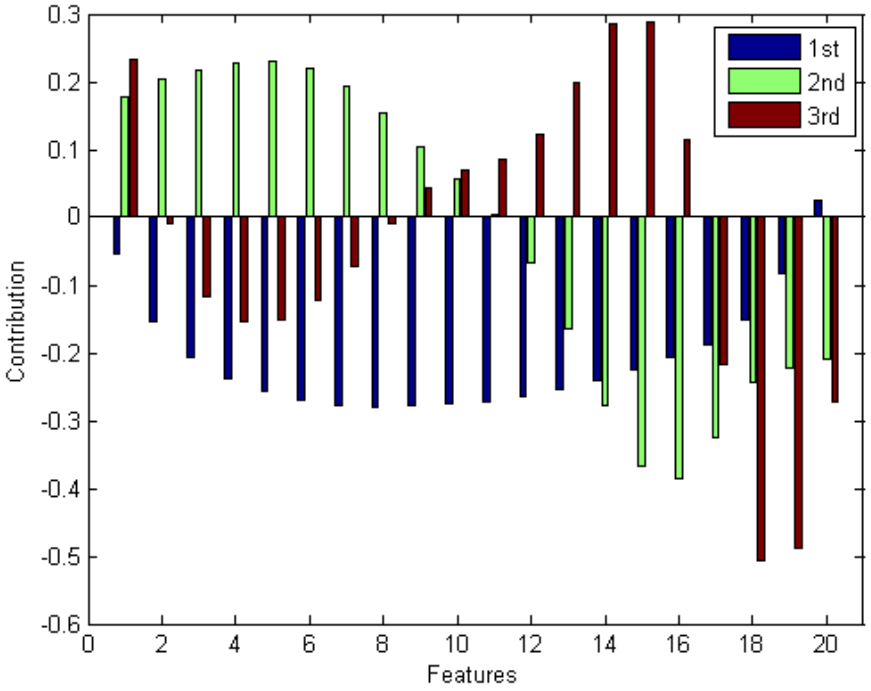


(b) Cumulative energy and thresholding for PCA

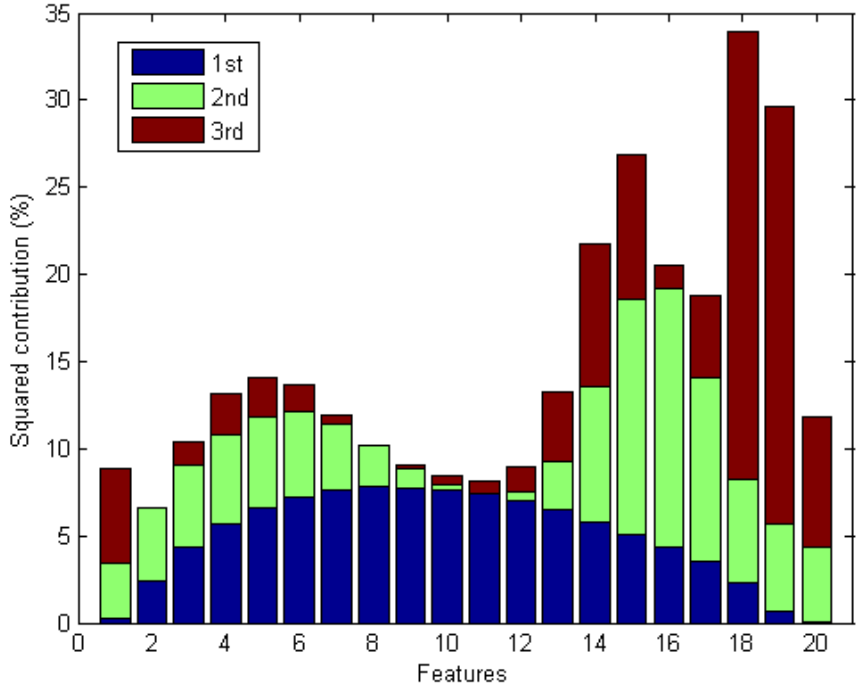
FIGURE 3.2: Eigenvalues of principal components and their cumulative energy

The contribution of each variable to the first three PCs is shown in Figure 3.3(a). While most variables have similar negative contribution to the first PC, the 14th to the 19th variables contribute more to the second and third PCs. The squared values of contribution are plotted in Figure 3.3(b), with contribution summation of 1 for each PC. It shows that the 8th and 9th points have the largest variance, followed by points at first quarter and third quarter of the overall time domain, and finally the near end points (18th and 19th).

After PCA, the N-point ($N = 20$) feature is reduced to a feature vector of three elements. The feature space becomes a 3-dimensional (3-D) space. Figure 3.4(b) shows the group scatter plot in the 3-D feature space. For easier visualization, the scattering of the first two PCs is shown as a 2-D plot in Figure 3.4(a). When whistle distance (or similarity) is viewed as the Euclidean distance between the



(a) Contribution of variables for PCA

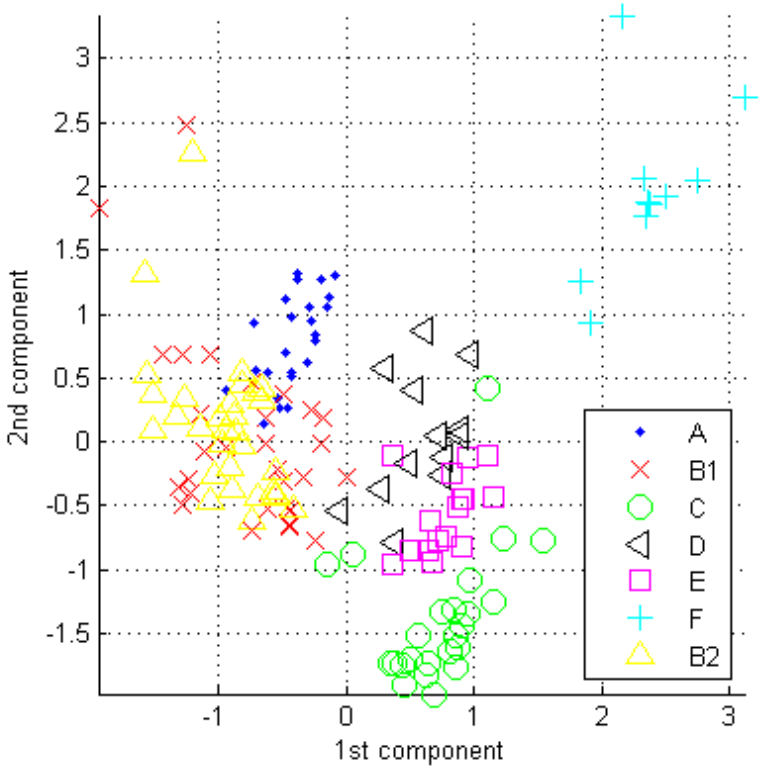


(b) Squared contribution in percentage

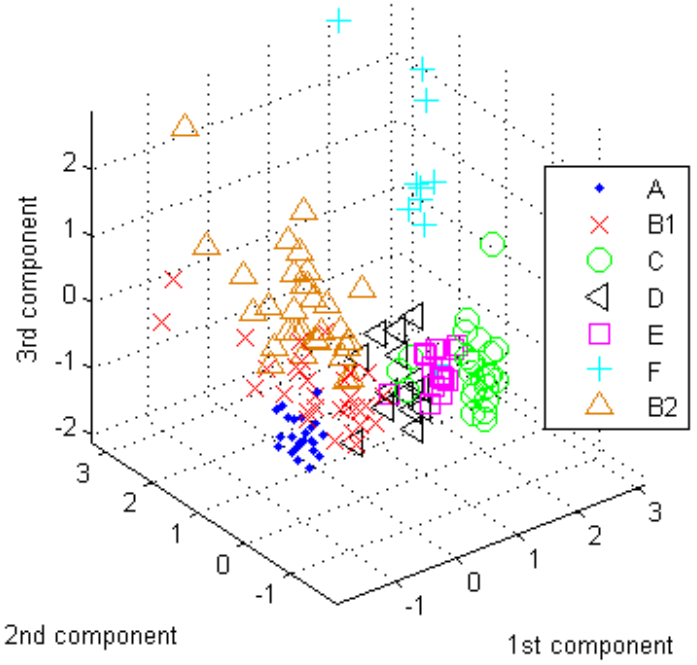
FIGURE 3.3: Contribution of variables for PCA

data points in the feature space, a clearly clustered distribution of whistle types will lead to a better classification result. Several observations are

- Type F occupies a clear region at the right top of the 2-D plot.
- Type B1 and B2 are mostly mixed.
- Type A, C and E are partially clustered since all of them have some regions overlapping with other groups.



(a) First two principal components



(b) First three principal components

FIGURE 3.4: Group scatter plot of principal components

3.3 Pairwise Similarity

A feature space is constructed by the feature vector selected. For N-point feature, the feature space is of N dimensions. Similarly a 3-D feature space is constructed by the three PCs. In the feature space, the distance between whistles describes how far apart the two whistles are. Let the feature vectors of two whistles be $\mathbf{x}_m$ and $\mathbf{x}_n$, the distance between them $d(\mathbf{x}_m, \mathbf{x}_n)$ is usually expressed as

$$d(\mathbf{x}_m, \mathbf{x}_n) = (\sum_{i=1}^{N} |x_{m,i} - x_{n,i}|^p)^{1/p} \tag{3.2}$$

This is called the *p-norm* distance. The commonly used Euclidean distance is *2-norm* where $p = 2$. When $p = 1$, the distance is the sum of absolute differences between features.

Another example of pairwise distance is the cosine distance

$$d(\mathbf{x}_m, \mathbf{x}_n) = 1 - \cos(\angle(\mathbf{x}_m, \mathbf{x}_n)) \tag{3.3}$$

where $\angle(\mathbf{x}_m, \mathbf{x}_n)$ is the angle between these two vectors $\mathbf{x}_m$ and $\mathbf{x}_n$.

In general case, the pairwise distance indicates the dissimilarity between two whistles. It should always be positive and symmetric as $d(\mathbf{x}_m, \mathbf{x}_n) = d(\mathbf{x}_n, \mathbf{x}_m)$. The more similar two whistles are, the smaller their distance is; hence *pairwise similarity* is an equivalent term as pairwise distance. It should be positive between two different whistle points, and is zero precisely when $\mathbf{x}_m = \mathbf{x}_n$.

A *dissimilarity matrix* is used to record the pairwise distances (or similarities) among all dolphin whistles; its entry $[i, j]$ is the distance between the ith and jth whistles. Figure 3.5 shows the color-coded pairwise distances in the dissimilarity matrix plot by the three PCs. The matrix is symmetric since $d(i, j) = d(j, i)$ by Euclidean distance and has a zero-valued diagonal line since $d(i, i) = 0$. Each whistle type is marked by the whistle number of the last whistle; hence Type A is from Whistle 1 to 24, Type B is from Whistle 25 to 55, and so forth.

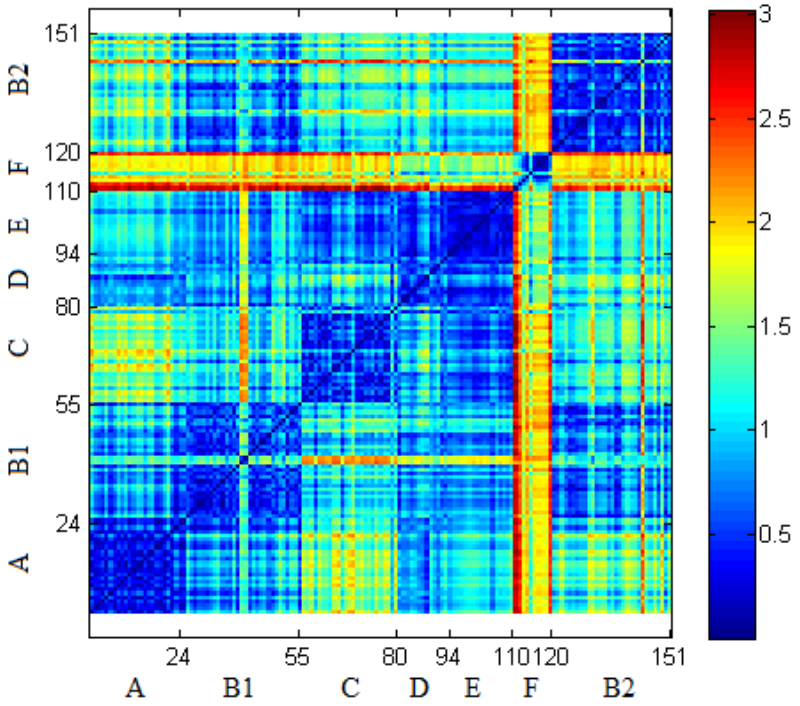


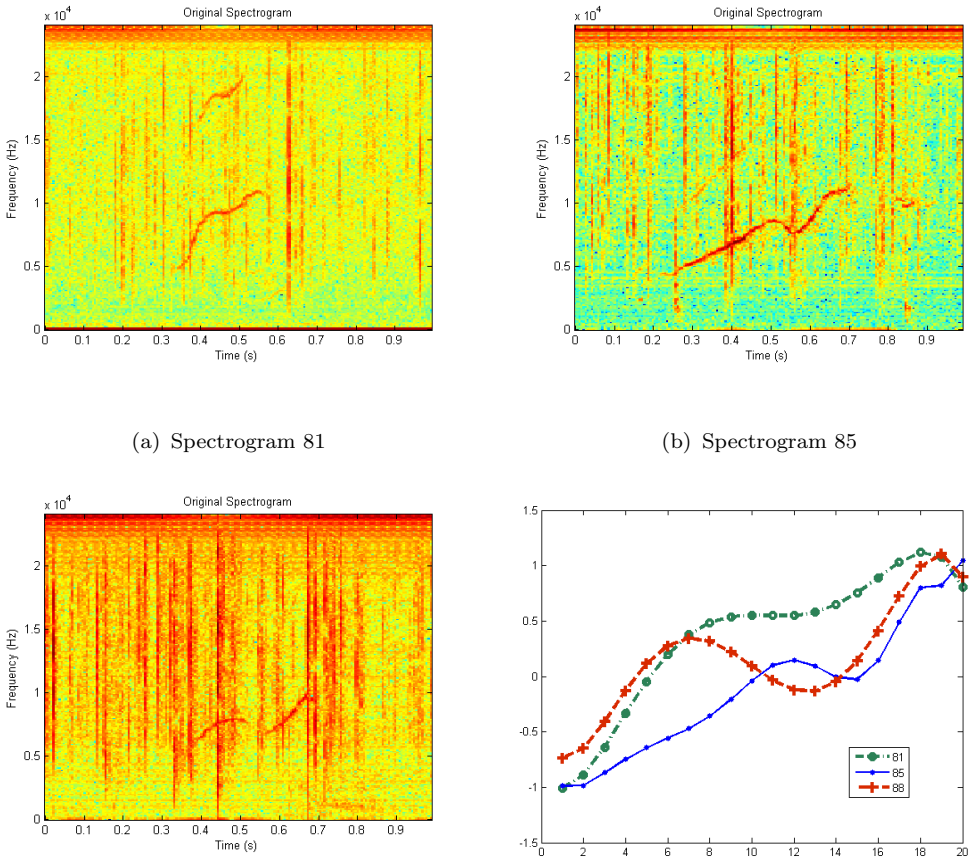
FIGURE 3.5: Dissimilarity plot for N-point feature after PCA

Along the diagonal line in Figure 3.5, it can be seen that whistles of the same type have small pairwise distances (blue patches). For example, a blue patch appears from $[1, 1]$ to around $[24, 24]$, although it has some overlap with the second blue ending at around $[55, 55]$. However, only whistles in Type F have much larger distances with whistles from other types; whistles from other types do not always have significant larger distance for whistles of different types. This indicates some whistles might be misclassified between B1 and B2, C and D.

3.4 Shape Contexts

Taking the three whistle in Figure 3.6 for example, they are different in intensity, duration and frequency modulation. They appear different when compared with

the 20-point sample (Figure 3.6(d)). However, when regarded as shapes, they would appear similar to the human observer.



(a) Spectrogram 81

(b) Spectrogram 85

(c) Spectrogram 88

(d) 20-point feature vector plot

FIGURE 3.6: Various whistle contours of the same type

Shape context is a novel descriptor for image recognition [2]. Shape matching by shape context is invariant to rotation, transformation and scale changing. Shape context considers the relative position among sample points and takes the relative distribution as the feature. With shape context, the sampled points are not presented by their frequency values but form a coarse log-polar distribution as the rest of the shape with respect to other points [2]. This descriptor expresses the configuration of the entire shape relative to each sample point as a reference.

For each sample point, 5 bins for $\log r$ and 12 bins for θ are used, where r is the length of the log-polar diagram and θ is the angle width. This diagram for capturing surrounding pixel density is demonstrated in Figure 3.7. The maximum r is twice of the mean distance between sample points; the minimum r is selected to be 80% of the mean distance. The histogram counts the number of other points falling into the bins formed by $\log r$ and θ. In this experiment, the bin size is thus $12 \times 5 = 60$. The whistle features consist of the log-polar histograms of all sample points.

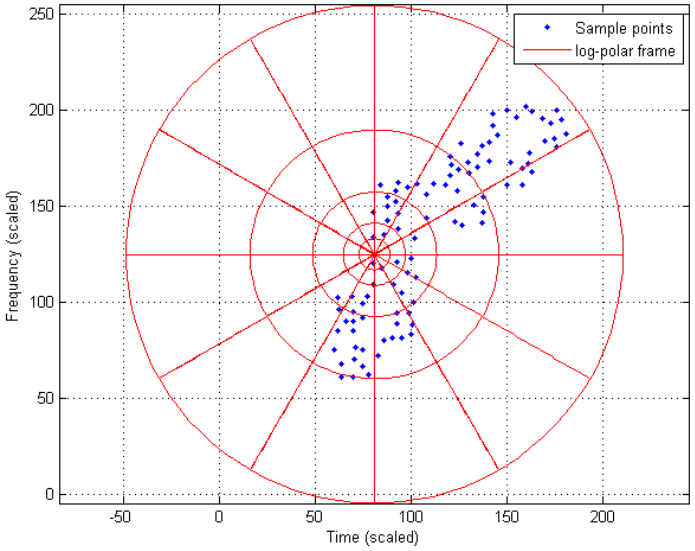


FIGURE 3.7: Diagram of log-polar histogram centering at a sample point of whistle traces

To measure the dissimilarity between whistles, a shape context distance d_{SC} is defined as a sum of shape context costs over best matching pairs. These costs are found from a shape context cost matrix C_{SC}. C_{SC} is a weighted sum of the cost matrices of shape difference C_{shape} and shape gradient difference C_θ:

$$C_{SC} = (1 - \omega_\theta)C_{shape} + \omega_\theta C_\theta. \tag{3.4}$$

Each entry $C_{shape}(i, j)$ is the histogram difference of the ith and jth sampling points from the two whistles. It is the obtained by χ^2 test statistics [12]. Matrix C_θ has a similar structure; each entry records the difference of the orientation measured at the two sampling points. When points are sampled at the shape edge by the Canny edge detector [11], the orientation is the derivative of the edge curve. Hence the entries of matrix C_{SC} record a combination of pairwise shape difference and gradient difference. Given C_{SC} between two whistles Q and T, the best matching finds the correspondences $H(Q, T)$ between points with the minimum total cost of matching subject to one-to-one mapping

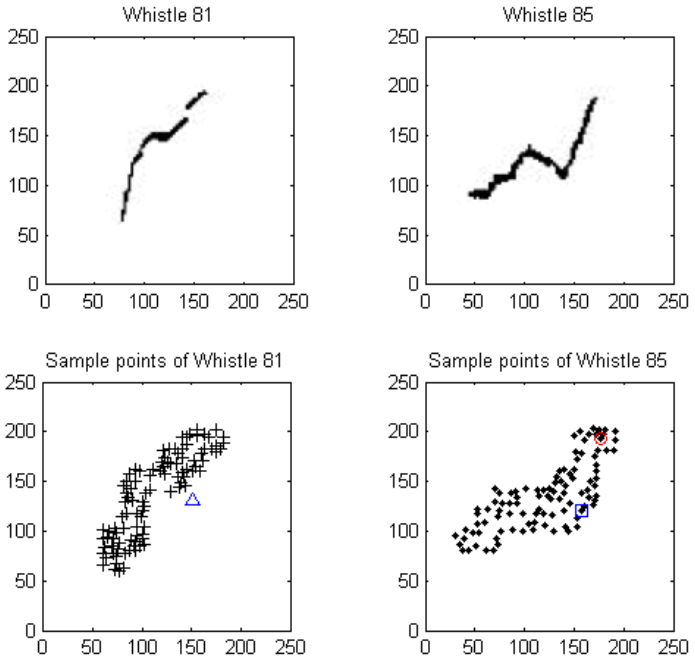
$$H(Q, T) = \min(\sum_i C_{SC}(i, w(i))) \tag{3.5}$$

where i denotes a point in Q and $w(i)$ denotes the warped matching point in T. This minimum total cost is d_{SC}. This is called 'weighted bipartite matching' by the Hungarian method [40]. A more efficient algorithm [24] can also be used to assign the matching pairs.

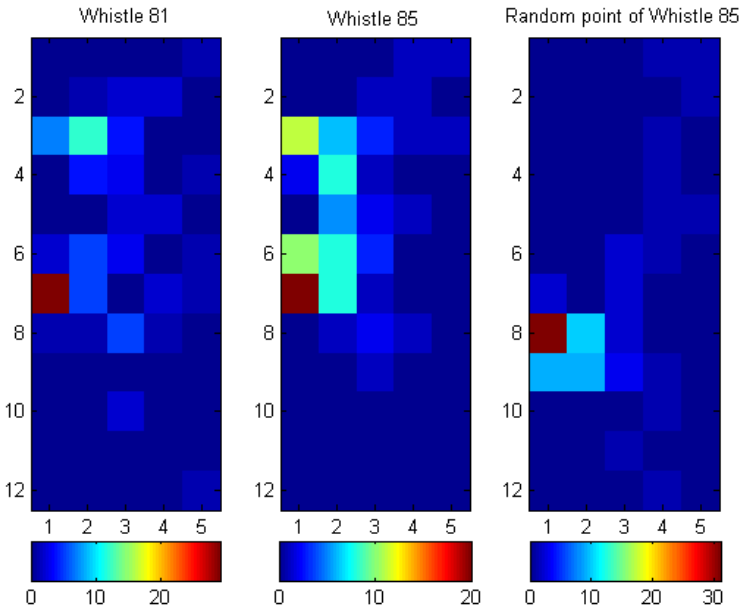
In [2], there are two more types of costs to be considered: *image appearance distance* d_{IA}, and *bending energy* E_{bend}. The image appearance distance d_{IA} is the sum of squared brightness differences after normalization. The bending energy E_{bend} is estimated from the thin plate spline model, which models the changes in biological forms.

The details of the shape contexts and code are available in [2, 3]. Previously shape context was used to assess the similarity between contoured shapes such as

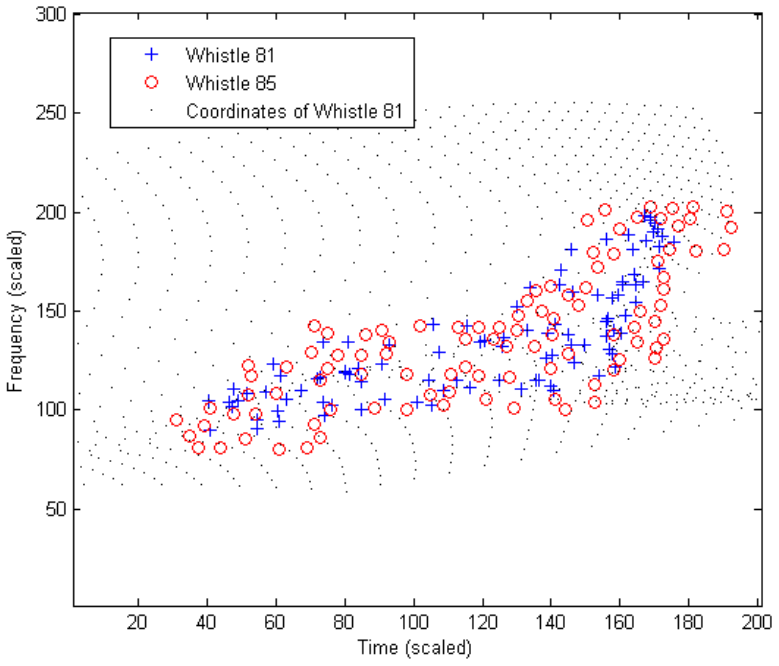
handwriting digits. It is modified for our dolphin whistle application. For a whistle (Whistle 81 for instance), one whistle from the same type and two whistles from different types are randomly chosen for testing. Figure 3.8(a) shows the segmented spectrograms and 100 sample points along the contour of two whistles. It is called '2-D shape context'. In Figure 3.8(b), the first two log-polar histograms are for the points in similar positions of the two whistles (□ and △ in Figure 3.8(a)); they are similar to each other. The third histogram in Figure 3.8(b) is for a randomly picked point (∘ in Figure 3.8(a)) and appears different from the first two. Figure 3.8(c) shows the warped matching between whistles 81 and 85. While the coordinates are for points of whistle 85, the dotted lines are the warped coordinates for points of whistle 81.



(a) Segmented whistle spectrograms (first row) and their 100 sampling points along the edges (second row). Axes are scaled to ratio. A pair of corresponding points is shown in △ and □; one random point is ◯.



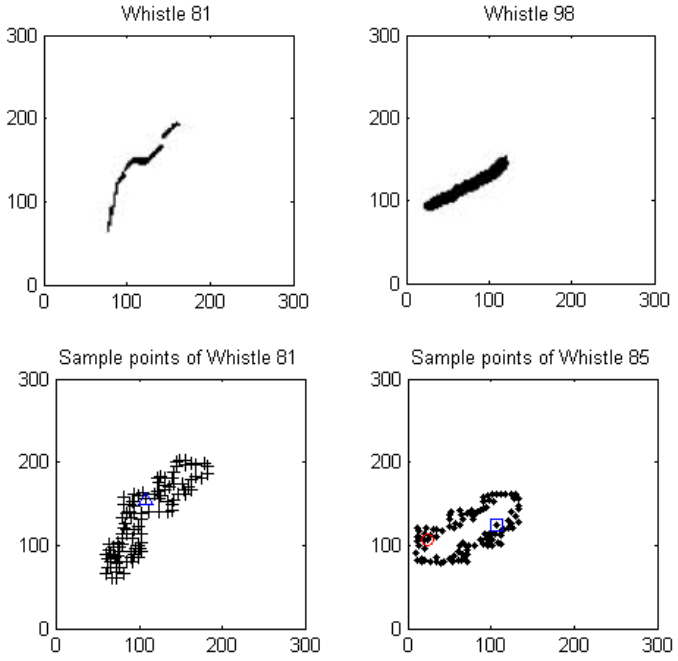
(b) Log-polar histograms for the sample points with twelve bins for θ (y-axis) and five bins for $\log r$ (x-axis): the histogram is for points $\triangle$, $\square$ and $\bigcirc$ from left to right.



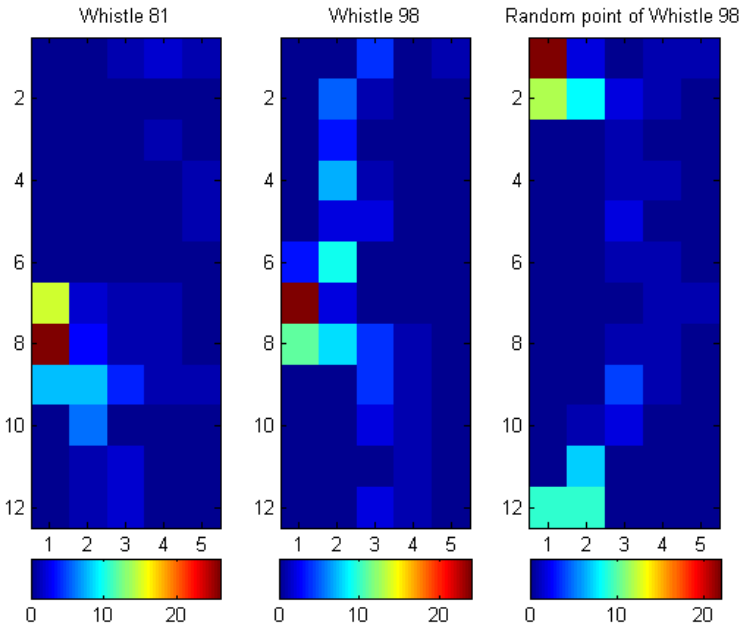
(c) Warped matching by bipartite graph matching [24]: x/y-axes are coordinates (scaled to ratio) of Whistle 85 while the black dots are warped coordinates for Whistle 81

FIGURE 3.8: 2-D shape context computation and matching for the same type: Whistle 81 vs. 85

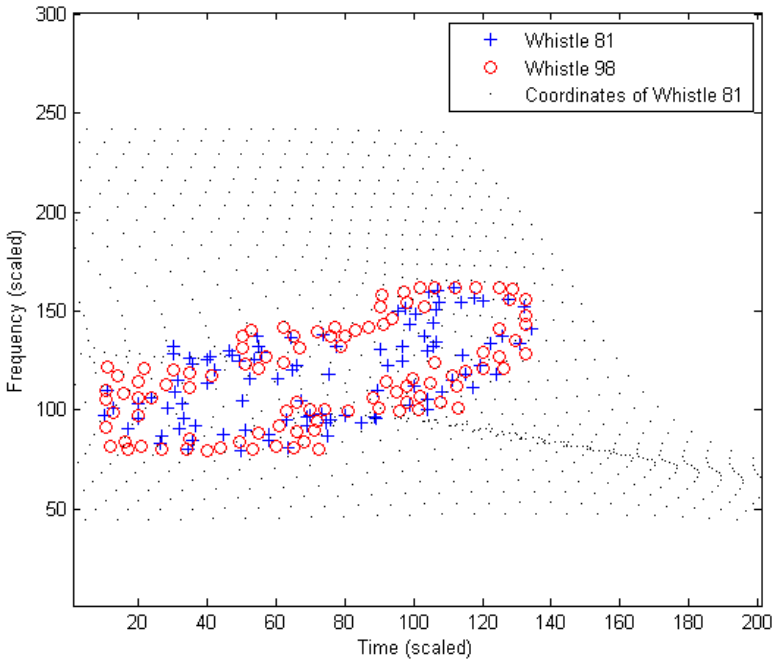
In Figure 3.9 and Figure 3.10, Whistle 81 is compared with whistles from different types using shape context. The various costs of shape context is presented in Table 3.1. The shape matching firstly finds the best set of correspondences from C_{SC}, which gives d_{SC}. The values of d_{shape} and d_θ in Table 3.1 are the costs from the best matching and averaged by the length of the longer sequence in the pair. The image appearance difference and warping cost are then computed.



(a) Segmented whistle spectrograms (first row) and their 100 sampling points (second row) along the edges. Axes are scaled to ratio. A pair of corresponding points in △ and □; one random point ◯.

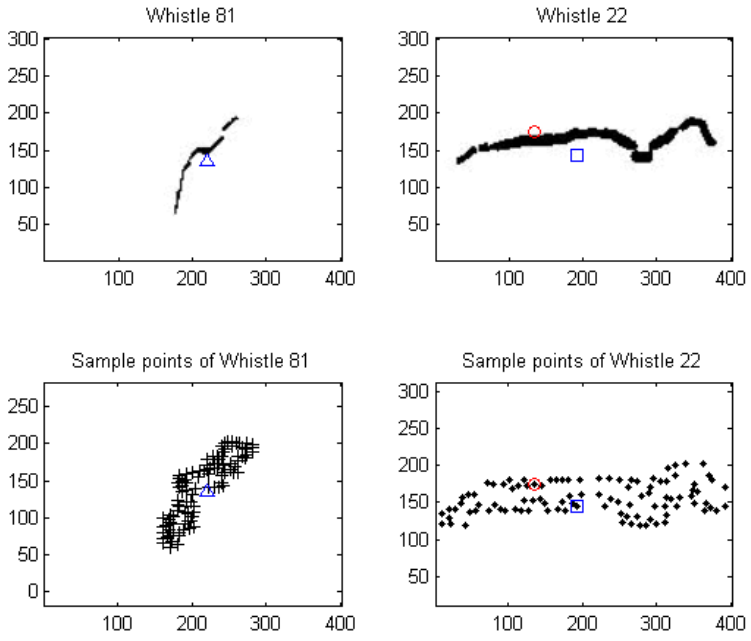


(b) Log-polar histograms for the sample points with twelve bins for θ (y-axis) and five bins for $\log r$ (x-axis): the histogram is for points $\triangle$, $\square$ and $\bigcirc$ from left to right.

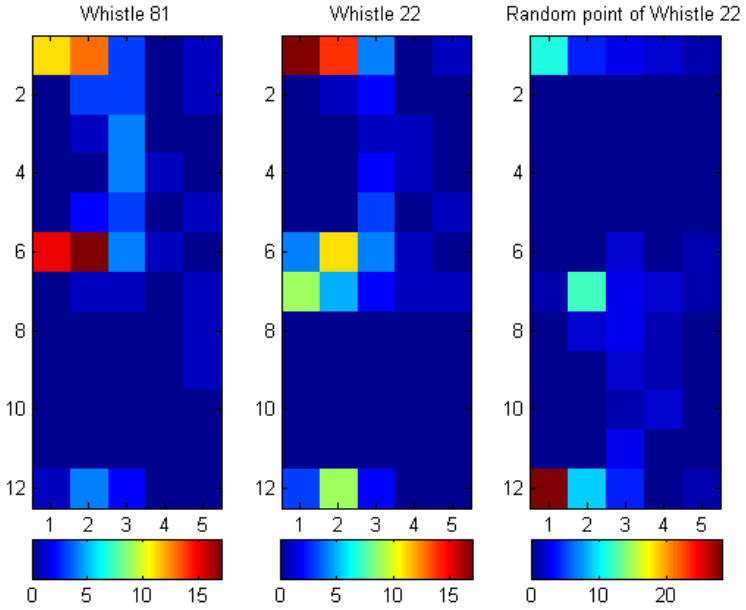


(c) Warped matching by bipartite graph matching [24]: x/y-axes are coordinates (scaled to ratio) of Whistle 98 while the black dots are warped coordinates for Whistle 81

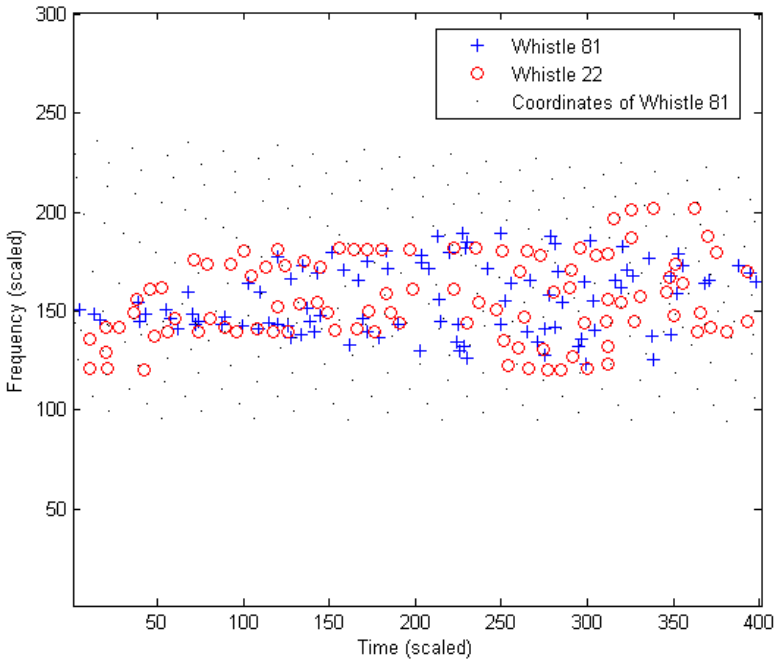
FIGURE 3.9: 2-D shape contexts computation and matching for different types: Whistle 81 vs. 98



(a) Segmented whistle spectrograms (first row) and their 100 sampling points (second row) along the edges. Axes are scaled to ratio. A pair of corresponding points in △ and □; one random point ◯.



(b) Log-polar histograms for the sample points



(c) Warped matching by bipartite graph matching [24]: x/y-axes are coordinates (scaled to ratio) of Whistle 22 while the black dots are warped coordinates for Whistle 81

FIGURE 3.10: 2-D shape contexts computation and matching for different types: Whistle 81 vs. 22.

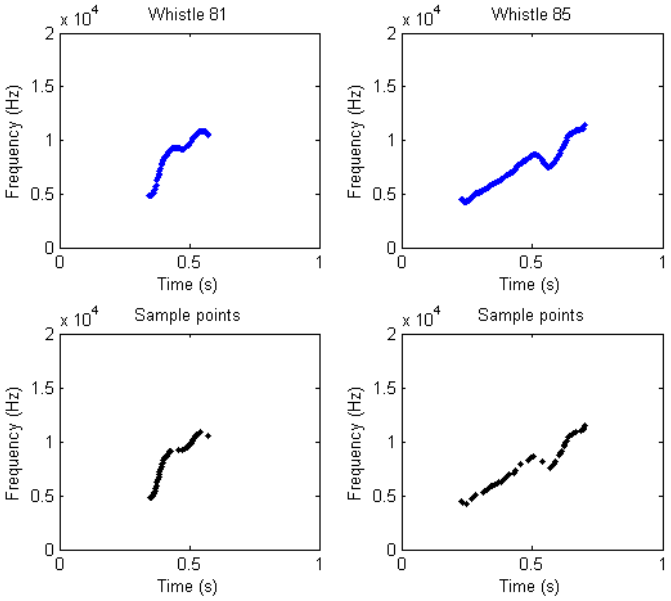
TABLE 3.1: Shape context costs on 2-D matching of an example whistle (Whistle 81) with other whistles

	$\mathbf{C_{SC}}$			$\mathbf{d_{IA}}$	$\mathbf{E_{bend}}$	$\mathbf{d_{SC} + d_{IA} + E_{bend}}$
	$\mathbf{d_{Shape}}$	$\mathbf{d_\theta}$	$\mathbf{d_{SC}}$			
Whistle 85	0.1274	0.0007	0.1170	3.2992	1.3864	4.8026
Whistle 98	0.1052	0.0014	0.0993	1.8823	0.5915	2.5731
Whistle 22	0.1241	0.025	0.1192	5.6345	1.2249	6.9785

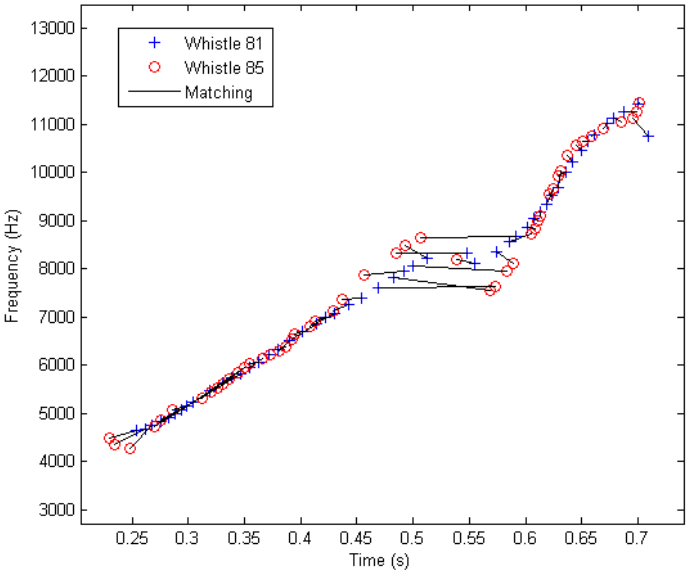
From Figure 3.9 and Figure 3.10, it is seen that whistles are over-warped in both cases. The bending energy of Whistle 85 is much larger than Whistle 91 and 22, which are much flatter and easier to bend. The orientation weight

w_θ for C_θ in Equation 3.4 is set to 0.5. The last column in Table 3.1 shows an example of identical weights. It shows that Whistle 81 has a much smaller distance with Whistle 98 than Whistle 85. The image appearance distance d_{IA} here again evaluates Whistle 81 to be more similar to Whistle 98. The brightness in the spectrogram indicates the whistle energy, whose effect on deciding whistle types is unknown up to this thesis. One possible approach is to study the training set of whistles and find the best combination of these costs for test set classification.

Different from applications in [2, 3], the TFR as a 1-pixel whistle tracing in Section 3.1 is also tried out for shape contexts. In contrast to the whistle contour as '2-D shape context', the TFR for shape context is called '1-D shape context'. It is much simpler than 2-D shape context since the image appearance and edge gradient do not apply in 1-D shape context. The three sets of comparison plot of Whistle 81 with other whistles are shown again in Figures 3.11, 3.12 and 3.13. In each set, the original TFR and sample points for two whistles (left and right columns) are shown first. Whistle 81 is warped to match other whistles for minimum matching cost by the bipartite graph matching shown in the second figure. The log-polar histograms for the sample points may be sparse and only have non-zero surrounding pixel density at two angular bins for almost constant frequency changing. We can see that the bending energy of Whistle 98 is still much less than Whistle 85 since Whistle 98 is straight and it takes less energy to warp Whistle 81 to a straight line. The shape distance in the last column in Table 3.2 is the sum of the shape context distance and bending energy.

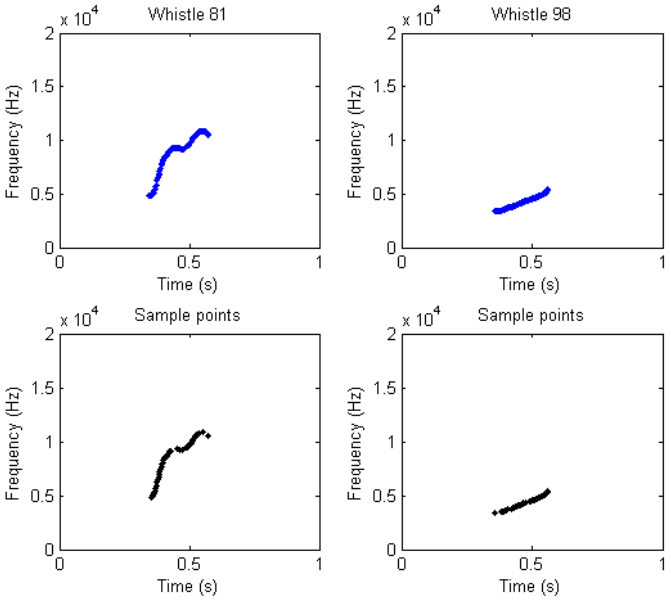


(a) One-pixel whistle traces (first row) and their 50 sampling points (second row).

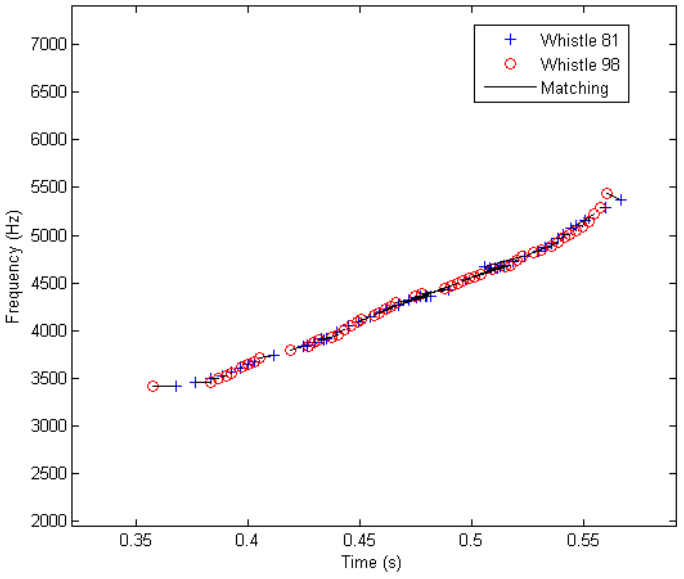


(b) Warped matching by bipartite graph matching [24]: time of Whistle 81 is warped to match Whistle 85.

FIGURE 3.11: 1-D shape contexts computation and matching for the same types: Whistle 81 vs. 85

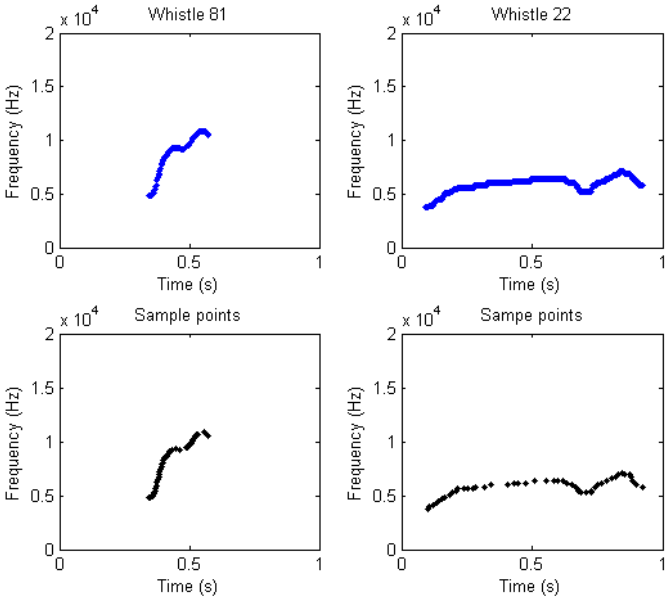


(a) One-pixel whistle traces (first row) and their 50 sampling points (second row).

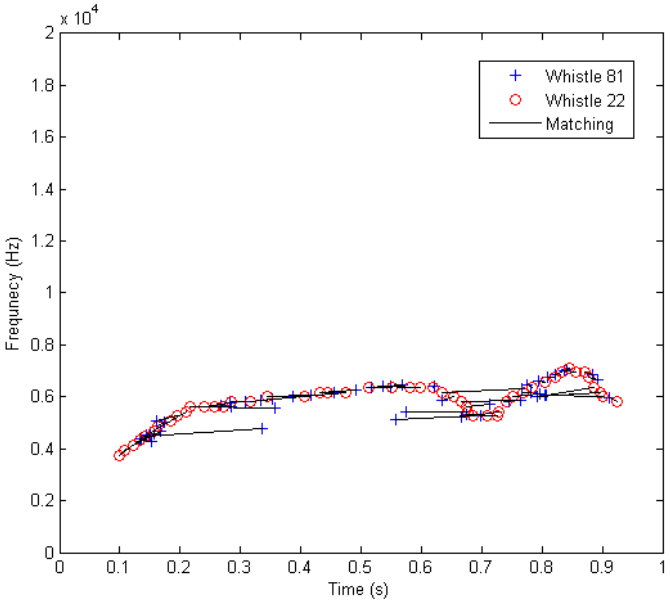


(b) Warped matching by bipartite graph matching [24]: time of Whistle 81 is warped to match Whistle 98. It takes less energy to warp Whistle 81 to a relatively straight whistle curve (Whistle 98)

FIGURE 3.12: 1-D shape contexts computation and matching for different types: Whistle 81 vs. 98



(a) One-pixel whistle traces (first row) and their 50 sampling points (second row).



(b) Warped matching by bipartite graph matching [24]: time of Whistle 81 is warped to match Whistle 22.

FIGURE 3.13: 1-D shape contexts computation and matching for different types: Whistle 81 vs. 22

TABLE 3.2: Shape context costs on 1-D matching of an example whistle (Whistle 81) with other whistles

	$\mathbf{d_{SC}}$	$\mathbf{E_{bend}}$	$\mathbf{d_{SC} + E_{bend}}$
Whistle 85	0.029	0.078	0.107
Whistle 98	0.017	0.040	0.057
Whistle 22	0.141	0.241	0.382

In summary, some disadvantages of this method for whistle matching are listed. Firstly, shape context for a sample point can be rich and unique among all others when the image contour of interest is complicated. Examples in [2] are handwritten digits and alphabets, giving more contour lines for sampling points. Whistle in this project are too simple with only one tracing line or simple contour. Secondly, the orientation of whistle curves is fully ignored in this method. Points are matched according to the distribution of surrounding pixels. In 1-D curve matching, the angular variation is too sparse. Meanwhile, the matching correspondence is one-to-one but not in order of time or frequency, whereas the sequence and changing of frequencies are important in defining whistle types. The matching of 1-D tracing points between whistles 81 and 85, 81 and 22 is undesirable.

The shape context describes whistles with the distribution of the surrounding pixels, yet introduces much over-warping. DTW could be more suitable for nonlinear mapping on a data sequence in the time domain, and hence applicable to a whistle spectrogram curve. The idea of DTW will be explored in Chapter 5 and Chapter 6. Although some information may be lost after scaling and shifting, a sequence of time-frequency points is still the most direct and basic description

of a whistle curve in the spectrogram. It is easy to construct the feature space from the sequence of these frequency points. In the next chapter (Chapter 4 about classification methods), the sample points on TFR and their principal components are used as the feature vector.

Chapter 4

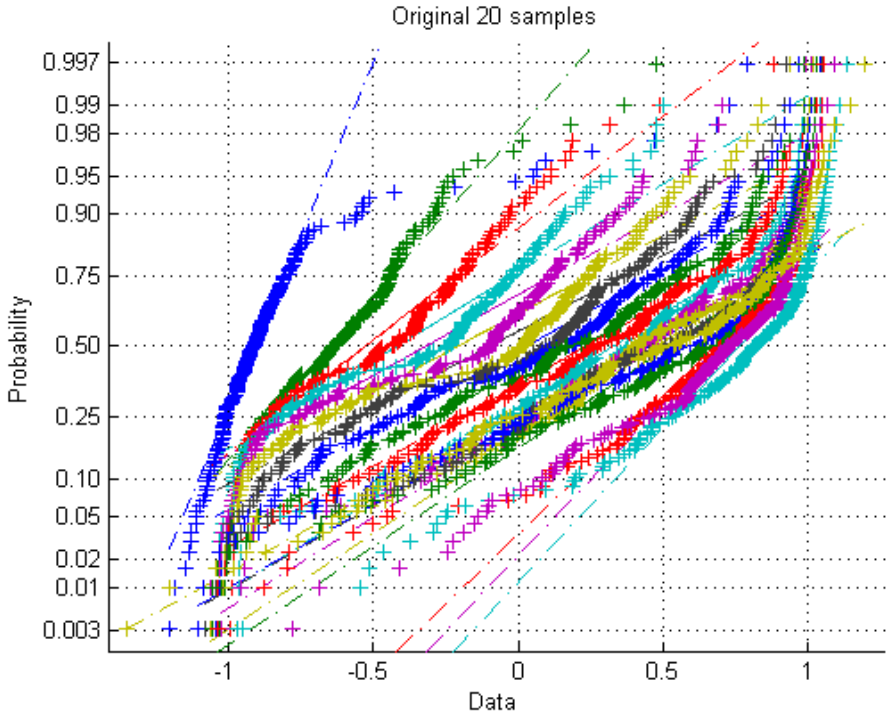
Classification Methods

A classification method is used to classify whistles using the features and similarity measurement selected. Classification methods are generally divided into two types: *supervised learning* and *unsupervised learning*. Supervised learning is a machine learning technique for deducing a classification from the training data, which comes together with the labeled classes. On the other hand, unsupervised learning seeks to determine how the data can be organized without any labels. It is also known as *clustering*, and involves grouping data into classes based on the measure of the inherent similarity. Some typical classification methods are simply experimented on the traditional feature vector - TFRs. Sections 4.2, 4.3 and 4.4 give two examples of supervised learning while Sections 4.5 and 4.6 give examples of unsupervised learning.

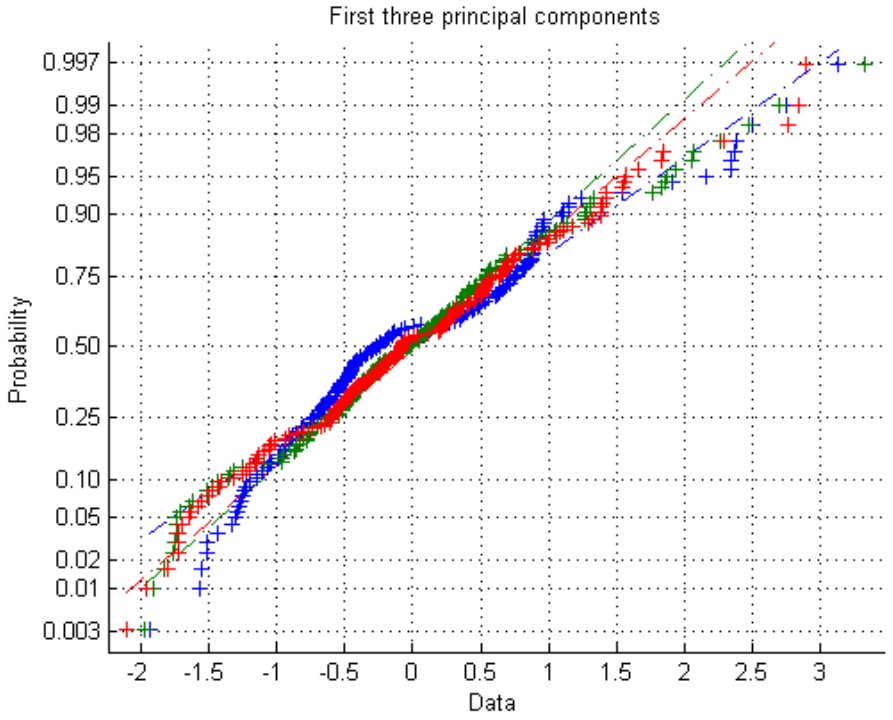
4.1 Data Normality Test

Without knowing the characteristics of features, most classification methods assume data is Gaussian distributed. A normality test is firstly implemented to test the validity of this assumption.

Figure 4.1 shows the normality plots of the feature data from all the whistles in Appendix A. The feature data comprises the original 20-point feature and their first 3 principal components (PCs). The normality plot assesses the normality of each variable (or feature) in the feature vector. It plots the normal inverse cumulative distribution probability (CDP) of the data when fitting the first and third quartiles of data versus theoretical quartiles of a normal distribution with a line in red. The closer these data are to the line, the more likely it is that the data distribution is normal. Figure 4.2(a) shows that the normality plots for most of the 20-point feature are not linear. However, their first 3 PCs are fairly close to the linear fits for a normal distribution in Figure 4.2(c). The PCs come from the dimensions where the set of data has the largest variance. This test shows that the data distribution in the first 3 PCs are approximately Gaussian distributed and can be used for classification methods with the Gaussian assumption.



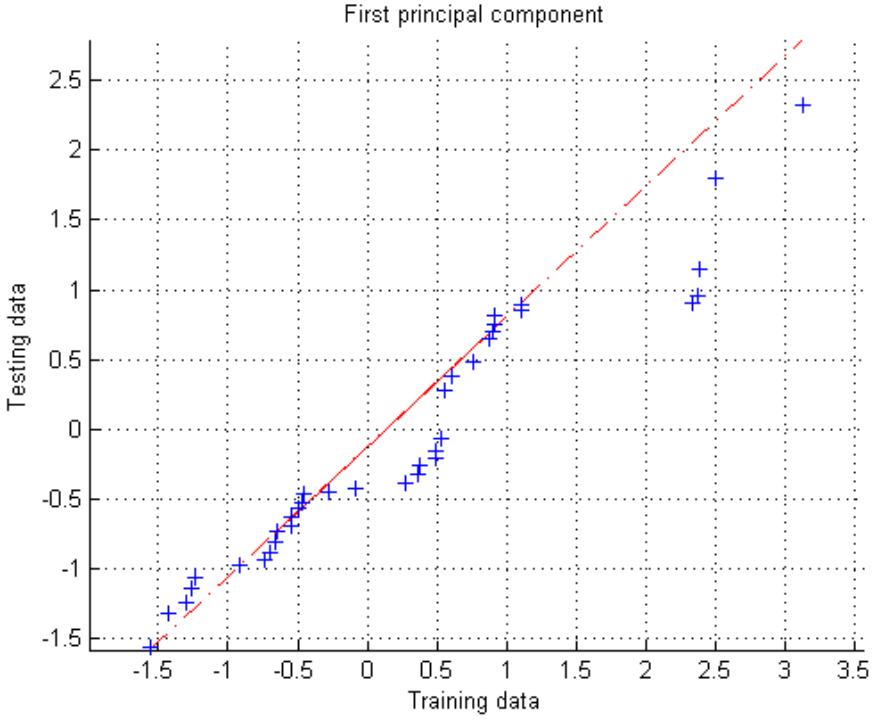
(a) Normality plot of original 20-point feature



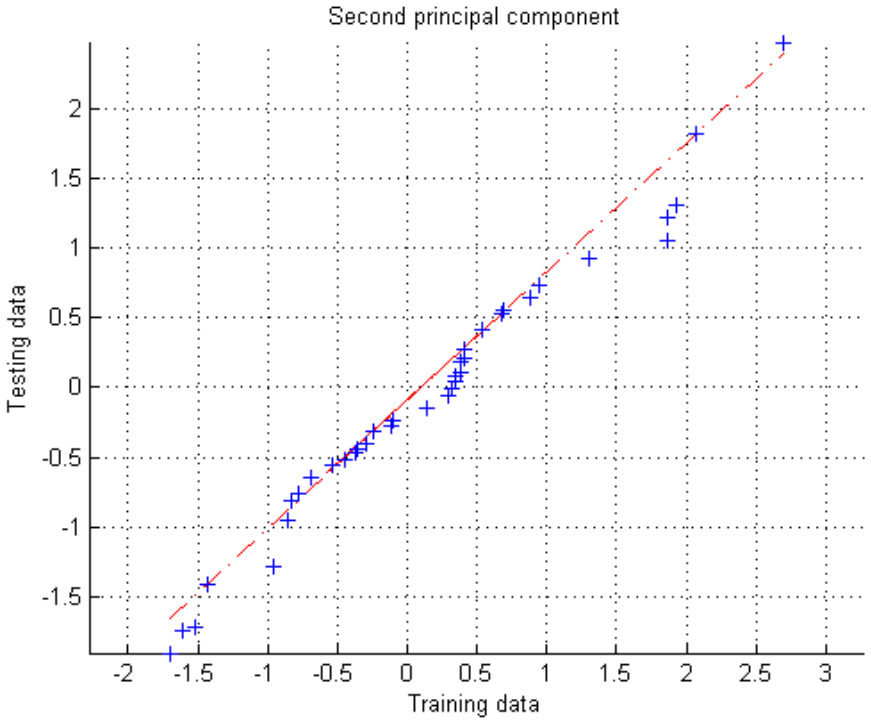
(b) Normality plot of the first 3 PCs

FIGURE 4.1: Normality test of feature data before and after PCA

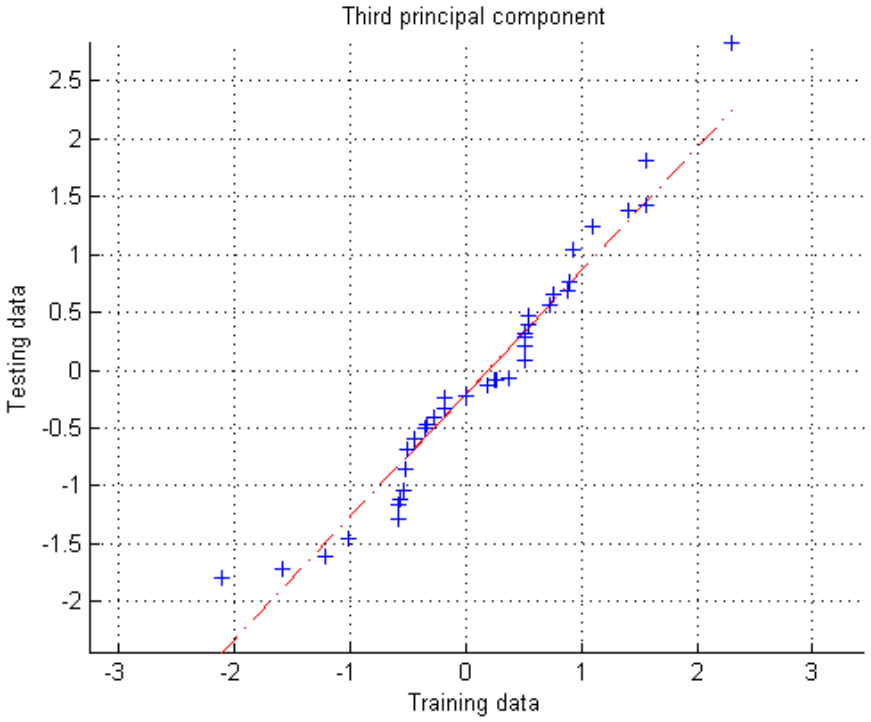
Supervised classification classifies test data according to the characteristics or distribution of the training data (with class labeled); it assumes the test and training data have the same distribution. The supervised classification used in this Chapter selected about 20% of the whistles in Appendix A as the training set and took the remaining as the test set. A normality test is needed to check whether the training data and test data are balanced. Figure 4.2 shows the quantile-quantile plots (Q-Q plot) of the first 3 PCs between training and test set. If the data in blue + is close to the linear fit in red line, the data from the two sets comes from a similar distribution.



(a) Q-Q plot of first principal component



(b) Q-Q plot of second principal component



(c) Q-Q plot of third principal component

FIGURE 4.2: Q-Q plot of the first three principal components

To compare the effect of feature reduction by PCA, the classification results on 8 principal components and the full 20-point feature are also shown (Appendix B) followed by a discussion.

4.2 Linear/Quadratic Discriminant Analysis

Linear discriminant analysis (LDA) and the related Fisher's linear discriminant (FLD) method use the training data to find a linear combination of features to characterize and separate different types. In the case of $c = 2$ classes with N features, a linear discriminant classifier [54] is defined as

$$g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0 \tag{4.1}$$

where $\mathbf{w} = [w_1, w_2, ..., w_N]^T$ is known as the weight vector and w_0 as the threshold, and $\mathbf{x}$ is the N-dimensional feature vector. By assigning $g(\mathbf{x}) = 0$, we obtain a hyperplane which separates these two classes. Training samples are used to find this hyperplane using various methods such as the perceptron algorithm and mean square error estimation [47].

When $c > 2$, we have c linear discriminant functions of the form

$$g_i(\mathbf{x}) = \mathbf{w}_i^T\mathbf{x} + w_{i0}, i = 1, 2, ..., c. \tag{4.2}$$

We assign sample $\mathbf{x}$ to class i if $g_i(\tilde{X}) > g_j(\tilde{X}), \forall j \neq i$. In this case, this linear classifier divides the feature space into exactly c decision regions. For each input

feature vector $\mathbf{x}$, the corresponding desired output response, that is, the class labels $\mathbf{y} = [y_1, ..., y_c]$ are chosen so that $y_i = 1$ and $y_j = 0$ if $\mathbf{x}$ belongs to class i rather than any other class j. The matrix $\mathbf{W}$ has as columns the weight vectors $\mathbf{w}_i$ and hence is of size $N \times c$. The mean squared error (MSE) criterion is to minimize the norm of the error vector $(\mathbf{y} - \mathbf{W}^T\mathbf{x})$, that is,

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W}} \mathbb{E}[\|\mathbf{y} - \mathbf{W}^T\mathbf{x}\|^2] = \arg\min_{\mathbf{W}} \sum_{i=1}^{M} (y_i - \mathbf{w}_i^T\mathbf{x})^2. \tag{4.3}$$

where $\mathbb{E}[\cdot]$ denotes the expected value. This is equivalent to c independent minimization problems. LDA fits a multivariate normal density to each group with the training data set, assuming all groups have identical covariance. LDA requires enough information to be able to estimate a full-rank covariance matrix. More observations (size of training data set) than number of features (N) in training data are required. Hence the dimension of the features is firstly reduced by PCA. The LDA is closely related to PCA in that both look for linear combinations of variables which best explain the data [34]. The first 3 PCs extracted in Section 3.2 are used for classification. In the discriminant analysis of classification, two types of errors are defined:

1. *Classification error*: the ratio of misclassified samples over all test set, and

2. *Re-substitution error*: the ratio of misclassified samples over all training set if the classification is re-applied on training set using the parameters extracted over the training set with their class labels.

With 20% of whistles in Appendix A as training data, LDA gives a classification error of 24.56% (28 misclassified out of 114 test whistles) and a re-substitution error of 21.62% (8 misclassified out of 37 training samples). A *confusion matrix* displays the predicted (classified) class labels of the data against the known class labels. In the confusion matrix of the test data in Table 4.1, each entry counts the number of whistles with a predicted class label in the column and at the same time the pre-classified or known class label in row. Hence the diagonal entries give the number of correctly classified whistles for each type. Similarly the confusion matrix of training data in Table 4.2 shows the classification result from re-classifying the training data.

TABLE 4.1: LDA: confusion matrix of test data from classification

		Predicted Class Label						
		A	**B1**	**C**	**D**	**E**	**F**	**B2**
Known Class Label	**A**	19	0	0	0	0	0	0
	B1	5	16	0	2	0	0	2
	C	0	0	16	2	2	0	0
	D	1	0	0	6	2	0	0
	E	0	0	0	1	10	0	0
	F	0	0	0	0	1	4	0
	B2	0	10	0	0	0	0	15

If B1 and B2 are considered to be the same type, LDA gives 13.51% of re-substitution error and 15.79% of classification error for the 6 types in all. These error rates are lower than the ones for 7 groups.

LDA separates the space into regions divided by lines and assigns different regions to different types. Figure 4.3(a) shows the regions divided by LDA in the

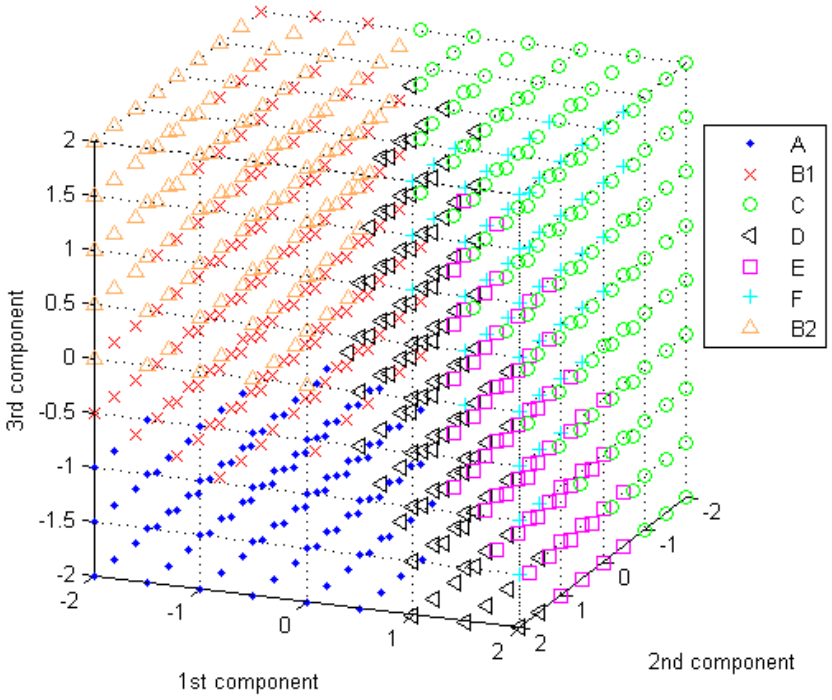
TABLE 4.2: LDA: confusion matrix of training data from re-distribution

		Predicted Class Label						
		A	**B1**	**C**	**D**	**E**	**F**	**B2**
Known Class Label	**A**	4	1	0	0	0	0	0
	B1	1	4	0	0	0	0	1
	C	0	0	4	1	0	0	0
	D	0	0	0	4	1	0	0
	E	0	0	1	0	4	0	0
	F	0	0	0	0	0	5	0
	B2	0	2	0	0	0	0	4

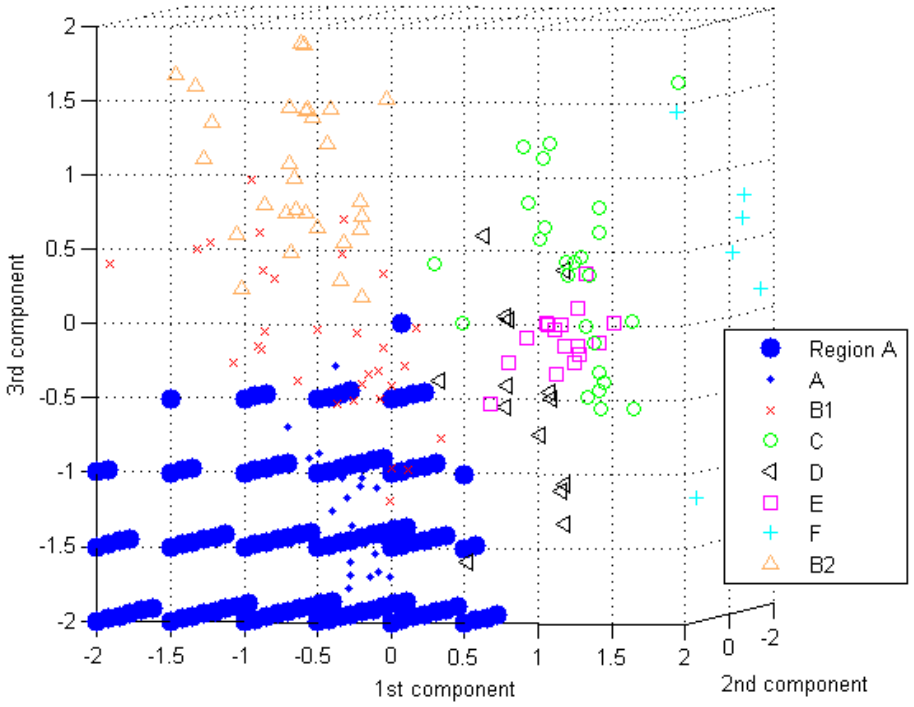
feature space spanned by the first 3 PCs of the 20-point sampling. These regions are separated by planes in the 3-dimensional space. Taking the region for Type A as an example, Figure 4.3(b) shows that most data points from Type A fall in the Region A found by LDA. However, some data points from Type B1 also fall into this area. This explains the classification errors of Type B in the confusion matrix shown in Tables 4.1 and 4.2. This happens for other types.

LDA assumes an identical covariance for all classes. This is not easy to verify with a small number of whistles in the case in this thesis.

There are various types of discriminant functions [44]. Their results are compared with LDA in Table 4.3. In a quadratic discriminant analysis (QDA), normal distribution is assumed for features with different covariance. Diagonal linear discriminant analysis (DLDA) is the diag-linear discriminant analysis. It is similar to LDA but with a diagonal covariance matrix estimate. This is also called *naive Bayes classifiers*. Similarly, DQDA is the quadratic discriminant analysis with a diagonal covariance matrix estimate. The Mahalanobis distance [31] is also used



(a) Classification regions by LDA



(b) Region for Type A and data points

FIGURE 4.3: Classification regions by LDA

for covariance estimates. We can see that QDA, DQDA and the Mahalanobis have quite an inconsistent performance for the training and test sets. The training set for quadratic discriminant analysis might not be representative.

TABLE 4.3: Comparison of various types of discriminant analysis: e_R is the re-substitution error; e_C is the classification error.

	7 **Types**		6 **Types**	
	e_R	e_C	e_R	e_C
LDA	21.65	24.56	13.51	15.79
DLDA	21.62	31.93	16.22	14.91
QDA	2.70	34	5.41	19.30
DQDA	13.51	27.19	8.11	16.67
Mahalanobis	13.51	35.09	13.51	25.44

There are other more advanced supervised classification methods. For example, support vector machines (SVM) constructs a hyperplane which has the largest distance to the nearest training data points of any class. However, multi-class SVM is needed in dolphin whistle classification. This multi-class requires reducing the single multi-class problem into multiple binary classification by normal SVM. Furthermore, parameter settings and kernel function selection also makes the supervised classification complicated. SVM has the potential to be used but is outside the scope of this thesis.

4.3 Bayesian Classification

The Bayes classifier is quite popular in many complex real-world situations in spite of the over-simplified assumptions. The simplified assumptions are: the

classifier has strong independence among features, and the probability density function (PDF) of each class is Gaussian. PCA is suitable for the first assumption. The distribution of PCA data is examined here to investigate the applicability of the Bayesian classifier. The histograms of the 3 PCs are plotted separately in Figure 4.4. The histograms of each type based on the first 2 PCs only (for easier visualization) are plotted in Figure 4.5.

In Figure 4.4, only Type F has an isolated feature distribution (the first PC). The second PC shows partial separation for Type C. When all types are plotted together in Figure 4.5(h), it is clear that Types D, E and F are well grouped and separated from other types. This implies possibility of good classification of these types. Notice that the adjacency of Types D and E implies some misclassification between them.

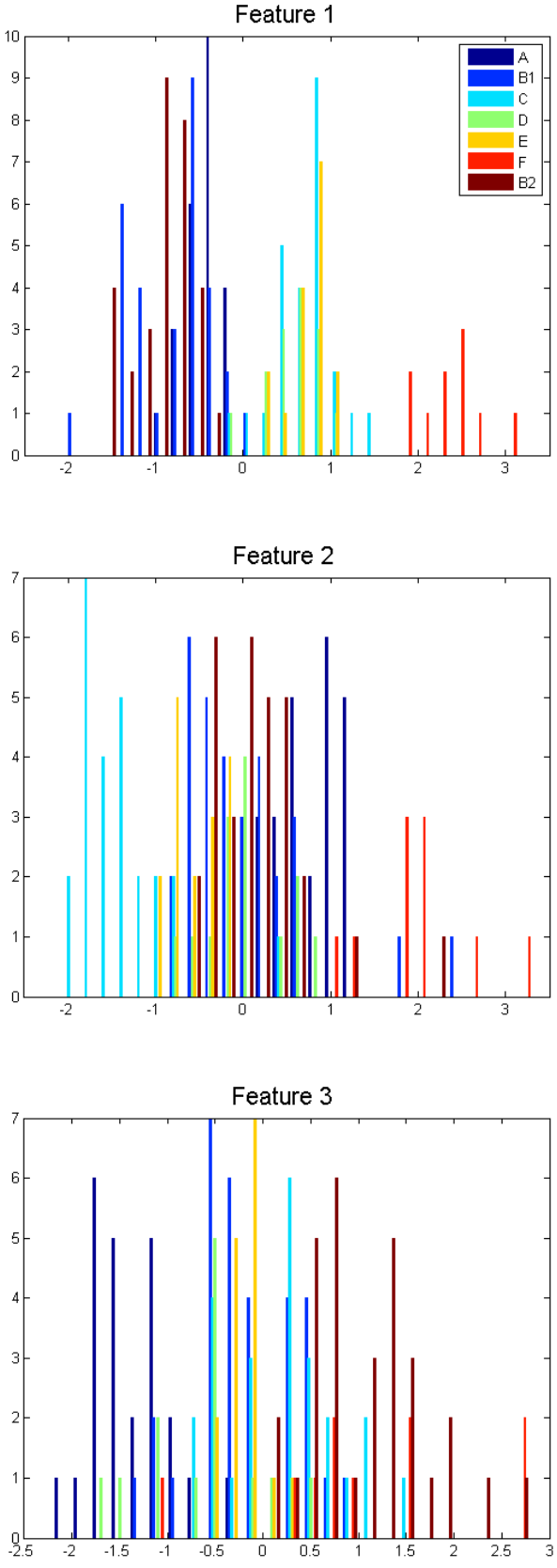


FIGURE 4.4: Histograms of whistle types for first three principal components from 20-point feature for the distribution of each feature

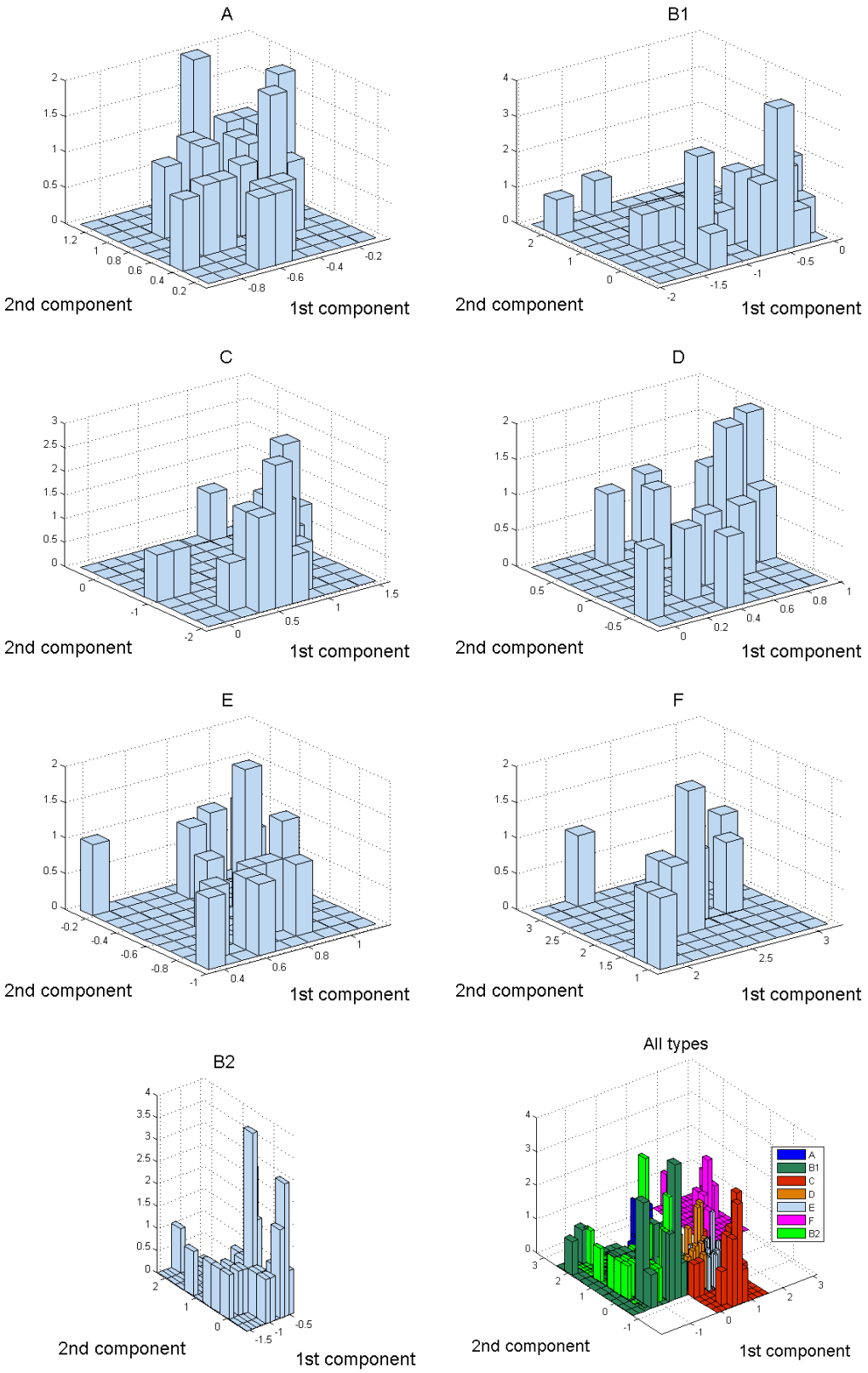


FIGURE 4.5: Histograms of first two principal components of 20-point feature for each whistle type for the distribution of whistles in each type

Section 4.1 has shown that the training and test sets are of similar distribution in the first 3 PCs (though the distribution similarity of the first PC is worse than the second and third). However, the histogram based on the first 2 PCs for all whistle types is still not Gaussian distributed and sufficiently separated for good classification. The third PC is yet not displayed. The Bayes classifier applies the PDF determined from the training whistles on the testing set, and computes the error rates in supervised classification. Table 4.4 shows the classification error of 21.93% and Table 4.5 shows the re-substitution error of 21.62%.

TABLE 4.4: Bayesian classifier: confusion matrix of test data from classification

		Predicted Class Label						
		A	**B1**	**C**	**D**	**E**	**F**	**B2**
Known Class Label	**A**	18	1	0	0	0	0	0
	B1	5	17	0	1	0	0	2
	C	0	0	16	2	2	0	0
	D	1	0	0	7	1	0	0
	E	0	0	0	1	10	0	0
	F	0	0	0	0	0	5	0
	B2	0	9	0	0	0	0	16

TABLE 4.5: Bayesian classifier: confusion matrix of training data from re-substitution

		Predicted Class Label						
		A	**B1**	**C**	**D**	**E**	**F**	**B2**
Known Class Label	**A**	4	1	0	0	0	0	0
	B1	1	4	0	0	0	0	1
	C	0	0	4	1	0	0	0
	D	0	0	0	4	1	0	0
	E	0	0	1	0	4	0	0
	F	0	0	0	0	0	5	0
	B2	0	2	0	0	0	0	4

4.4 K Nearest Neighbors (KNN) and Probabilistic Neural Network (PNN)

From the previous supervised classification, a set of training data with known categories are used to train the classification with an estimate of the probability of the class membership. K nearest neighbors (KNN) classifies samples based on the closest training examples in the feature space. It is amongst the simplest of all machine learning algorithms: a sample is classified by a majority vote of its neighbors (training samples). If $k = 1$, then the sample is simply assigned to the class which its nearest neighbor belongs to. A classification error of 22.81% is scored by $k = 1$ with 26 samples misclassified by the nearest training samples (Table 4.6).

TABLE 4.6: KNN: confusion matrix of test data ($k = 1$)

		Predicted Class Label						
		A	**B1**	**C**	**D**	**E**	**F**	**B2**
Known Class Label	**A**	18	1	0	0	0	0	0
	B1	4	17	0	0	0	0	4
	C	0	1	17	0	2	0	0
	D	1	1	1	5	1	0	0
	E	0	0	0	1	10	0	0
	F	0	0	1	0	0	4	0
	B2	0	8	0	0	0	0	17

When k is set to a larger integer, the classification error increases. This can be explained by the drawbacks of KNN. The basic "majority voting" tends to be dominated by the classes with the more frequent training samples. However, we do not have equal numbers of whistles in each types. This may mislead the voting. Another problem is, when k is larger than 1, there might be two or more classes equally voted by training samples. One way to overcome this problem is to weigh the classification by the distance from the test data to each of its k nearest neighbors.

The probabilistic neural network (PNN) [53] is a typical way to weigh the distance between test and training samples. This network learns to estimate the probability density function (PDF) by separating the training data into their associated classes. In the PNN, there are at least three layers: the input layer, the radial basis layer and the competitive layer. The input layer computes the distances from the input test sample to the training samples. The radial basis layer is a hidden layer. It uses a Gaussian kernel function (also called the *radial basis function (RBF)*) α to compute the influence of the training samples from their distance to the test input. Hence, the nearer the training sample is to the input test data, the more influence it has in the decision of the class that the test data is assigned. The kernel function can be expressed as:

$$\alpha(\mathbf{x}, \mathbf{x}_i) = \exp\left(-\frac{d(\mathbf{x}, \mathbf{x}_i)}{2\sigma^2}\right) \tag{4.4}$$

where the distance between the input test sample $\mathbf{x}$ and the training sample $\mathbf{x}_i$ uses Euclidean distance here, the σ is the spread of the gaussian distribution.

Finally the competitive layer choses the class label of the input test sample based on the summation from the hidden layer for each class label. Table 4.7 shows the result of choosing a spread value of 0.1.

TABLE 4.7: PNN: confusion matrix of test data

		Predicted Class Label						
		A	**B1**	**C**	**D**	**E**	**F**	**B2**
Known Class Label	**A**	18	1	0	0	0	0	0
	B1	4	16	0	0	1	0	4
	C	0	1	18	0	1	0	0
	D	1	1	0	6	1	0	0
	E	0	0	0	0	11	0	0
	F	0	0	0	0	0	5	0
	B2	0	8	0	0	0	0	17

It is also found that the classification error increases with a larger spread in PNN. This is because the weight decreases slowly with the distance between the input sample and the training samples, which makes the distant training samples more influential. The choice of spread value can only be optimal when the distribution (or inter- and intra-class variation) is known. Another disadvantage of PNN is the high memory it requires for the input layer. It increases with the number of training data.

Conceptually PNN is similar to KNN. Both of them assign a sample to the category whose members have closest distances with this sample. However, PNN uses a radial basis function (RBF) to compute the weight for the neighboring points, while KNN only takes the direct distance and counts the numbers of nearest

training data. The Gaussian function is a common choice for RBF for multivariate analysis, and the sigma value of the Gaussian function determines the spread of the RBF function. Comparing Table 4.6 and Table 4.7, PNN mixes more whistles between Types B1 and B2. This is because whistles from B1 and B2 are mixed in the feature space and their RBFs overlap.

4.5 K-means Clustering

While Section 4.2, 4.3 and 4.4 discussed the supervised learning for classification with information from labeled training data. In this section onwards, the unsupervised learning is explored for natural clustering.

The feature vector of length N extracted from the whistle spectrogram represents one observation in N-dimensional feature space. Thus each whistle has a representation point in the space and there are n whistle points to be clustered. The k-means algorithm [30] partitions these n whistle points into k clusters where the value of k is predefined. Each cluster is parameterized by its mean, and whistle points are assigned to the cluster whose mean vector is the closest. After assignment, the cluster mean is updated, and the whistle points are reassigned. This iterative two-step algorithm continues until there is no change in clustering or the number of iterations is reached.

The feature vector used here is the N-point sampling evenly along the polynomial fit of the whistle curve in Chapter 3. If the value of k is set to 7, we have the clustering in shown in Table 4.8. The classification result is plotted using the

original whistle TFRs in Figure 4.6. In Table 4.9, the classification error is defined as the percentage of the misclassified samples among their labeled group. If we consider Types B1 and B2 as belonging to the same group, they are quite well grouped. This is shown for $k = 6$; the result in Table 4.10 shows that Type B1 and B2 are closely related.

TABLE 4.9: Classification error of k-means clustering ($k = 7$) on N-point sampling

Whistle Type	**Classification Error (%)**
A	8.33
B1	19.35
C	8.00
D	21.43
E	31.25
F	10.00
B2	6.45

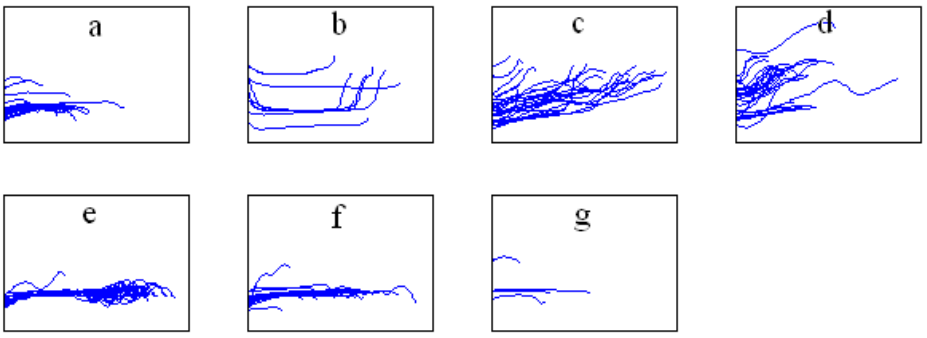


FIGURE 4.6: Plot of original whistles by k-means into 7 groups

To determine the optimal number of classes in k-means, a percentage reduction δ is used to represent the cost of k clustering:

$$\delta = \frac{\hat{J}_{e,1} - \hat{J}_{e,k}}{\hat{J}_{e,1}} \times 100 \tag{4.5}$$

TABLE 4.8: K-means clustering ($k = 7$)

Whistle Type	**a**	**b**	**c**	**d**	**e**	**f**	**g**
Whistle ID.	26($B1$)	111 ∼ 114(F)	63(C)	56 ∼ 62(C)	1 ∼ 5(A)	6(A)	40(B1)
	50(B1)	116 ∼ 120(F)	74(C)	64 ∼ 73(C)	7 ∼ 24(A)	23(A)	41(B1)
	51(B1)		79(C)	75 ∼ 78(C)	25(B1)	27 ∼ 38(B1)	31(B1)
	53(B1)		82 ∼ 87(D)	80(C)	39(B1)	42 ∼ 49(B1)	144(B2)
	121 ∼ 124(B2)		89 ∼ 92(D)	93(D)	88(D)	52(B1)	
	126 ∼ 130(B2)		94(D)	100(E)		54(B1)	
	132 ∼ 136(B2)		95 ∼ 99(E)	101(E)		55(B1)	
	139 ∼ 143(B2)		102 ∼ 106(E)	107(E)		81(D)	
	145 ∼ 149(B2)		108(E)	109(E)		125(B2)	
	151(B2)		115(F)	110(E)		137(B2)	
						138(B2)	
						150(B2)	

TABLE 4.10: K-means clustering ($k = 6$)

Whistle Type	**a**	**b**	**c**	**d**	**e**	**f**
Whistle ID.	$1 \sim 24$(A)	$56 \sim 62$(C)	47(B1)	$111 \sim 114$(F)	40(B1)	23(B1)
	25(B1)	$64 \sim 73$(C)	63(C)	$116 \sim 120$(F)	41(B1)	$26 \sim 38$(B1)
	39(B1)	$75 \sim 78$(C)	74(C)		51(B1)	$43 \sim 46$(B1)
	42(B1)	80(C)	79(C)		123(B2)	$48 \sim 50$(B1)
	55(B1)	93(D)	$81 \sim 87$(D)		130(B2)	$52 \sim 54$(B1)
	88(D)	100(E)	$89 \sim 92$(D)		131(B2)	121(B2)
		101(E)	94(D)		133(B2)	122(B2)
		107(E)	$95 \sim 99$(E)		134(B2)	$124 \sim 129$(B2)
		109(E)	$102 \sim 106$(E)		136(B2)	132(B2)
		110(E)	108(E)		139(B2)	135(B2)
			115(F)		$140 \sim 142$(B2)	137(B2)
					$144 \sim 147$(B2)	138(B2)
					149(B2)	143(B2)
					151(B2)	148(B2)
						150(B2)

where the normalized SSE $\hat{J}_{e,k}$ for k classes is defined in Equation 2.5. Figure 4.7 shows the percentage reduction with respect to the number of classes k. It is seen that the best choice of k should be $k = 4$. This is much smaller than the subjective classification of 6 or 7 classes. This is due to the overlapping of whistle points in feature space using the selected N-point features.

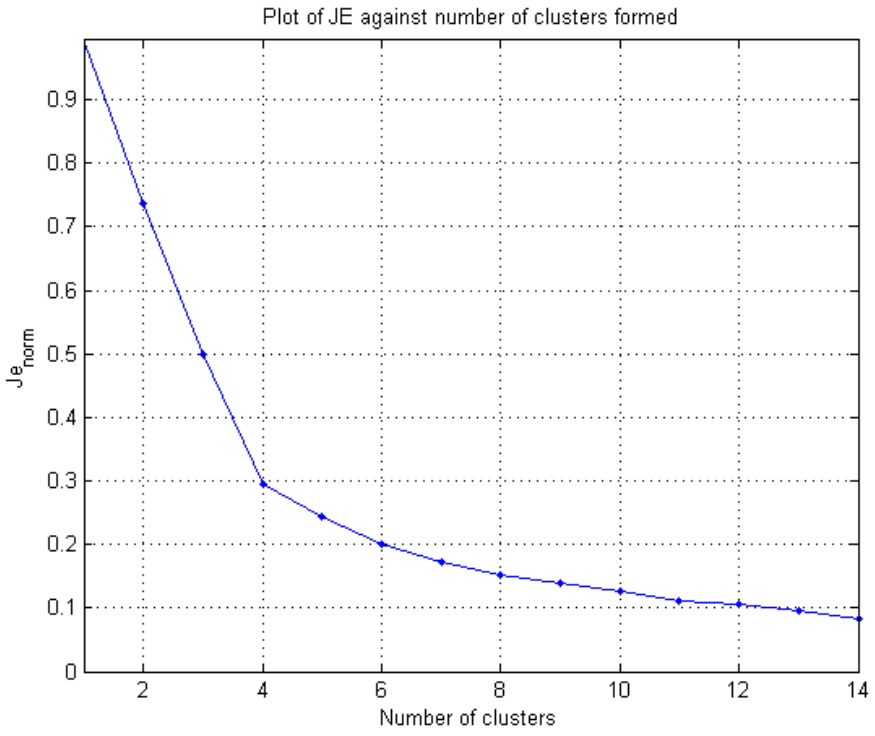
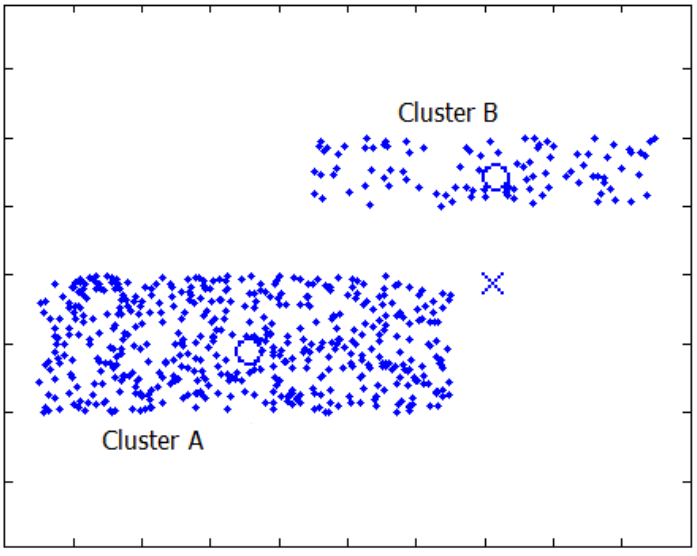


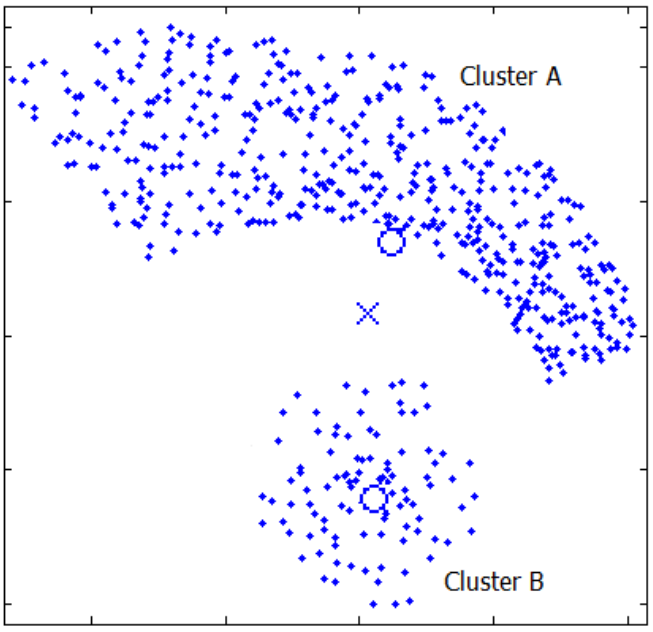
FIGURE 4.7: Normalized SSE J_e against number of clusters

Some drawbacks are noted for the k-means algorithm. Firstly, Euclidean distance used as distance measure between whistles and clusters might not be the right metric since it does not consider the cluster shape and data distribution within the cluster. Taking 2-dimensional feature space as example, the data points $(\cdot)$ may cluster in a non-elliptical way as the two cases shown in Figure 4.8. In each case the data points can be obviously divided into two clusters with their cluster

mean (∘). However the clustering of some points would be wrong if they are assigned to the nearest cluster mean [46]. For a data point (×) outside the clusters in Figure 4.8(a), although it is nearer to Cluster A, the distance to the cluster mean of Cluster B is smaller than to the cluster mean of Cluster A. This is because Cluster A has a larger cluster radius; points at the edge of the distribution are thus further from the cluster center. An alternative measurement can be the shortest distance or the average distance between a data point and a cluster centroid. The shape of the cluster distribution also affects cluster mean location. In Figure 4.8(b), the cluster mean of Cluster A falls almost outside its data distribution. Though the point (repeatedly a ×) is nearer to the mean point of Cluster A and has almost the same nearest distance to both clusters, it might belong to Cluster B if the cluster shape is considered.



(a) the data point in cross is nearer to the mean point of the smaller rectangular cluster but it obviously belongs to the larger rectangular group



(b) similar scenario when cluster mean of the larger group falls out of its cluster region

FIGURE 4.8: Demonstration of clusters in 2-D feature space: the data point in $\times$ is too be classified with other data points in $\cdot$ of two clusters by distribution; $\circ$ shows the mean point of each cluster.

A dynamic modeling method *Chameleon* [26] for cluster representation has been proposed to consider the cluster configurations in data mining applications. This could be useful in dealing with large amount of whistles, provided that the feature vector is precise enough to represent whistles. The clusters formed by many whistles might be of arbitrary shape, proximity, orientation and varying densities. *Chameleon* introduces relative inter-connectivity and relative closeness as dynamic criterion in the agglomerative hierarchical clustering and thus does not depend on a static, user-supplied model such as the metric space formed by selected features. Another advantage of *Chameleon* is that it operates on a sparse graph in which

nodes represent data items and weighted edges represent similarities among the data items. It enables the data that are available only in similarity space and not in metric spaces.[1]

4.6 Competitive Learning and Self-Organizing Map (SOM)

Artificial neural networks are often used to model complex relationship between the inputs and outputs. When the input is the feature data of dolphin whistles and output is the class label, neural networks can be used to find whistle patterns.

A basic competitive learning network consists of an input layer and a competitive layer. Similar to other neural networks, an input pattern at the input layer is a sample point in the N-dimensional feature space. The output nodes indicate the classes and each output node represents a pattern category. With k classes, the neurons with weighting vectors $\mathbf{w}_i (i = 1, ..., k)$ in the competitive layer learn to represent different regions of the input space. Every time an input pattern is fed in, the neuron associated with the nearest distance with the input pattern becomes the winner. The weight vector $\mathbf{w}^{winner}$ of the winner will be updated by attracting the data input $\mathbf{x}$ with the strength that is decided by the distance

[1]Data sets in a metric space have a fixed number of attributes for each data item. For example, the descriptive features from whistle spectrograms. Data sets in a similarity space only provide similarities between data items.

$d(\mathbf{w}^{winner}, \mathbf{x})$ between them:

$$\Delta\mathbf{w}^{winner} = \alpha(\mathbf{w}^{winner} - \mathbf{x})d(\mathbf{w}^{winner}, \mathbf{x}) \tag{4.6}$$

where $\alpha(\mathbf{w}^{winner} - \mathbf{x})$ is the learning rate and regulates how fast the weighting neuron moves towards the data input. With 3 PCs, the feature space for whistle feature data $\mathbf{x}$ and weight $\mathbf{w}$ is 3-D. For easier visualization, only the first 2 PCs are shown; the third component has less variance then the first two (Figure 3.2(a)). In Figure 4.9, the initial neuron is shown in a black solid circle in the center of the data region. After 100 epochs, the neurons are trained to move to the center of clusters. The resulting positions are plotted in blue solid circles (Figure 4.9) and marked as $\mathbf{w}_i, i = 1, ..., 7$. We can see that neuron $\mathbf{w}_6$ is a good representation of Type A, neuron $\mathbf{w}_1$ is a good representation of Type F, and neuron $\mathbf{w}_2$ is a good representation of Type C; there are 3 neurons ($\mathbf{w}_3$, $\mathbf{w}_5$ and $\mathbf{w}_7$) in regions of Types B1 and B2; neuron $\mathbf{w}_4$ seems to be located between Types D and E.

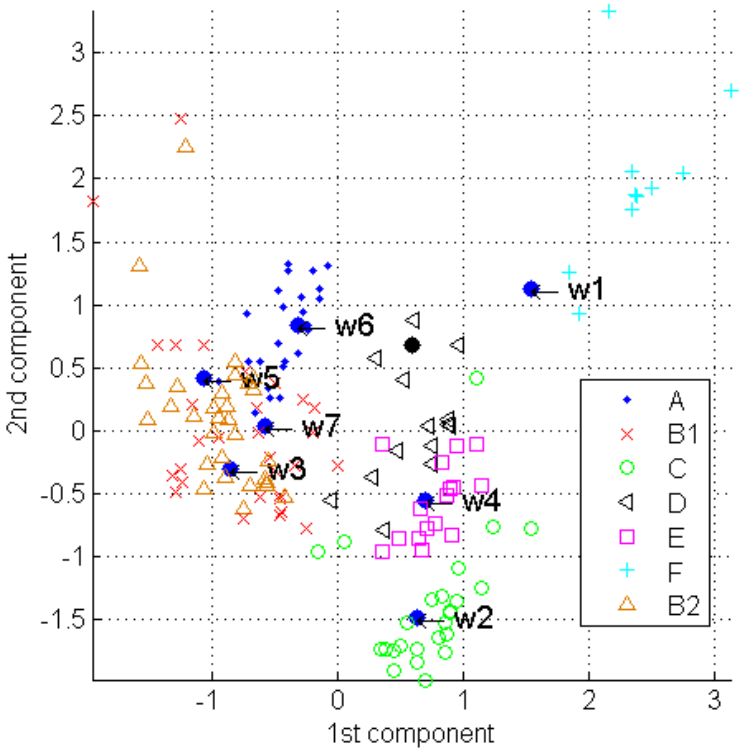


FIGURE 4.9: Clustering by competitive learning: learning neurons (solid circles $w_i, i = 1, ..., 7$) after 100 epochs and whistle data with labels

The clustering result is shown in Table 4.11. The classified whistle types are defined by the neurons trained. We can see that the clustering confirms the competitive learning in Figure 4.9: neuron $\mathbf{w}_6$ groups most whistles from Type A, neuron $\mathbf{w}_1$ contains all whistles from Type F, neuron $\mathbf{w}_2$ contains most of Type C, whistles from B1 and B2 spread over neuron $\mathbf{w}_3$, $\mathbf{w}_5$ and $\mathbf{w}_7$.

In principle, all the neurons move in the general direction of nearby data points, ending up in positions as representatives of clusters. However, neurons in competitive learning are allowed to move freely in feature space; the relationship between clusters is unknown. A clever variety of competitive learning is the self-organizing map (SOM).

TABLE 4.11: Clustering result by competitive learning

Whistle Type	$\mathbf{w}_1$	$\mathbf{w}_2$	$\mathbf{w}_3$	$\mathbf{w}_4$	$\mathbf{w}_5$	$\mathbf{w}_6$	$\mathbf{w}_7$
	79(C)	46(B1)	26(B1)	63(C)	40(B1)	$1 \sim 3$(A)	4(A)
	87(D)	$56 \sim 62$(C)	33(B1)	$82 \sim 86$(D)	41(B1)	5(A)	6(A)
	89(D)	$64 \sim 78$(C)	34(B1)	92(D)	51(B1)	$7 \sim 10$(A)	11(A)
	90(D)	80(C)	38(B1)	94(D)	130(B2)	$12 \sim 14$(A)	15(A)
	91(D)	93(D)	43(B1)	95(E)	131(B2)	16(A)	18(A)
	$111 \sim 120$(F)	97(E)	49(B1)	96(E)	134(B2)	17(A)	19(A)
		98(E)	50(B1)	99(E)	136(B2)	20(A)	$22 \sim 24$(A)
		100(E)	53(B1)	$102 \sim 106$(E)	139(B2)	21(A)	25(B1)
		101(E)	54(B1)	108(E)	$140 \sim 132$(B2)	88(D)	$27 \sim 32$(B1)
Whistle ID.		107(E)	$121 \sim 129$(B2)		$144 \sim 147$(B2)		$35 \sim 37$(B1)
		109(E)	132(B2)		149(B2)		39(B1)
		110(E)	133(B2)		151(B2)		42(B1)
			135(B2)				44(B1)
			137(B2)				45(B1)
			138(B2)				47(B1)
			143(B2)				48(B1)
			148(B2)				52(B1)
			150(B2)				55(B1)
							82(B1)

The SOM describes a mapping from a higher dimensional input space (i.e. the feature space) to a lower dimensional map space [29]. It has been applied in speech recognition [55] and many other vocalizations [52] [15]. The inputs are still the data to be classified. However the trained neurons (or nodes) form a grid map and each is associated with a weighting vector of the same dimension as the input vectors and a position in the map. Hence the neurons are not moving freely; the constraints or grid connection between them show only the relationship between clusters represented by these neurons. When there is an input data, the Euclidean distances of the input data and all neurons are computed. A best matching unit (BMU) is the winning neuron $\mathbf{w}^{winner}$ that is most similar to the input. While only the winning neuron is updated in competitive learning, a neighborhood function is used to update all the neurons within certain neighborhood. It preserves the topological properties of the input space. This can be seen in the neuron updating function:

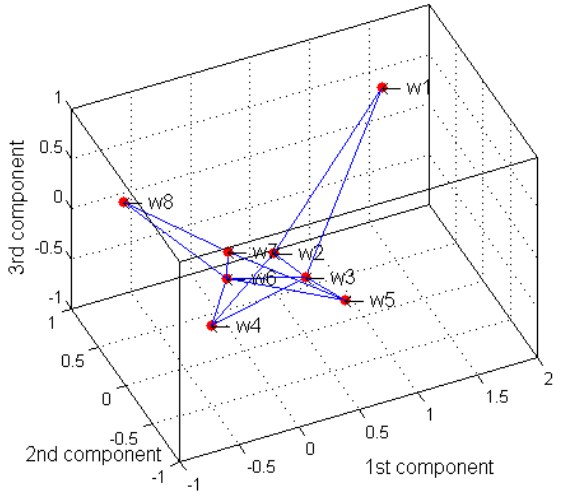
$$\Delta \mathbf{w}_i = \alpha(\mathbf{w}_i - \mathbf{x}) h(\mathbf{w}_i, \mathbf{w}^{winner}) d(\mathbf{w_i}, \mathbf{x}) \tag{4.7}$$

where α is a learning coefficient and $\mathbf{x}$ is an input vector. The term in Equation 4.7 for SOM that does not exist in Equation 4.6 for competitive learning is the neighborhood function $h(\mathbf{w}_i)$. It is equal to 1 when neuron $\mathbf{w}_i$ is the BMU $\mathbf{w}^{winner}$ itself; it depends on the lattice distance (i.e. the number of links between neuron $\mathbf{w}_i$ and the BMU).

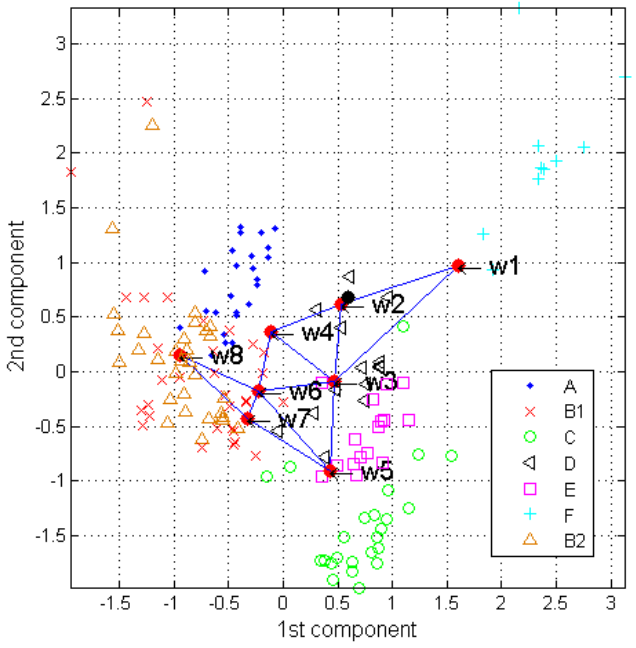
Neurons in SOM are interconnected with each other and display the relationship among clusters. After several epochs, the map is updated and learns to detect the regularities and correlations in the input space. SOM considers the distance

of each input from all the neurons rather than the closest one (in the case of k-means). It is more sophisticated by using a neighborhood function (for example, the Gaussian function is a common choice) and maintaining a relationship between clusters. K-means requires the number of clusters to fit the data by users while SOM requires the shape and size of a network of clusters. However, SOM does not force as many clusters as the number of neurons, since it is possible for a node to have no associated input vectors (considered as empty).

With a 3-D feature space, the map space is set to 2-D for 8 classes (and hence 2×4). The trained neurons with lattices are displayed in the 3-D feature space (Figure 4.10(a)). The first 2 PCs are shown in Figure 4.10(b). Again the classification result agrees with the observation on the neurons. For example, neuron $\mathbf{w}_1$ is near the center of Type F, $\mathbf{w}_8$ is in area of Types B1 and B2, neuron $\mathbf{w}_4$ is the one nearest to most of Type A but also near to some whistles of Type B1, and neuron $\mathbf{w}_5$ represents Types C and E. The result is in Table 4.12.



(a) Learning neurons in 3-D plot (solid red circles $w_i, i = 1..8$)



(b) Learning neurons after 500 epoches and labeled whistle data

FIGURE 4.10: Clustering by SOM

Despite its wide applications in classification and data mining, SOM remains a black box. The variables that SOM requires increase the complexity of clustering.

TABLE 4.12: Clustering result by SOM (8 classes)

Whistle Type	$\mathbf{w}_1$	$\mathbf{w}_2$	$\mathbf{w}_3$	$\mathbf{w}_4$	$\mathbf{w}_5$	$\mathbf{w}_6$	$\mathbf{w}_7$	$\mathbf{w}_8$
Whistle ID.	111 ∼ 114(F)	79(C)	63(C)	1 ∼ 22(A)	56 ∼ 62(C)	23(A)	34(B1)	26(B1)
	116 ∼ 120(F)	82(D)	83 ∼ 85(D)	24(A)	64(C)	29 ∼ 32(B1)	54(B1)	27(B1)
		86(D)	89(D)	25(B1)	65(C)	35(B1)	66(C)	33(B1)
		87(D)	90(D)	28(B1)	67 ∼ 78(C)	37(B1)	93(D)	36(B1)
		91(D)	94(D)	39(B1)	80(C)	43 ∼ 46(B1)	122(B2)	38(B1)
		115(F)	95(E)	42(B1)	92(D)	49(B1)	123(B2)	40(B1)
			96(E)	47(B1)	97(E)	52(B1)	125(B2)	41(B1)
			99(E)	55(B1)	98(E)	81(D)	126(B2)	48(B1)
			103(E)	88(D)	100 ∼ 102(E)		137(B2)	50(B1)
			105(E)		104(E)		138(B2)	51(B1)
			106(E)		107(E)		148(B2)	53(B1)
			108(E)		109(E)			121(B2)
					110(E)			124(B2)
								127 ∼ 136(B2)
								139 ∼ 147(B2)
								149 ∼ 151(B2)

This causes difficulties in parameter evaluation for optimal clustering when no vocalization categories are known *a priori* to be biologically meaningful. These variables include: grid topology, number of neurons, dimensionality of layers, weight tuning of neurons and neighborhood function, etc. It appears that competitive learning gives better classification than SOM. One reason is that neurons in competitive learning move freely and hence captures isolated clusters (Types A, C and F) when most whistle data lie and overlap in the center area. However, comparing the neuron plots and the classification results, both competitive learning and SOM tell us that the distribution of data itself guides the clustering. If whistles of different types overlap in the feature space, it is very difficult for these artificial neural networks to differentiate them. Hence the selection of feature vectors and similarity measure matter in the first place.

This chapter has reviewed the typical classification methods for supervised and unsupervised learning. Most of the methods explore the data distribution for clustering. The selection of the classification method depends the type of feature and its similarity.

Chapter 5

Dynamic Time Warping (DTW)

As discussed in Chapter 2 and Chapter 3, dolphin whistle classification involves proper selection of feature vector and similarity measurement using prior knowledge on the significance of features for categorization. Just like human speech, dolphin whistles of the same type may vary in speed. Since N-point feature is evenly sampled along whistle traces in time domain, the direct matching of these feature vector might not be optimal (This has been demonstrated in Figure 3.6). In this chapter, dynamic time warping (DTW) is proposed to solve the nonlinear mapping between whistles with local time variation. The basic DTW is outlined in Section 5.1. Modifications to features and similarity will be shown in Section 5.2 and Section 5.3. These modifications are quite similar to the way humans observe and recognize the dolphin whistle patterns.

5.1 Dynamic Time Warping (DTW)

Consider two whistle feature vectors of different lengths: one is called the *query whistle* Q of length m and the other is the *template whistle* T of length n. A difference matrix $\mathbf{D}$ is firstly constructed. The element $\mathbf{D}(i,j)$ is the difference between the ith feature in query whistle and the jth feature in the template whistle. An example of the feature difference could be the absolute frequency difference between any pair of points from the two TFRs respectively:

$$\mathbf{D}(i,j) = d(\mathbf{D}(i) - \mathbf{T}(j)) = |\mathbf{Q}(i) - \mathbf{T}(j)| \tag{5.1}$$

where $i \in [1,m]$ and $j \in [1,n]$. The distance between the query and template whistles is

$$d(Q,T) = \frac{1}{m} \min_{w} \{ \sum_{i=1}^{m} |Q(i) - T(\xi(i))| \} \tag{5.2}$$

where i is the index of query element while $j = \xi(i)$ is the corresponding index of the template element. The matching cost is the sum of differences of all paired elements. The final matching path is found with the minimum matching cost. The whistle distance is the sum of the element differences along the matching path normalized by the length of the query sequence.

To find the matching path with minimum dissimilarity, a cost matrix $\mathbf{C}$ is constructed on the difference matrix $\mathbf{D}$ by dynamic programming [43], where a running tab updates the entries of cost matrix $\mathbf{C}$ along each row by accumulating the minimum cost measured previously. In the basic DTW algorithm, the running tab adds the current difference element $\mathbf{D}(i,j)$ for that position (or node) to the

minimum of the three previously determined elements $\mathbf{C}(i-1, j-1)$, $\mathbf{C}(i, j-1)$ and $\mathbf{C}(i-1, j)$ of the cost matrix:

$$\mathbf{C}(i,j) = min \begin{pmatrix} \mathbf{C}(i-1,j-1) \\ \mathbf{C}(i-1,j) \\ \mathbf{C}(i,j-1) \end{pmatrix} + \mathbf{D}(i,j). \tag{5.3}$$

which is called 0°-45°-90° warping shown in Figure 5.1. The tab at current position $[i, j]$ looks backwards for the path with minimum cost to add on until it reaches $[m, n]$. On the cost matrix $\mathbf{C}$, the matching path is found from the last pair $\mathbf{C}(m, n)$ to the beginning pair $\mathbf{C}(1, 1)$ by tracking the nodes with minimum accumulated costs.

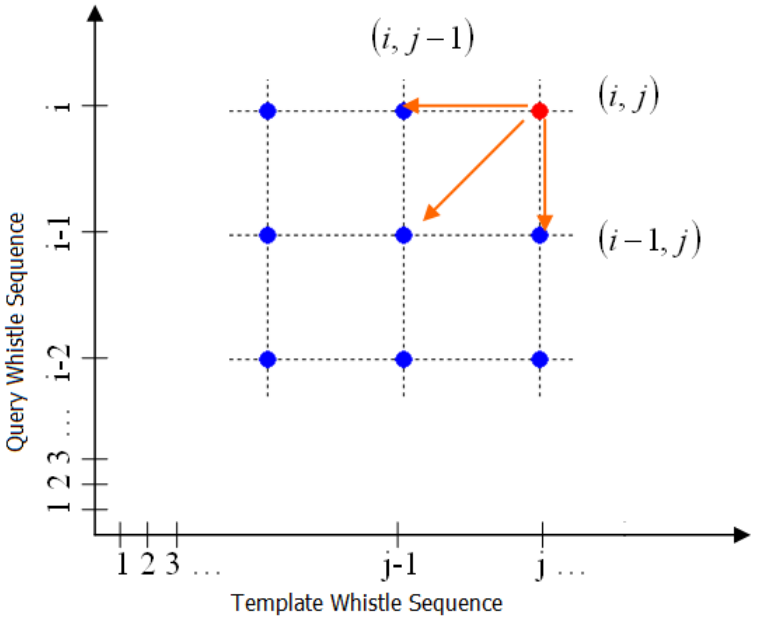


FIGURE 5.1: Cost matrix calculation in basic DTW: Cost of matching is accumulated from the minimum of the previous three in 0°-45°-90° direction (yellow arrows)

An example of DTW matching is shown in Figure 5.2. For clear visualization,

the time and frequency are shifted and only every other 3 pairs are shown. The matching is nonlinear. For example, the lowest valley frequencies in middle of query whistle are all matched to one lowest frequency in template whistle.

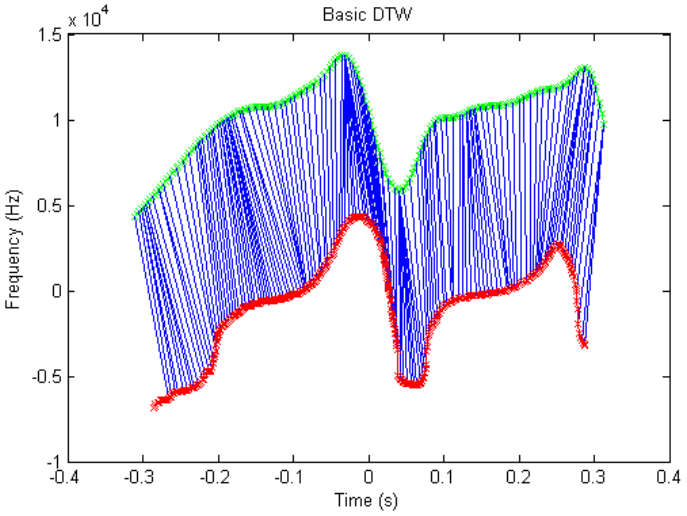


FIGURE 5.2: An example of basic DTW matching: a query whistle (red) is matched to template (green) with matching lines (blue)

5.2 Modified DTW

The standard DTW can be altered to suit application. From an earlier publication [20], we know that the template whistle traces are well defined while the query whistle traces, extracted from the automated method, may have noise and breaks. The tracing noise can be viewed as 'outliers' describing either

1. *tracing points that have a low likelihood of being consistent with the rest in frequency,* or

2. *tracing points that are far from the main body in time domain.*

In Section 5.1, the warping is 0 °-45 °-90 ° warping. In this section, the warping choice is modified to 0 °-27 °-45 °-63 °-90 ° warping in Equation 5.4. It is illustrated in Figure 5.3. This allows one-to-many mapping for sequences of different lengths with local variations. A single frequency outlier can be ignored in the 27 °-63 ° direction.

$$\mathbf{C}(i,j) = \begin{pmatrix} \mathbf{C}(i-1,j-1) \\ \mathbf{C}(i-2,j-1) \\ \mathbf{C}(i-1,j-2) \\ \mathbf{C}(i,j-1) \\ \mathbf{C}(i-1,j) \end{pmatrix} + \mathbf{D}(i,j). \tag{5.4}$$

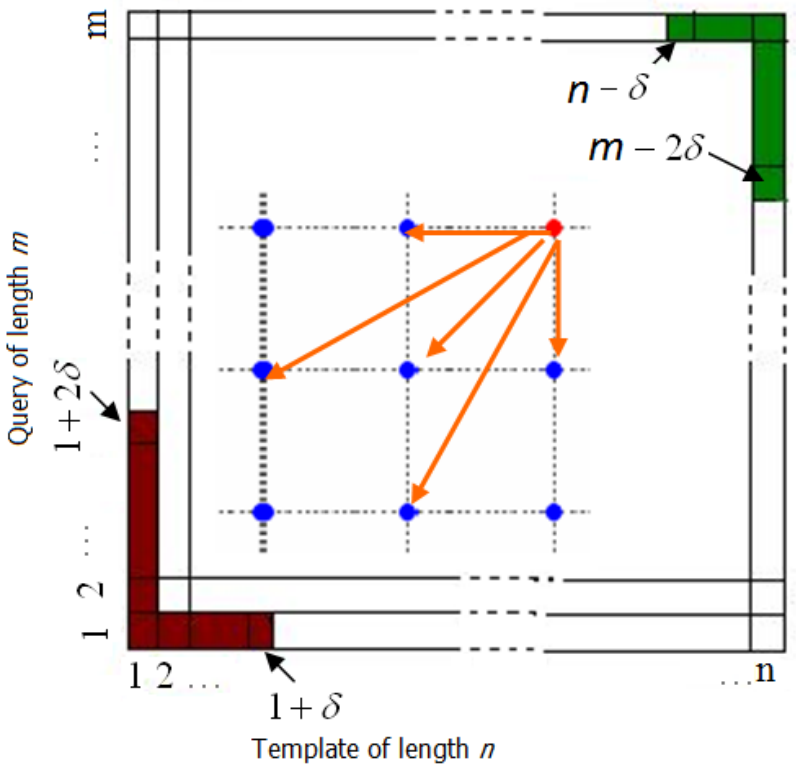


FIGURE 5.3: Cost matrix calculation in modified DTW: Cost of matching is accumulated from the minimum of the previous five in 0 °-27 °-45 °-63 °-90 ° directions (yellow arrows) with adaptive selection areas

To exclude the outliers outside the whistle duration from matching, the starting point of the matching path is chosen as being the minimum difference pair in the range colored in green when searching back on the cost matrix (Figure 5.3); the matching path ends at any pair in the range colored in dard red. These two ranges are defined as:

$$\begin{cases} 1 \leq w(1) \leq \delta + 1 \\ n - \delta \leq w(m) \leq n \\ min[w(i) = 1], 1 \leq i \leq 1 + 2\delta \\ max[w(i) = n], m - 2\delta \leq i \leq m \end{cases} \tag{5.5}$$

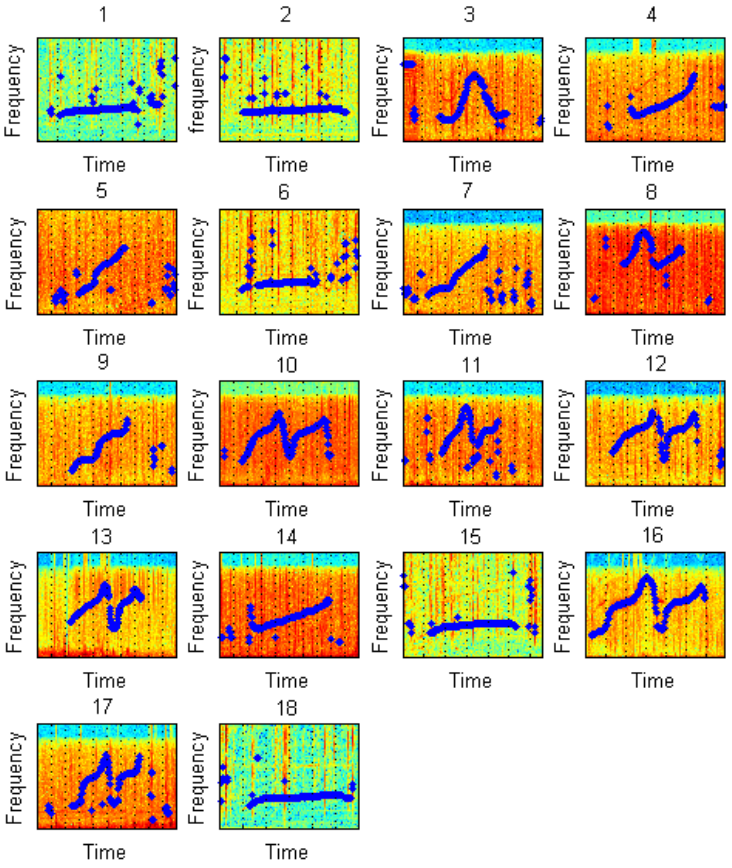
where δ is an adaptive parameter defined from the lenght of the query whistle sequences:

$$\delta = m/12 \tag{5.6}$$

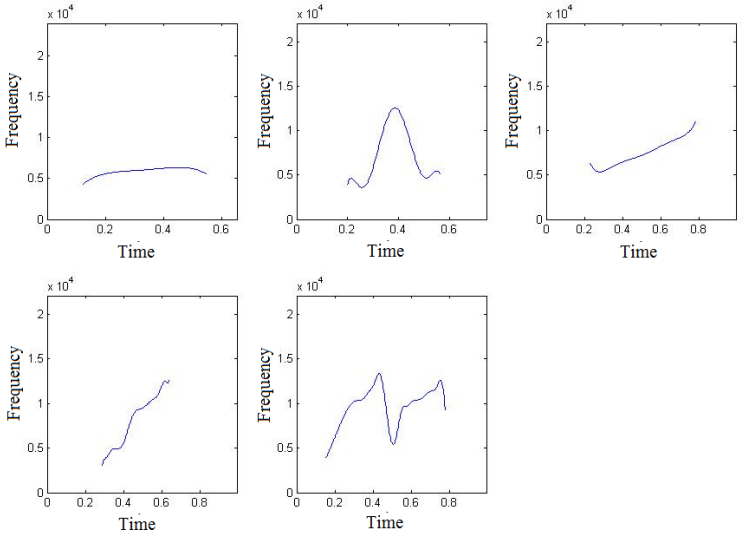
To be frequency invariant for the frequency-modulated (FM) dolphin whistles, the curve traces are shifted by the median frequency of all TFRs. The use of the median over the mean is driven by the consideration of robustness to outliers. Since the query whistle is not fully matched due to the flexible selection of ending pair, the accumulated difference is only normalized by the matching ratio $n/|C|$, where n is the length of template sequence and $|C|$ is the length of the matching path.

The modified DTW was compared with the basic DTW and McCowan's 20-point feature [35]. 18 query whistles in Figure 5.4(a) are to be matched to 5 artificially synthesized templates in Figure 5.4(b). In Figure 5.4(a), the noise remains

as outliers in time or frequency in most whistle traces derived from automated whistle extraction.



(a) 18 query whistles with imperfect tracing: the frequency is from 0 Hz to 20 kHz, the time ticks are marked every 0.1 seconds



(b) 5 templates to match

FIGURE 5.4: Query and template whistles

The first type of outliers has a high standard deviation from the mean frequency; the second type might be consistent with the main body in frequency, yet occurring before and after whistles. This makes outlier detection difficult to decide in the presence of breaks within whistle curve. One measure of *tracing error* records the percentage of outliers and breaks, compared with commonly agreed manual traces. Normalized root mean squared error (RMSE) evaluates the tracing error compared with *spline* [14] interpolated manual traces. The tracing error between the auto-traces and reference is measured at the time instances at the tracing points that the former has. The tracing error is defined as

$$e_{tracing} = \sqrt{\frac{\sum_{t_1}^{t_{end}} |f_R - f_A|^2}{n}} / bw \tag{5.7}$$

where n is the number of total sampling instances, f_A is the frequencies of the automated tracing. The reference tracing has frequency points f_R and duration starting at t_1 and ending at t_{end}. The scaling factor bw is the bandwidth of the whistle. Hence the tracing error measures the impact of average tracing error on the whistle frequency bandwidth. If the tracing error is larger than 1, it means that the noise that is present is on average overwhelming the whistle frequency range. There are two other measures on tracing error: *missed* measures the percentage of duration that are missed by automated tracing; and *extra* measures the percentage of outliers in both time and frequency over all traces. The tracing errors of the 18 query whistles are listed in Table 5.1.

TABLE 5.1: Tracing error of the 18 query whistles

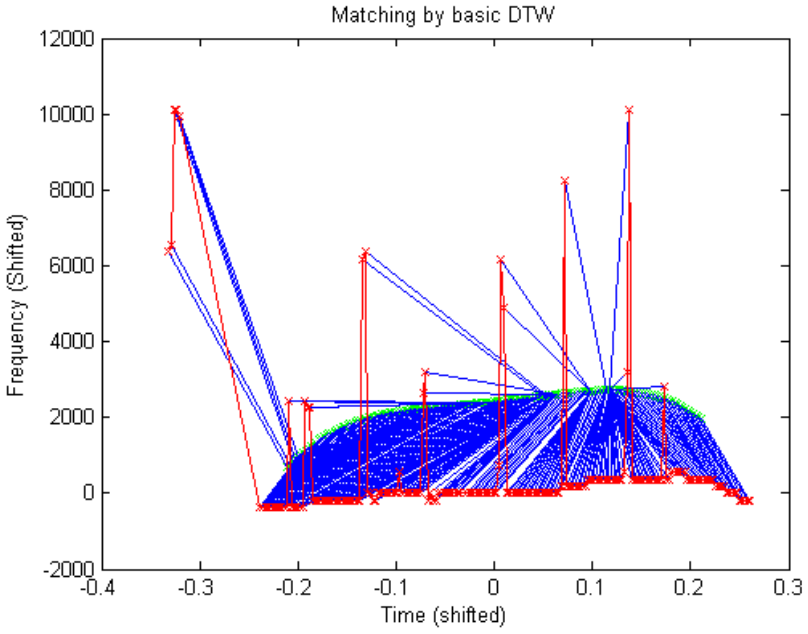
ID.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Error	0.705	1.415	0.059	0.006	0.060	1.054	0.070	0.073	0.006
Missed	0.06	0.021	0	0.049	0	0.219	0.026	0	0
Extra	0.205	0.11	0.217	0.138	0.387	0.211	0.285	0.209	0.116
ID.	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**
Error	0.018	0.216	0.055	0.132	0.27	0.300	0.013	0.217	0.35725
Missed	0	0	0.068	0	0	0.116	0.0152	0.006	0.064
Extra	0.072	0.217	0.073	0.122	0.24	0.086	0	0.379	0.087

It is observed that single outlier in frequency occurs quite frequently in the error tracing. The breaks on steep slope (for example, Whistle 8, 13 and 16) are not real break in time domain; they are due to the large frequency change in short time (two consecutive time bins). Whistle 6 has the highest missing rate; its break is obviously seen in Figure 5.4(a). The tracing error are more than 1 for Whistle 2 and 6; the outliers in frequency have frequency error much larger than the bandwidth of the whistle itself. These measurements describe the tracing

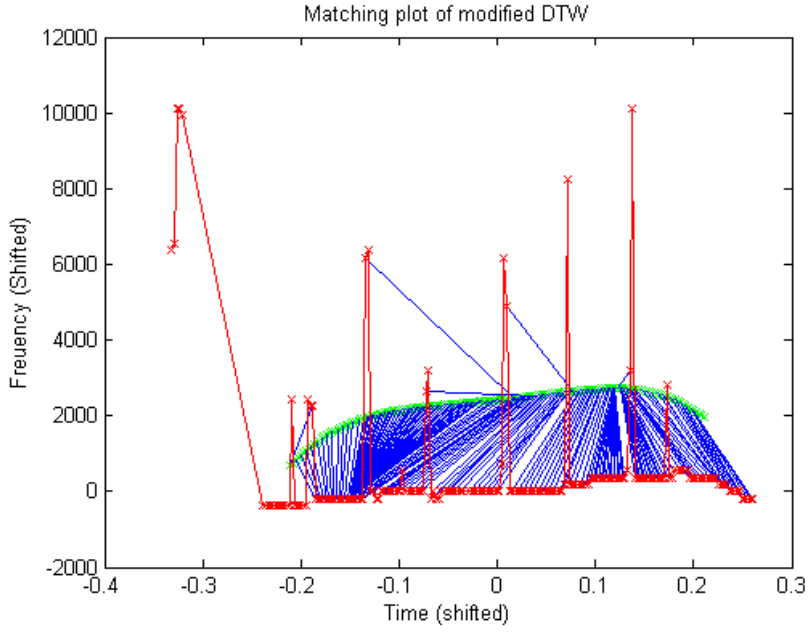
performance in general. They will affect the template matching if these errors are too much.

5.2.1 DTW for Template Matching

Figure 5.5 shows one example of template matching. While outliers might mislead the basic DTW in Figure 5.5(a) for an over-mapping, the modified DTW tolerates most outliers in both frequency and time from query whistle 1. Despite only ignoring the single outlier inside the traced whistle, the modified DTW improves matching performance.



(a) Template matching by basic DTW



(b) Template matching by modified DTW

FIGURE 5.5: A matching example of modified DTW vs. basic DTW: query whistle in red is matched to template whistle in green

Table 5.2 shows the matching result of the 18 query whistles against the templates.

TABLE 5.2: Template matching result of the 18 query whistles

ID.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Labels	1	1	2	3	4	1	4	2	4
N-point	1	1	2	4	2	1	1	1	5
Basic DTW	3	1	2	2	2	2	2	5	2
Modified DTW	1	1	2	3	4	1	2	2	4
ID.	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**
Labels	5	5	5	5	3	1	5	5	1
N-point	1	5	4	5	4	1	5	1	1
Basic DTW	5	5	5	5	2	2	5	5	2
Modified DTW	5	5	5	5	3	1	5	5	1

A measurement is needed to define the ability of these similarity measures in recognizing the correct template from others. Let the dissimilarity (or distance)

of the query whistle with the correct template be d_c and with other template be d_o. The differentiation ability (DA) is defined as:

$$DA = \frac{\min(d_o) - d_c}{d_c}. \tag{5.8}$$

DA should be positive for correct classification. Larger DA indicates an easier decision in selecting the matching template. If $\min(d_o) > d_c$, whistle will be matched to a wrong template. Figure 5.6 compares DA for basic DTW, modified DTW and Euclidean distance with N-point feature. When the query whistle is matched to the wrong template, the DA is negative and hence not shown in the log-scale.

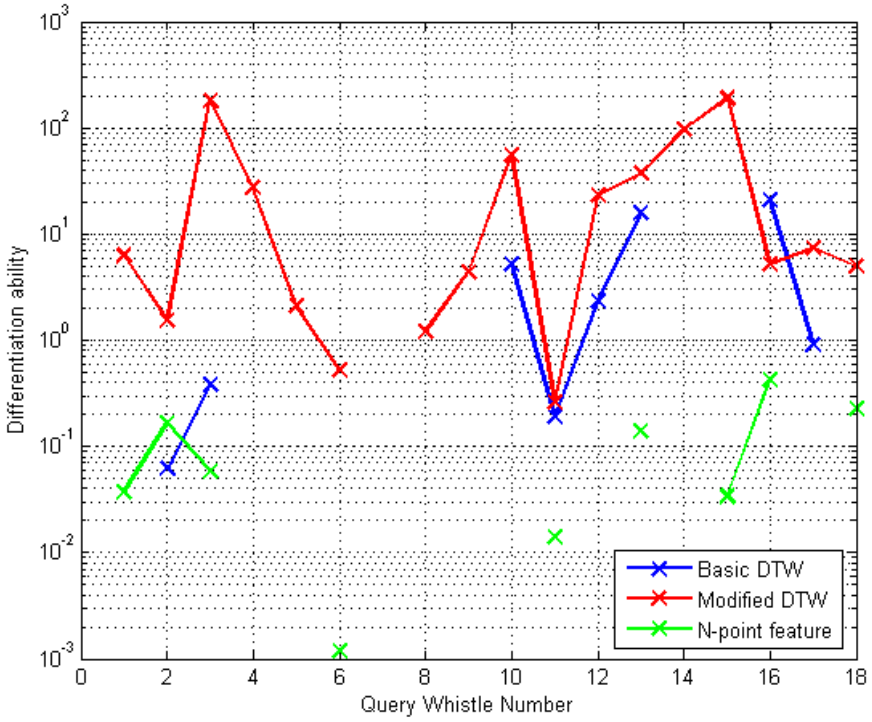


FIGURE 5.6: Differentiability ability plot

The misclassification of Whistle 7 by modified DTW is due to overwhelming

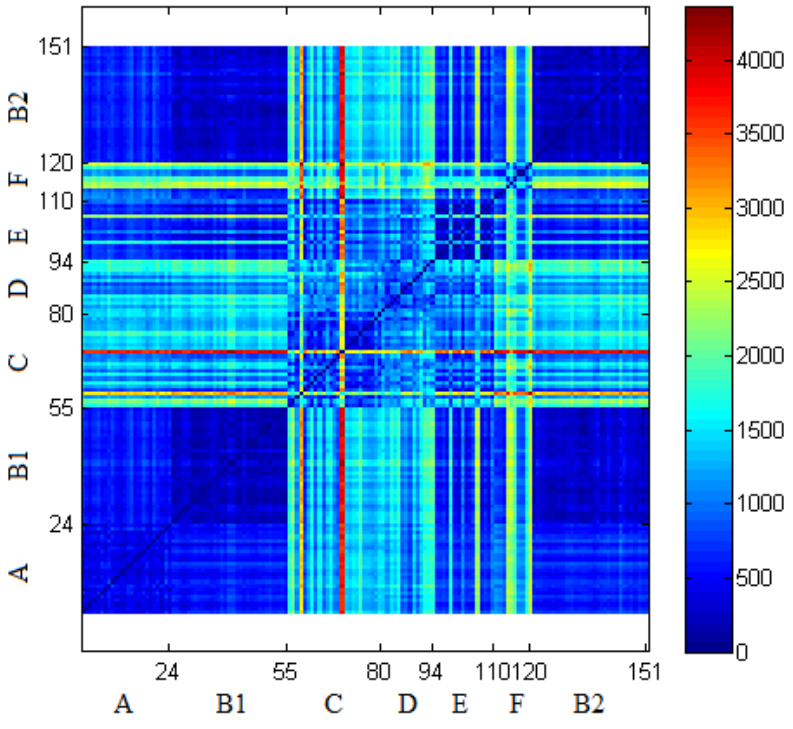
noise in traces. The value of the noise makes the sequence of pixels more like Template 2. It is difficult to control the tracing accuracy for all whistle recordings under different conditions. From here on, whistle traces of good quality are assumed.

5.2.2 DTW for Natural Clustering

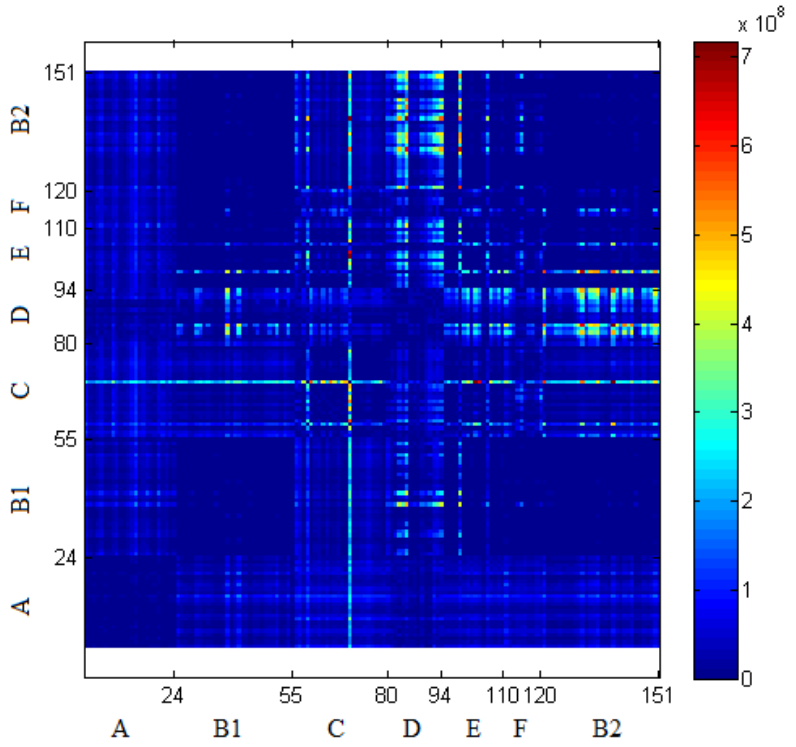
As a similarity measure, DTW only gives pairwise dissimilarity between whistles. This similarity cannot be shown in feature space. However, a dissimilarity matrix recording the pairwise distances between whistles can be constructed by DTW. For natural clustering of a set of dolphin whistles, there are no query and template whistles; all the whistles are unknown. There are two points to be noted for clustering compared with template matching: One, both whistles in the comparison pair are query whistles. The noisy traces remaining may be complex. Hence whistles are assumed of good tracing quality. Two, in the comparison pair, whistle of the shorter length is matched to the one with longer length and the normalization term is the length of the shorter whistle.

A dissimilarity plot for the whistles in Appendix A is shown first using the modified DTW (Figure 5.7(b)). The ticks on axes are the whistle numbers on which each whistle type ends. The small matrix along the diagonal line, for example, between whistles 25 to 55, shows the dissimilarities within Type B1. Both dissimilarities have a significant line at Whistle 70, which shows further distance from other whistles than from the ones in the same type. Surprisingly, the dissimilarity matrix of the modified DTW (Figure 5.7(b)) does not show clear differences

between the clusters. Whistles of different types have fairly similar color-coded distances as whistles of the same type. The Euclidean distance for the 20-point feature vector (Figure 5.7(a) for comparison) clearly has better clustering. This is due to an over-warped matching from DTW (both basic and modified DTW). Figure 5.8 shows two examples of over-warping, resulting in very small dissimilarity values for whistles of different types. DTW gives too much flexibility in warping when there is no noisy trace; to be more accurately saying, the whistle traces have too much redundancy for one-to-many mapping. The next step is to use a shorter and more compact feature vector to eliminate the over-warping. It will show that DTW matching is improved by reduction of whistle features in the next section.



(a) Euclidean distance on 20-point sampling



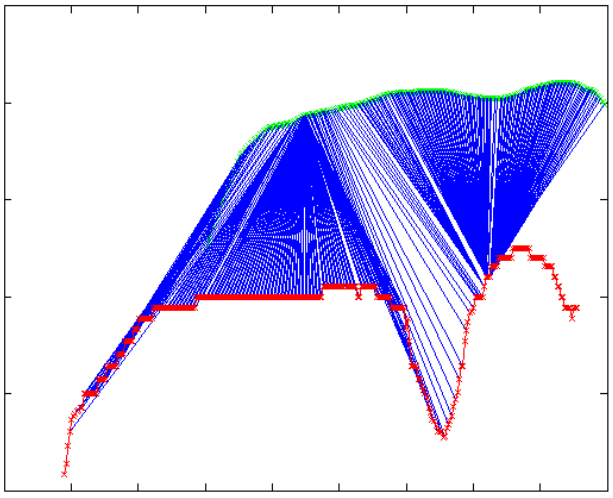
(b) Modified DTW

FIGURE 5.7: Dissimilarity plot of Euclidean distance and modified DTW

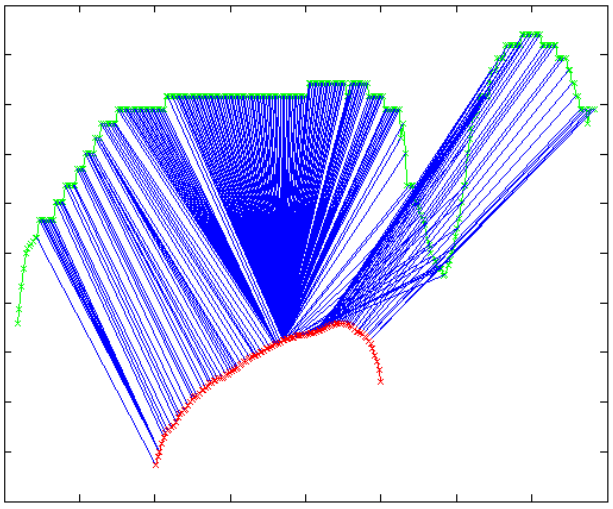
Nevertheless, if we can get a good dissimilarity matrix plot, multidimensional scaling (MDS) can be used to transform the distances to a coordinate representation [5] depending on the requirement of the classification method afterwards.

5.3 Line Segment Dynamic Time Warping for Template Matching

In this section, whistles are represented by a series of segments. As the noisy traces tend to mislead segment approximation in an automated process, whistle



(a) Whistle 1 vs. 28



(b) Whistle 1 vs. 125

FIGURE 5.8: Over-warped matching by DTW, too much one-to-many mapping

traces are assumed to be of good quality. A local feature difference is proposed for the segments, and is used by DTW for pairwise whistles similarity.

5.3.1 Whistle Curve Segmentation

There are two ways of whistle curve segmentation:

1. *Top-to-bottom*: the whistle curve is approximated by a single line segment first and splits until the required number of segments or approximation error is reached.

2. *Bottom-to-up*: the basis segments are built from tracing points and are merged into set of segments when requirements are reached.

The bottom-to-top merging is used for whistle segmentation. Defining K_s as the number of segments, the whistle TFR by is approximated by K_s straight lines. Each segment represents a period of rising, falling or flat frequency. The segmentation initializes basis segments by connecting every consecutive pair tracing points. The merging cost of a segment is the potential approximation error by merging it with its neighbor (the next segment in time domain). Each time, the segment with the lowest merging cost is merged with its neighbor and the piecewise segment approximation is updated. The bottom-to-top merging stops when the required K_s is reached. To discourage disturbance of noisy traces and encourage merging of short segments, normalized segment length is used as the weight of the merging cost.

Figure 5.9 shows an example of whistle segmentation. Breaking down the whistle curve into a collection of straight line segments has many advantages over the N-point representation [13]: it is compact and perceptually meaningful; it naturally corresponds to parts or features; and it allows us to define and use the cues such as frequency modulation and local variation.

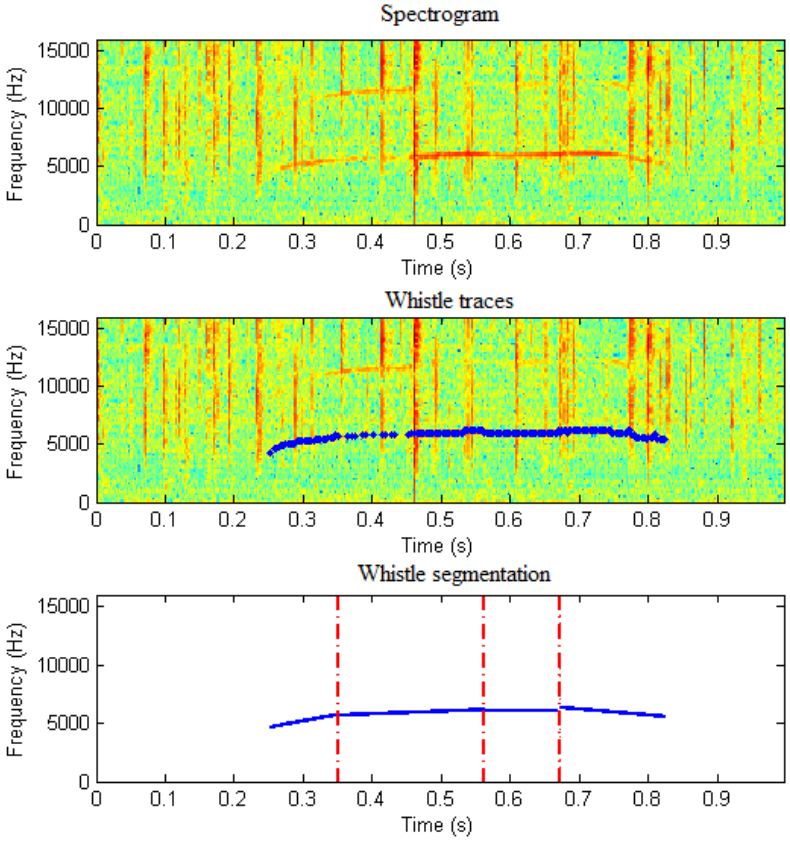


FIGURE 5.9: Example of whistle spectrogram segmentation

5.3.2 Line Segment Distance Measure

The distinctiveness of a whistle curve lies in the inter-relationships between its line aspects such as the length and steepness of each segment. Matching is based on its relationship with other segments or its position in a global view such as the

relative location, relative orientation as well as the adjacency and parallelism of these segments separated by frequency peaks and valleys.

In the segmentation, a smaller K_s possibly overlooks the original shape of whistle traces while a larger K_s is more prone to noisy traces and introduces redundant fragments. In an automatic template matching procedure, the compactness of segmentation on query whistles is not known. An integrated squared perpendicular distance (ISPD) is proposed in this dissertation to compare the similarity of query segments with well segmented template whistles. ISPD integrates the squared point-to-line distance along template segment. This is illustrated in Figure 5.10.

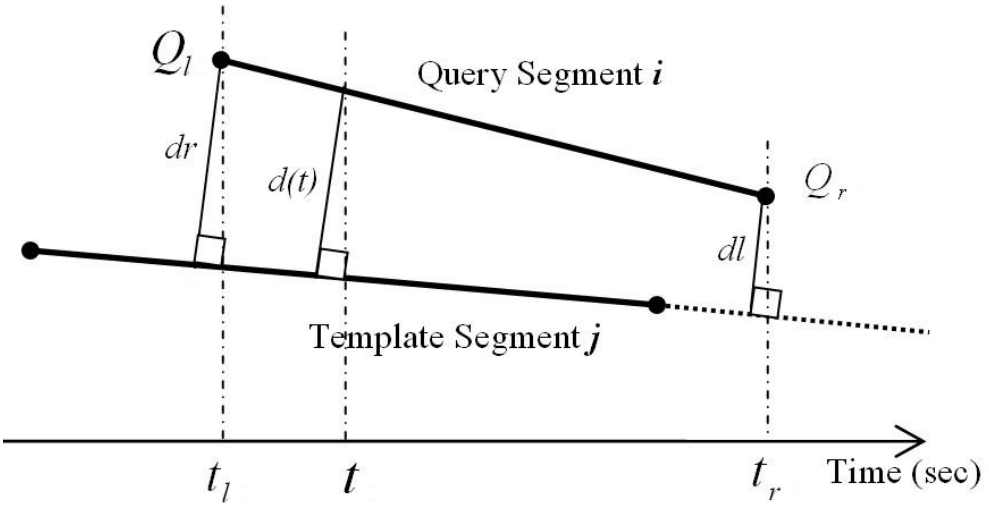


FIGURE 5.10: Illustration of ISPD between segments from query and template whistles

In Figure 5.10, the left and right endpoints of segment i from query whistle are denoted as Q_l and Q_r. These two endpoints occur at time t_l and t_r respectively. When projected to a template segment j or its extension, Q_l and Q_r have signed perpendicular distances d_l and d_r, where the sign depends on their relative side with respect to the template whistle. Any point at time t along the query template

has the perpendicular distance $d(t)$. This $d(t)$ can be expressed by d_l and d_r and their relative time:

$$d(t) = d_l + \frac{(d_r - d_l)}{t_r - t_l} \times (t - t_l). \tag{5.9}$$

The ISPD between segments from query and template whistles is hence the integration of squared $d(t)$ and expressed as

$$\int_{t=t_l}^{t_r} d(t)^2 = \frac{1}{3}(t_r - t_l)(d_r^2 + d_l^2 + d_r d_l). \tag{5.10}$$

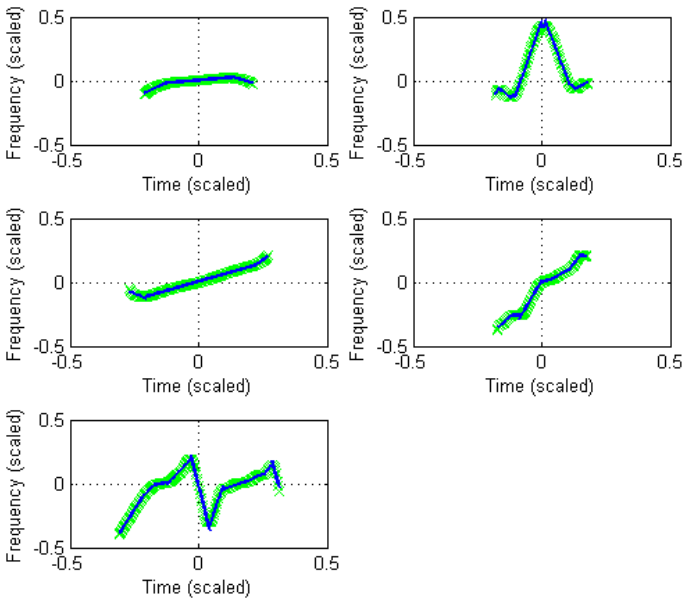
5.3.3 Line Segment Dynamic Time Warping (LSDTW)

The IPSD incorporates the time factor into the local feature distance. The pairwise similarity between whistles adopts DTW for the dynamic warping and also the sequential order of mapping. Each entry in the difference matrix $\mathbf{D}(i, j)$ is the ISPD between the ith query segment and the jth template segment. Since K_s for the query whistle is set to have more than enough segments to represent itself and also have more than the number of segments of its template, the many-to-one matching is used to allow the combination of fragmented segments to correspond to one template segment. The warping is set to look for the minimum distance on 45° and 90° paths on the difference matrix $\mathbf{D}$, as the direction of the decision area decides the matching pattern [20]. Thus this warping constraints ensure that at least 1 query segment is matched to each segment from the template whistle.

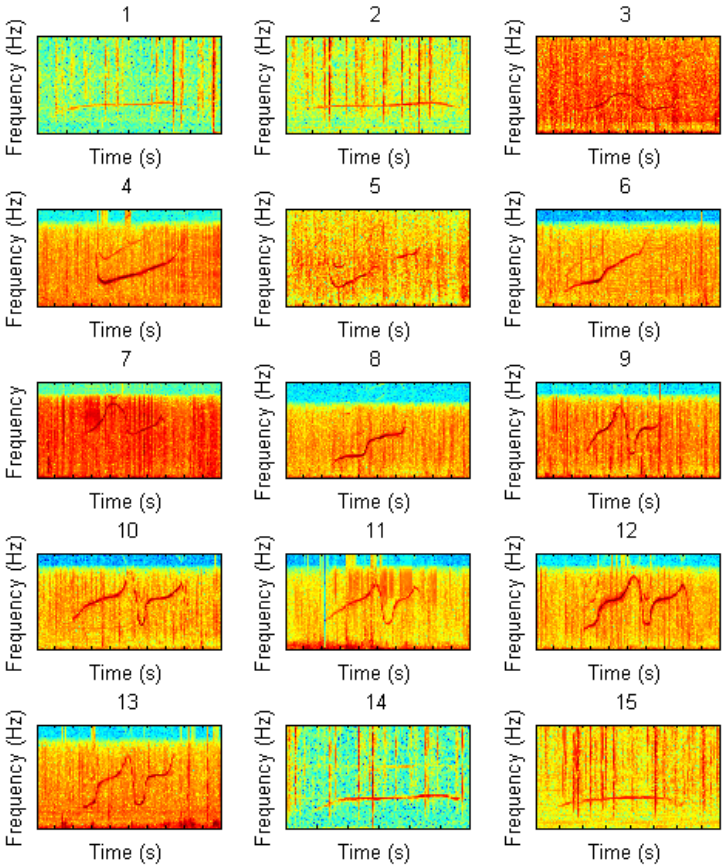
5.3.4 LSDTW for Template Matching

In the experiment, 15 whistles were matched to one of the 5 templates. These templates are well traced and concisely represented by a set of lines. To preserve shape while treating whistle spectra as image curves, frequency and time are normalized to $[0, 1]$ by the same frequency and time ranges. Figure 5.11 shows the segmentation of whistles and their matching results. Query whistles were shifted upwards for easier visualization of matching. We can see that the line segments represent most whistles in a concise and descriptive manner and hence reduce the computation load to the order of segments number (usually $K_s < 20$). The misclassification of Whistle 3 is mainly because Whistle 3 has a smaller range of frequency than Template 2 with the same shape and hence has a very different slope (Figure 5.12). Although scaling by the whistle's respective ranges finds Whistle 3 the correct template, it distorts most whistle shapes by magnifying their small changes.

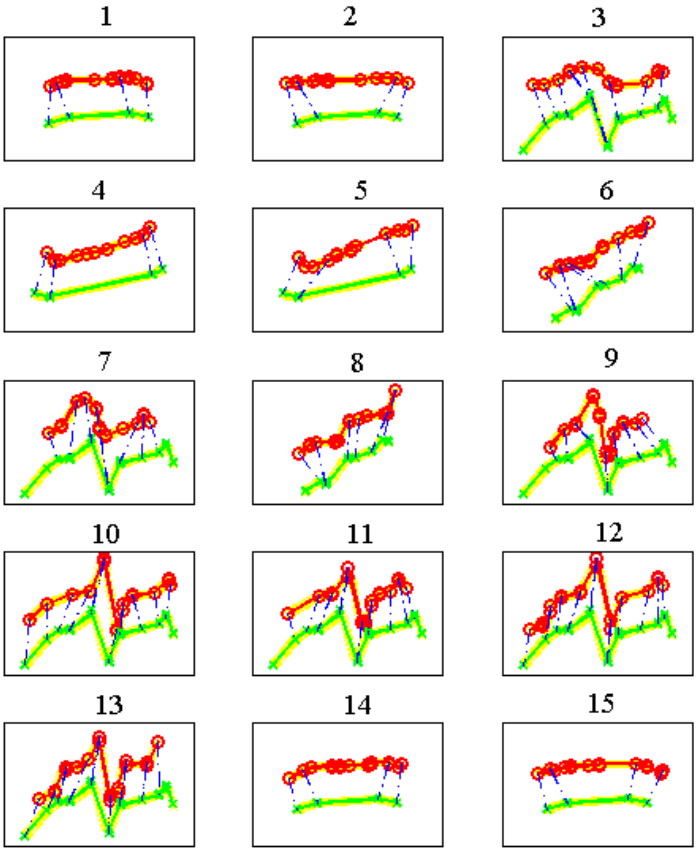
This distance measure is based on the template whistle. It is very likely to be different if the distance measure is based on the query whistle.



(a) Segmented template whistles



(b) Query whistle spectrograms: the frequency ranges from 0 Hz to 20 kHz, the time ticks are marked every 0.1 second



(c) Matching result: templates in green while queries in red

FIGURE 5.11: LSDTW template matching

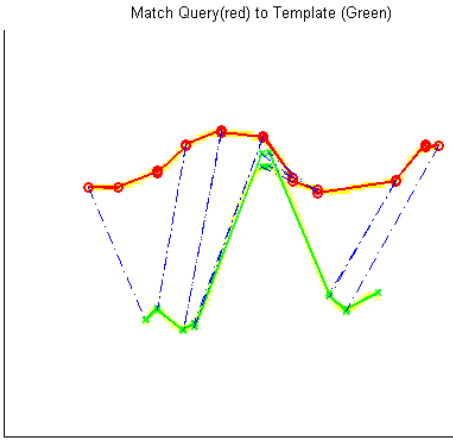


FIGURE 5.12: False matching by LSDTW: Whistle 3 in red is matched to Template 2 in green

5.3.5 LSDTW for Natural Clustering

The LSDTW is also tested for natural clustering in Chapter 7. Before that, the dissimilarities among all test whistles are plotted in Figure 5.13. It is built from pairwise matching where the longer whistles are projected to the short ones. The length here refers to the number of segments. Compared with the DTW on traces in Figure 5.7(b) and the N-point dissimilarity plot in Figure 5.7(a), The dissimilarity matrix in Figure 5.13 has a better perceptual grouping.

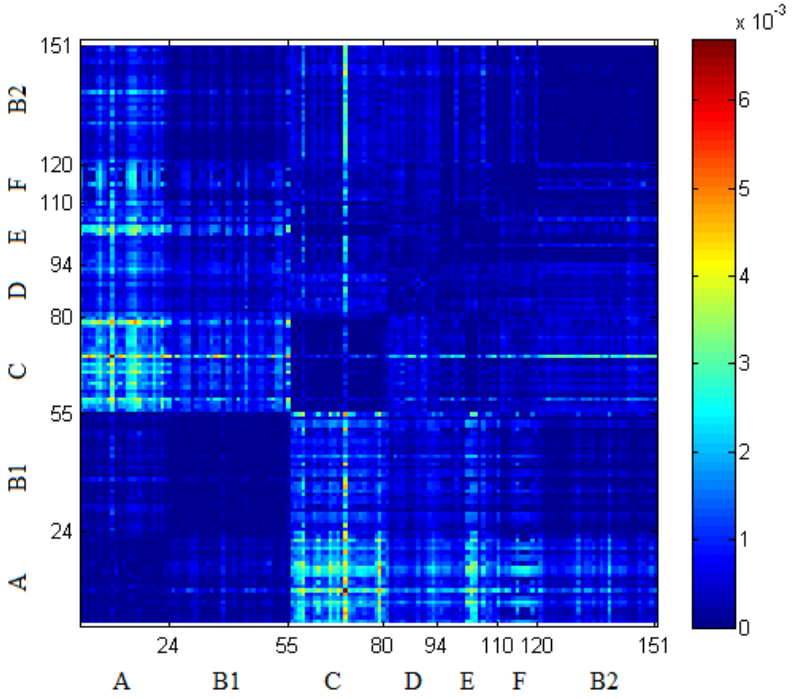


FIGURE 5.13: LSDTW dissimilarity plot

The series of segments has a much smaller size than the N-point feature vector, which reduces the computation of DTW yet keeps more information about the whistle curve. During segmentation, the adjacent segments which are more similar are merged first, and the remained adjacent segments are relatively different from each other. Hence the possibility of over-mapping by DTW is eliminated. However

this is only guaranteed by proper segmentation, where at least one whistle of the pair is represented by segments as compact as possible. The dissimilarity by DTW depends on the segmentation resolution, that is, the number of segments for whistle representation. Another concern regarding LSDTW for natural clustering is: since ISPD measures the absolute frequency difference between segments, the local variation of frequency within the same whistle type is not tolerated. Further improvements will be introduced in the next chapter.

Chapter 6

Pattern Recognition Using Natural Clustering

As discussed in Chapter 3, pre-processing such as scaling and normalization eliminates the information recorded in the absolute frequency of whistles, such as relative frequency variation. On the other hand, as discussed in the previous chapter, simply keeping the absolute frequency values as a feature vector would easily mislead the comparison of whistle shapes. In this chapter, a new feature vector and method with advantages over DTW in finding the dynamic warped matching are proposed to solve these problems.

6.1 Line Segment Curvature

The local features for comparison should contain the tonal changes of vocalization. The curvature of the whistle traces is proposed to characterize the frequency

variation without scaling. A sequence of curvatures at sample points along the whistle curve can form a feature vector. In the case of segmented whistle curves, the curvatures are formed between the adjacent segments of uniform length. It is approximated as the reciprocal of the averaged circle radii. When fitting a circle to the adjacent segments, we have curvature $k = 2/(r_1 + r_2)$, where r_1 and r_2 are defined in Figure 6.1.

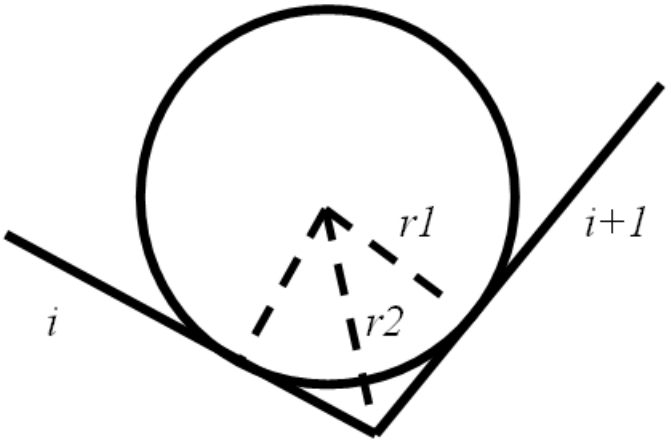


FIGURE 6.1: Curvature on segmented whistle curve: $r1$ and $r2$ are the distances to segments and their intersection from center of the fitting circle

The local feature distance $f(i, j)$ between two whistles is the absolute difference between curvatures from different whistles plus a smoothing factor λ which will be discussed later. This forms the distance matrix $\mathbf{D}$ whose entry $[i, j]$ is denoted as:

$$f(i, j) = |k_i - k_j| + \lambda. \tag{6.1}$$

6.2 Optimal Path by Fast Marching Method

In the conventional dynamic programming for DTW [43] discussed in Chapter 5, the cost matrix is constructed without being weighted by the path warping at each node. Path warping occurs due to one-to-many mapping, or skipping of one or more elements. Furthermore, as a simple sum of all the minimum local feature differences, the matching path could be significantly different if the resolution of local features changes. When the feature vector comprises N samples, the resolution is decided by N. When the feature is the sequence of curvatures from segmented whistle curve, the resolution is decided by the length of the segments (and hence also the number of segments) for whistle curve segmentation. Fast marching [45] is applied in this chapter for a smoother matching path with less sensitivity to the feature resolution compared with DTW. For example, Figure 6.2 shows the matching path under different segmentation resolution. The matching paths are plotted on the distance matrix on the four left image plots and on the cost matrix on the four right contour plots. The first row is the distance and cost matrix constructed by DTW while the second row is by fast marching. The two whistles for comparison are plotted in the last row. With a different segmentation resolution (segment length is changed from 0.02 to 0.04), the matching path by DTW changes significantly while fast marching retains a smooth and relatively consistent matching path.

The total cost of the matching path is the sum of all the differences of the matched pairs along the matching path. However, those matching differences should be weighted by the cost of the nonlinear matching, that is, the cost of

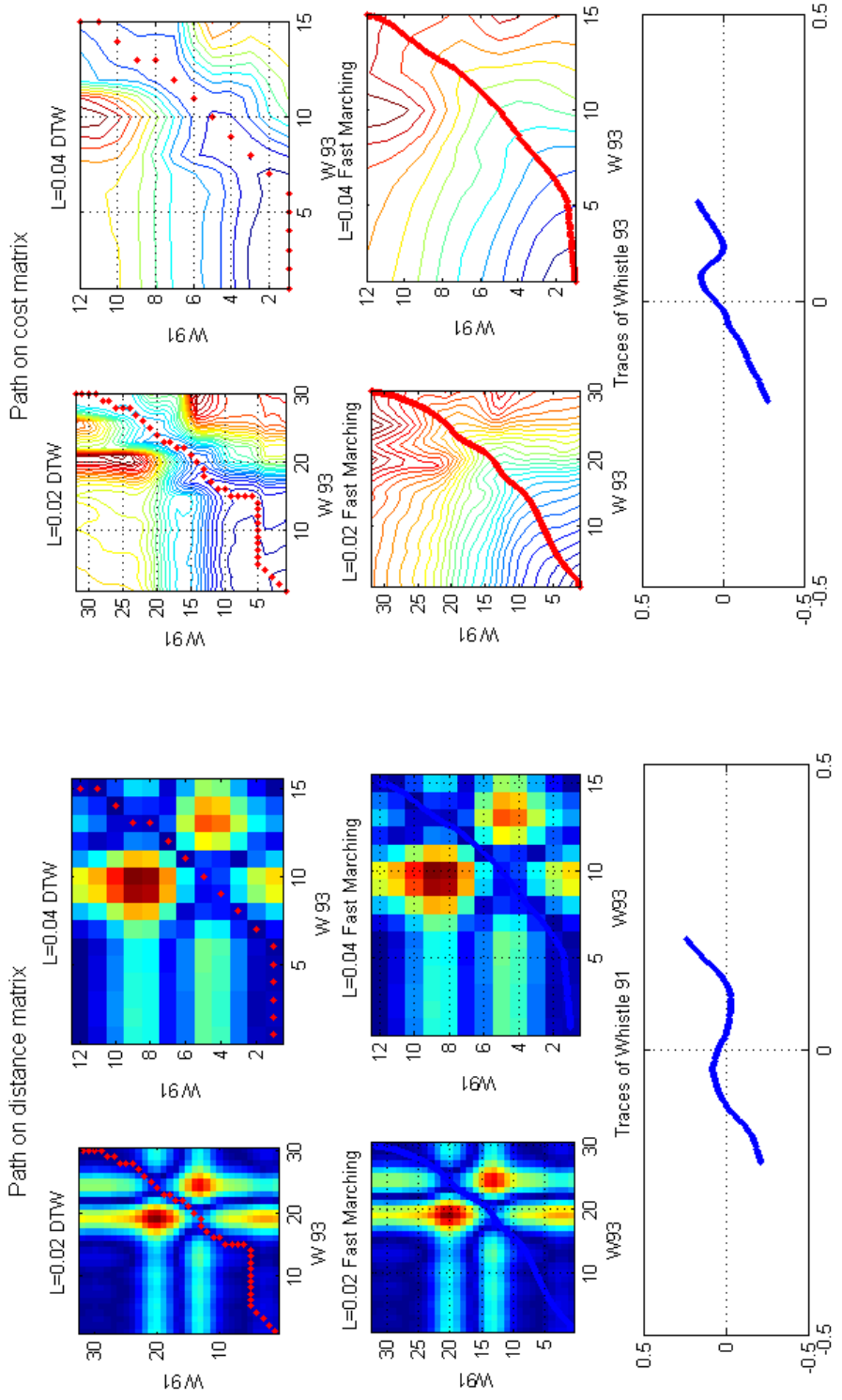


FIGURE 6.2: Comparison between DTW and fast marching method with different feature resolution. Row 1: matching path on distance matrix (left) and cost matrix (right) for DTW. Row 2: fast marching with different segmentation resolution

deforming one sequence to the other (for example, a feature element could be ignored in 27° or 63° mapping). Denoting $[i, j]$ as the matching of the ith and jth local features from two whistles respectively, whistle dissimilarity is the integral of differences along the matching path up to node $[i, j]$. A cost matrix $\mathbf{T}$ stores the minimum cost at every node $\mathbf{T}(i, j)$ for whistles dissimilarity up to pair $[i, j]$. The minimum cost is hence accompanied with an optimal matching path C_p. The entire cost matrix is constructed as:

$$\mathbf{T} = \min \int_{C_p} f(i, j)dc. \tag{6.2}$$

This can be viewed as a surface gradient function when cost matrix $\mathbf{T}$ is plotted as a 2-D surface:

$$|\nabla T(i, j)| = f(i, j). \tag{6.3}$$

Fast marching is an $O(N \log N)$ technique to solve Equation 6.3 [45]. The surface gradient at node $[i, j]$ can be approximated in discrete form:

$$(\max(|\frac{\partial \mathbf{T}(i, j)}{\partial x}|, 0))^2 + (\max(|\frac{\partial \mathbf{T}(i, j)}{\partial y}|, 0))^2 = f(i, j)^2 \tag{6.4}$$

where x and y are the grid lengths along the two dimensions of the cost matrix. Since the curvature comes between segments, the grid lengths can be viewed as the uniform segment length. Hence the integration in Equation 6.2 is along the whistle curves rather than whistle time domain. With a uniform segment length

L, we can re-write Equation 6.4 as:

$$(\max(|\frac{\mathbf{T}(i,j)-T_1}{L}|,0))^2+(\max(\frac{\mathbf{T}(i,j)-T_2}{L}|,0))^2=f(i,j)^2 \tag{6.5}$$

where $T_1=\min(\mathbf{T}(i-1,j),\mathbf{T}(i+1,j))$ and $T_2=\min(\mathbf{T}(i,j-1),\mathbf{T}(i,j+1))$. This is a quadratic problem in solving $\mathbf{T}(i,j)$. Sethian's fast marching method [45] symmetrically computes $\mathbf{T}(i,j)$ in one direction, that is, from smaller values on $\mathbf{T}$ to the larger values. A fronting band consisting of a set of grid points on the cost matrix $\mathbf{T}$ is used to march forward. Every time the minimum node in the fronting band is selected to update $\mathbf{T}$ by the largest possible solution among all its neighbors to Equation 6.5. Hence the fronting band hence is marching at every update. The details are explained in [45]. When cost matrix $\mathbf{T}$ is fully updated, the path is searched backward in steps smaller than the grid length by gradient interpolation. The gradient between nodes is bi-linearly approximated. To avoid over-warping, the search space is confined by 3 types of warping boundaries (hence the 6 white lines in the left top plot of Figure 6.4). There are two cases when the dissimilarity between two whistles is set to infinity:

1. If the gradient along the matching path is too small, a local minimum trap is found, or

2. When the number of marching steps exceeds a maximum threshold, a distorted matching can be detected. The threshold is set as twice of the summed lengths from the two whistles.

6.3 Smoothing Factor

The optimal path is the one that incurs the minimum matching cost. It looks for the smallest gradient along the cost matrix $\mathbf{T}$. The gradient indicates the change of curvature difference with respect to matching length. When searching backwards, the gradient has to be always positive for a monotonically decreasing path; it makes sure that the fast marching method continues even when the local feature difference is zero. Thus from Equation 6.1, λ should always be positive. Otherwise, the path would search backward when λ is negative (the gradient $f(i,j)$ is pointing downwards in a reverse manner), or the path would stop searching when λ is zero (the local difference $f(i,j)$ would be zero and the cost matrix surface is flat). The positiveness of the smoothing factor λ is to drive band marching when the local difference is zero. It also has an effect of smoothing out the solution [18]. Let $F_{x,y}$ denote the element-wise curvature differences along the feature vector at position $[x,y]$, when m and n are the lengths of the two feature vectors. It is automatically defined to be comparable to the magnitude of the curvature difference [18]:

$$\lambda = \frac{1}{mn} \int \int F_{x,y} dx dy. \tag{6.6}$$

In a discrete case, the local curvature difference is $|k_i - k_j|$ at $x = i$ and $y = j$.

The final dissimilarity between whistles subtracts the smoothing factor $|C_p|\lambda$ from the cost of the matching path C_p when $|C_p|$ is the length of the optimal matching path.

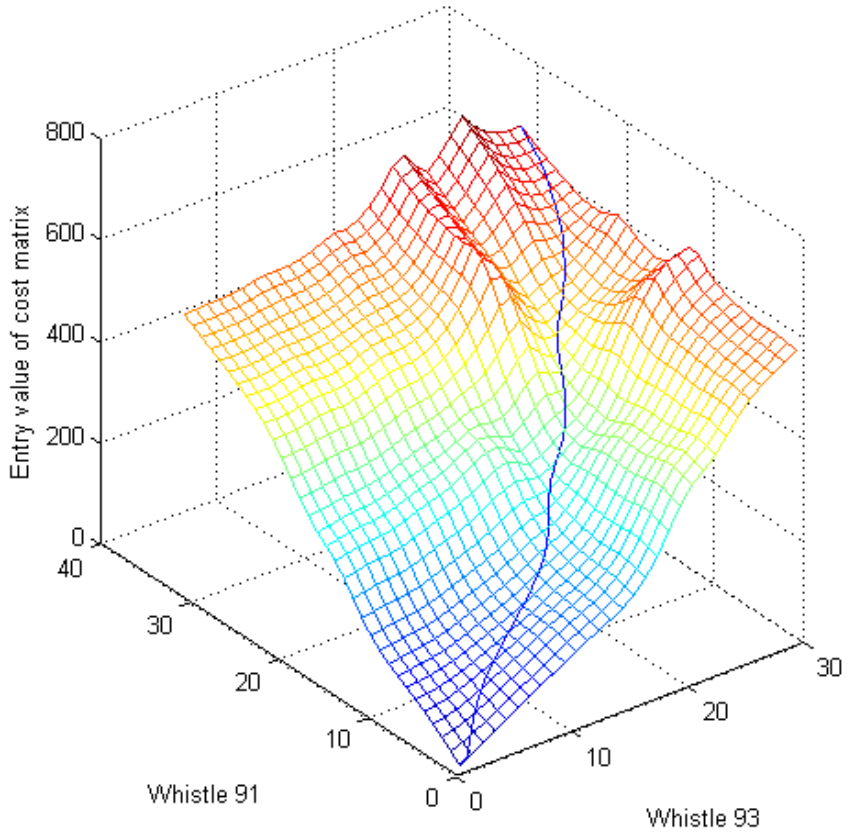


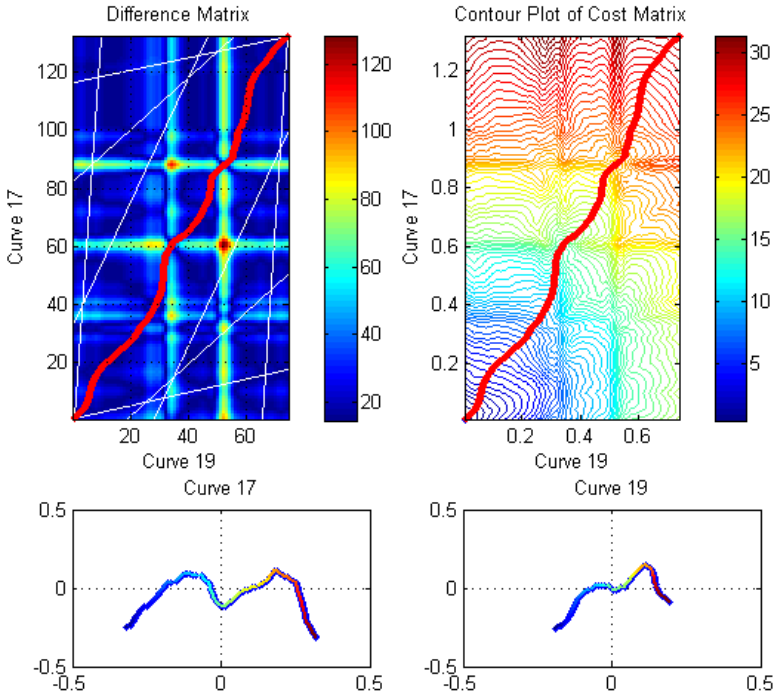
FIGURE 6.3: Path searching along cost matrix with smoothing factor

Figure 6.3 shows an example of path searching on the cost matrix with smoothing factor automatically constructed. Path searching starts from the end pair at entry $[30, 32]$ and looks back till the beginning pair for the smallest gradient. It is clear that the gradient is always positive when searching backwards.

6.4 Examples

Figure 6.4 shows examples of pairwise whistle matching of same and different types respectively. In the second row of each comparison plot, the matching between two whistles is color-coded.

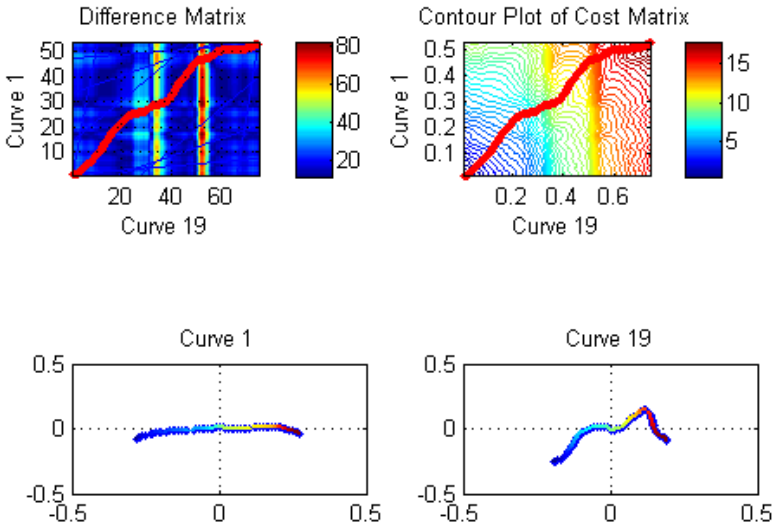
Table 6.1 shows various matching differences of these two pairs. The accumulated difference is the sum of differences from matched element pairs, which is the curvature difference in our case. The average difference is the accumulated difference normalized by the matching path. The matching ratio is the ratio of curve lengths that whistles are matched. At first, Whistle 1 and 19 of different types have a small accumulated difference value; it is because Whistle 1 has a shorter curve. After averaging by the matching path, Whistle 1 displays larger difference with Whistle 19 than Whistle 17 does.



(a) Whistle 17 and 19

TABLE 6.1: Fast marching method on curvatures (Example 1)

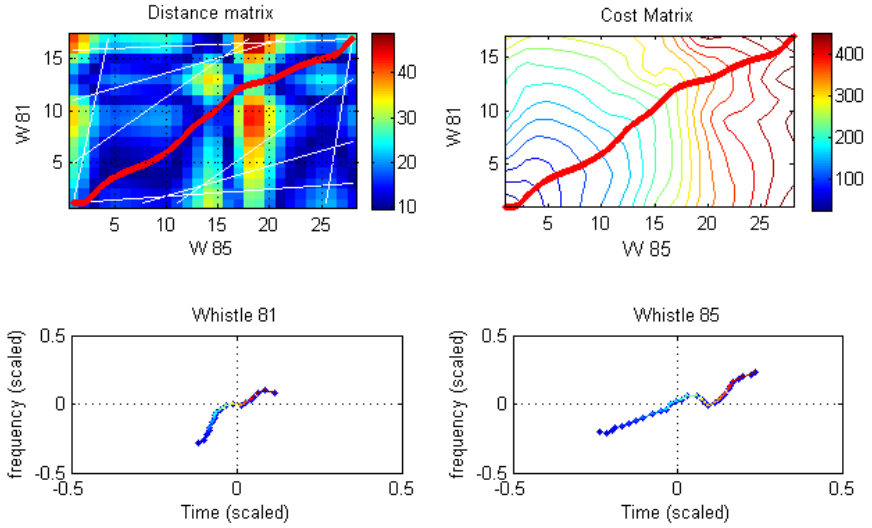
	Whistle 17 vs. 19	**Whistle 1 vs. 19**
Accumulated difference	29.9612	18.2394
Average difference	5.4136	8.4632
Average difference + matching ratio	5.4482	8.5134



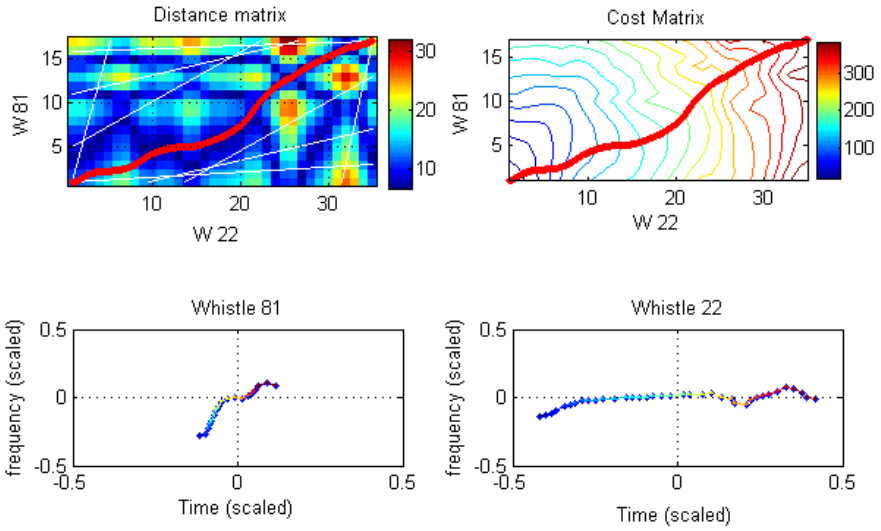
(b) Whistle 1 and 19

FIGURE 6.4: Fast marching method on curvatures (Example 1)

Another example is shown in Figure 6.5. It compares Whistle 81 with Whistle 85 (the same type), 98 and 22 (different types). Interestingly Whistle 81 is more similar to Whistle 22 in terms of the average difference. Whistle 81 and 22 do have similar sequences of curvatures, however their frequency trend is much different. This can be considered as the orientation of the whistle curve, which shows the general trend of whistle frequency. We see that frequency of Whistle 81 is generally increasing while frequency of Whistle 22 is generally flat. The descriptor of curvature sequence is as orientation-free as shape context, yet in a simple way. We need also consider the orientation difference of whistles when using the curvature features. The way that the orientation difference is added will be explored in Section 7.2.



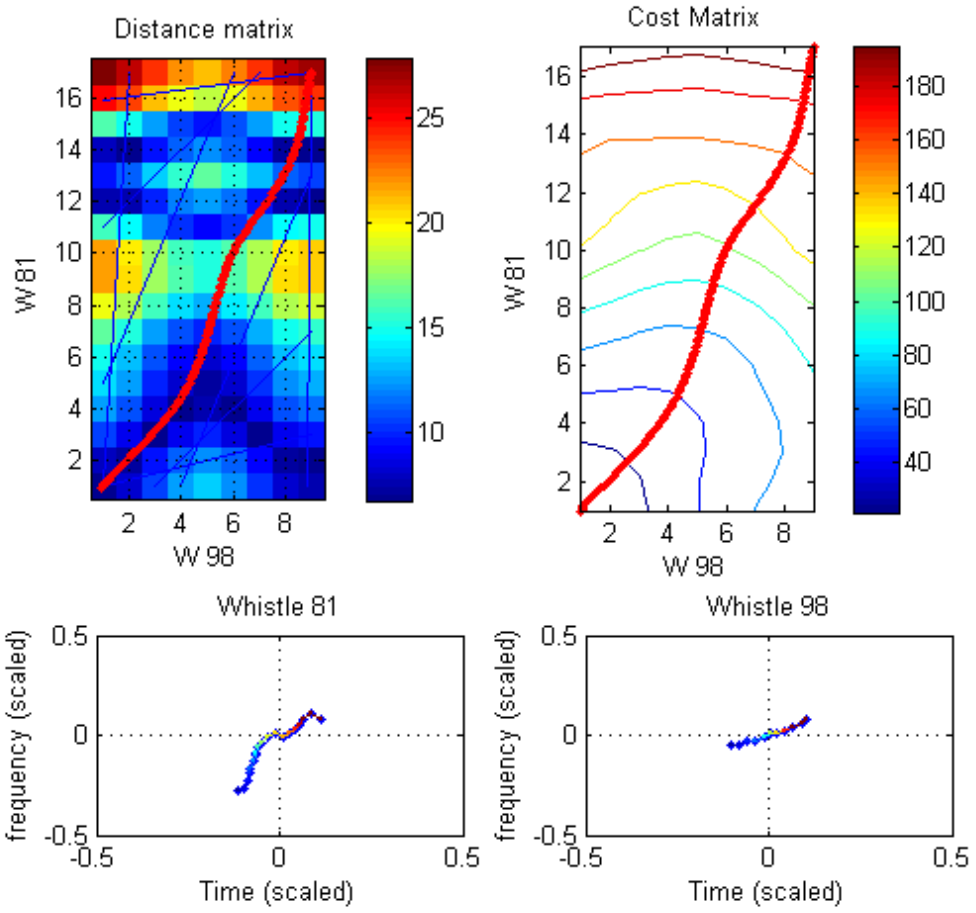
(a) Whistle 81 and 85



(b) Whistle 81 and 22

TABLE 6.2: Fast marching method on curvatures (Example 2)

	Whistle 81 vs. 85	**Whistle 81 vs. 22**	**Whistle 81 vs. 98**
Accumulated difference	23.0885	22.8431	11.4424
Averaged difference	10.6187	7.9621	11.6730
Average difference + matching ratio	10.6462	7.9847	11.6997



(c) Whistle 81 and 98

FIGURE 6.5: Fast marching method on curvatures (Example 2)

Chapter 7

Comparative Results for Clustering

In this chapter, different features and similarity measurements are firstly compared using hierarchical clustering. They are the commonly used N-point feature, the LSDTW proposed in Chapter 5, and the image-based method in Chapter 6. The importance of selecting the correct features and similarity measurement are shown in these progressive results. Secondly, the proposed image-based method using hierarchical clustering is compared with a dolphin whistle classification proposed in [37].

7.1 Hierarchical Clustering

Hierarchical clustering "grows" the largest possible decision tree by merging data into groups (or nodes) through the pairwise similarity among individuals. The

number of nodes can be decided by users. Since every pair is matched with distinct warping and some of them are recognized as "infinitely" dissimilar, the feature space is impossible to construct as the basis for most classification methods. Hierarchical clustering is selected to cluster whistles with pairwise similarity. It is also useful in the initial recognition of whistle patterns by providing the entire hierarchy map about the clustering of a large amount of whistles. Hierarchical clustering also has a distinct advantage that any valid measure of similarity (or distance in opposite) can be used; the observations and the feature space are not necessary. The only disadvantage of hierarchical clustering is the heavy computation which increases with the number of whistles.

Firstly, the hierarchical clustering of the 20-point feature is shown in Figure 7.1. Each node indicates one cluster, whereas the labeled class is in the brackets behind whistle number. Dolphin whistles are plotted as clusters under each node with the starting time aligned at zero. The single-whistle clusters are noted with whistle number. The hierarchical tree shows the relationship between clusters if they are further merged. The dissimilarity matrix in Figure 5.7(a) is based on the Euclidean distance. It is very clear that the N-point feature only relies on the frequency values. Though Type A and B have different patterns, they are likely to be categorized together since they occupy the same frequency band. It is the same logic for Type E and D.

On the other hand, the over-warped matching by DTW on whistle traces has been shown in the dissimilarity matrix in Figure 5.7(b). This is because there are too many redundant frequencies deteriorating the warping. The line segments form

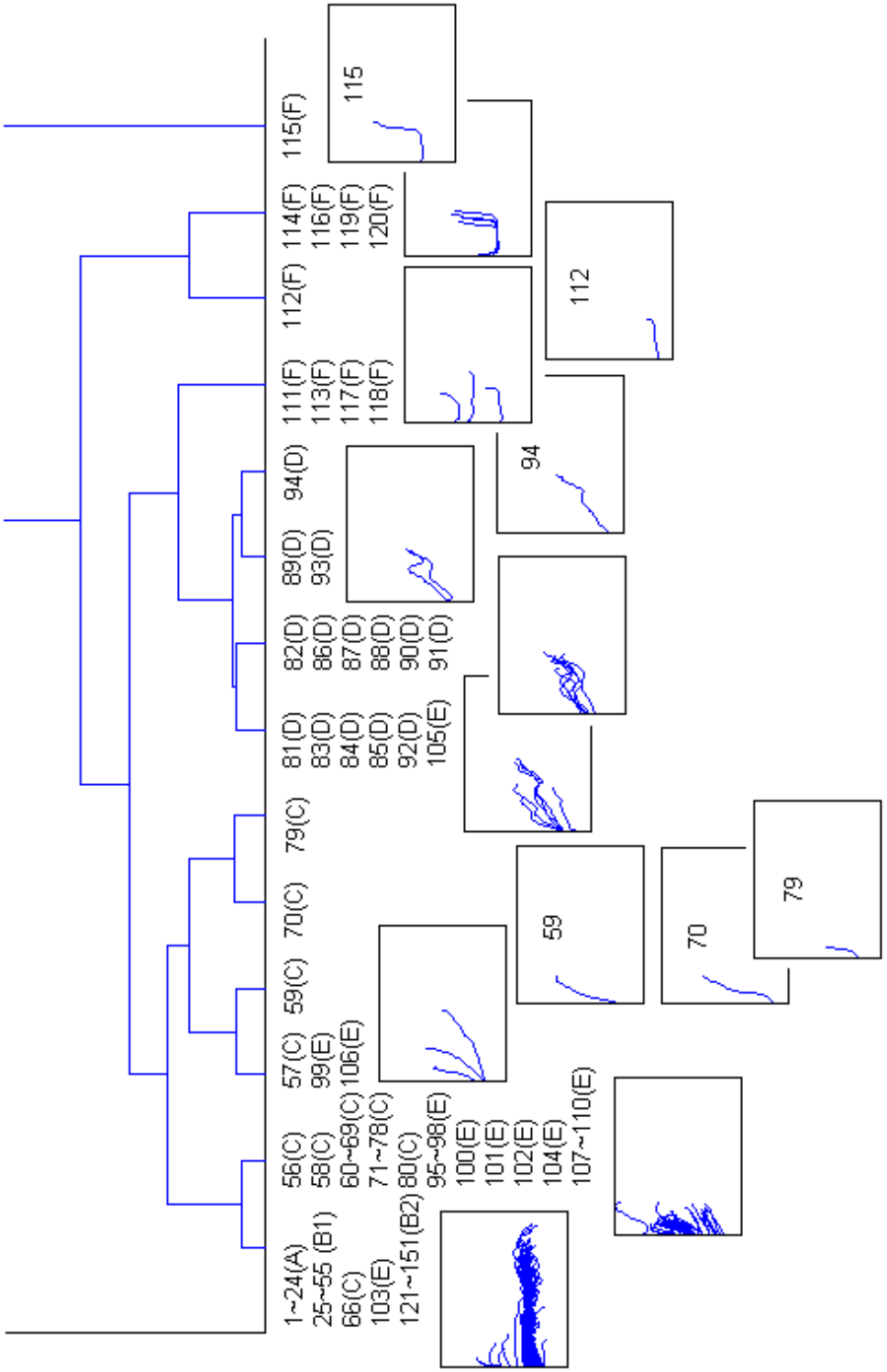


FIGURE 7.1: Hierarchical clustering on N-point with 14 leaf nodes

a more compact and simpler representation. It can utilize the dynamic warping as well for similarity measure.

A hierarchical clustering result using the LSDTW is also shown and analyzed in Table 7.1. In the result, the cluster type is defined by the types that it includes most. The square brackets indicate the misclassified group of whistles by LSDTW. Figure 7.2 shows a relatively better categorization by LSDTW compared with the clustering by the N-point feature vector in Figure 7.1. This can be predicted by comparing the dissimilarity plot of LSDTW in Figure 5.13 with the one by DTW of the N-point feature vector in Figure 5.7.

TABLE 7.1: Natural clustering result analysis of LSDTW

	Hierarchical clustering on LSDTW		
	Description	**Misclassified**	**Error rate** %
A	Mostly clustered	3,4,10,23 and 24[B]	20.8
B	B1 and B2 mixed	55[A]	1.6
C	Mostly clustered	80[D]	4.0
D	Split into 2 sub-groups, one mixed with E	81[C] and 88[F]	14.3
E	mixed with B,C,D	N.A.	100
F	split into 2 sub-groups	111-113[B]	30

It can be seen that whistles of Type E are still misclassified into Types B, C and D due to their similar frequency bands.

7.2 Image-based Method versus K-means

Among the feature vectors for k-means clustering proposed in [37], the N-point feature vector sampled from high order polynomial fit found the best separation

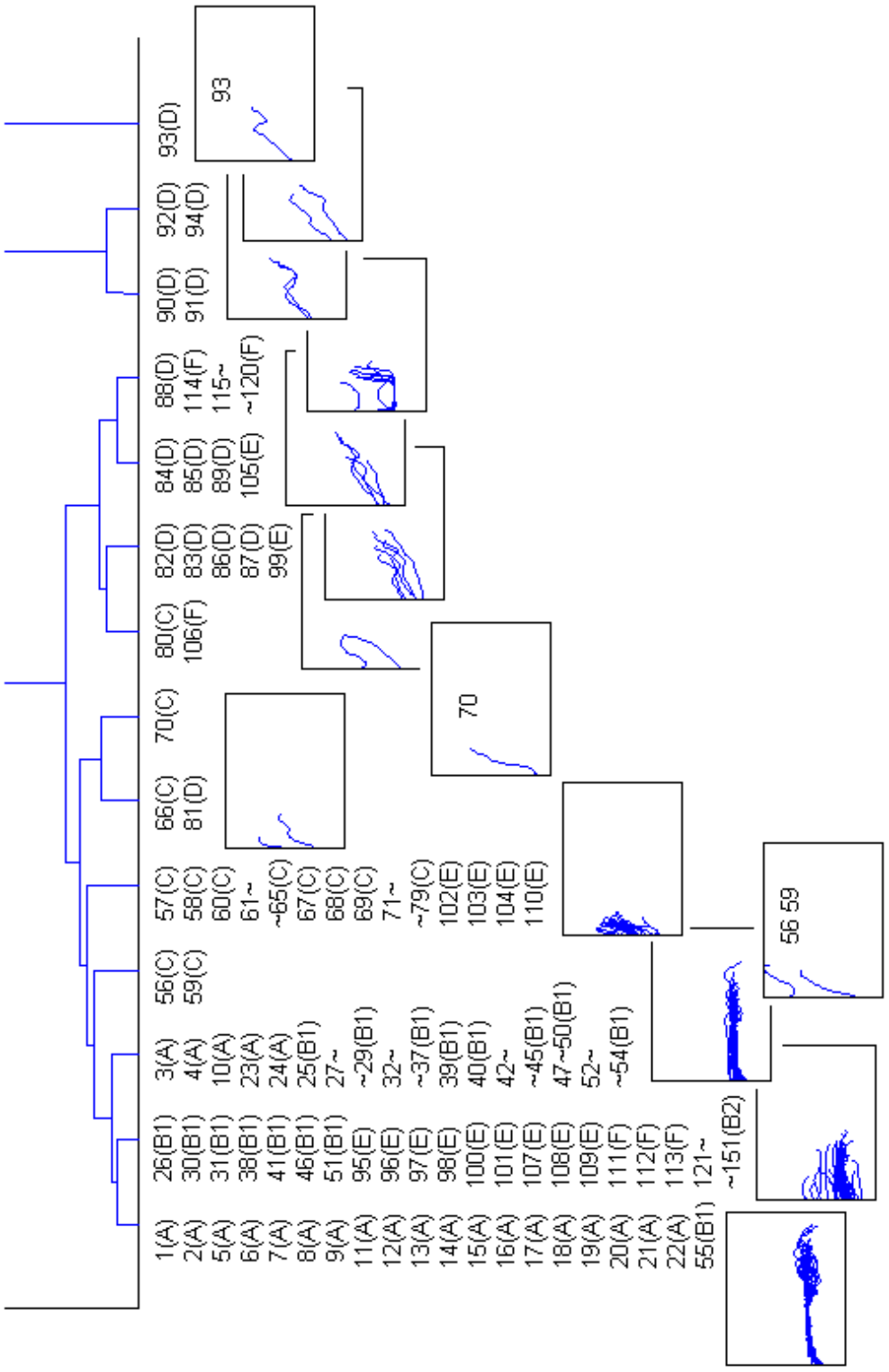


FIGURE 7.2: Hierarchical clustering on LSDTW with 14 leaf nodes

among different types. PCA was applied to reduce feature dimensions. Figure 7.3 shows the trend of normalized sum-of-squared error (SSE) with increasing number of classes and its percentage of reduction from the normalized SSE when all whistles form one class. Figure 7.2 shows the classification at $k = 14$, where the percentage of reduction reaches 90%. Each column represents one of the 14 clusters. The whistle labels by researchers are in the brackets behind whistle number. Figure 7.4 shows the clustering of the whistle contours.

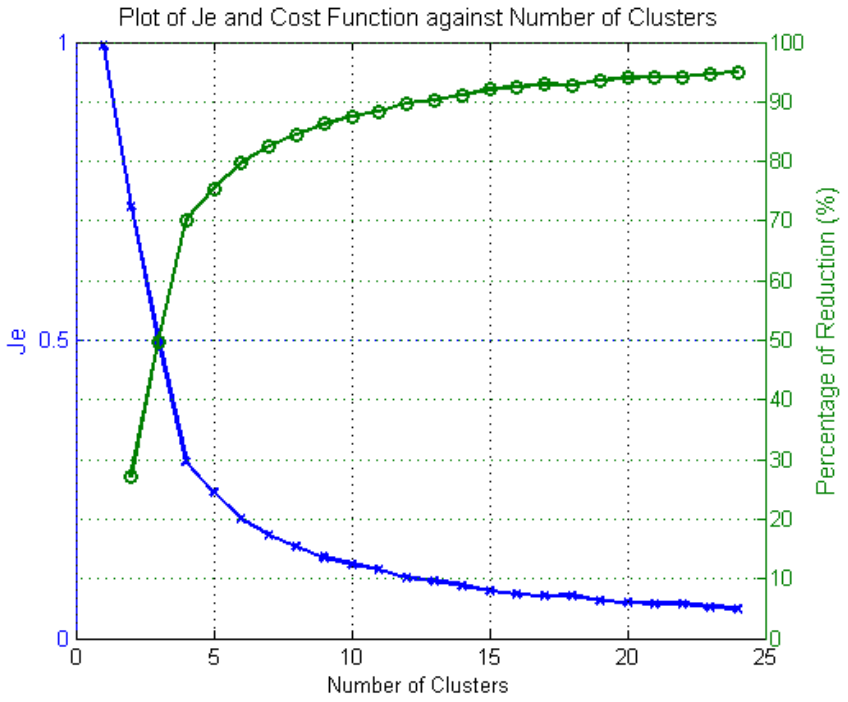


FIGURE 7.3: Normalized SSE and percentage of reduction vs. number of clusters

From the k-means clustering result in Table 7.2 we can see that Type F is well grouped except for Whistle 115, which is strangely grouped with some of Type D. The reason for mixture of Types D and E, and Types C and E is that they occupy similar ranges of frequency distribution. It is the same reason for the mixture in Column **e** for Types A and B.

TABLE 7.2: K-means clustering ($k = 14$) on 20-point feature (after PCA)

Whistle Type	a	b	c	d	e	f	g	h	i	j	k	l	m	n
Whistle ID.	113	111	1	123	2(A)	84(D)	6(A)	29(B1)	26(B1)	40	131	56	57(C)	79(C)
	114	112	5	130	3(A)	85(D)	23(A)	30(B1)	33(B1)	41	144	58	61(C)	82(D)
	116	120	8	133	4(A)	89(D)	27(B1)	32(B1)	38(B1)	(B1)	(B2)	60	62(C)	83(D)
	117	(F)	9	134	7(A)	90(D)	28(B1)	34(B1)	50(B1)			64	63(C)	86(D)
	118		10	136	11(A)	93(E)	31(B1)	35(B1)	51(B1)			65	72(c)	87(D)
	119		12	139	15(A)	94(E)	36(B1)	43(B1)	53(B1)			67	73(C)	91(D)
	(F)		13	140	18(A)	95(E)	37(B1)	45(B1)	121(B2)			68	74(C)	115(F)
			14	141	19(A)	96(E)	44(B1)	46(B1)	122(B2)			69	76(C)	
			16	142	22(A)	97(E)	48(B1)	49(B1)	124(B2)			70	92(D)	
			17	145	24(A)	98(E)	55(B1)	52(B1)	126(B2)			71	100(E)	
			20	147	25(B1)	99(E)		54(B1)	127(B2)			75	101(E)	
			21	149	39(B1)	103(E)		59(C)	128(B2)			77	102(E)	
			88	151	42(B1)	105(E)		66(C)	129(B2)			78	104(E)	
			(A)	(B2)	47(B1)	106(E)		81(D)	132(B2)			80	107(E)	
						108(E)		125(B2)	135(B2)			(C)	109(E)	
								137(B2)	138(B2)				110(E)	
									143(B2)					
									146(B2)					
									148(B2)					
									150(B2)					

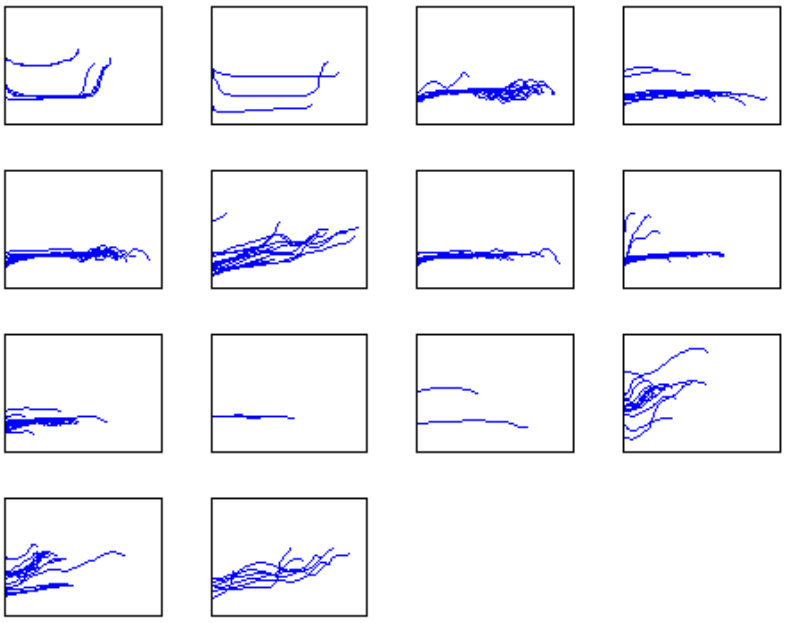


FIGURE 7.4: Plot of whistle contours by k-means into 14 groups

In terms of geometry, the curvature is the amount by which a curve deviates from being flat; it has no sense on the orientation of the curve. In the clustering by warped matching on sequences of curvature, whistle curve orientation θ is added to avoid the rotation invariance. This orientation is defined by the slope of its first order polynomial approximation. The overall whistle dissimilarity is then the weighted sum of these two factors:

$$D(C_1, C_2) = W_d d(C_1, C_2) + W_\theta |\theta_1 - \theta_2|. \tag{7.1}$$

The weight factors are used to combine the influence of both the matching difference and orientation difference. Hence the ratio of these two factors is importance.

Taking $W_d = 1$, the weight factor for orientation difference W_θ should be comparable to the average magnitude of the matching difference. It is hence taken as the average of the matching differences over the dissimilarity matrix. We have:

$$W_\theta = \frac{1}{mn} \sum_{i=1..m, j=1..n} d(C_i, C_j) W_d = 1. \tag{7.2}$$

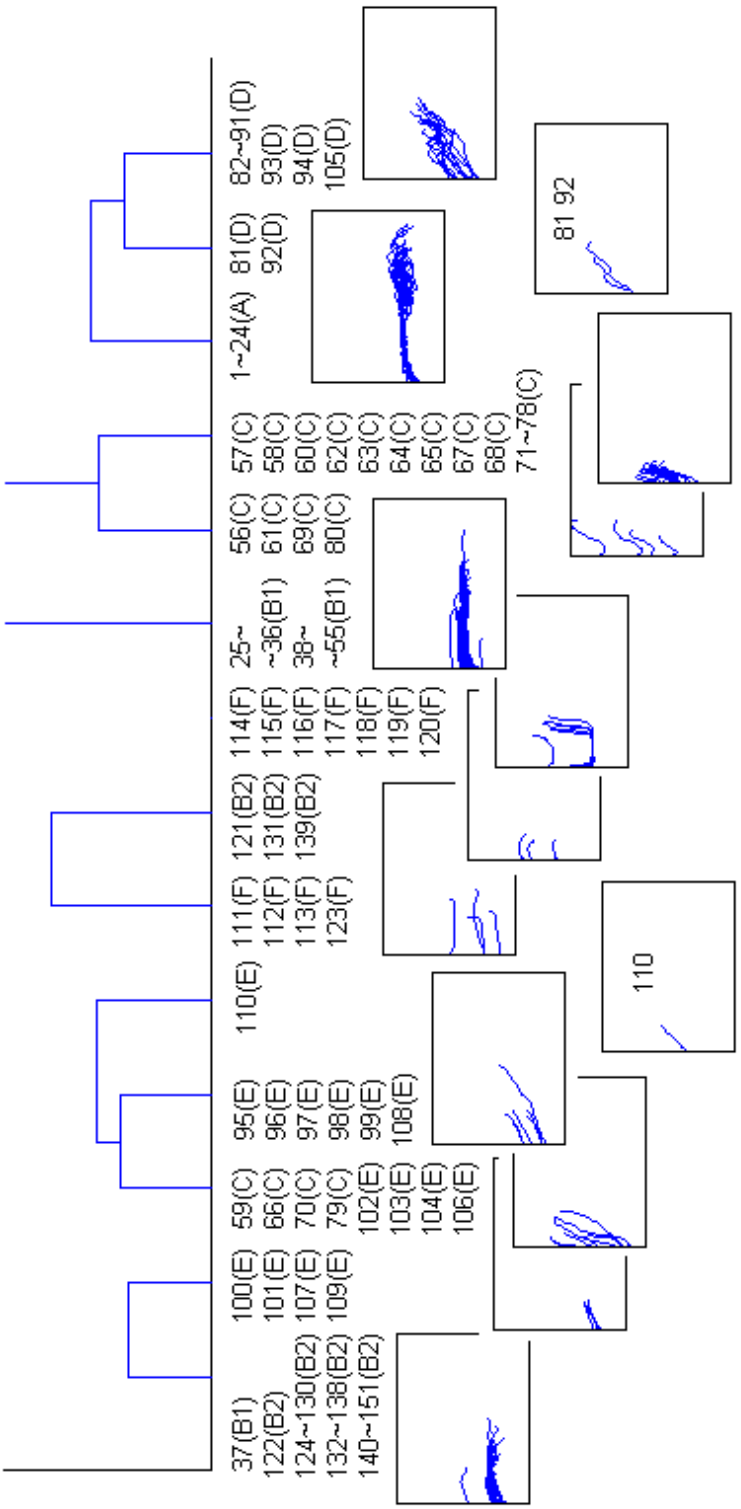
In any of the two cases discussed in Section 6.2, pairwise whistles have infinite dissimilarity value and hence should be excluded. Figure 7.5(a) shows the hierarchical clustering on the weighted sum of the matching difference and orientation difference.

By the weighted sum of matching and orientation differences shown in Figure 7.5(a), Types A, B1, C and D are mostly classified correctly. Whistle 100, 101, 107 and 109 are different in slope from the other whistles of Type E; they are found nearer to Type B2 in terms of both curvature and whistle orientation. Type F is found to have two different shapes - one is formed by Whistle 111 to 113 and the other consists of Whistle 114 to 120. Few whistles of Type B are separated to other types.

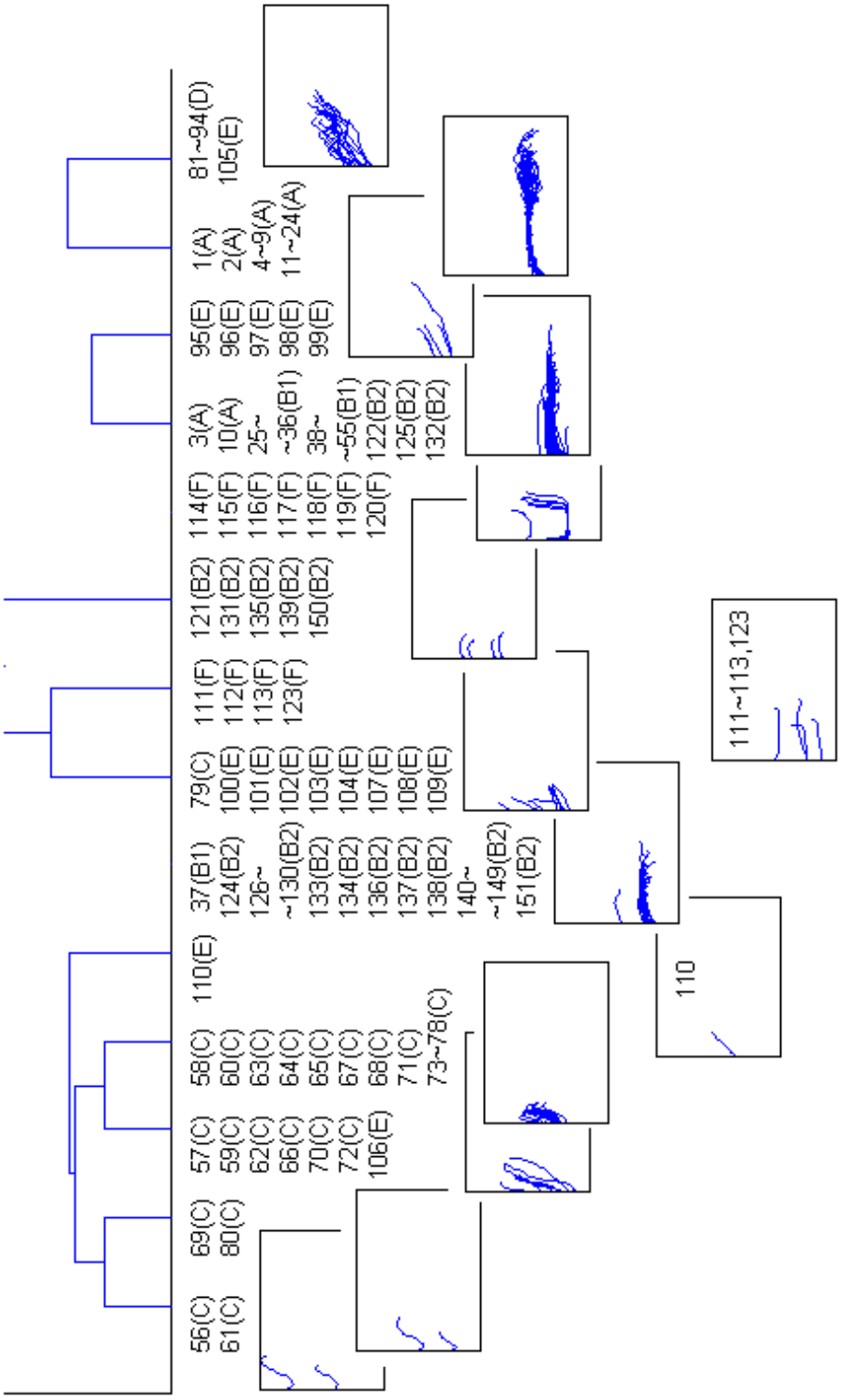
Figure 7.5(b) uses the orientation difference to scale the matching distance between whistles as

$$D(C_1, C_2) = W_s d(C_1, C_2). \tag{7.3}$$

Since the orientations of whistles differ at 90 degree at most, we define the scaling term as



(a) Weighted sum of warped matching and orientation difference



(b) Matching difference scaled by orientation difference

FIGURE 7.5: Hierarchical clustering on image-based method with 14 leaf nodes

$$W_s = \tan(|\theta_1 - \theta_2|) + 1 \tag{7.4}$$

which ranges from one to infinity. There is no more mixture of Types C and E, which occurs in Figure 7.5(a). The clustering result has no longer a mixture of Types C and E and hence is better than the one by the weighted sum of the matching and orientation differences.

To find the best clustering result by the proposed image-based method, we adjust the length of segment to $L = 0.02$. This is the maximum segment length required to represent the shortest whistle compactly in the data set. We continue to use the orientation factor as a scaling term since it shows more promising result in Figure 7.5(b) compared with Figure 7.5(a). In this case we have the result in Figure 7.6.

We compare this with the k-means result in Table 7.3. For each type of whistle, if some whistles are misclassified or grouped with other types, we define its class by voting of the whistle classified to this class. If the labeled whistles in one type are somehow equally split, we will discuss and evaluate it. Since there are more class numbers than the pre-defined classes, sub-classes not mixing with other types are also accepted. The square brackets behind the misclassified whistles indicate the results by this classification method.

With both k-means method and the image-based method, Type F has clearly 2 sub-groups according to the beginning and ending frequencies. Whistles of Type E are totally split and mixed with other types of whistles in the k-means using

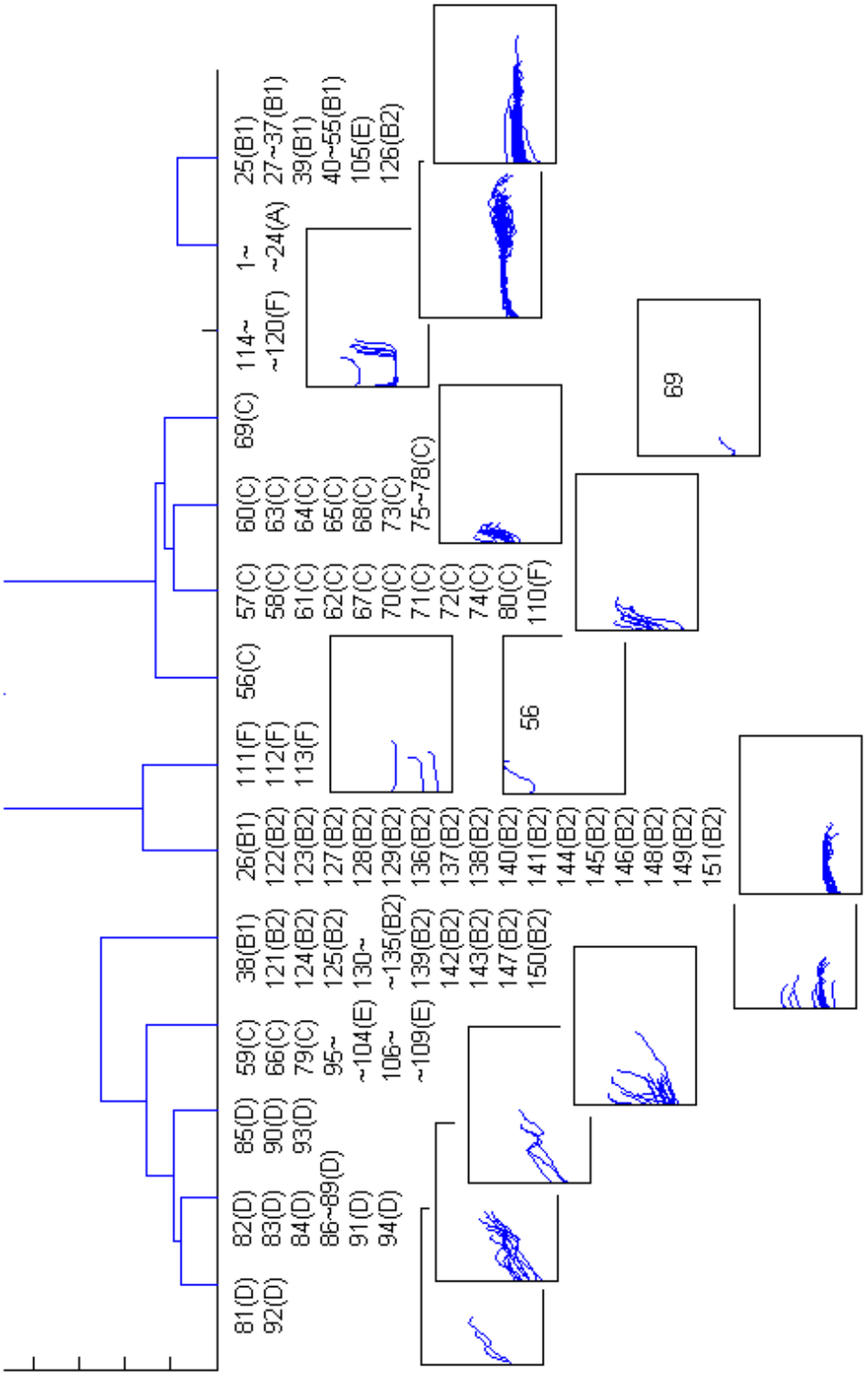


FIGURE 7.6: Best result: hierarchical clustering on image-based method with 14 leaf nodes. The segment length is adjusted and orientation difference is used as scaling factor.

TABLE 7.3: Natural clustering result analysis of k-means and fast marching method (FMM)

	k-**means on** N **points**			**Fast marching on segment curvature**		
	Description	**Misclassified**	**Error rate** %	**Description**	**Misclassified**	**Error rate** %
A	2 subgroups	6,23[B]	8.4	All correct	None	0.0
B	B1,B2 mixed	25,39,42,47[A]	6.45	B1 mostly correct	26 and 38[B2]	0.0
				B2 has 2 subgroups	126[B1]	
C	3 subgroups,8 whistles mixed with Type E	59,66[B],79[D]	44.0	4 subgroups in one	59, 66 and 79[E]	12.0
D	3 subgroups, 6 mixed with Type E	81[B],88[A] and 92[C/E]	64.3	3 subgroups in one	None	0.0
E	totally mixed with Type C and D	N.A.	100	Mostly correct	110[C] and 105[B]	12.5
F	2 subgroups	115[D]	10.0	2 subgroups	None	0.0

the N-point feature. This is mainly because of the scaling on frequency domain such that Type E is stretched to span similar frequencies with Type C and D. Type C and D are also affected by this problem. Type B is fine as the groups of B1 and B2, except for a few, are misclassified as Type A. Again Whistle 23 has small frequency change and is nearer to Type B of constant frequency by N-point distance.

The fast marching on segment curvature shows significantly better result. Type A, B, and D are all correctly classified. The sub-class of Type B are also nicely divided into Type B1 and B2. The misclassified whistles are mainly from Types C and E, we can see some ambiguities. For example, Whistle 79 does not have the flat frequencies in the beginning and end as Type C. It is closer to Whistle 106 in Type E. It is demonstrated that with proper segmentation length, whistle clustering by the image-based method can have fairly good agreement with human classification. It also helps researchers to find the possible sub-types and exceptions.

Chapter 8

Conclusion and Future Work

This thesis presents a systematic analysis of dolphin whistle classification. In this thesis, three steps in classification of dolphin whistles were summarized and explored: feature selection, similarity measure and classification methods. The selection of whistle features and their similarity measure are important in characterizing whistles and pairwise similarity. They also affect the classification at the third step. Some commonly used features and similarity measurements were reviewed first, followed by the classification methods. It was found that when whistles are to be compared, the feature sequence might not be linearly mapped for the same type. The feature space and their Euclidean distance for similarity measures are not optimal for whistle matching. In supervised learning, the selection of training whistles is also a critical factor. In unsupervised learning, the inter-class and intra-class variations are unknown, which presents difficulties in deciding boundaries of whistle types.

The methods proposed in this thesis use the idea of dynamic warping in speech recognition. DTW was modified to nonlinearly map whistles with expected tracing noise. However, whistles are easily over-warped by DTW matching due to the information redundancy in traces. A series of segments was initially attempted to emulate human observations on dolphin whistles. They are the compact encoders for the whistle curve. An integrated squared perpendicular distance was introduced to record the relative difference between whistle segments. However, with more inter- and intra-class variations, both the N-point feature and segment sequence are limited by the frequency values. By considering the curvatures of segmented whistle curves, whistles with different scales can be classified according to their relative frequency changes. Fast marching was adopted for smoother matching with a sub-resolution accuracy to tolerate difference in the segmentation resolution. It also prevents over-warping by counting the warping cost and providing matching boundaries. This treats the whistle curve as image curves for matching and is hence named as *image-based method*. In a contrast to the shape context, this image-based method conserves the sequential mapping during nonlinear warping. The whistle orientation representing the overall tonal trend is also included adaptively for whistle dissimilarity. With this pairwise similarity, dolphin whistles of different lengths and different frequencies can be stretched and warped appropriately for comparison. The hierarchical clustering has successfully found the whistle patterns and explored the level of clustering among the set of 151 whistles.

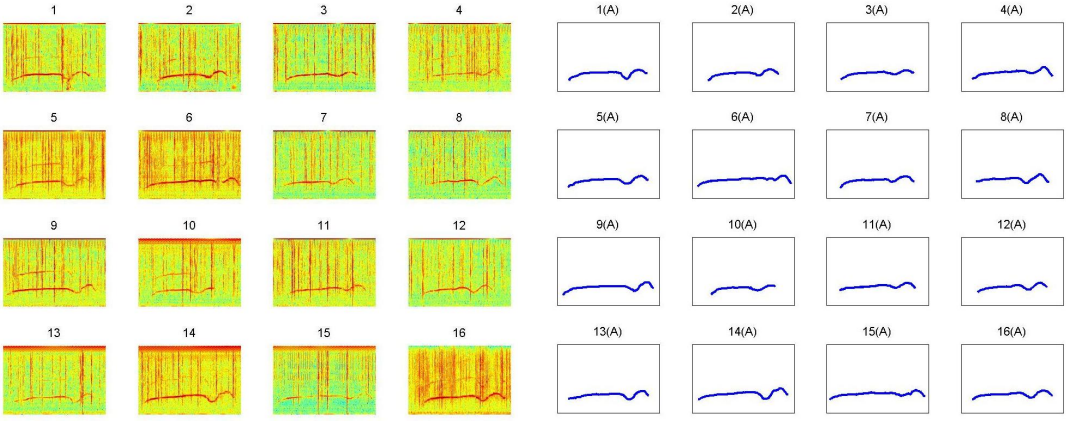
In terms of computation, dolphin whistles represented by a series of segments

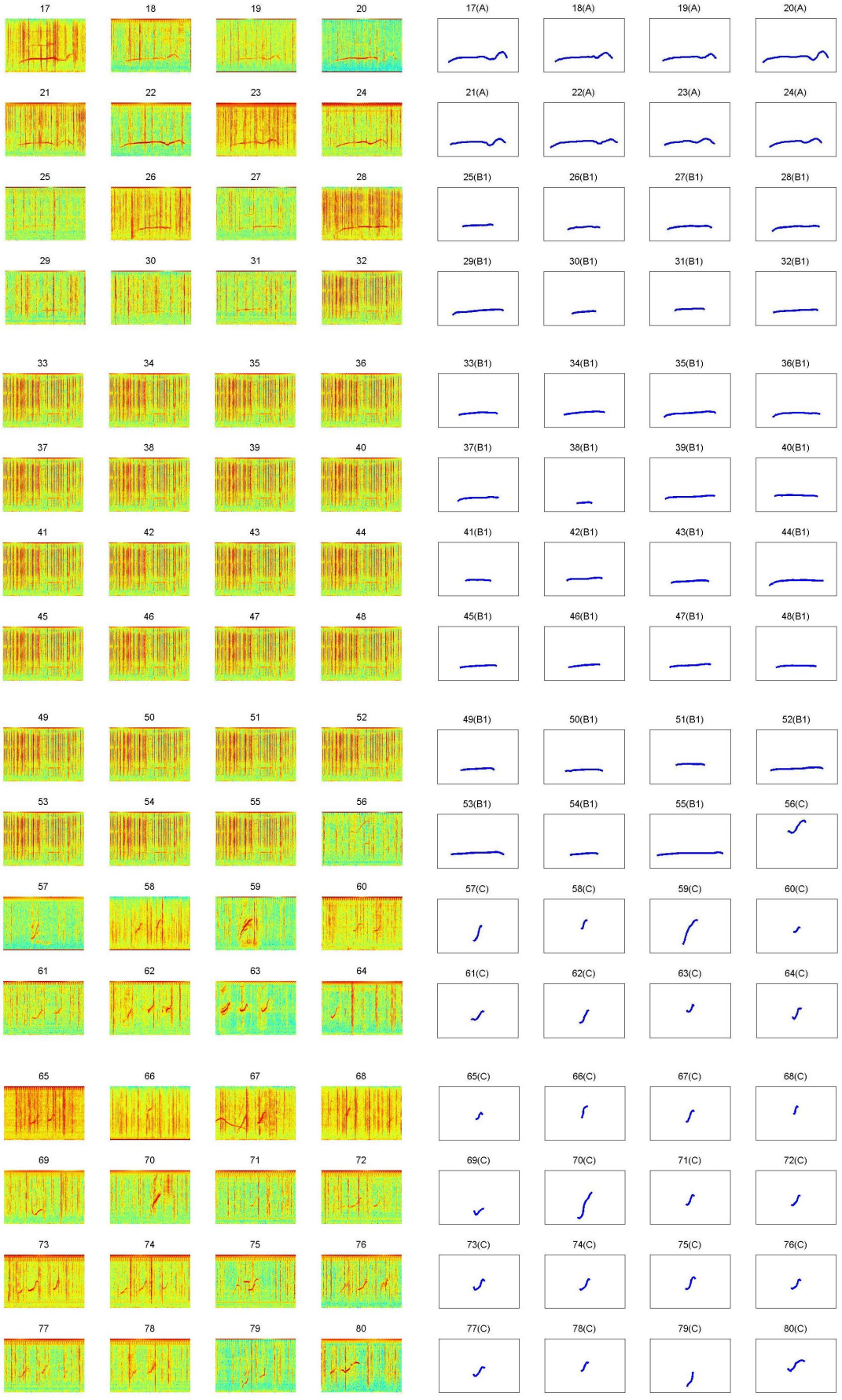
gives a shorter feature vector yet keeps more information and N-point feature (although only three components remain after PCA). Hierarchical clustering still incurs high computation depending on number of dolphin whistles. A more efficient classification method is needed. A user-friendly software in visualizing, extracting and classifying dolphin whistles would need to be set up for real-time application. Together with the whistle detection and tracing of the first stage, this whistle classification can be used to automatically recognize whistle patterns in a way that agrees with human criteria. This is very useful when dolphin researchers are training dolphins, and exploring dolphin behaviors.

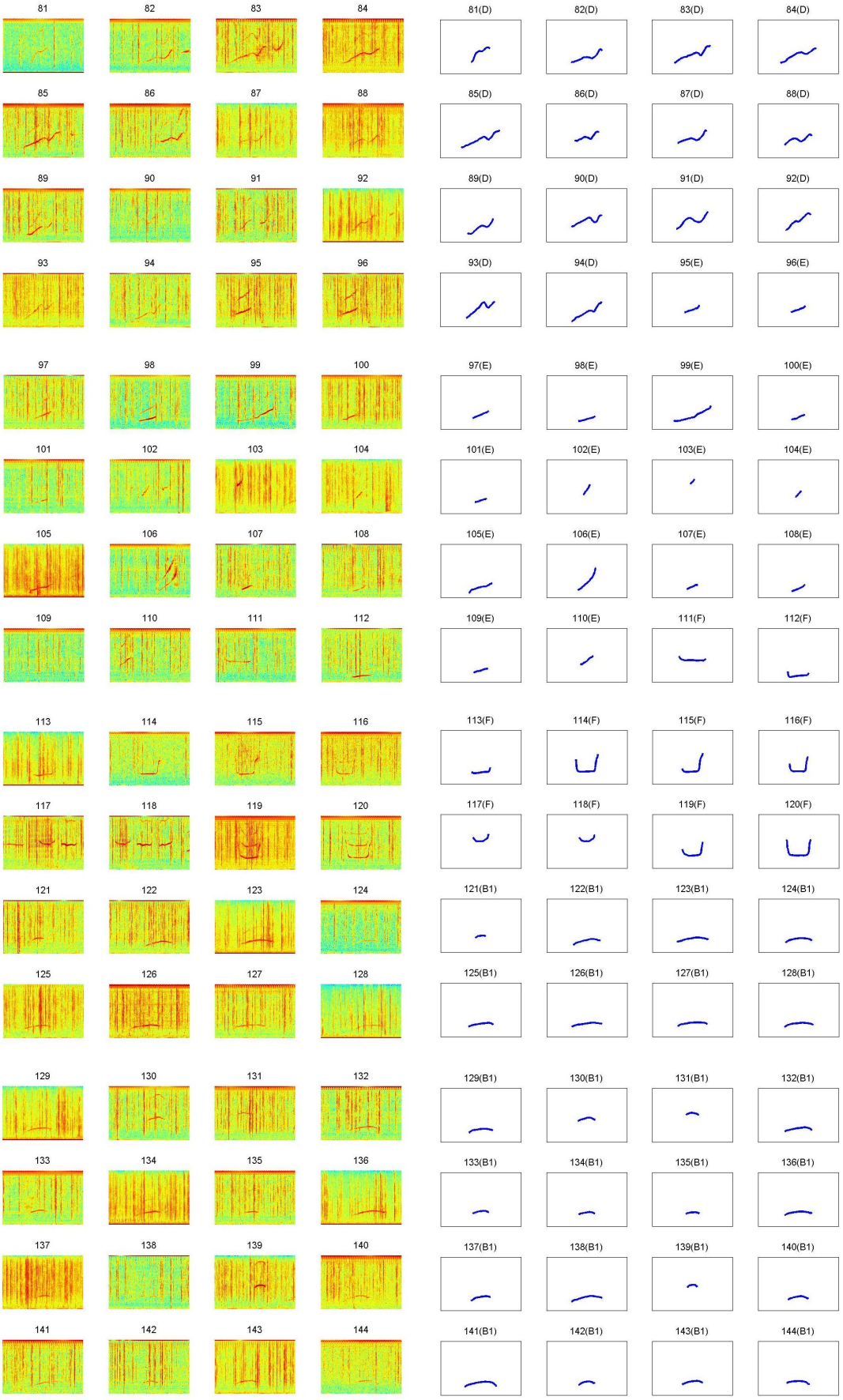
Appendix A

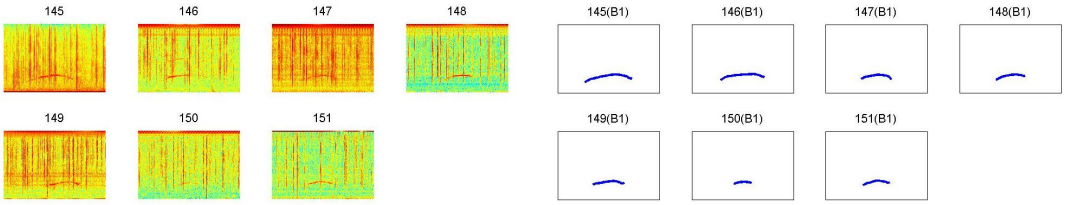
Whistle Recordings and Traces

Below are the dolphin whistles extracted from underwater recordings of Indo-Pacific humpback dolphins (*Sousa Chinesis*) at the Dolphin Lagoon Sentosa, Singapore. On the left shows the original spectrogram after short-time Fourier transform (STFT). On the right is the time-frequency presentation (TFR) by whistle traces (centered).




17
18
19
20
17(A)
18(A)
19(A)
20(A)
21
22
23
24
21(A)
22(A)
23(A)
24(A)
25
26
27
28
25(B1)
26(B1)
27(B1)
28(B1)
29
30
31
32
29(B1)
30(B1)
31(B1)
32(B1)
33
34
35
36
33(B1)
34(B1)
35(B1)
36(B1)
37
38
39
40
37(B1)
38(B1)
39(B1)
40(B1)
41
42
43
44
41(B1)
42(B1)
43(B1)
44(B1)
45
46
47
48
45(B1)
46(B1)
47(B1)
48(B1)
49
50
51
52
49(B1)
50(B1)
51(B1)
52(B1)
53
54
55
56
53(B1)
54(B1)
55(B1)
56(C)
57
58
59
60
57(C)
58(C)
59(C)
60(C)
61
62
63
64
61(C)
62(C)
63(C)
64(C)
65
66
67
68
65(C)
66(C)
67(C)
68(C)
69
70
71
72
69(C)
70(C)
71(C)
72(C)
73
74
75
76
73(C)
74(C)
75(C)
76(C)
77
78
79
80
77(C)
78(C)
79(C)
80(C)


81
82
83
84
81(D)
82(D)
83(D)
84(D)
85
86
87
88
85(D)
86(D)
87(D)
88(D)
89
90
91
92
89(D)
90(D)
91(D)
92(D)
93
94
95
96
93(D)
94(D)
95(E)
96(E)
97
98
99
100
97(E)
98(E)
99(E)
100(E)
101
102
103
104
101(E)
102(E)
103(E)
104(E)
105
106
107
108
105(E)
106(E)
107(E)
108(E)
109
110
111
112
109(E)
110(E)
111(F)
112(F)
113
114
115
116
113(F)
114(F)
115(F)
116(F)
117
118
119
120
117(F)
118(F)
119(F)
120(F)
121
122
123
124
121(B1)
122(B1)
123(B1)
124(B1)
125
126
127
128
125(B1)
126(B1)
127(B1)
128(B1)
129
130
131
132
129(B1)
130(B1)
131(B1)
132(B1)
133
134
135
136
133(B1)
134(B1)
135(B1)
136(B1)
137
138
139
140
137(B1)
138(B1)
139(B1)
140(B1)
141
142
143
144
141(B1)
142(B1)
143(B1)
144(B1)


145
146
147
148
145(B1)
146(B1)
147(B1)
148(B1)
149
150
151
149(B1)
150(B1)
151(B1)

Appendix B

Classification Results of Whistle Data with Different Principal Components (PCs)

Table B.1 compares the supervised classification methods using different PCA reduction, namely, three principal components (PCs), eight PCs and the full 20-point feature for Chapter 4. These methods include: linear discriminant analysis (LDA), diag-linear discriminant analysis (DLDA), quadratic discriminant analysis (QDA), diag-quadratic discriminant analysis (DQLA), Mahalanobis distance, k nearest neighbors (KNN), probabilistic neural network (PNN). The *N.A.* indicates the estimated covariance matrix from training data is not positive definite.

TABLE B.1: Supervised classification (7 types) on different number of principal components (PCs): e_R is the re-substitution error; e_C is the classification error

Method	3 **PCs**		8 **PCs**		20 **points**	
	e_R	e_C	e_R	e_C	e_R	e_C
LDA	21.65	24.56	8.11	19.30	N.A.	N.A.
DLDA(Naive Bayes)	21.62	31.93	10.81	26.32	16.22	28.95
QDA	2.70	34	N.A.	N.A.	N.A.	N.A.
DQDA(Naive Bayes)	13.51	27.19	2.70	22.81	10.81	22.81
Mahalanobis	13.51	35.09	N.A.	N.A.	N.A.	N.A.
KNN($k = 1$)	0	22.81	0	21.93	0	15.79
PNN	0	20.18	0	22.81	0	15.79

It is observed that with more PCs in the feature vector, the non-positive-definite covariance matrix is more often estimated. In naive Bayesian classification, the eight PCs give the lowest classification error and re-substitution error. KNN and PNN have zero re-substitution error; the classification error decreases with more features in the feature vector.

In natural clustering, the k-means clustering on eight PCs and the full 20-point feature is listed in Table B.2 and Table B.3 to compare with the three PCs in Table 4.8. It is seen that the clustering by three PCs is worse than the one by eight PCs and full 20-point feature (the latter two give the same result). Hence the reduction in features by PCA does reduce information in clustering.

The competitive learning and SOM clustering by eight PCs and full 20-point feature are also shown below. Compared with the three PCs in Table 4.11 and Table 4.12. It is pretty difficult to see the effect of the feature reduction by PCA. Taking competitive learning for example, with more PCs (from three to eight PCs, and to full 20-point feature), Type A is better clustered and Type D is less mixed with Type F; however, clustering of Type C and E is getting worse. It becomes even more difficult to compare the clustering by SOM.

TABLE B.2: K-means clustering ($k = 7$): 8 PCs

Whistle Type	**a**	**b**	**c**	**d**	**e**	**f**	**g**
Whistle ID.	6(A)	1 ∼ 3(A)	44 ∼ 46(B1)	56 ∼ 58(C)	4(A)	23(A)	111 ∼ 114(F)
	25 ∼ 33(B1)	7(A)	59(C)	60 ∼ 65(C)	5(A)	34(B1)	116 ∼ 120(F)
	35(B1)	8(A)	66(C)	67 ∼ 78(C)	9(A)	36(B1)	
	37 ∼ 43(B1)	10(A)	79(C)	80(C)	11(A)	50(B1)	
	47 ∼ 49(B1)	12(A)	81 ∼ 94(D)		13(A)	51(B1)	
	52(B1)	14(A)	95 ∼ 110(E)		15(A)	53(B1)	
	55(B1)	17(A)	115(F)		16(A)	54(B1)	
		18(A)	137(B2)		19(A)	121 ∼ 136(B2)	
		21(A)			20(A)	138 ∼ 151(B2)	
		22(A)					
		24(A)					

TABLE B.3: K-means clustering ($k = 7$): 20-point feature

Whistle Type	**a**	**b**	**c**	**d**	**e**	**f**	**g**
Whistle ID.	6(A)	$1 \sim 3$(A)	$44 \sim 46$(B1)	$56 \sim 58$(C)	4(A)	23(A)	$111 \sim 114$(F)
	$25 \sim 33$(B1)	7(A)	59(C)	$60 \sim 65$(C)	5(A)	34(B1)	$116 \sim 120$(F)
	35(B1)	8(A)	66(C)	$67 \sim 78$(C)	9(A)	36(B1)	
	$37 \sim 43$(B1)	10(A)	79(C)	80(C)	11(A)	50(B1)	
	$47 \sim 49$(B1)	12(A)	$81 \sim 94$(D)		13(A)	51(B1)	
	52(B1)	14(A)	$95 \sim 110$(E)		15(A)	53(B1)	
	55(B1)	17(A)	115(F)		16(A)	54(B1)	
		18(A)	137(B2)		19(A)	$121 \sim 136$(B2)	
		21(A)			20(A)	$138 \sim 151$(B2)	
		22(A)					
		24(A)					

TABLE B.4: Clustering result by competitive learning: 8 PCs

Whistle Type	$\mathbf{w}_1$	$\mathbf{w}_2$	$\mathbf{w}_3$	$\mathbf{w}_4$	$\mathbf{w}_5$	$\mathbf{w}_6$	$\mathbf{w}_7$
Whistle ID.	63(C)	56 ~ 58(C)	1(A)	34(B1)	121 ~ 124(B2)	2 ~ 5(A)	28 ~ 30(B1)
	79(C)	60 ~ 62(C)	6(A)	46(B1)	128 ~ 134(B2)	7(A)	33(B1)
	82(D)	64(C)	9(A)	59(C)	136(B2)	8(A)	35 ~ 39(B1)
	84(D)	65(C)	14(A)	66(C)	138 ~ 145(B2)	10 ~ 13(A)	41 ~ 45(B1)
	86 ~ 91(D)	67 ~ 78(C)	15(A)	81(D)	147 ~ 149(B2)	16(A)	47(B1)
	111 ~ 120(F)	80(C)	17(A)	92 ~ 94(D)	151(B2)	18 ~ 24(A)	48(B1)
			25 ~ 27(B1)	95 ~ 104(E)		83(D)	54(B1)
			31(B1)	106 ~ 110(E)		85(D)	105(E)
			32(B1)				125 ~ 127(B2)
			40(B1)				135(B2)
			49 ~ 53(B1)				137(B2)
			55(B1)				146(B2)
							150(B2)

TABLE B.5: Clustering result by competitive learning: 20-point feature

Whistle Type	$\mathbf{w}_1$	$\mathbf{w}_2$	$\mathbf{w}_3$	$\mathbf{w}_4$	$\mathbf{w}_5$	$\mathbf{w}_6$	$\mathbf{w}_7$
Whistle ID.	63(C)	56(C)	4(A)	6(A)	25(B1)	$1 \sim 3$(A)	47(B1)
	69(C)	57(C)	7(A)	27(B1)	29(B1)	5(A)	58(C)
	$111 \sim 120$(F)	$60 \sim 62$(C)	15(A)	31(B1)	30(B1)	$8 \sim 14$(A)	59(C)
		64(C)	23(B1)	33(B1)	32(B1)	$16 \sim 24$(A)	68(C)
		65(C)	26(B1)	36(B1)	34(B1)		70(C)
		67(C)	28(B1)	38(B1)	35(B1)		$72 \sim 74$(C)
		71(C)	43(B1)	40(B1)	37(B1)		79(C)
		$75 \sim 78$(C)	44(B1)	41(B1)	39(B1)		$82 \sim 94$(D)
		80(C)	54(B1)	$48 \sim 51$(B1)	42(B1)		$95 \sim 110$(E)
			$121 \sim 123$(B2)	53(B1)	45(B1)		135(B2)
			$125 \sim 130$(B2)	124(B2)	46(B1)		137(B2)
			133(B2)	131(B2)	52(B1)		146(B2)
			$136 \sim 138$(B2)	132(B2)	55(B1)		150(B2)
			140(B2)	134(B2)	66(C)		
			142(B2)	135(B2)	81(D)		
			143(B2)	139(B2)			
			145(B2)	141(B2)			
			146(B2)	144(B2)			
			$148 \sim 149$(B2)	147(B2)			
			151(B2)	150(B2)			

TABLE B.6: Clustering result by SOM (8 classes): 8 PCs

Whistle Type	$\mathbf{w}_1$	$\mathbf{w}_2$	$\mathbf{w}_3$	$\mathbf{w}_4$	$\mathbf{w}_5$	$\mathbf{w}_6$	$\mathbf{w}_7$	$\mathbf{w}_8$
Whistle ID.	56 ~ 58(C)	34(B1)	29(B1)	28(B1)	2 ~ 5(A)	26(B1)	1(A)	52(B1)
	60 ~ 65(C)	46(B1)	30(B1)	36(B1)	7(A)	27(B1)	6(A)	55(B1)
	67 ~ 78(C)	59(C)	35(B1)	37(B1)	8(A)	33(B1)	9(A)	82(D)
	80(C)	66(C)	38(B1)	44(B1)	10 ~ 13(A)	41(B1)	14(A)	86 ~ 88(D)
	99(E)	79(C)	39(B1)	45(B1)	16(A)	49 ~ 51(B1)	15(A)	90(D)
	100(E)	83 ~ 85(D)	42(B1)	48(B1)	18 ~ 24(A)	144(B2)	17(A)	91(D)
	110(E)	89(D)	43(B1)	81(D)			25(B1)	111(F)
		92 ~ 94(D)	47(B1)	121 ~ 151(B2)			31(B1)	112(F)
		95 ~ 98(E)	54(B1)				32(B1)	114 ~ 120(F)
		101 ~ 109(E)					40(B1)	
		113(F)					53(B1)	

TABLE B.7: Clustering result by SOM (8 classes): 20-point feature

Whistle Type	$\mathbf{w}_1$	$\mathbf{w}_2$	$\mathbf{w}_3$	$\mathbf{w}_4$	$\mathbf{w}_5$	$\mathbf{w}_6$	$\mathbf{w}_7$	$\mathbf{w}_8$
Whistle ID.	27(B1)	4(A)	29(B1)	1 ∼ 3(A)	56(C)	83(D)	57(C)	69(C)
	31(B1)	6(A)	30(B1)	5(A)	58 ∼ 59(C)	84(D)	60 ∼ 64(C)	111 ∼ 120(F)
	33(B1)	7(A)	32(B1)	8 ∼ 14(A)	65(C)	88(D)	72 ∼ 74(C)	
	36(B1)	15(A)	34(B1)	16 ∼ 22(A)	67 ∼ 68(C)	90(D)	76 ∼ 77(C)	
	38(B1)	23(A)	35(B1)	24(A)	70 ∼ 71(C)	91(D)	79(C)	
	40 ∼ 41(B1)	26(B1)	43(B1)	25(B1)	75(C)	105(E)	82(D)	
	48 ∼ 51(B1)	28(B1)	45 ∼ 46(B1)	37(B1)	78(C)		85 ∼ 87(D)	
	53(B1)	44(B1)	52(B1)	39(B1)	80(C)		89(D)	
	121(B2)	54(B1)	55(B1)	42(B1)	92 ∼ 93(D)		94(D)	
	124(B2)	125(B2)	66(C)	47(B1)	97(E)		95 ∼ 96(E)	
	127 ∼ 135(B2)	136(B2)	122(B2)	81(D)	100 ∼ 102(E)		98 ∼ 99(E)	
	139 ∼ 147(B2)	138(B2)	123(B2)		107(E)		103 ∼ 104(E)	
	149(B2)	151(B2)	126(B2)		109(E)		106(E)	
	150(B2)		137(B2)				108(E)	
			148(B2)				110(E)	

Bibliography

[1] W. W. Au, "Echolocation signals of the atlantic bottlenose dolphin (*Tursiops truncatus*) in open waters," in *Animal Sonar Systems*, R. G. Busnel, Ed. New York: Plenum Press, 1980, pp. 251–282.

[2] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509–522, Apr. 2002.

[3] ——, "Matching with shape contexts," Oct. 2001. [Online]. Available: http://www.eecs.berkeley.edu/Research/Projects/CS/vision/shape/sc_digits.html

[4] C. Blomqvist and M. Amundin, "High-frequency burst-pulse sounds in agonistic/aggressive interactions in bottlenose dolphins, *Tursiops truncatus*," in *Echolocation in Bats and Dolphins*, R. G. Busnel, Ed. The University of Chicago Press Chicago, 2004, ch. 60, pp. 425–431.

[5] I. Borg and P. Groenen, *Moderm Multidimensional Scaling*. Springer Series in Statistics, Dec. 1996.

[6] J. C. Brown, A. Hodgins-Davis, and P. J. O. Miller, "Classification of vocalization of killer whales using dynamic time warping," *JASA Express Letters*, vol. 119, no. 3, Feb. 2006.

[7] J. R. Buck and P. L. tyack, "A quantitative measure of similarity for tursiops truncates signature whistles," *J. Acoust. Soc. Am.*, vol. 94, no. 5, pp. 2497–2506, Nov. 1993.

[8] D. K. Caldwell and M. C. Caldwell, *Mammals of the sea: Biology and Medicine*. Springfield, Illinois: Charles C. Thomas, Publisher, 1972, ch. Senses and communication, pp. 466–502.

[9] M. C. Caldwell and D. K. Caldwell, "Statistical evidence for individual signature whistles in pacific white-sided dolphins, *Lagenorhynchus obliquidens*," *Cetology*, vol. 3, no. 3, pp. 1–9, 1971.

[10] M. C. Caldwell, D. K. Caldwell, and P. L. Tyack, "A review of the signature whistle hypothesis for the atlantic bottlenoise dolphin, *Tursiop truncatus*," in *The Bottlenose Dolphin*, S. Leatherwood and R. R. Reeves, Eds. San Diego, California: Academic Press, Inc., 1990, pp. 199–234.

[11] J. Canny, "A computational approach to edge detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679 –698, Nov. 1986.

[12] H. Cramer, *Mathematical methods of statistics*. Princeton University Press, 1946.

[13] S. Datta and C. Sturtivant, "Dolphin whistle classification for determining group identities," *Signal Processing*, vol. 82, no. 2, pp. 251 – 258, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/B6V18-44P6W8N-1/2/e0dfa0fba57ab46ffd2bb5438039c884

[14] C. de Boor, *A practical guide to splines*. New York : Springer-Verlag, 1978.

[15] V. B. Deescke and V. M. Janik, "Automated categorization of bioacoustic signals: Avoiding perceptual pitfalls," *J. Acoust. Soc. Am.*, vol. 119, no. 1, pp. 645–653, Jan. 2006.

[16] C. Ding and X. He, "K-means clustering via principal component analysis," in *Proc. of Int'l Conf. Machine Learning (ICML 2004)*. University of California Press, Jul. 2004, pp. 225–232.

[17] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. Wiley-interscience Publication, Nov. 2000.

[18] M. Frenkel and R. Basri, "Curve matching using the fast marching method," in *Energy Minimization Methods in Computer Vision and Pattern Recognition 4th International Workshop*, ser. Lecture Notes in Computer Science, 2003, pp. 35–51.

[19] D. Fripp, C. Owen, E. Quintana-Rizzo, A. Shapiro, K. Bucksaff, K. Jankowski, R. Wells, and P. Tyack, "Bottlenose dolphin (*Tursiops truncatus*) calves

appear to model their signature whistles on the signature whistles of community members," *the Journal of Experimental Biology*, vol. 8, no. 1, pp. 17–27, Jan. 2005.

[20] R. Gao, M. Chitre, S. H. Ong, and E. Taylor, "Automatic template matching for classification of dolphin vocalizations," in *Proc. of MTS/IEEE Oceans'08, Kobe, Japan*, 2008.

[21] M. Greco, F. Gini, and L. Verrazzani, "Analysis and modeling of acoustic signals emitted by mediterranean bottlenose dolphins," in *Signal Processing and Information Technology*, Dec. 2003, pp. 122–125.

[22] L. Hong, T. B. Koay, J. R. Potter, and S. H. Ong, "Estimating snapping shrimp noise in warm shallow water," in *Oceanology International'99, Singapore*, 1999.

[23] V. M. Janik, "Pitfalls in the cauterization of behaviors: a comparison of dolphin whistle classification methods," *Animal behaviours*, vol. 57, pp. 133–143, 1999.

[24] R. Jonker and A. Volgenant, "A shortest augmenting path algorithm for dense and sparse linear assignment problems," *Computing*, vol. 38, no. 4, pp. 325–340, Mar. 1987.

[25] H. Kaprykowsky and X. Rodet, "Globally optimal short-time dynamic time warping application to score to audio alignment," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, 2006, pp. 249–252.

[26] G. Karypis, E.-H. Han, and V. Kumar, "Chameleon: hierarchical clustering using dynamic modeling," *Computer*, vol. 32, no. 8, pp. 68–75, Aug. 1999.

[27] E. J. Keogh and M. J. Pazzani, "Scaling up dynamic time warping for datamining applications," in *Proceedings of the sixth ACM, Boston, Massachusetts, US*, 2000, pp. 285–289.

[28] H. Khanna, S. L. L. Gaunt, and D. A. McCallum, "Digital spectrographic cross-correlation: tests of sensitivity," *Bioacoustics*, vol. 7, no. 3, pp. 209–234, 1997.

[29] T. Kohonen, *the Self-Organizing Maps*, 3rd ed. New York: Springer, 2000.

[30] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. of the 5th Berkeley Symposium on Mathematical Statistics and Probability*. University of California Press, 1967, pp. 281–297.

[31] P. C. Mahalanobis, "On the generalised distance in statistics," in *Proc. of the National Institute of Sciences of India*, vol. 2, no. 1, 1936, pp. 49–55.

[32] A. Mallawaarachchi, "Spectrogram denoising for the automated extraction of dolphin whistle contous." M. Eng. thesis, the National University of Singapore, 2007.

[33] A. Mallawaarachchi, S. Ong, M. Chitre, and E. Taylor, "A method for tracing dolphin whistles," in *OCEANS'06 - Asia Pacific*, May 2006, pp. 1–5.

[34] A. Martinez and A. Kak, "PCA versus LDA," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 228–233, Feb. 2001.

[35] B. McCowan, "A new quantitative technique for categorizing whistles using simulated signals and whistles from captive bottlenose dolphins (Delphinidae, *Tursiops truncatus*)," *Ethology*, vol. 100, no. 3, pp. 177–193, January-Decemeber 1995.

[36] B. McCowan, L. Marino, E. Vance, L. Walke, and D. Reiss, "Bubble ring play of bottlenose dolphins (*Tursiops truncatus*): implications for cognition," *Journal of Comparative Psychology*, vol. 114, no. 1, pp. 98–106, March 2000.

[37] S. C. Nanayakkara, M. Chitre, S. H. Ong, and E. Taylor, "Automatic classification of whistles produced by indo-pacific humpback dolphins (*Sousa chinensis*)," in *Proc. of Oceans'07*, vol. 7, Jun. 2007, pp. 1–5.

[38] L. Ong, "The description and analysis of bottlenose dolphin (*Tursiops truncatus*) whistles," 1996, a thesis submitted to the National University of Singapore in partial fulfilment of the Degree of Bachelor of Science with Honours in Zoology.

[39] J. N. Oswald, J. Barlow, and T. F. Norris, "Acoustic identification of nine delphinid species in the eastern tropical pacific ocean," *Ultrasonics*, vol. 19, no. 1, pp. 20–37, Jan. 2003.

[40] C. Papadimitriou and K. Stieglitz, *Combinatorial Optimization: Algorithms and Complextiy*. Prentice Hall, 1982.

[41] L. Rabiner, A. Rosenberg, and S. Levinson, "Considerations in dynamic time warping algorithms for discrete word recognition," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 26, no. 6, pp. 575–582, Dec. 1978.

[42] M. J. Russell, R. K. Moore, and M. J. Tomlinson, "Some techniques for incorporating local timescale variability information into a dynamic time-warping algorithm for automatic speech recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 8, Apr. 1983, pp. 1037–1040.

[43] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 26, no. 1, pp. 43–49, Feb. 1978.

[44] G. A. F. Seber, *Multivariate Observations*. John Wiley & Sons, Inc., 1984.

[45] J. A. Sethian, "A fast marching level set method for monotonically advancing fronts," *Proc. Nat. Acad. Sci*, vol. 93, no. 4, pp. 1591–1595, Feb. 1996.

[46] M. Steinbach, L. Ertz, and V. Kumar, "The challenges of clustering high-dimensional data," in *New Vistas in Statistical Physics: Applications in Econophysics, Bioinformatics, and Pattern Recognition*. Springer-Verlag, 2003.

[47] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 4th ed. Burlington, MA ; London : Academic Press, 2009.

[48] P. L. Tyack, "Communications and cognition," in *Biology of Marine Mammals*, I. J. E. Reynolds and S. A. Rommel, Eds. Smithsonian Institution Press, Washingtong, D.C., 1999, pp. 287–323.

[49] F. van der Heijden, R. Duin, D. de Ridder, and D. M. J. Tax, *Classification, parameter estimation, and state estimation:an engineering approach using MATLAB*. John Wiley & Sons, Inc., Nov. 2004.

[50] S. M. van Parijs and P. J. Corkeron, "Evidence for signature whistle production by a pacific humpback dolphin, *Sousa Chinensis*," *Marine Mammal Science*, vol. 17, no. 4, pp. 944–949, Oct. 2001.

[51] ——, "Vocalizations and behavior of pacific humpback dolphins *Sousa Chinensis*," *Ethology*, vol. 107, pp. 701–716, 2001.

[52] A. Walker, R. Fisher, and N. Mitsakakis, "Classification of whalesong units using a self-organizing feature mapping algorithm," *J. Acoust. Soc. Am.*, vol. 100, no. 4, p. 2644, Oct. 1996.

[53] P. D. Wasserman, *Advanced Methods in Neural Computing.* John Wiley & Sons, Inc. New York, USA, 1993.

[54] A. Webb, *Statistical pattern classification*, 2nd ed. London: Arnold, 1999.

[55] L. Yuan, L. Zhou, and Z. Liu, "The self-organizing feature map used for speaker-independent speech recognition," in *3rd International Conference on Signal Processing*, vol. 1, 14-18 1996, pp. 733 –736.