MASTER'S THESIS

Ground Vehicle Acoustic Signal Processing Based on Biological Hearing Models

by Li Liu
Advisor: John S. Baras

M.S. 99-6

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
1999		**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Ground Vehicle Acoustic Signal Processing Based on Biological Hearing Models	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Office of Naval Research,One Liberty Center,875 North Randolph Street Suite 1425,Arlington,VA,22203-1995**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**		**88**	

ABSTRACT

Title of thesis: GROUND VEHICLE ACOUSTIC SIGNAL PROCESSING BASED ON BIOLOGICAL HEARING MODELS:

Degree candidate: Li Liu

Degree and year: Master of Science, 1999

Thesis directed by: Professor John S. Baras

Institute of Systems Research

This thesis presents a prototype vehicle acoustic signal classification system with low classification error and short processing delay. To analyze the spectrum of the vehicle acoustic signal, we adopt biologically motivated feature extraction models – cochlear filter and A1-cortical wavelet transform. The multi-resolution representation obtained from these two models is used in the later classification system. Different VQ based clustering algorithms are implemented and tested for real world vehicle acoustic signals. Among them, Learning VQ achieves the optimal Bayes classification performance, but its long search and training time make it not suitable for real time implementation. TSVQ needs a logarithmic search time and its tree structure naturally imitates the aggressive hearing in biological hearing systems, but it has a higher classification error. Finally, a high performance parallel TSVQ (PTSVQ) is introduced, which has classification performance close to the optimal LVQ, while maintains logarithmic search time.

Experiments on ACIDS database show that both PTSVQ and LVQ achieve high classification rate. PTSVQ has additional advantages such as easy online training and insensitivity to initial conditions. All these features make PTSVQ the most promising candidate for practical system implementation.

Another problem investigated in this thesis is combined DOA and classification, which is motivated by the biological sound localization model developed by Professor S. Shamma: the Stereausis neural network. This model is used to perform DOA estimation for multiple vehicle recordings. The angle estimation is further used to construct a spectral separation template. Experiments with the separated spectrum shows significant improvement in classification performance. The biologically inspired separation scheme is quite different from traditional beamforming. However, it integrates all 3 biological hearing models into a unified framework, and it shows great potential for multiple target DOA and ID systems in the future.

GROUND VEHICLE ACOUSTIC SIGNAL PROCESSING BASED ON BIOLOGICAL HEARING MODELS:

by

Li Liu

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Science
1999

Advisory Committee:

Professor John S. Baras, Chair
Professor Steven I. Marcus
Professor P.S. Krishnaprasad

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude and thanks to my advisor, professor John S. Baras, for his advice, support, guidance and sponsorship throughout my dissertation research at University of Maryland, College Park.

I would like to give my sincere thanks to Professor Shihab Shamma, at the Center of Audio and Acoustic Research (CAAR), for his constant supports throughout my research. Without his help, the whole thesis would not be possible.

I would like to give my special thanks to Professor P.S.Krishnaprasad, Professor Steven Marcus, Mr. Tien Pham, Mr. Varma, and other staffs at CAAR, for their excellent suggestions during group meetings, helpful advice and comments, and for serving on my committee.

This work was supported by ONR-MURI Center for Auditory and Acoustic Research contract. (Contract #: N000149710501EE)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1 Introduction

Researchers have long been working on automated target detection and recognition systems. For ground vehicles, acoustic signals are useful for classification purposes. The classification problem is defined as assigning an unknown vehicle sound into one of a pre-specified class based on the extraction of significant features or attributes [10]. Such a simple problem to a human is not so simple when we want to make a machine perform the task. In order to be able to classify its input, the machine has to process the input sound, measure its similarity and decide which vehicle class that input belongs to. We may say that a pattern classification problem is a pattern recognition problem and that recognition is the ability to classify.

Human beings have an outstanding ability to recognize natural sounds. Normally a musician can easily tell the 1Hz difference between two tones. Since biological perceptual nervous systems are basically self-trained classification machines, and have a superior performance than most existing classification systems, the knowledge about how signal processing is done in the nervous system has attracted significant attention from researchers. In this thesis, we will study several state of the art biological signal processing models, and use these models to extract multi-resolution features from vehicle

acoustic signals. Based on these hierarchical feature representations, an aggressive unsupervised TSVQ algorithm is implemented to classify the acoustic inputs. Furthermore, we make some modifications to the existing binaural hearing model, which provides us with new features very important for multi-vehicle ID systems. Through this research, we hope to gain more insight into the potential application of biological models in acoustic pattern recognition systems design.

1.1 Research Background

The Army Research Laboratory (ARL) has created the Acoustic-seismic Classification Identification Data Set (ACIDS) for vehicle classification research. This database contains 9 types of vehicles, as shown in table 1.1. In the ACIDS database, each vehicle has dozens of runs, corresponding to different speed and gear, different terrain (desert, arctic, normal roadway, and etc), and different recording systems. This database represents an ideal opportunity for classification research.

Type 1: heavy track vehicle
Type 2: heavy track vehicle
Type 3: heavy wheel vehicle
Type 4: light track vehicle
Type 5: heavy wheel vehicle
Type 6: light wheel vehicle
Type 7: light wheel vehicle
Type 8: heavy track
Type 9: heavy track

Table 1.1 Different vehicles in the ACIDS database.

Pattern recognition is an inexact science involving many areas and disciplines. A typical pattern recognition system consists of the following standard parts as shown in fig.1.1. Later in this thesis, the design of each part will be described in detail.

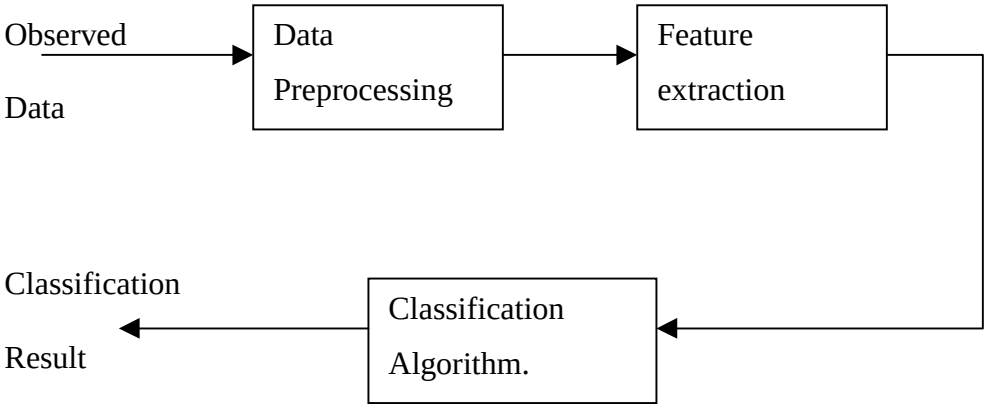


Figure 1.1 Block diagram of acoustic signal classification system

1.2 Survey of previous research

The overall acoustic signal of a vehicle arises from several sources including engine, gear, fan, cooling system, road-tire interaction, exhaust and air movement. Historically, the most extensive study of this kind of signals was carried out by scientists who were working on ground traffic control problems.

1.2.1 Algorithm based on time domain feature extraction:

In [14], Sampan of Virginia Tech used block-averaging of the time domain signal to classify vehicles into 4 different classes: cars, trucks, heavy trucks, and trailers. Basically, his method is based on the short time strength of the acoustic signal, and little spectrum information is used. So their method can not distinguish two different cars with nearly the same size and engine power. In [14], the best performance is 96% correct classification. In [28], Scott used a similar approach and obtained better performance.

However, the time domain features limit this method to coarse classification of vehicle types. For more precise classification, features extracted from the frequency domain must be considered.

1.2.2 Wavelet and filter bank based features:

In [17], Dress and Kercel suggest that: “due to the non-stationary nature of the vehicle acoustic signal, parameter based methods such as ARMA models are likewise unsuccessful, and a time –frequency approach seemed more likely to succeed." In their approach, the FFT of wavelet subspace signals are used as features. In [18], Choe *et al.* use combined STFT and wavelets as features, and for a database containing 2 vehicles, he achieved a 98% correct classification. For a larger database, their method doesn't guarantee the same performance.

1.2.3 Classification algorithm:

Many types of classification algorithms have been used in vehicle signal classification. In [18], Choe *et al.* use an HMM-ANN fused classifier, in [17], Dress and Kercel use fuzzy set membership and ANN, in [14], Sampan uses Fuzzy logic. Classical K-nearest neighbor and radial basis function networks are also found in the literature [17].

In [3], Baras and Wolk introduced a tree structured vector quantization (TSVQ) algorithm for the ship radar return classification. They demonstrated that a cascade of Wavelet transform followed by a TSVQ clustering algorithm can achieve a progressive classification scheme. In their experiments, the ‘parallel’ TSVQ provided a performance very close to the optimal Bayes LVQ classifier. Furthermore, their model provides a

natural way to imitate the hierarchical physiological hearing in the human nervous system. Related discussions of this scheme can be found in [6][9][19].

1.2.4 Previous work on biological hearing models

- Periphery auditory processing models:

The sound signal undergoes a series of transformations in the early stage of auditory processing, and people developed various kinds of biophysical models or approximate computational algorithms to simulate the cochlear processing. In [4], Shamma *et al.* integrated the earlier approach and introduced a new framework of 3 stage cochlear processing. Using this auditory model he successfully reconstructed the original sound from different stages of the auditory representation. Since the cochlear model shows strong spectral analysis capability, in [29][30], Kumar *et al*. used it as a front-end of a speech feature extraction system.

- Cortical processing model

In human nervous system, the stimuli from the peripheral auditory system are transmitted to the cortex for further processing. In [5], Shamma *et al.* suggested that the cortex analyses the input auditory spectral pattern along three independent dimensions: a logarithmic frequency axis, a local symmetry axis and a local ripple frequency axis. It is shown that this processing is equivalent to performing an affine wavelet transform of the spectral pattern while preserving both the amplitude and phase information. In our research, we use a constant Q filter bank as a simplified cortical model to decompose the auditory spectrum into a multi-scale representation. This multi-scale representation, combined with the TSVQ algorithm, provides us a hierarchical classification scheme as

suggested before. In this sense, the whole classification system will be consistently biological based.

- Stereausis binaural hearing model

In [10], Shamma introduced a binaural hearing model - Stereausis, to explain the spatial hearing and sound localization in human physiology. This model is unique in that its output purely depends on the cross-correlation of different filter banks, and no neural delay pathway is involved in the network. In the stereausis network, an unbalanced sound input will cause the network response to shift away from the main diagonal. A proper measure of this shift can be used to calculate the impact angle of arrival signal.

Later in this thesis, we will examine the Stereausis network based on a small array for its DOA estimation performance for multi-vehicle recordings. From this DOA estimation, a signal separation scheme similar to traditional beamforming will be introduced and discussed in detail. Through these approaches, we hope to integrate the vehicle ID, DOA estimation, and multi vehicle signal separation problems into a unified framework.

1.3 Contributions and scope of research

Our research goal is:

- Develop a prototype vehicle signal classification system with low classification error and short classification delay.
- Test and modify the biology based hearing model as a practical feature extraction system.

- Explore the VQ based classification algorithm, improve its overall performance such as low classification error, short search time and easy online training.

The following contributions have resulted from this thesis:

- A prototype vehicle acoustic signal classification system is implemented and tested. The suggested classifier can achieve above 90 percent correct classification, while only using logarithmic search time.
- A combined DOA and classification system, in which significant classification gain is obtained through Stereausis based DOA estimation.
- Feature extraction from biological hearing models proved successful for ground vehicle classification purposes. This result should lead to wider usage of such models in various speech processing applications.
- A new signal separation algorithm, different from traditional beamforming. This algorithm is based on DOA estimation and performs very well for small arrays.
- A new method to initialize the LVQ classifier, which helps the LVQ classifier to overcome local minimal points in a rapid manner.
- A thorough analysis and comparison of VQ based classification algorithms, which may lead to further development of a tree structured LVQ algorithm.
- An entropy based classification confidence measure. This measure fits well with all VQ based classifiers, and shows great potential in providing reliable confidence suggestions to the end user.

Chapter 2 Feature extraction with biological hearing models

Human beings have a strong ability to recognize acoustic signals. In recent years, researchers have carefully studied this biological hearing capability, hoping to find beneficial structures or useful models to assist the research of pattern recognition and signal classification. Some of the biological research results and findings have already been used in speech recognition systems, such as the Mel-frequency scale[22], adaptive mechanisms[23][24], and compressive non-linearity[25]. In recent years, Shamma *et al.* presented a series of mathematical models to mimic the structure of the peripheral and cortical auditory systems. His models not only proved to be successful in explaining the mechanisms of the biological nervous system, but also showed remarkable ability in spectral enhancements and noise suppression. In this chapter, Shamma's peripheral auditory model [4] and central cortex hearing model [5] will be introduced. Later, these models will be extensively used to perform feature extraction for vehicle acoustic signal.

2.1. Biological hearing models

2.1.1 Peripheral auditory processing model

For human beings, the sound signal undergoes a complex series of transformations in the early stage of auditory processing. In [4], Shamma divides the total procedure into 3 concatenated stages: analysis stage, transduction stage, and reduction stage. The whole mathematical model is plotted in figure 2.1.

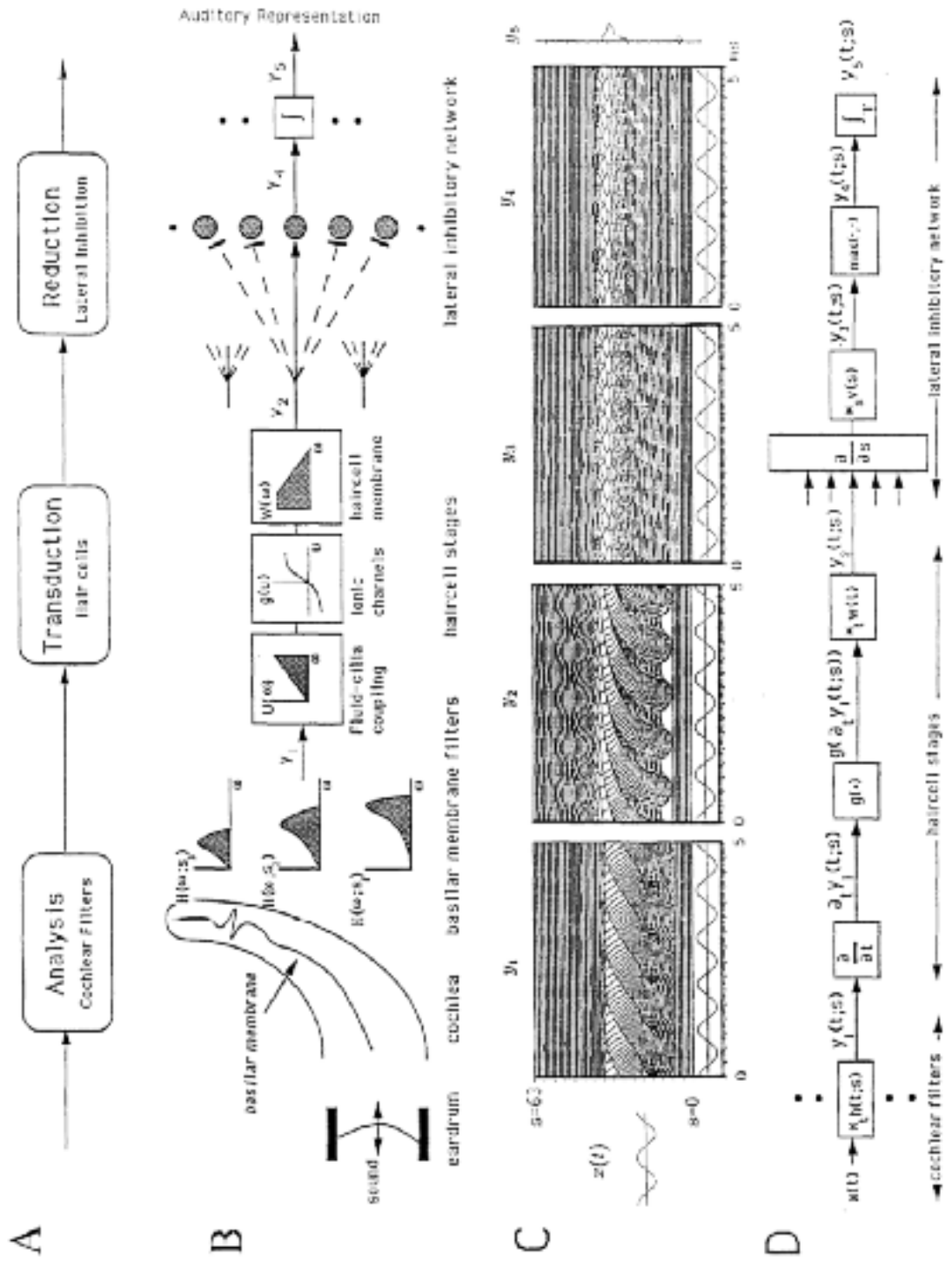


Figure2.1 Peripheral auditory model: (a) block digaram of the three basic stages in the early auditory system, (b) Quasi-anatomic sketches of the suditory stages, (c) mathematic models of each stages.

In the analysis stage, the cochlear is modeled as a parallel bank of band-pass filters. Along the logarithmic frequency, the transfer function of each band appears approximately invariant except for a translation, i.e., a constant Q filter bank. Therefore, it is natural to interpret the outputs of the cochlear filters as affine wavelet transforms of the input signal. The biological counterpart of this module is the spatially distributed basilar membrane along the length of the cochlea. Vibrations evoked by a single tone appear as traveling waves that propagate up the cochlea, reach maximum amplitude before slowing down and decaying rapidly. Thus basilar membranes at different locations of the cochlea appear to be band-pass filters sensitive to particular frequency stimuli.

The transduction stage is modeled by a three-step process: The first part is a time derivation, followed by a nonlinear transform (normally a sigmoid function), and end with a low pass filter. Each part has a corresponding physiological process associated with it. From the information processing point of view, these complex transforms merely convey hair cell potentials to the cochlear nucleus. Under the high gain limit assumption, this stage can be totally ignored[5].

The Reduction stage performs the spectral estimation function. Its dominant part is a lateral inhibitory network (LIN), which is common in all nerve sensory systems. In Shamma's model, this stage is further decomposed into 3 parts. The first part is a derivative (or differential) structure with respect to the spatial axis of the cochlea, which models the lateral inhibition effects among the LIN neurons, which essentially enhances the sensitivity to spatial discontinuities of the input pattern. A half wave rectifier is the second part. It models the threshold non-linearity in the neuron models of the LIN network. The final part is a long time constant (10-20ms) integrator. This step is based on

the fact that central auditory neurons can not follow rapid temporal modulations higher than a few hundred Hertz.

The processes in this module can be summarized into the following formulas:

$$y_1(t,x) = x(t) \otimes_t b(t,x)$$

$$y_2(t,x) = g(y_1(t,x))$$

$$y_3(t,x) = \partial_x(y_2(t,x))$$

$$y_4(t,x) = \max(y_3(t,x),0)$$

$$y_5(t,x) = y_4(t,x) \otimes_t \Pi(t)$$

where x(t) is the input acoustic signal, b(t, x) is the impulse response of the wavelet filter at location (or scale) x; g(.) is a sigmoid function; Π(.) is a temporal integration window, and $y_i(t,x), i = 1,\cdots,5$ correspond to the output of different stages in Fig2.1.

To summarize, this module transforms the time domain acoustic signal into a log-frequency spectrum profile. This 1-D spectrum profile maximally reduced the data volume with minimal loss of perceptual information [4]; thus it is suitable for various applications such as low bit-rate speech compression or automatic speech recognition.

2.1.2 Cortical processing model

The auditory spectrum generated from the auditory module is fed into cortical nerves for further processing. In [5], Shamma uses a wavelet transform to model this cortical function. A spatial frequency measure: ripple frequency, Ω, is introduced as sinusoidally modulated magnitude spectrum in the log-frequency domain. It represents the number of cycles in one octave. The relation between scale (log-frequency) and ripple

domain is analogous to the relation between time and frequency domain. Therefore, the outcome of this module is the complex-valued representation of the input auditory spectrum at different resolutions (different ripple frequency). This 2-D cortical representation is given by:

$$r(x,\Omega) = y_5(x) \otimes_x w(x,\Omega)$$

Where $y_5(x)$ is the input 1-D long-term averaged auditory spectrum from the previous auditory model, $w(x,\Omega)$ is the impulse response of the cortical filter at a given ripple frequency Ω. $r(x,\Omega)$ represents the auditory spectrum at the particular resolution Ω. In the complex-valued 2-D pattern, the real part represents the In-phase component of the cortical response, while the imaginary part is the corresponding quadrature component. Since the In-phase component contains all information concerning the classification, only the real-valued cortical representation will be preserved for later usage.

2.2 Implementation issues:

- Auditory Module:

It is difficult to implement the entire function blocks in fig. 2.1, therefore we make some simplifications. After preprocessing, segmented acoustic data with zero-mean and unit variance are fed into this module. A 128-band constant Q filter bank serves as the cochlear filters, some of the filter responses are plotted in fig.2.2. The nonlinear compression function g(.) is dropped and replaced with a linear function. Since the following cortical model requires a 1-D spectrum as input, we collapse the T-F presentation onto the ripple frequency axis by calculating the mean value along the time axis to obtain an 1-D auditory spectrum. The window for short time average is roughly

250 ms, within such short time, the stationary assumption will hold for most vehicle acoustic signals. However, severe fluctuations do happen in many situations. To compensate the short time fluctuation, a decision fusion unit is implemented in the classification system, which will be discussed in detail in Chapter 3. All the other parts of the auditory model, such as the half wave rectifier and LIN, are the same as in [4].

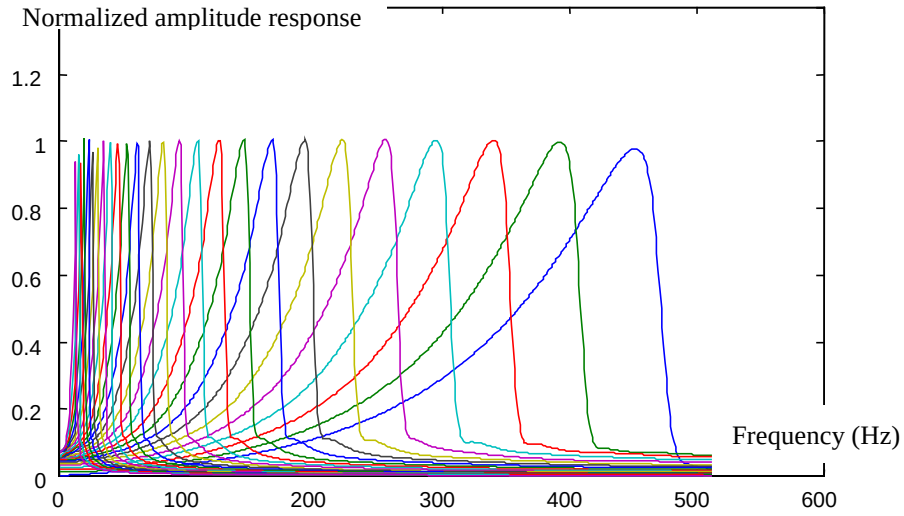


Figure 2.2 Frequency response of cochlear filter banks

- Cortical module:

This stage is implemented by a series of constant Q (ripple) band-pass filters, with each filter tuned around a characteristic ripple frequency. Actually, this mapping of spectral ripples onto a scale axis is very similar to the logarithmic mapping of an acoustic frequency onto the spatial axis of a cochlear filter. This suggests that the sequence of cochlear and cortical analysis of acoustic signal is conceptually a form of a double affine wavelet transform, which is very similar to the cepstral analysis. After this stage, we obtain a multi-resolution representation of the auditory spectrum.

2.3 Experiments on feature extraction

(1) Simulation on auditory processing:

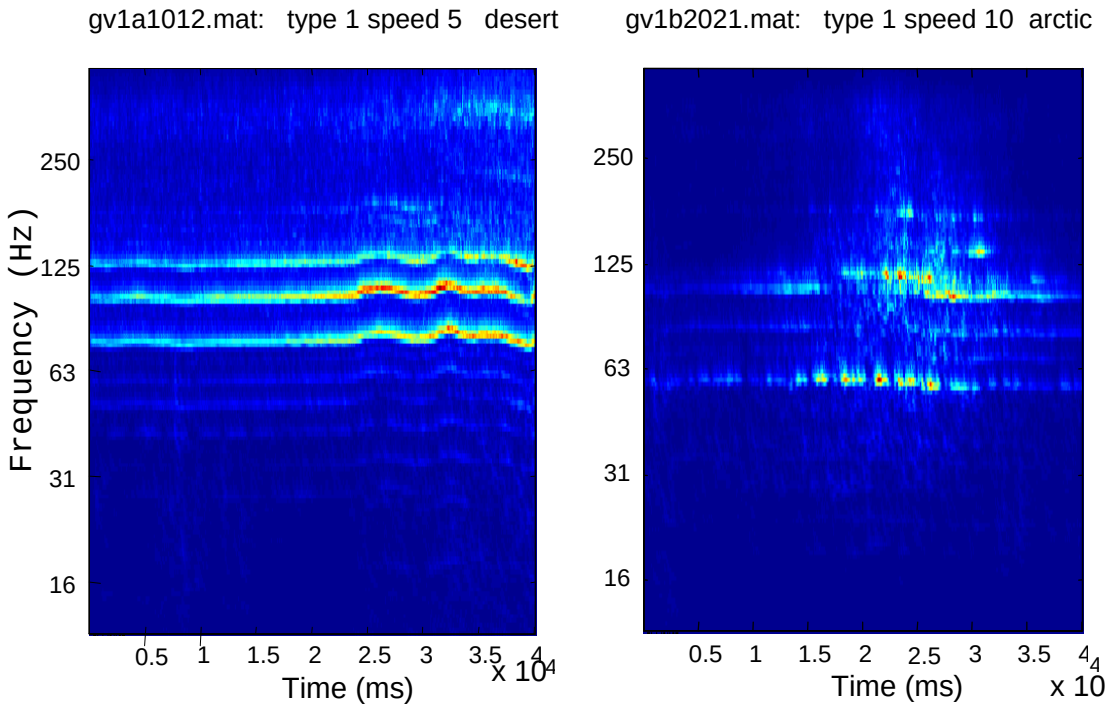


Figure 2.3 cochlear pattern for vehicle signals (a): vehicle type 1, speed 5km/hr. (b): vehicle type 1, speed 10km/hr

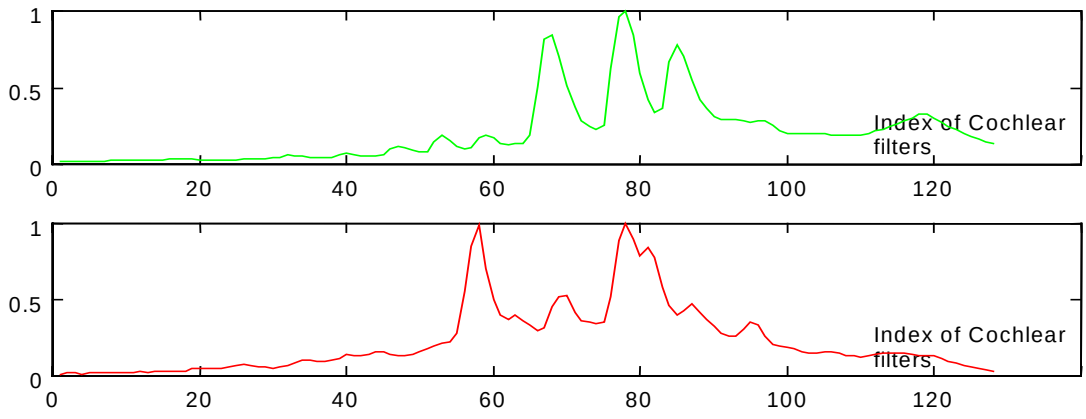


Figure 2.4 Vehicle signal auditory spectra. The horizontal axis is index of cochlear filters, the vertical axis is the amplitude of the normalized auditory spectrum. Top: vehicle type 1, 5km/hr. Bottom: vehicle type 1, 10km/hr

Fig.2.3 (a) shows a typical auditory time-frequency representation obtained by passing a vehicle signal through the cochlear filter banks. From this T-F representation,

we find that the vehicle acoustic signal is approximately confined to the range of 20 to 200 Hz, and is dominated by salient low frequency harmonics parallel to the time axis. Fig.2.3 (b) is the same type of vehicle running a little faster and on a different ground. It is obvious that several harmonics have disappeared and reappeared repeatedly during the 40 seconds recording period. This non-stationarity is a very common phenomenon in vehicle acoustic signals. In general, vehicle signal maintains stationary within a 250 ms or shorter window. If longer than that, many harmonics will gradually shift away or even disappear. Sometimes, they shift upward or downward in a synchronized manner as in Fig. 2.3(a). More frequently, they show quite random shifting pattern as in Fig2.3 (b). This kind of fluctuation within 1 second can be classified as short term non-stationarity.

Fig.2.4 shows the 1-D spectrum obtained by collapsing Fig2.3's Time-Frequency representation along the time axis (after LIN). Although the second recording is only 5 km/hr faster than the first one, we observe significant difference between the two signals. Clearly there are a new harmonics appeared around 60 Hz in the 10-km/hr case. When vehicles runs at different speeds, with different gears, the sound will change accordingly. For vehicle classification, this varying spectrum causes even more troubles than the short term non-stationary effect, because one vehicle running at one speed may have similar spectrum as another type of vehicle running at a different speed. During the transitory states that a vehicle engine changes its working state, the problem becomes even more complicated.

To summarize, two types of spectrum fluctuations exist in vehicle signal, the first is **short-term** non-stationarity, the second is **long-term** spectrum variation caused by different vehicle speeds or different gears.

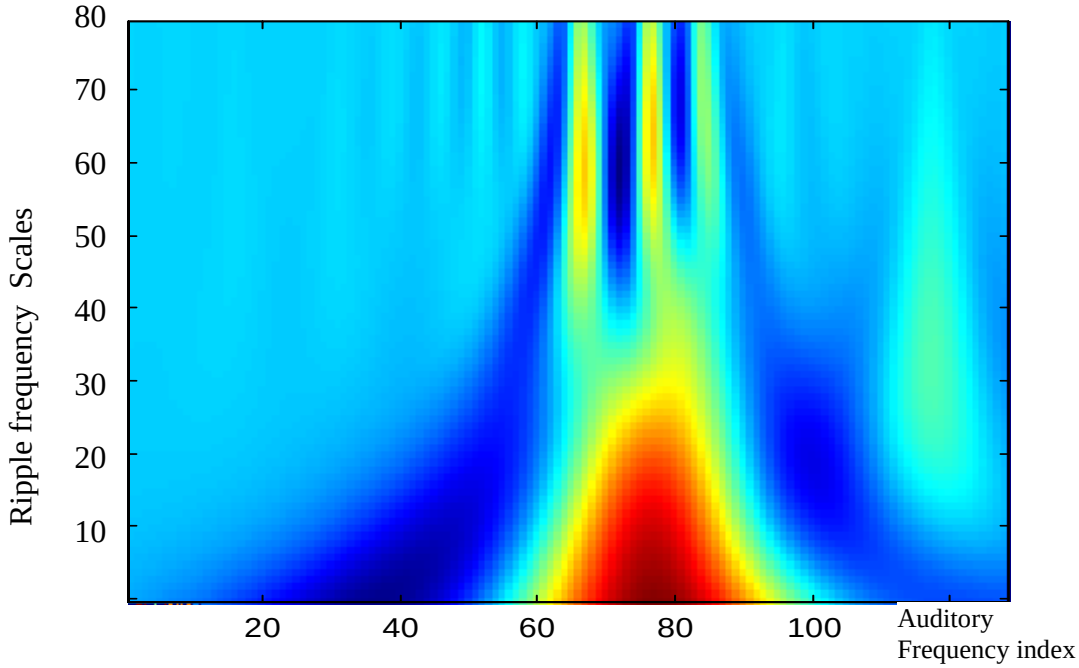


Figure 2.5 Multi-resolution representation from cortical module

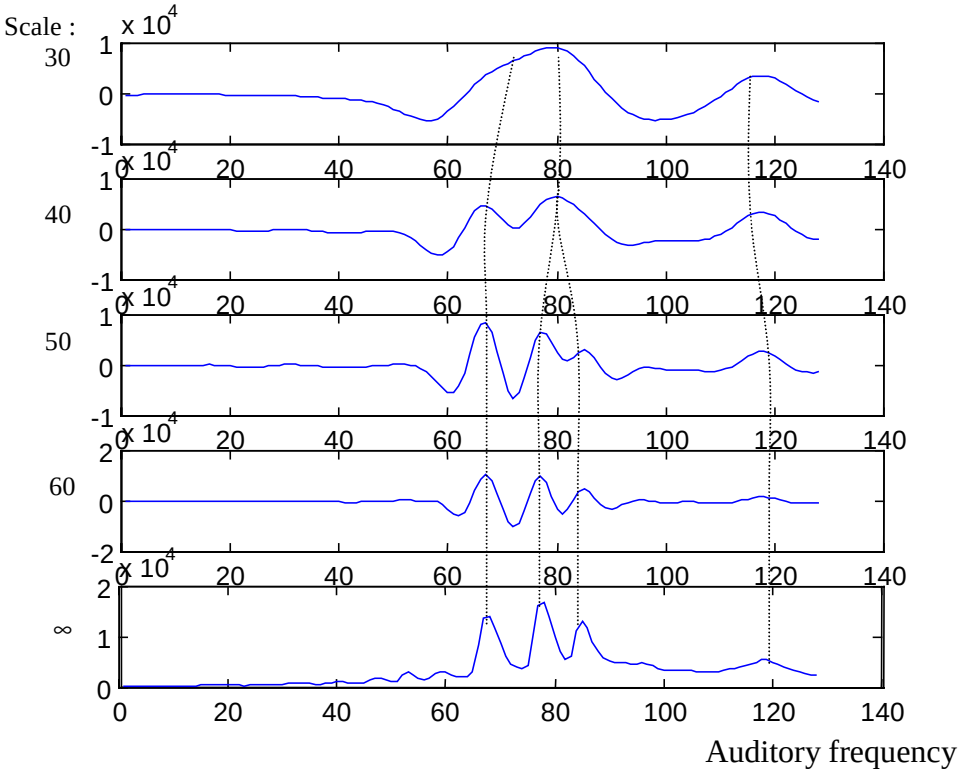


Figure 2.6. Cortical representation at different scales. From top to bottom are scales 30, 40, 50, 60, and '∞'. Where '∞' means raw spectrum from auditory module.

(2) Simulation on cortical processing.

Fig. 2.5-6 give the cortical processing pattern for the auditory spectrum. This pattern clearly demonstrates the following properties:

- The cortical processing of the auditory spectrum is conceptually an affine wavelet transform. Since the auditory wavelet also use logarithmic frequency scale like other wavelet transform, its harmonics are evenly distributed on the frequency axis.
- The coarse scale (low ripple frequency) captures the broad and skewed distribution of energy in the auditory spectrum, while the finer scale captured the detailed harmonics structure. In the other intermediate cortical scales (such as scales 30 to 60), the dominant harmonics are highlighted while the weaker ones are suppressed. For example, in Fig.2.6, the weak oscillation between band 40 to 60 in raw auditory spectrum (scale ∞) is not observable for scales between 30 to 60. Thus these intermediate scales emphasize the most valuable perceptual features within the signal.
- From fig.2.6, we can clearly see the multi-resolution character of the cortical representation. This figure reminds us of the same phenomenon as in the radar return research [3]. Along the artificial vertical lines we gradually extract all the harmonics just the same way as we extract local peaks in ship radar returns; the only difference is that a biological model-based wavelet transform, instead of an orthogonal wavelet transform, is used here.
- The cortical filter is a redundant representation, not all the scales are necessary for the classification algorithm. Fig.2.6 clearly suggests that 3~5 scales are sufficient. Since we know that higher scale cortical representation preserves more harmonic details than lower scale, while lower scale (here ‘lower’ refers to scales between 0 and 20) is a better

descriptor of the spectral contour. Normally, the spectral contour at lower scales is relatively insensitive to speed variations, which is a valuable characteristic for classification problems. Nevertheless, since most vehicle harmonics are crowded within the range of 20Hz to 200 Hz, these coarse scale spectral contours are very similar to each other. Due to the limited resolution, the classification decision is not very reliable if based solely on low resolution information. Meanwhile, since most intermediate scales highlight the perceptually important components in the auditory spectrum, they are better candidates for invariant features. Therefore, in future research, the scale [30 40 50 ∞] will be consistently used in the classification system.

Chapter 3. VQ based classification algorithm

3.1 Motivation

Once we implemented the biological hearing model as in Chapter 2, the job left is to implement a suitable classification algorithm. Up to now, the features we obtained are multi-resolution auditory spectrum. As physiological and psychological experiments show [26], cortical neurons exhibit certain organizational characteristics that reflect systematic response selectivity to various stimulus features. Those response areas sensitive to different ripple frequencies are organized topographically across the surface of the cortex. This topographic organization leads to the natural aggressive recognition capability, as we experience it in life daily. Normally, the best way to model this aggressive hearing capability would be a tree-structured multi-resolution classifier. At lower levels (coarse resolution) of the tree, the cortex only performs preliminary and indecisive classifications. As the sound becomes clearer, and more information becomes available, the cortex will carry out more precise and decisive classification. In this chapter, the TSVQ based classification algorithm will be consistently utilized; its tree structure is the best imitation of the cortical system because it is both hierarchically layered and topologically distributed. In this sense, our system is more biologically motivated than other systems, such as systems based on fuzzy logic membership or genetic algorithms (GA).

In our research, we studied 3 different VQ based classifiers: LVQ, TSVQ and a parallel TSVQ (PTSVQ) algorithm. Among them, LVQ is an optimal Bayes classifier and the slowest one, while the other two algorithms are not optimal but much faster and more efficient to implement. Generally, VQ is known as a tool for multidimensional data

compression, however, classification and compression have long been known to be highly correlated problems. Recent work has lead to very beneficial cross-fertilization between the two fields, in particular, between TSVQ compression and classification trees [3]. In general, classification of different features can be viewed as a form of compression since it associates each input vector with a class label. Conversely, compression can be viewed as a special form of classification, since it assigns a template or code word in a small set to the input features drawn from a large set in such a way as to provide a good approximation. All inputs sharing the same code word can be deemed as a common class. In this sense, the VQ compression algorithm can be considered as an unsupervised classifier. Although its classification performance is not optimal in the Bayes sense, it offers significant advantages such as memory saving and fast searching and training. This is true especially for the tree structured VQ algorithms. In this thesis, our goal is to improve TSVQ's classification performance as close to the optimal LVQ as possible. The basic idea behind is to design a combined system that takes advantage from both systems.

3.2 Learning Vector quantization (LVQ)

Learning vector quantization (LVQ) is a non-parametric method of pattern classification. As a supervised learning neural network, LVQ works in two stages: In the training stage, it uses a set of training data to divide the feature space into non-overlapped Voronoi cells. Later during the testing stage, it applies the nearest neighbor rule to classify the new input. The following section outlines the basic LVQ algorithm:

Define input vector x, training data population N, codebook of size K with Voronoi vectors m_i, i=1, 2, …, K. Then x is decided to belong to Voronoi cell c if

$$c = \arg \min_{i=1:K} \left(\left\| x - m_i \right\|^2 \right)$$

In the training process, the m_i are updated using the following equations:

$$\begin{cases} m_c(t+1) = m_c(t) + \alpha(t)[x(t) - m_c(t)] & \text{if x(t) and } m_c \text{ belong to the same class} \\ m_c(t+1) = m_c(t) - \alpha(t)[x(t) - m_c(t)] & \text{if x(t) and } m_c \text{ belong to different classes} \\ m_i(t+1) = m_i(t) & \text{if x(t) does not belong to cell c} \end{cases}$$

.

Here $0<\alpha(t)<1$ is the learning rate, it may be constant or decrease monotonically with time. After repeating the above training process sufficient times, the algorithm converges to a stable state. In [1] and [2], Baras and Lavigna proved that the classification error of LVQ converges to the optimal Bayes classification error as long as the volume of the Voronoi cells goes to zeros as $K\to\infty$, provided we have $\underset{N\to\infty}{Lim}(K/N)\to 0$. Therefore, LVQ serves as an optimal calssifier in our research. It provides a upper bound for the achievable classification performance.

One weakness of LVQ is that it is extremely difficult to be trained to the global optimal state, especially when a huge volume of data is used as the training set.In [21] [27], Kohonen points out that the convergence of the LVQ network depends on the following factors: initial node allocation among different classes, initial Voronoi vector position, learning rate and simulated annealing schemes, and times of presenting the training data to the network. Direct training of the LVQ from a random initial state is normally not successful, the most widely used way of training a LVQ network is using a VQ algorithm to pre-cluster the training data, then the LVQ algorithm inherits the Voronoi vectors from it, and continues the training until the LVQ algorithm converges. In this sense, a robust and effective VQ classification algorithm is very important for LVQ, because VQ with poor classificarion performance can not help LVQ to overcome the local minima in a large vector space.

3.3 Tree structure vector quantization (TSVQ)

3.3.1. Definitions

In this section, we Define the VQ as an unsupervised clustering algorithm: An N-dimensional vector quantizer consists of an encoder γ mapping an N-dimensional vector space X to a set of code symbols F and a decoder δ mapping these code symbols to a reproduction alphabet A. For a given code symbol $F \in F$ if we let l(F) denote its length (in bits) then we can define the average rate R in bits per vector of a given encoder γ by R=E[$l(\gamma(X))$], where the expectation arises from our chosen probabilistic model for the random vector X. The distortion between any input vector $x \in X$ and its reproduction $\delta(\gamma(x))$ is defined as d(x, $\delta(\gamma(x))$), with which one defines the average distortion of a given VQ to be E[d(X, $\delta(\gamma(X))$)]. In this thesis, we take the widely used squared error as distortion measure because of its simplicity: $d(X, \delta(\gamma(X))) = \|X - \delta(\gamma(X))\|^2$, where X=(X(1), …, X(k)) is a k-dimensional vector.

3.3.2 The classic LBG algorithm

This is the most common approach to VQ training. It repeatedly uses clustering techniques to minimize the average distortion subject to the constrains on bit rate and code structure. The LBG clustering algorithm can be summarized in the following steps:

- Given a codebook $\{ m_i \}$, the optimal partition $\{ R_i \}$ of the signal space that minimizes distortion D_{ave} is based on the nearest neighbor rule. In our case, it is the minimum mean square error (MMSE) rule.

$$R_i = \{ x : d(x, m_i) \le d(x, m_j), \forall j \}$$

where d(.) is the distance measure and D_{ave} is the average distortion.

$$D_{ave} = E(d(x, m_x))$$

- For a given partition *P*, the optimal decoder assigns to each index i the conditional centroid of all input vectors X for which γ(x)=i. In our case of squared distance, the representative of the current partition region is the conditional expectation E(x|γ(x)=i).
- For a given initial partition, we repeat the two steps, until a saturated state is reached.

3.3.3 TSVQ based on LGB algorithm

We are especially interested in tree structured VQ (TSVQ) because it is consistent with the aggressive perceptual model and it represents a natural way to use multi-resolution feature vectors. Furthermore, TSVQ has a logarithmic search time compared to the linear search time of a full search VQ, making TSVQ the most effective and widely used technique for reducing search complexity. In TSVQ, the search is performed at different scales. At each scale a substantial subset of candidate Voronoi cells is eliminated. In a binary balanced tree with depth L, we only need 2L comparisons before we find the best match.

Normally, a TSVQ tree is grown by successively splitting nodes and then optimally pruning them until the desired rate is reached. In our research, we follow the greedy method described in [3] to construct the tree. The basic problem here is whether the splitting should be done in the current layer or down to a new layer.

When we get the multi-resolution representation of the features, we first partition the feature space into non-overlapping Voronoi cells by repeatedly applying the LBG algorithm. LBG is first applied to the coarsest resolution, the resultant distortion is

determined by the mean squared distance metric, and is computed using the finest resolution representation of the data. The cell that contributes most to the total average distortion is the cell which is split in the next application of LBG. A new Voronoi vector is found near the Voronoi vector for the cell to be split and is added to the Voronoi vectors previously used for LBG. LBG is applied to the entire population of data vectors, again using the coarsest representation of each vector. These steps are repeated until the percentage reduction in distortion for the entire population falls below a predetermined threshold. Then the partition in the coarsest resolution is fixed, and further partitioning continues by splitting the existing cells based on finer representation of the data in the cell. The algorithm then iterates until the allotted number of cells has been reached[3].

The whole process can be summarized into the following statements:

- For a given block I which contains J cells at scale M, we compute the average distortion $D_{I,J}^{M} = \frac{\sum_{j=1}^{J} \sum_{x_I^M \in cellj} \left\| X_I^M - m_{I,j}^M \right\|^2}{N}$,Where m_{Ij}^M is the centroid of cell j at scale M and N is the total number of observations.

- Compute $\Delta D_{IJ}^{M} = \frac{D_{I,J-1}^{M} - D_{IJ}^{M}}{D_{I,J-1}^{M}}$, if it is larger than a prefixed threshold, than new centroid J+1 is added at the same scale, otherwise goes down to scale m+1.

After the training stage, all Voronoi cells are labeled using majority voting, i.e. if class k dominate Voronoi cell j, then in the testing stage, all samples falling into cell j will be classified as class k. After the above training procedure, an hierarchical multi-resolution classifier is available.

3.4 Parallel TSVQ (PTSVQ)

In [3][6], Baras and Wolk introduced a Parallel TSVQ structure that shows superior classification performance. The algorithm works in the following way: during the training stage, features from different vehicles will be used to construct independent subtrees, generally one tree for each type of vehicle. After that, the algorithm goes into testing stage, each new input vector will be presented to all the subtrees in parallel, and being processed in the usual way it is processed in the former TSVQ. Once settled in leaf nodes in all the subtrees, we calculate the minimal distances between the new input vector and the centroid of its settled Voronoi cell for each subtree. If the subtree that has the minimal distance corresponds to type k vehicle, we declare a type k classification. In the following part of this thesis, we will refer to this method as Parallel TSVQ (PTSVQ) method, while the traditional method will be referred to as Global TSVQ (GTSVQ).

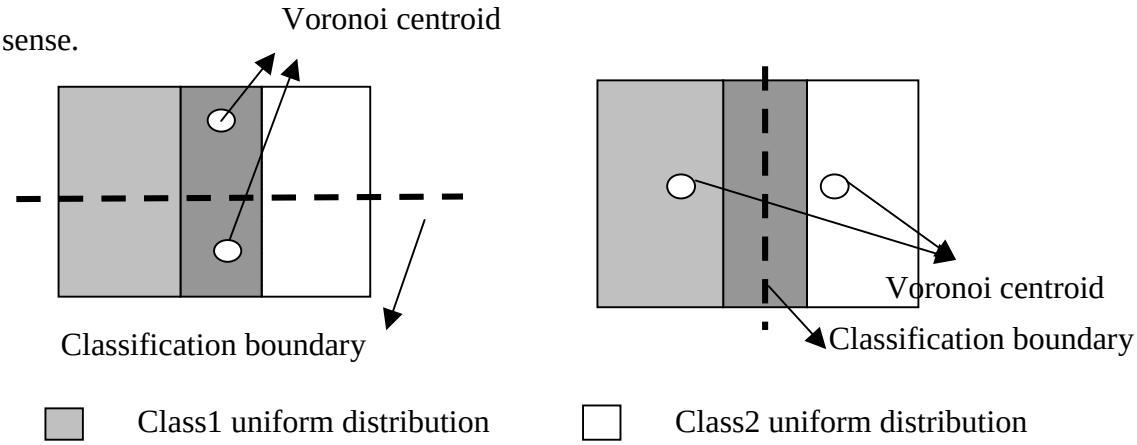
The PTSVQ has been shown to be successful in ship radar return classification. It achieves classification rate close to the optimal LVQ, while the search time is comparable to the logarithmic search time of the traditional GTSVQ. Furthermore, since only one subtree will be involved in the training stage, on-line training can be easily implemented, and the “new target insertion” in real time systems will also be possible.

The primary problem associated with PTSVQ is that it is totally heuristic. In the next chapter, we will examine this algorithm through a series of simulations for the vehicle classification problem. In this way, we hope to gain more insight from it and provide some useful results for later theoretical study.

3.4.1 PTSVQ vs. GTSVQ

The superior classification performance of PTSVQ originates from two aspects. First, the 'one subtree for each pattern' structure will approximate individual pattern density more precisely than the global tree structure. Secondly, PTSVQ uses a little more search time than GTSVQ. Here, an example is presented to illustrate the first aspect.

In Fig.3.1, two classes exist in the 2-D vector space. Assume the two classes have the same prior probability, both patterns are of spatial uniform distribution within their definition boundary, and their density functions overlap in the middle. In the overlapped region, we have higher compound density than the rest of the region. After using LBG algorithm to assign two Voronoi nodes to this vector space, we get a result shown in Fig.3.1 (a), If we perform LBG for these two classes separately (as in PTSVQ), we get a result shown in Fig.3.1 (b). Using the nearest neighbor partition, we obtain the classification boundary. Obviously the two resultant partition boundaries are different, and the parallel subtree scheme yields a correct classification boundary in the Bayes sense.



(a) Clustering by Global LBG (b) Clustering by independent subtree

Figure 3.1 Classification gain from independent clustering of different classes

In general, when two patterns are highly overlapped, PTSVQ will achieve better performance. A heuristic explanation of this phenomenon is that: since the LBG algorithm distributes Voronoi nodes according to the underlying density functions, if we apply LBG to the whole sample space, the nodes distribution will approximate the compound density function. If we apply LBG to each pattern independently, the node distribution will approximate the density function of each individual class. When classification is concerned, node distribution according to individual density function will lead to more meaningful classification boundary. E.g., the LBG algorithm will put more Voronoi cells on the high compound probability density areas, while these areas may just lie on the Bayes classification boundary. Therefore, PTSVQ can be deemed as supervised algorithm; all ID information is incorporated into the training process.

We should also notice that PTSVQ could not approach the optimal Bayes classification, even when the number of Voronoi nodes goes to infinity. This can be proved using a simple example. In Fig.3.2(a) two patterns exist in the 2-D vector space, both patterns are of spatial uniform distribution within some rectangle regions, and their density functions overlap in the left rectangle. Assume the two classes have equal prior probabilities. In the overlapped region, Class B has higher regional density than Class A, therefore by Bayes criterion, the whole left rectangle should belong to class B's classification region. After using LBG algorithm to these two patterns independently (as in PTSVQ case), we get a classification partition as in Fig3.2 (d). As a result, class A will always have some nodes left in the left rectangle. Finally, some areas in the left rectangle are mistakenly assigned to class A. Therefore, PTSVQ can not approach Bayes classification in this case. The underlying reason is that, in PTSVQ, LBG is carried out

independently for individual class, it doesn't concern which class has a higher relative a posterior probability for an interested region. While Bayes classification is based on the maximal a posterior criterion, it carefully examines which class has the highest a posterior probability within interested region, and will declare a classification for that class.

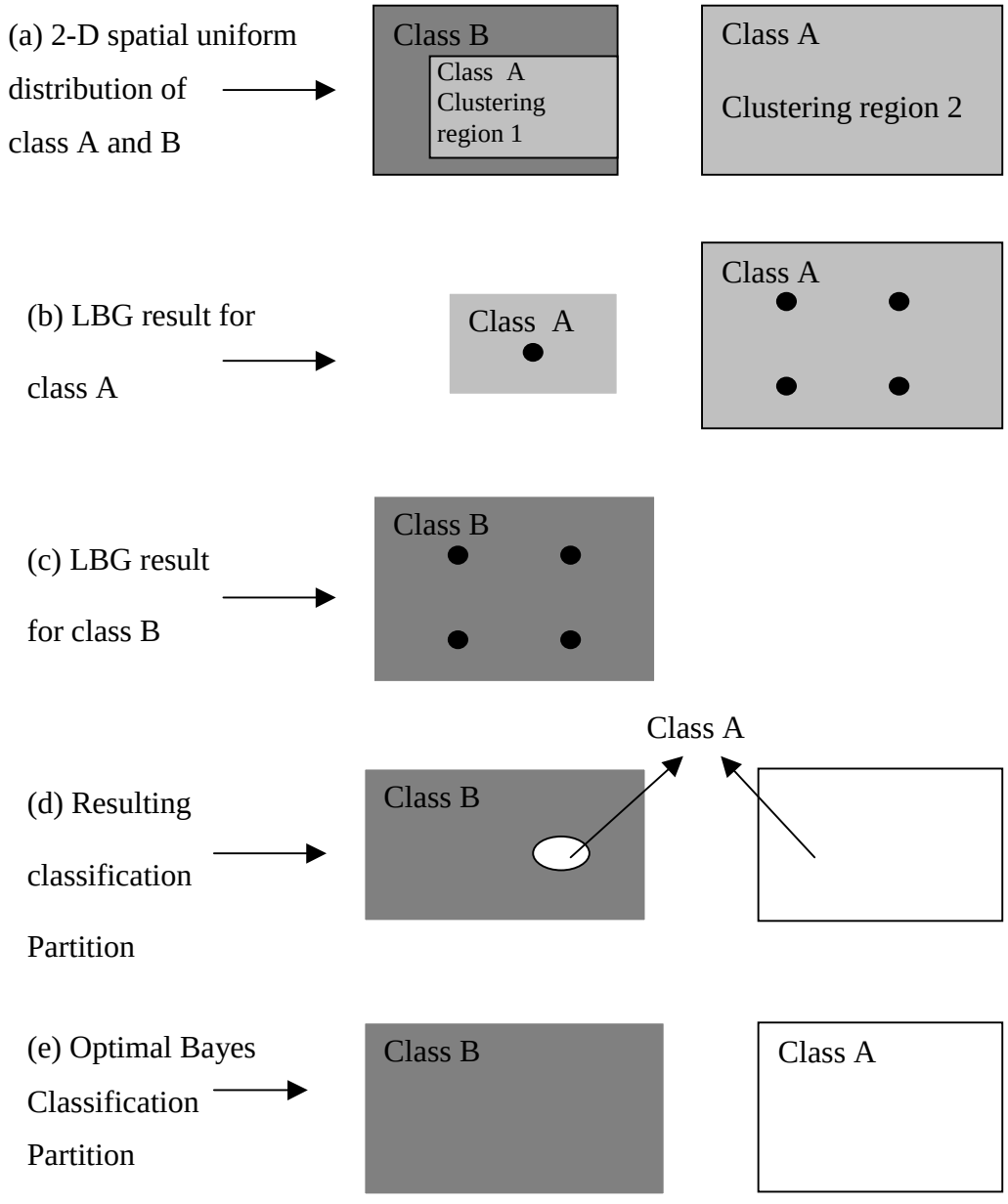


Figure 3.2 Difference between PTSVQ and Bayes optimal classification

3.4.2 Comparison in search time

In this section we will prove that PTSVQ uses a logarithmic search time. To keep the derivation simple, we assume that all trees constructed are symmetric full-balanced trees. Later experiments will show that this assumption will not seriously affect our result.

(1) The GTSVQ case: For an L scales multi-resolution representation, if we assign F leaf nodes to an M-ary full balanced tree, then:

$$M^L = F \quad \text{or} \quad M = F^{1/L}$$

On the average, each parent node has $F^{1/L}$ children. In each step, the input vector should examine all the children in the next layer to find out the next branch to go. The average search time for the GTSVQ tree is:

$$\overline{S}_{CTSVQ} = L * M = L * F^{1/L}$$

(2) The PTSVQ case: In total, there are N classes, each has its own subtree. To make the comparison fair, the same number of leaf nodes are assigned to the two algorithms, so in average, we have F/N leaf nodes for each subtree. If each subtree is also fully balanced, the search time for each subtree will be:

$$\overline{S_i} = L * M_i = L * (F / N)^{1/L} \qquad for \qquad i = 1,2,\cdots N$$

Therefore, the total search time for PTSVQ is:

$$\overline{S}_{PSTSVQ} = N * \overline{S_i} = N * L * (F / N)^{1/L}$$
$$= N^{\frac{L-1}{L}} \overline{S}_{CTSVQ}$$

Compared to the GTSVQ case, the PTSVQ has a factor of $N^{\frac{L-1}{L}}$. When N and L are fixed, the PTSVQ has a logarithmic search time with respect to the number of leaf

nodes F. In our case, N=9 (9 classes), L=4(4 scales), the search time of PTSVQ is roughly 5.19 times that of GTSVQ.

In GTSVQ, there are $\sum_{i=1}^{L-1} M^{i} = \sum_{i=1}^{L-1} F^{i/L}$ intermediate nodes. While in PTSVQ, the number of intermediate nodes is:

$$N * \sum_{i=1}^{L-1} M_{i}^{\ i} = N * \sum_{i=1}^{L-1} (F/N)^{i/L} > \sum_{i=1}^{L-1} F^{i/L}$$

At the same time, the total search path goes from 1 (in the GTSVQ case) to N (PTSVQ case). To sum up, the PTSVQ keep a logarithmic search time with respect to the number of leaf nodes, while it searches more branches and a little more intermediate nodes during the testing stage. In practical, the greedy TSVQ algorithm may lead to many complicated unbalanced tree, so some assumptions here may not be valid, but later experiment shows the above conclusion are very close to the true value and can be used for coarse evaluation of the search speed.

3.4.3 Node allocation schemes for PTSVQ

How to allocate leaf nodes among all subtrees is still an unsolved problem in PTSVQ. In a VQ based classifier, classification decision is based on the nearest neighbor criterion, therefore, the more nodes one class gets, the better the classification for this class, and the worse the other classes will be. In this thesis, we tried the following ad hoc node allocation strategies. By comparison between these schemes, it is hopeful to gain more insight into PTSVQ algorithm which may be helpful in the future theoretical study.

(1) Allocation according to sample a prior probability:

This is a straightforward approach. The basic idea behind is that the class having more training and testing samples should have more nodes assigned to it.

(2) Allocation according to equal distortion of each TSVQ subtree.

This method is based upon the assumption that classes with a condensed distribution would need less nodes to represent than classes with a sparse distribution. In the extreme case, all samples belong to one class fall into one point in the N-dimensional vector space, then one Voronoi centroid vector is enough to represent this class, no matter how many samples it has. Therefore, a possible 'fair' way to distribute the leaf nodes would be the one that, after the node allocation, all subtrees have the same mean square distortion.

(3) One subtree for each speed:

The dominant difficulty of vehicle acoustic signal classification lies in the fact that the auditory spectrum changes with different working conditions. Studies on ACIDS data show most spectrum fluctuations are caused by speed changes, while the terrain has less severe influences on it. Normally, when a vehicle changes speed, either new harmonics show up or disappear, which corresponds to gear change, or the harmonics gradually shift their relative position on the frequency axis, which corresponds to varying engine vibration period. Therefore, the auditory spectrum from vehicles with the same speed turns to group together in the vector space, and the whole feature space appears to be a combination of several clustering areas, each cluster corresponds to a particular vehicle running at a specific speed. In this case, it is a natural attempt to construct a subtree for each such clustering area. Another advantage of this scheme is that "one tree for each speed" maximally approximates the cortical processing, which may be just topologically distributed neuron sensitive to specific stimuli. Finally, One subtree for each speed may be a better candidate for real time on-line training, since in this case, the

size of each subtree is further reduced. When new training data come, only the corresponding subtree are updated, all the other subtree remain unaffected.

3.4 Decision fusion

As a last step in our classification system, we perform a simple decision fusion operation to improve the classification performance. Fig.3.3 illustrates this approach. Each one-second input signal is segmented in 250 ms block, each block goes through the proposed classifier, and provides a sub-decision. 4 such consecutive sub-decisions in a row are feed into the decision fusion unit, where a majority voting operation will settle the final classification. The basic idea behind this scheme is that the vehicle signal often has severe short time fluctuations in the spectrum, and a majority voting can alleviate the associated short time fluctuation. Finally, this scheme introduces 1-second delay in the overall system, such a small cost is generally affordable in practical system design.

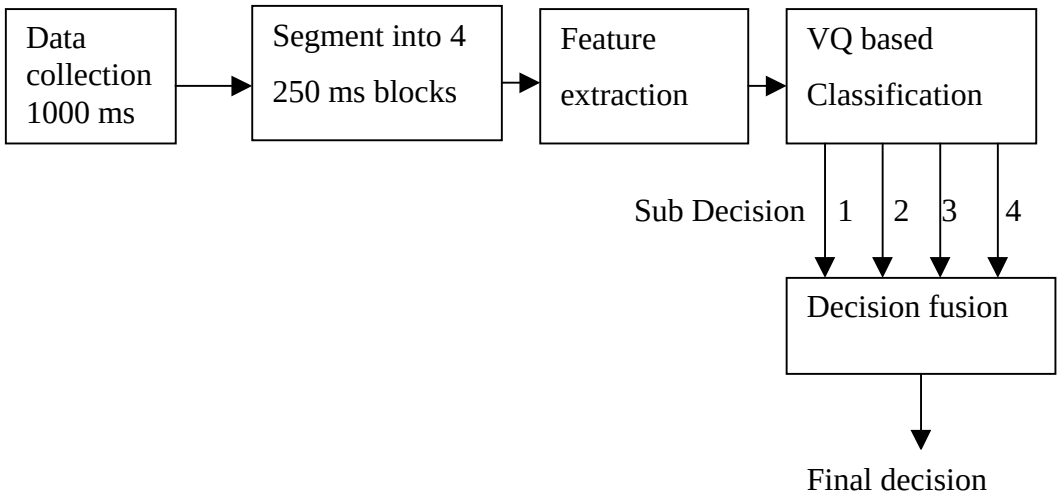


Figure 3.3 Decision fusion unit

Chapter 4 System implementation, Simulation and Discussion

In this chapter, all proposed VQ based classifiers are trained, tested, and compared with each other. The performance is measured in both classification rate and search time. When comparing different classifiers, we use the same sets of training and testing samples, and the same amount of Voronoi cells, thus make the comparison fair. In all the experiments, the feature extraction system is based on the biological models introduced in Chapter 2.

4.1 Data preprocessing

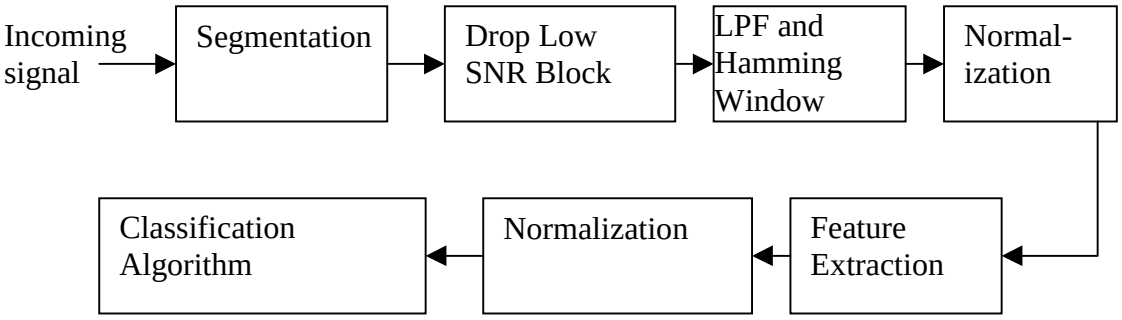


Figure 4.1 Data preprocessing in the system

Incoming signal waveform is first segmented into equal length blocks. For a classification system, short block length is preferred since it leads to small classification delay. In our classification system, the block size is fixed to 250 point, shorter than that will make the followed processing such as filtering and spectrum analyses unreliable. Since the sampling rate is 1025 Hz, one such frame corresponds to roughly 250 ms. As discussed in Chapter 2, vehicle signals are approximately “stationary” in such short

duration. Combined with a decision fusion unit that corrects any burst error from short time fluctuation, 250 point is proved to be an appropriate processing window.

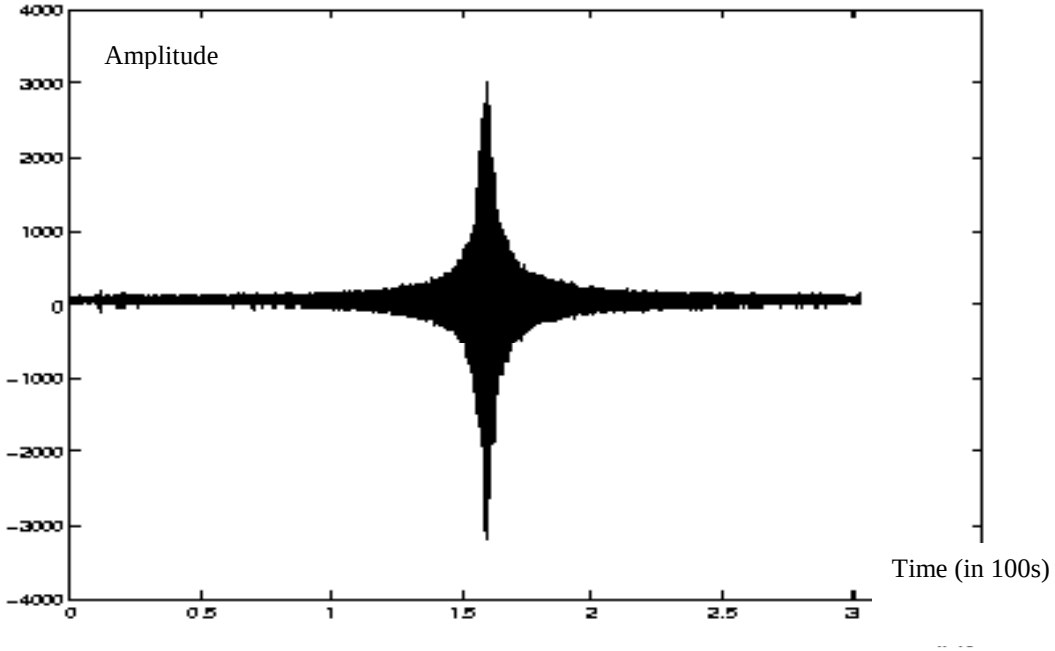


Figure 4.2 A typical vehicle acoustic signal waveform

Fig 4.2 shows a typical recording in the ACIDS database. Although the whole recording lasts more than 300 seconds, most part is too weak for classification purpose, some even undistinguishable from the background noise. For this reason, after computing the energy of each block, only the strongest 40 seconds from each recording are processed, all the other low SNR blocks are dropped. Next, a low pass filter with 450Hz stop frequency gets rid of the high frequency wind noise, and a hamming window added to the raw data reduces the spectral side lobe. Before entering the feature extraction system, each block is normalized into zero mean and unit variance frame. Each frame of data is processed through cochlear and cortical filter banks as discussed in Chapter 2. Through above procedures, a multi-resolution auditory spectrum is available. Before this

representation enters the classification system, it is normalized to zero mean and unit variance again. This second normalization is very important because our VQ based classifier uses L^2 norm as distance measure; un-normalized feature vectors will make the classification unfair for different samples.

4.2 TSVQ for aggressive classification

This part will demonstrate the aggressive classification capability of the system. Fig. 4.3 and 4.4 show the tree constructed by the method introduced in Chapter 3.3. Here 6 types of vehicle are employed to evaluate the TSVQ algorithm. For each type, 3 recordings of different speed and different ground condition are used. Therefore, even under stationary assumption, there are 18 perceptually different sounds present to the system (similar to speaker-independent phoneme recognition in speech recognition). Since our goal is preliminary evaluation of TSVQ algorithm, a reduced size database is used. Fig.4.3 illustrates the resulting Voronoi centroid in each cell, and Fig.4.4 illustrates the histogram in each cell.

In the top layers of the tree, the TSVQ algorithm clusters acoustic signals according to their spectral profile, most cells are occupied by multiple classes. As we move to finer resolution, detailed harmonics structure is available, and the situation gets better. When we reach the leaf layer, most Voronoi cells are occupied by samples from a specific class. This phenomenon also confirms our hypothesis in section 2.3 that fine resolution cortical representation is more reliable in separating different classes than coarse resolution representation.

Fig.4.5 gives a detailed example of how cells are split. In cell 1-3-0, patterns of class 1 and 2 coexist. As we move to next layer (layer 2), class1 dominates cell 2-3-0,

while class 2 dominates cell 2-3-1, and clear difference can be observed between the feature vectors within the two cells.

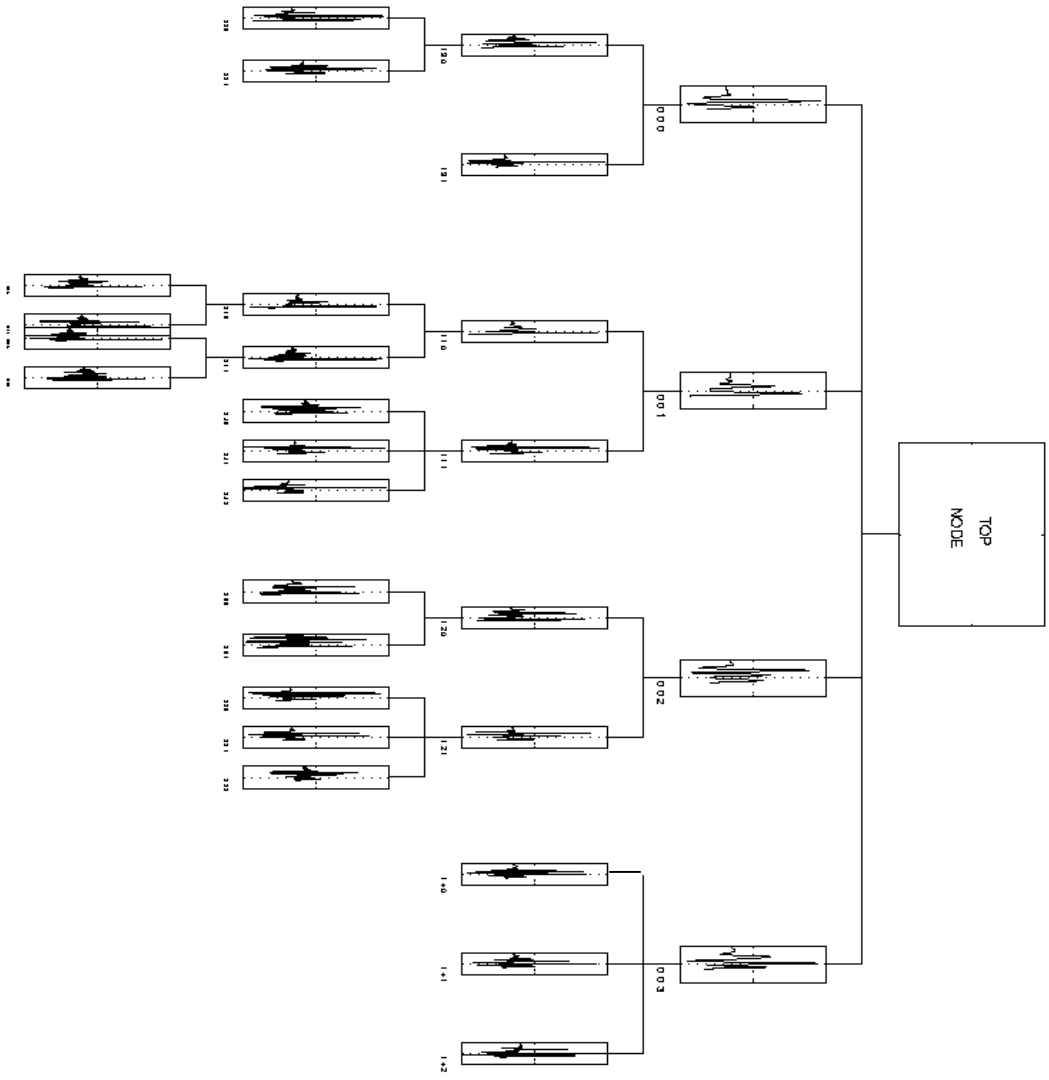


Figure 4.3 Multi-resolution tree constructed by the TSVQ algorithm, the voronoi centroid vector are plotted in each cell

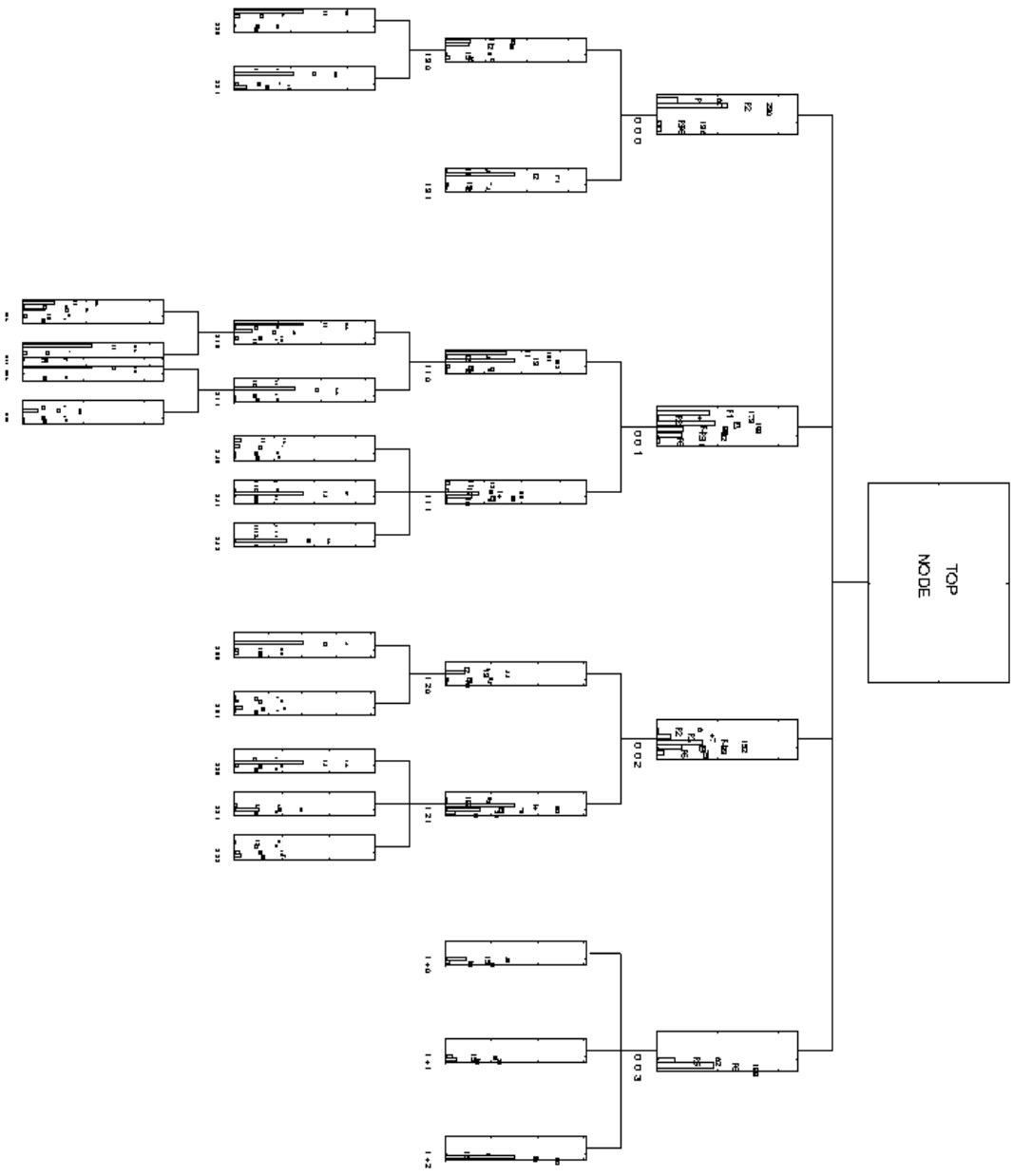


Figure 4.4 Multi-resolution tree constructed by the TSVQ algorithm, the histogram of each cell are plotted correspondingly.

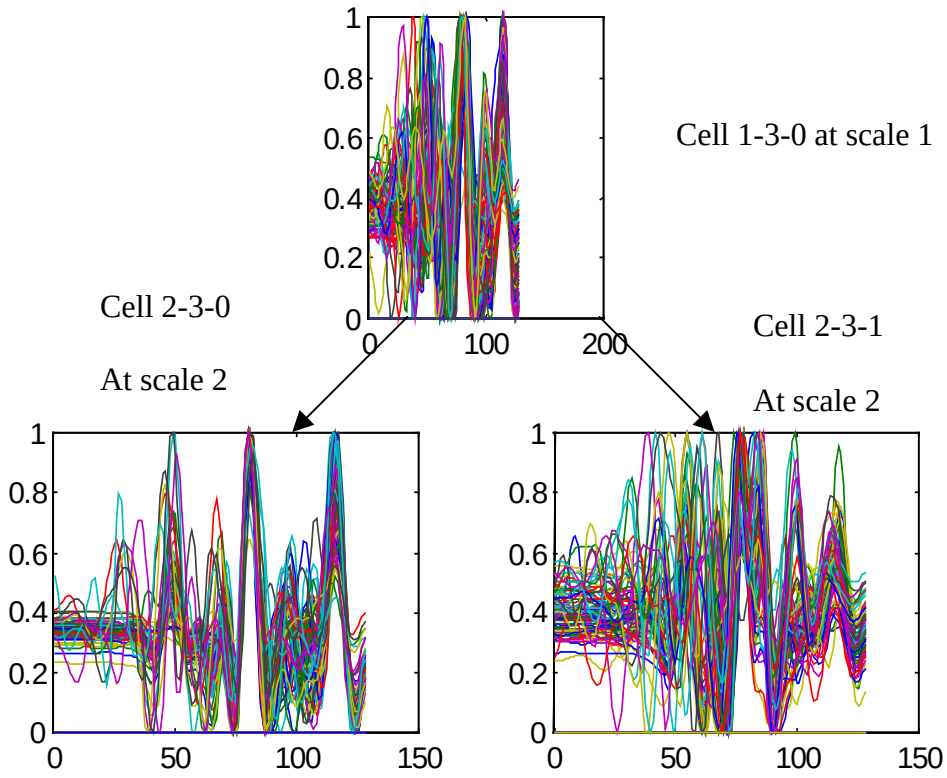


Figure 4.5 Cell 1-3-0 is split into cell 2-3-0 and 2-3-1

4.3 Different node allocation schemes

From now on, the entire ACIDS data will be used. In total, 43840 multi-resolution feature vectors from 274 recordings of all 9 types of vehicles are available. Among them, 70% is used for training and the rest 30% used for testing. Three VQ based classifiers are examined here. For GTSVQ, since the whole tree is labeled automatically using majority voting, no specific node allocation scheme is needed. The LVQ algorithm is initialized using the Voronoi centroid resulting from PTSVQ, therefore its node allocation scheme is the same as PTSVQ. Here, we briefly introduce 3 different node allocation schemes for PTSVQ as in chapter 4.

- Allocation based on sample a prior probability:

The number of leaf nodes of each class is proportional to the prior probability of each class, as shown in table 4.1

Class	1	2	3	4	5	6	7	8	9	Total Node
Train samples	3487	2068	515	1504	2167	2082	397	1990	1134	
Case1	24	14	4	10	15	14	3	14	8	106
Case2	31	18	5	13	19	19	4	18	10	137
Case3	35	22	5	16	23	22	4	21	12	160
Case4	47	28	7	20	29	28	5	27	15	206
Case5	50	30	7	22	30	30	6	29	16	220
Case6	62	37	9	27	39	37	7	36	20	274

Table 4.1 Node allocation according to sample prior probability

- Node allocation based on equal distortion:

This method is based on the hypothesis that the average distortion for each subtree should be the same after vector quantization. To implement this scheme, we first compute 5~6 rate-distortion pairs for each subtree, and interpolate them into a complete rate-distortion curve. When all 9 rate-distortion curves are ready, we fix a common distortion for all subtrees, and use the rate-distortion curves to find corresponding rate (number of leaf nodes) for each subtree. The rate distortion curves for all 9 classes are plotted in fig.4.6, The resulting node allocation scheme is given in table4.2.

- Node allocation according to vehicle speed.

In this scheme, we build a subtree for each speed of each vehicle. In the ACIDS database, there are 4 different speed values: 5, 10, 15, 30km/hr. Since 10 km/hr recordings are rare, they are grouped into the 5 km/hr category. In total, there are 27 subclasses, corresponding to 27 subtrees. Finally, we assign leaf nodes to these 27 subtrees according to a prior probability of each subclass. The resulting scheme

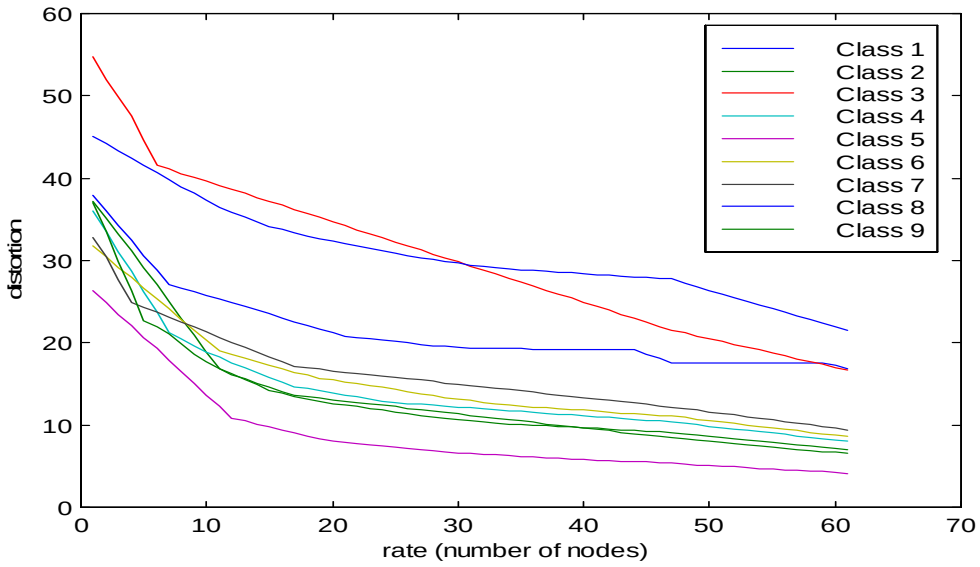


Figure 4.6 rate distortion curves for 9 subtrees

Node	Tree	1	2	3	4	5	6	7	8	9	Total
Case 1	Rate (Node)	8	8	33	6	3	5	5	32	6	106
	Distortion	30.7	29.2	29.8	31.1	27.7	30.4	30.4	30.1	29.9	
Case 2	Rate (Node)	10	9	35	8	3	8	6	51	7	137
	Distortion	27.1	27.0	26.9	26.2	27.8	26.6	27.7	26.8	26.3	
Case 3	Rate (Node)	15	10	43	8	5	9	7	56	7	160
	Distortion	24.8	25.0	24.9	26.2	24.9	25.4	24.9	25.1	26.3	
Case 4	Rate (Node)	24	12	52	10	8	12	13	65	10	206
	Distortion	20.8	20.9	20.8	21.2	20.7	21.6	27.3	21.0	21.1	
Case 5	Rate (Node)	29	12	53	12	8	13	15	67	11	220
	Distortion	19.9	20.9	20.1	19.6	20.6	20.2	20.1	20.1	19.8	
Case 6	Rate (Node)	50	14	61	15	10	17	19	74	14	274
	Distortion	18.1	18.9	18.1	18.2	17.9	18.1	18.3	17.9	17.7	

Table 4.2 Node allocation according to equal distortion.

is given in table 4.3. For vehicle without specific speed, we assign 0 node to it.

Tree	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6
1	9	11	13	18	18	23
2	8	11	13	15	18	22
3	7	8	10	12	14	17
4	3	4	5	6	7	9
5	5	7	8	10	11	14
6	5	7	8	10	11	14
7	0	0	0	0	0	0
8	2	3	3	6	4	5
9	2	3	2	3	4	4
10	0	0	0	0	0	0
11	5	7	8	9	11	14
12	5	6	8	9	10	13
13	0	0	0	0	0	0
14	8	10	12	15	17	21
15	7	9	11	12	14	18
16	0	0	0	0	0	0
17	9	11	13	15	18	23
18	5	7	8	12	11	14
19	0	0	0	0	0	0
20	2	3	3	6	4	5
21	2	3	2	3	3	2
22	0	0	2	3	2	3
23	12	14	14	21	22	26
24	2	3	4	6	5	6
25	0	0	0	0	0	0
26	5	6	8	9	10	13
27	3	4	5	6	6	8
Total	106	137	160	206	220	274

Table 4.3 Node allocation according to vehicle speed.

4.4 Classification performance and discussion

Our measure of performance is average probability of correct classification and total search time. Average probability of correct classification is defined as the total number of correct classification divided by total test population. While the total search time is defined in the following formula:

Total Search Time = $\sum_{i=1}^{test} \sum_{j=1}^{tree} \sum_{k=1}^{scale} S_{i,j,k}$

Where i: index of current testing sample j: index of current searching subtree

k: index of current searching layer *tree*: total subtree number

scale: all layer used in the searching until reach the leaf node

S: number of siblings need to be compared in current scale.

test: testing data population

The following definition is used to denote different classification scheme:

GTSVQ: Global TSVQ

PTSVQ (1): PTSVQ, node allocation according to sample a prior distribution,

PTSVQ (2): PTSVQ, node allocation according to equal distortion,

PTSVQ (3): PTSVQ, one subtree for each speed of each vehicle.

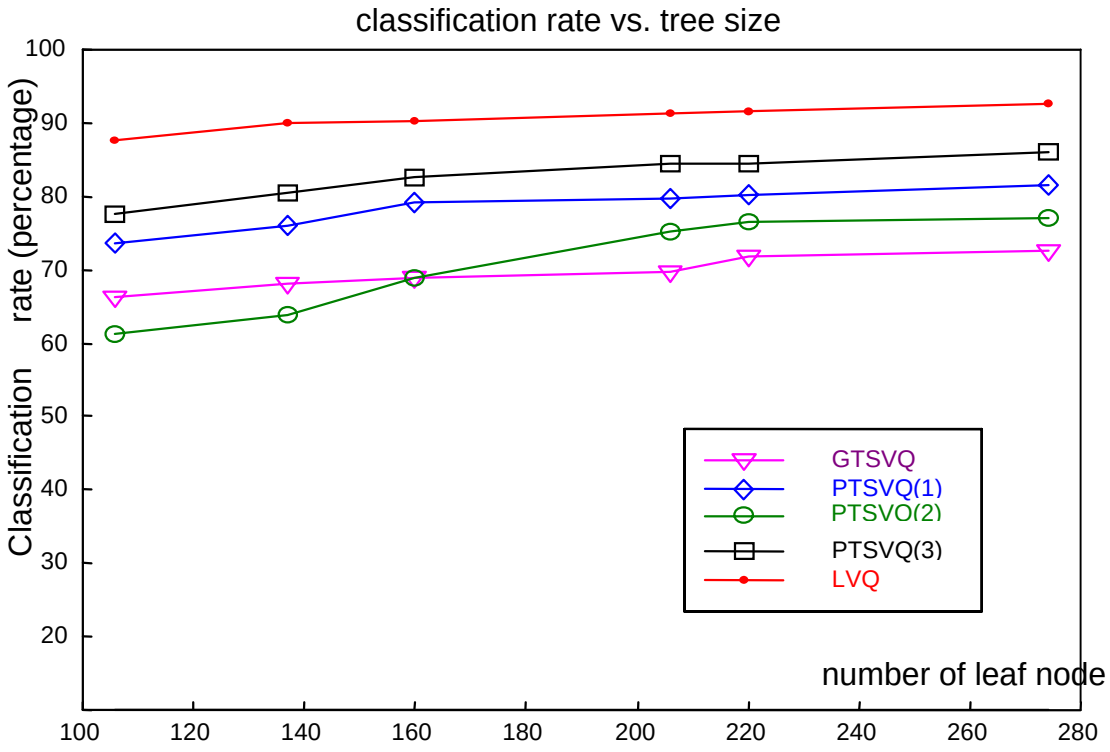


Figure 4.7 Classification performance for different classifiers

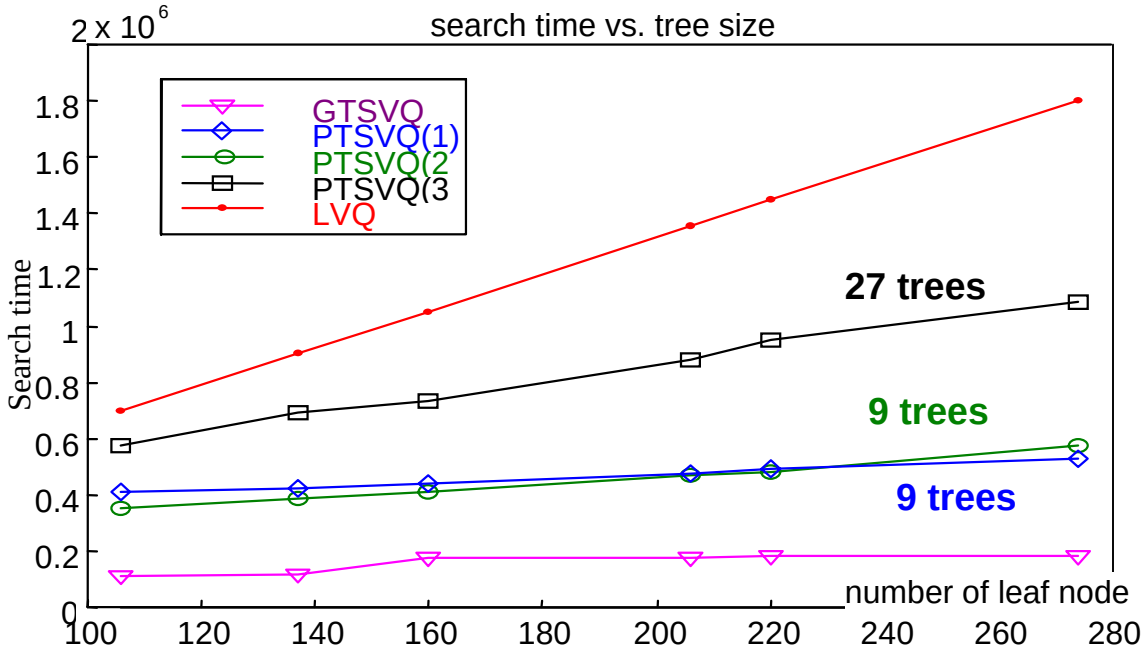


Figure 4.8 Total search time for different classifiers

Leaf Node	GTSVQ	PTSVQ(1)	PTSVQ(2)	PTSVQ(3)	LVQ
106	66.29	73.57	61.25	77.49	87.64
137	68.04	76.00	63.81	80.37	90.05
160	68.86	79.21	68.89	82.45	90.12
206	69.56	79.56	75.32	84.31	91.18
220	71.75	80.09	76.46	84.41	91.50
274	72.42	81.46	77.07	86.12	92.61

Table 4.4 Classification performance for different classifiers

The overall system performance is given in Fig. 4.7, 4.8 and Table 4.4. Based on these results, we summarize the outstanding features of these classifiers.

1. The LVQ has the best classification performance while GTSVQ has the worst performance. PTSVQ is an intermediate state between the two. In the simulation, all 3 PTSVQ schemes are better than GTSVQ. PTSVQ(3) provides about 13 percent

classification gain over GTSVQ, and PTSVQ(3) is about 7 percent lower than LVQ. We should notice that the comparison is not absolute 'fair' for PTSVQ (3), because LVQ uses PTSVQ(3)'s result as initial condition for further training, if equal amount of training time is devoted to PTSVQ, it may be further improved.

2. PTSVQ(1) and PTSVQ(2) use about 2 times the search time of GTSVQ, and their search time increases very slowly as total number of leave nodes increases, this result confirms the logarithmic search time hypothesis in chapter 4. PTSVQ(3) will use a little more time because the total number of leave nodes assigned to it is insufficient to build full balanced subtrees, as more training samples and more vehicles involved, the full balanced tree assumption will hold, and PTSVQ(3) will fall into the same category as PTSVQ(1). In addition, PTSVQ(3) has the highest level parallelism. In this experiment, if assign 1 CPU for each subtree, the total search time of PTSVQ(3) should be divided by 27, thus it will use far less search time than GTSVQ. So when classification speed is concerned, PTSVQ is the most promising scheme.

3. A serious problem with LVQ is that direct training of its neural network can not overcome local minimum. In our experiment, we have tried to directly implement LVQ from random initial conditions, but since both the training population and the dimensionality of input vector are fairly large (21920*0.7*128), neither the MATLAB LVQ tools nor Kohonen's LVQ-Pak software package converges to the global optimal. In the simulations, LVQ never achieved more than 80% classification from direct training. The convergence of LVQ network relies on too many factors, such as initial node allocation among classes, initial Voronoi centroid position, learning rate, simulate annealing scheme, and the times of presenting the training data to the network. TSVQ

algorithm, on the other hand, can easily converge to a stable state that corresponds to global minimal total distortion. Using results from GTSVQ as initial condition for further LVQ training shows improvement than direct training from random initial condition. However, since GTSVQ can only provide around 70 percent classification, their voronoi node is still far from optimal. In most situations, the convergence to global optimal point is not guaranteed.

For PTSVQ, when constructing a subtree, only a small subset of all training data will be used. Therefore, the input data dimensionality is greatly reduced for each individual class, and each subtree can approach to its global optimal state of minimal distortion in extremely short time. This quality makes PTSVQ remarkably insensitive to initial training condition. Given in addition the near-optimal classification performance, PTSVQ serves as the best candidate for the initialization of LVQ network. In our experiment, we adopted the final result of PTSVQ as the initial states for further LVQ training, and after only a few training cycles (1000~3000), LVQ converges to a saturated state. A problem with this scheme is that LVQ can not directly use the multi-resolution features. In the simulation, we must carefully adjust PTSVQ network to make most of its leaf nodes appear on the finest resolution. In the future, we hope to develop a tree structured LVQ algorithm, so that it can directly use the multi-resolution representation from the cortical model.

4. Another advantage of PTSVQ is its online training ability. When new target shows up, only relevant subtrees need to be retrained, all the other subtrees maintain their state. From online training point of view, this scheme may be the only possible candidate for practical system design.

5. Among the 3 node allocation schemes in PTSQV, PTSVQ(2) has the worst performance. To account for this result, we propose a heuristic explanation through a simple example, as shown in Fig.4.9.

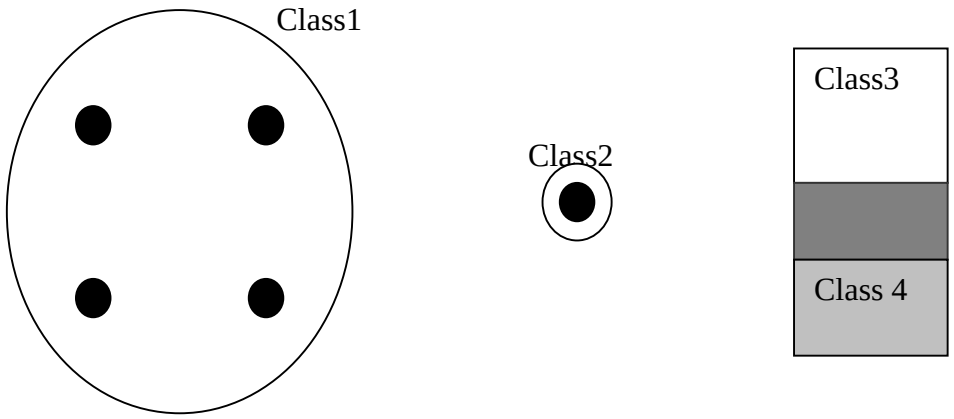


Figure4.9 Failure of node allocation according to equal distortion

In this figure, samples from class 1 are sparsely distributed in the 2-D space, while class 2 are more compactly clustered. According to equal distortion criteria, we need more nodes for class 1 than class 2 to achieve equal distortion. However, one node located in the center of each circle will be enough to separate class 1 and 2, since their spatial distribution is not overlapped. In this case, more nodes should be reserved for class 3 and class 4 since their spatial distribution is severely intersected. Generally, the non-optimal nature of PTSVQ prevents any allocation schemes from absolute ‘fair’. For ACIDS database, class 8 vehicles have the sparsest distribution, so when total number of leaf nodes is small, nearly one third of all leaf nodes will be allocated to this class (as shown in table4.2), thus seriously deteriorate the classification of all the other classes.

4.5 Further improvement of Classification

The decision fusion unit in section 3.4 is implemented here. Each sub-decision comes from preceding PTSVQ(3) and LVQ classifiers, the final performance is listed in table4.5. This table concludes our final performance: among all the 1644 one-second testing samples, 91.01 (or 96.35) percent samples are correctly classified using a 274-cell PTSVQ(3) (or LVQ) classifier.

Total Nodes	106	137	160	206	220	274
Original PTSVQ(3)	77.49	80.37	82.45	84.31	84.41	86.12
After Fusion	81.72	84.95	86.77	89.32	89.36	91.01
Original LVQ	87.64	90.05	90.12	91.18	91.50	92.61
After Fusion	91.24	93.86	94.56	95.07	95.19	96.35

Table 4.5 Classification gain using decision fusion

The decision fusion unit successfully reduces about 4 percent short time error caused by burst oscillation within vehicles signals, the cost is 750 ms more processing delay. In practical system, if we use a higher sampling rate, we may segment the input data into shorter frames, thus the classification delay can be further reduced.

4.6 Experiments with independent testing data

So far in our experiments, the training and testing samples is from the same set of recordings, i.e., for 21920 available samples, 70 percent samples are randomly selected as training data, so nearly every recording has some frames picked into the training data set. In this simulation environment, the classifier has experience with all available recordings. However, in real battlefield condition, the classifier must recognize new input which may come from unexpected speed and ground condition that it never encounters before.

Therefore, we must reexamine the classification performance of previous algorithms with totally new recordings.

In this experiment, the old ACIDS database is used to train the VQ based classifiers. Once the training is finished, the classifier is fixed and a new set of recording is used to test the classification performance. In Table 4.6, 4.7 and 4.8, the confusion matrix of several classifier is presented.

Predicted\True	1	2	3	4	5	6	7	8	9
1	**93.0769**	37.5	18.9583	2.2917	0	0	11.5625	14.1667	4.25
2	5.9615	**36.25**	2.7083	1.0417	0	0	3.125	0.2083	19
3	0.1923	0	**43.9583**	0	0	0	5.3125	0	2.25
4	0	1.25	1.875	**78.5417**	0	0	38.4375	0.625	1.75
5	0	25	0.8333	0.625	**0**	0	1.25	0.4167	4.75
6	0	0	0.625	0	0	**0**	6.875	4.1667	6
7	0	0	4.5833	3.9583	0	0	**6.25**	0.2083	1.75
8	0.1923	0	3.125	3.5417	0	0	20.9375	**79.5833**	8.75
9	0.5769	0	23.3333	10	0	0	6.25	0.625	**51.5**
Total %	100	100	100	100	100	100	100	100	100

Overall Score 46/71 Correct

Table 4.6. 137-cell LVQ classifier, classification performance on high SNR 40 seconds of the acoustic data, all value in percentage, a classification result is reported for each second.

Predicted\True	1	2	3	4	5	6	7	8	9
1	**68.4615**	34.375	9.7917	0.2083	0	0	1.875	4.1667	0
2	6.1538	**17.5**	1.875	0.8333	0	0	1.25	0.2083	4.25
3	0.7692	1.25	**42.7083**	0	0	0	5.9375	0	3
4	6.1538	20	10.8333	**87.0833**	0	0	43.75	2.5	8
5	0.1923	23.75	1.6667	0	**0**	0	6.5625	0.8333	5.25
6	3.0769	0	1.4583	0	0	**0**	2.5	3.125	7.5
7	0.9615	0.625	8.9583	0.8333	0	0	**5**	0	0.75
8	13.2692	1.875	3.9583	1.875	0	0	22.5	**88.5417**	16
9	0.9615	0.625	18.75	9.1667	0	0	10.625	0.625	**55.25**
Total %	100	100	100	100	100	100	100	100	100

Overall Score 49/71 Correct

Table 4.7. 401-cell LVQ classifier, classification performance on high SNR 40 seconds of the acoustic data, all value in percentage, a classification result is reported for each second.

Predicted\True	1	2	3	4	5	6	7	8	9
1	**91.9231**	46.875	41.0417	4.5833	0	0	23.125	28.75	19.5
2	5.3846	**26.875**	20.2083	6.25	0	0	9.6875	1.4583	8.75
3	0.1923	0.625	**13.5417**	0	0	0	0	0.625	1
4	0	0.625	1.6667	**55.625**	0	0	22.8125	0.4167	9.5
5	0	24.375	2.0833	7.5	**0**	0	15.9375	0.8333	8.25
6	0	0	0.625	0	0	**0**	2.1875	2.5	1.25
7	0.7692	0	3.75	5	0	0	**5**	0	0
8	1.7308	0	2.5	2.7083	0	0	11.25	**64.375**	15
9	0	0.625	14.5833	18.3333	0	0	10	1.0417	**36.75**
Total %	100	100	100	100	100	100	100	100	100

Overall Score 36/71 Correct

Table 4.8. 206-cell PTSVQ classifier, classification performance on high SNR 40 seconds of the acoustic data, all value in percentage, a classification result is reported for each second.

From above simulation, we can draw the following conclusions:

1. With the independent testing data, both LVQ and PTSVQ classifiers suffer from insufficient training. For class 1, 4, 8, 9, they still achieve a reasonable performance. The independent testing data doesn't include class 5 and 6 recordings. For class 3 and 7, which has the smallest training samples in the ACIDS database, these two classes are highly confused with other classes.

2. In average, LVQ classifier is still a little better than PTSVQ, however, LVQ is no longer optimal in the Bayes sense, and under certain situation, PTSVQ outperforms LVQ. For example, 206-cell PTSVQ classifier achieve better classification on class 1 vehicle than 401-cell LVQ classifier.

3. Unlike the old experiments, the classification performance doesn't increase as the number of leaf nodes increases. This clearly suggests that insufficiently trained VQ classifier is 'biased', i.e., the voronoi centroid only partially represent the real spatial distribution for each class.

4. To improve the performance, we need much more training data than current ACIDS database. In this situation, when new data is available, it should be inserted into the training set of particular subtree, and PTSVQ's parallelism will show great advantage over the LVQ algorithm.

4.7 Entropy based confidence measure

Using above proposed classifiers, we can make a classification decision on every new testing sample, however, this decision is not always reliable. Basically, when distribution of two vehicles are highly overlapped in feature vector space, it is better to skip making a decision rather than straightly given an unreliable decision. In this case, a confidence measure is needed.

For VQ based classifiers, a natural confidence measure is the 'pureness' of each voronoi cell. For example, after the training stage, if only one type of training samples exists in a specific voronoi cell, this implies no other classes have distribution function overlapped in the surrounding area. Therefore any decision from this cell is highly reliable, the confidence value of the decision should be high. From information theory, the best 'pureness' measure is entropy, which is defined using the following equation:

$$E(V_j) = -\sum_{i=1}^{9} p_i \log_2(p_i)$$

here V_j is the jth voronoi cell, p_i is the percentage of class i training samples among all training samples that ended up in this cell. Obviously, the lower the entropy, the purer this voronoi cell, and more reliable the decision based on this cell. After the tree construction stage, all training data are applied to all subtrees in parallel again, and the class ID and corresponding entropy for each leaf node are recorded. In the future testing stage, based on the leaf node where the testing sample ends up into, a confidence value

can be reported together with the vehicle ID. Fig.4.11 shows an example of a PTSVQ subtree with each leaf nodes labeled with entropy values. Fig. 4.11 gives the entropy histogram based on PTSVQ training data.

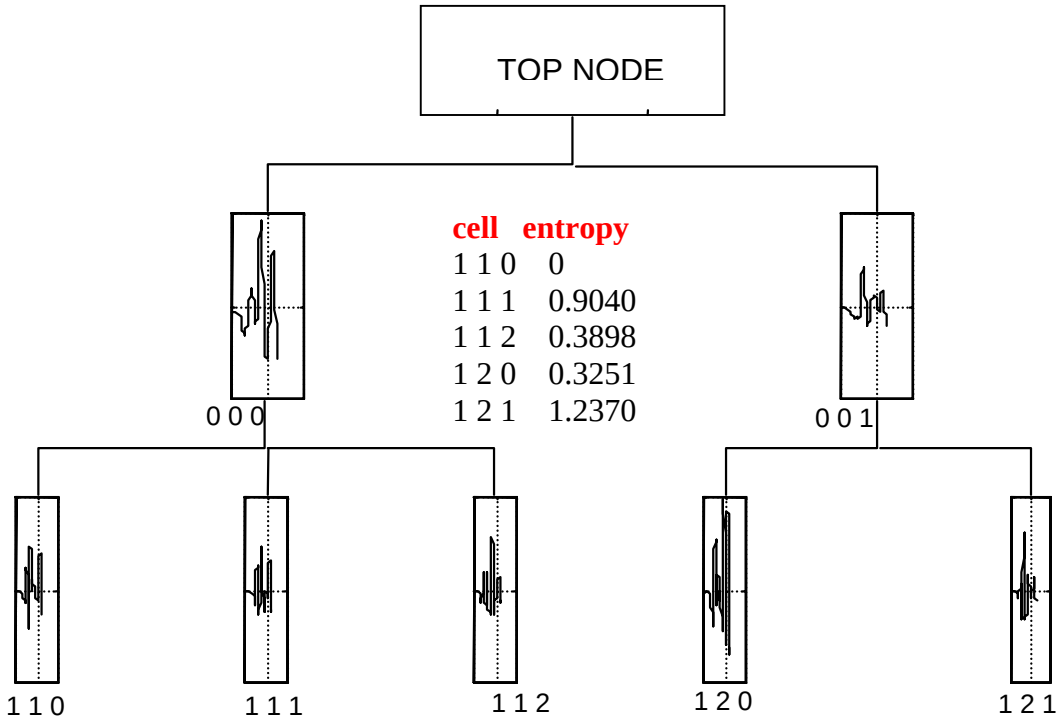


Figure 4.10 PTSVQ subtree for vehicle 7, each node labeled with a entropy value

In Fig. 4.11, most training samples end up into low entrpgy voronoi cells, therefore a straightforward approach is to drop the decisions corresponding to the high tail end of the histogram (e.g. drop the 15% high entropy cell decisions). In this way, only high confidence decision are kept for the end user. This scheme is applied to the independent testing experiment as discribed in section 4.6, the resulting entropy histogram is shown in Fig. 4.12, and after choping the 15% high entropy decision, the resultant confusion matrix is shown in table 4.9.

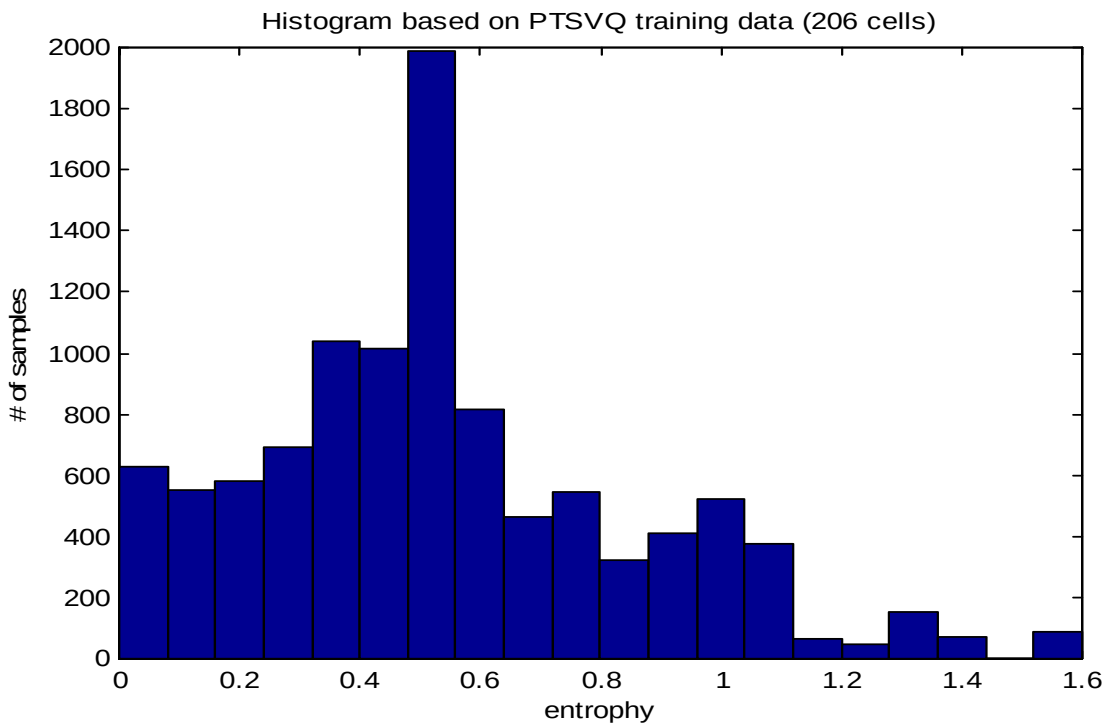


Figure 4.11 Entropy histogram of all classification decisions in PTSVQ training data

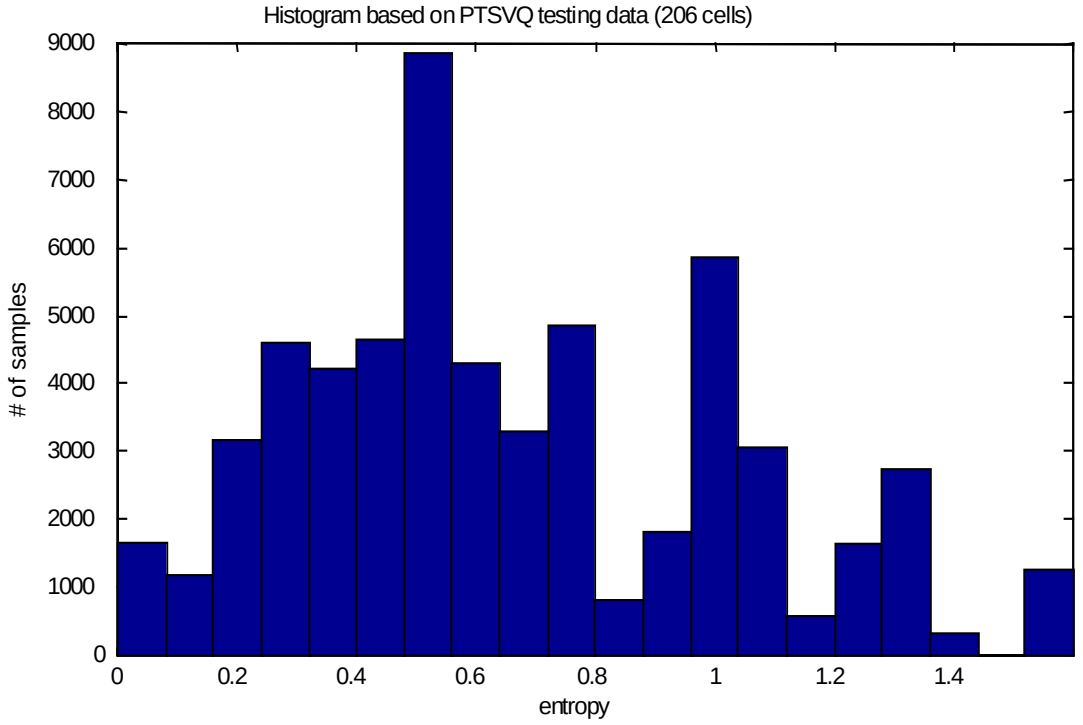


Figure 4.12 Entropy histogram of all classification decisions in PTSVQ testing data

Predicted\True	1	2	3	4	5	6	7	8	9
1	**87.1576**	30.9051	45.4476	2.7327	0	0	32.0975	29.3119	32.3543
2	7.1211	**41.8322**	13.9429	5.2802	0	0	8.8983	4.7811	11.6567
3	0.1217	0.3311	**6.5524**	0	0	0	0.5297	1.0724	1.5145
4	0.8521	4.1943	3.581	**57.573**	0	0	11.7585	1.6086	14.9151
5	0.5478	20.9713	4.0381	18.7124	**0**	0	10.911	1.2958	11.4273
6	0.213	0	1.1048	0.0463	0	**0**	1.9068	3.7534	1.4686
7	0.426	0.1104	3.5048	2.4085	0	0	**2.6483**	0.3128	0.3212
8	3.2562	0.6623	14.9333	1.8064	0	0	22.0339	**56.5237**	5.8284
9	0.3043	0.9934	6.8952	11.4405	0	0	9.2161	1.3405	**20.514**
Total %	100	100	100	100	100	100	100	100	100

Table 4.9 206-cell PTSVQ classifier, classification performance on the whole acoustic data, all decisions counted, a classification result is reported for each second.

Predicted\True	1	2	3	4	5	6	7	8	9
1	**88.2112**	34.4782	39.6243	1.8072	0	0	15.1424	17.8328	27.7655
2	6.5824	**34.7424**	13.3246	5.2711	0	0	9.2954	4.3732	15.0442
3	0.1488	0	**6.0288**	0	0	0	0.4498	0.8746	0.7743
4	0.8925	3.5667	2.3591	**59.488**	0	0	13.6432	0.8746	16.2611
5	0.595	25.2312	4.0629	20.8835	**0**	0	15.7421	1.0204	13.4956
6	0.1859	0	1.0048	0.0502	0	**0**	1.949	4.3732	0.9403
7	0	0.1321	0.3495	0	0	0	**0.5997**	0.0972	0.1659
8	3.0866	1.1889	25.9502	1.8574	0	0	31.1844	**69.5821**	3.4292
9	0.2975	0.66	7.2958	10.6426	0	0	11.994	0.9718	**22.1239**
Total %	100	100	100	100	100	100	100	100	100

Table 4.10 206-cell PTSVQ classifier, classification performance on the whole acoustic data, 15% high entropy decision dropped, a classification result is reported for each second.

From Fig. 4.12, it is obvious that more high entropy decision are made in the independent testing experiment, this is because the classifier is trained using high SNR data segment, while testing is carried out on the whole recording. In table 4.9 and 4.10, class 1,4,8,9 show apparent improvement with the low confidence decision dropped. However, class 2,3,7 show degradation in classification performance. A possible explanation for this result is that currennt PTSVQ classifier still suffers from insufficient training, many fixed vonoroi cell centroids are seriously biased, they can not represent

the true distribuation of each class within feactre space. As an example, class 3 and 7 are the most scarcely trained vehicles, as a result, their resulting entropy value for each node is also biased. To improve this entropy based confidence measure, a much larger traininig database is needed.

4.8 Conclusion on classification algorithms

The effectiveness of Shamma's biological feature extraction models is proved in above practical system. Among different VQ based classification algorithms, LVQ has the best performance but is also the slowest one. PTSVQ are found to be an intermediate state between LVQ and GTSVQ. It provides a classification performance close to the optimal LVQ, while maintains a logarithmic search time. After decision fusion, PTSVQ(3) is only 7% lower than LVQ. Meanwhile, PTSVQ(3) is the best parallel scheme to implement fast training, fast searching and online new target insertion. As a direct result, PTSVQ (3) will be the best candidate for practical system design. For the ACIDS database, a PTSVQ(3) scheme followed by a decision fusion unit can provide 91% correct classification.

On the other hand, in the independent testing experiment, all classifiers suffer from insufficient training, many Voronoi cells are biased. To solve this problem, more training data is needed, and PTSVQ's parallelism and new-target insertion capability will show great advantage during the online training. Finally, an entropy based confidence measure is proposed, although this confidence measure also suffers from biased voronoi centroid, it shows great potential in evaluating the reliability of current ID decision, and the efficiency of this measure will be a major research topic in the future.

Chapter 5 Combined Classification and DOA Estimation

So far, we have been focused on the classification for 'clean' vehicle acoustic signal. In the real battlefield condition, the vehicle signal is seriously polluted by all kinds of noise, especially by the sounds from nearby vehicles. For practical system, the signal must be putrefied before any further processing. The traditional and classic method for signal enhancement is beamforming based on array processing. With plenty amount of sensors, we can build a narrow acoustic beam in angular space that can extract the signal from the interested direction, thus enhance the signal for further processing. Right now, there are two serious problems associated with acoustic beamforming: first, it normally takes more than 10 sensors to get a beam with main lobe narrow enough to shield the sound from uninterested direction. Such a large sensor system is always difficult to build, very expensive and difficult to deploy, therefore, most available data sampling system is based on very small arrays. As an example, the current ACID database is recorded using only 3 microphones. The second problem is that acoustic signal, unlike most radar signals and some sonar signals, is broadband signal, therefore, signal with different frequencies will endure different phase shift even when the propagation delay between two sensors is the same. Right now, the common approach for broadband acoustic beamforming is based on frequency invariant adaptive algorithms, which involve complicated FIR filter bank design and various broadband array processing techniques, and they still can not guarantee a beam narrow enough for a small array of 3 sensors. On the other hand, biological hearing system shows remarkable sound

localization ability, which is widely known as cocktail party effect. As shown in Fig 5.1, a human being can easily identify the sound from different instrument, while the SNR from each instrument is far below 0dB. This remarkable ability is purely dependent on a small array of only 2 sensors (2 ears). From this phenomenon, we hope to investigate the localization ability within the biological system, with knowledge therein, we may find a unified framework for combined multi-target detection, ID and DOA system suitable for small arrays.

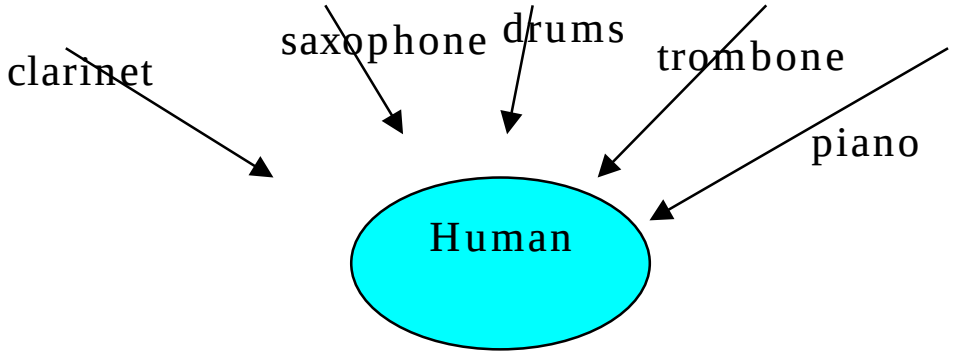


Figure 5.1 Cocktail party effect

5.1 Stereausis model for DOA estimation

There are several binaural hearing models that have been proved successful in accounting for biological sound localization, such as [31], [32] and [33]. However, all these models are based on a running-correlation measure between the cochlear outputs from the two ears at various time delays, yet there is no direct physiological support of the existence of spatially organized neural delays in the mammalian auditory system. Shamma's Stereausis model, on the other hand, utilizes the delays already present in the traveling waves of the basilar membrane to extract the correlation function, thus avoids involving undetected neural delay into the network. The two-dimensional stereausis neural network is plotted in Fig 5.2.

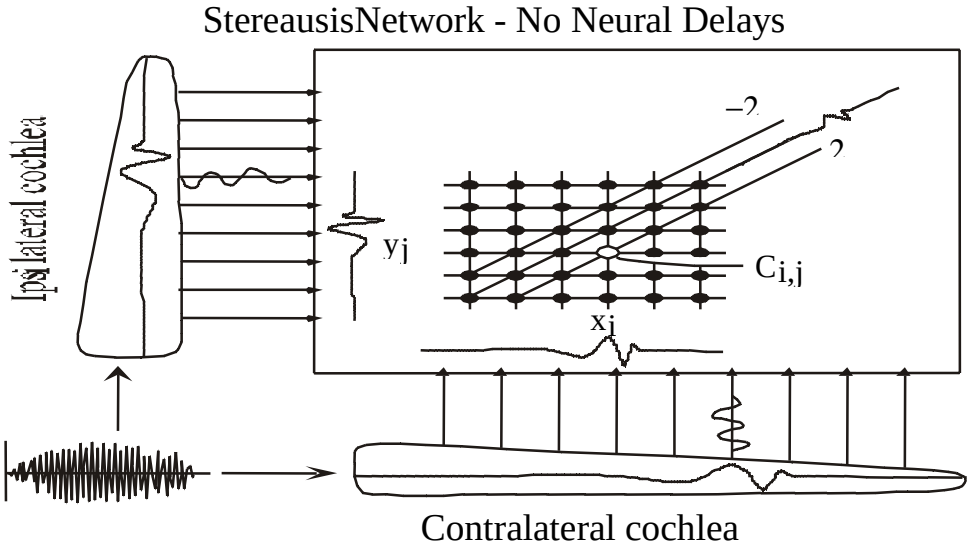


Figure 5.2 Stereausis neural network model

The stereausis network measures binaural differences by detecting the spatial disparities between the instantaneous outputs of two series of filter banks of the two ears. As shown in Fig. 5.3, the output of the cochlear filter banks from left ear is fed into the network from left side, the output of right ear is fed in from the bottom. The two side signals are cross-correlated inside the network, the output of the network is a 2-D image with one axis representing the characteristic frequency of one ear, and the other axis representing the other ear. As an example, a 2-D stereausis pattern is plotted in Fig. 5.3. In the stereausis pattern, a dominant peak of activity appears along the main diagonal (zero disparity). This diagonal equals the auditory spectrum in chapter 2. Parallel to the main diagonal, there are some ridges and valleys. These ridges and valleys are the result of different phase delay between neighboring bands. If the two bands are far apart, their correlation decays quickly, since their bands no longer has overlapped part, and the cross correlation between two signals with different carrier would be zero. When a tone is binaurally phase-shifted, the network pattern shifts accordingly. As the dominant ridge

shifts away from main diagonal and degrades into secondary ridges, the secondary ridges or valleys shift toward main diagonal and grows into the dominant peak. In this way, this model successfully explains the binaural localization ability in biological hearing.

To exploit the Interaural Time Delay (ITD), different recordings from different microphones are used as inputs for the Stereausis Network. The disparity plot in Fig. 5.3(b) shows the 1-D patterns of activity computed near and along the cross-sections which are perpendicular to the main diagonal. As discussed before, the more interaural time delay, the larger the disparity from the main diagonal. Based on this disparity, the following scheme is proposed to estimate the interaural phase difference.

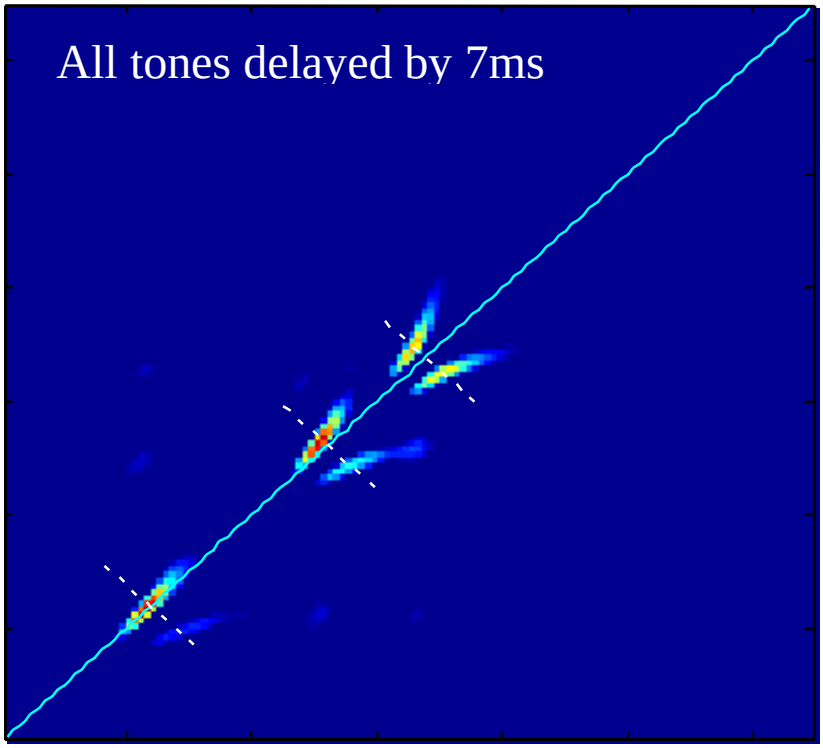


Figure 5.3 (a) Stereausis Pattern, 3 tones each with a ITD of 7ms.

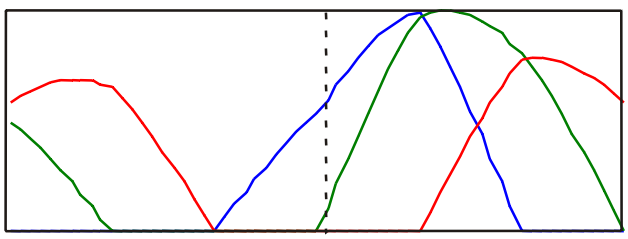


Figure 5.3 (b) Disparity plot, stereausis pattern along the bar perpendicular to the main diagonal

In the stereausis network, DOA estimation is performed on each cochlear filter bank, specifically, on the central characteristic frequency w_c of each band, here w_c is the characteristic frequency of band *c*. Assuming the maximal disparity happens to be M bands away from the main diagnal, then, this disparity corresponses to the maximal delay when the source is on the same line as the two sensors, and it also corresponses to a phase shift of $\pm\pi$ if w_c is the upper limit of spatial sampling frequency. Obviously, only the disparity within $\pm M$ should be considered, higher disparity, which corresponses to a larger phase delay more than $\pm\pi$, is caused by the nolinearity within the peripheral auditory system, and should not be used in our estimation. Let y_i and y_j be the cochlear filter response to a pure tone of frequency w_c, i.e.,

$$y_i(t) = A_i(w_c)\cos(w_c t + \theta_i(w_c)) \tag{5.1}$$

$$y_j(t) = A_j(w_c)\cos(w_c t + \theta_j(w_c) + \delta) \tag{5.2}$$

where A_i, A_j and $\theta_i(w_c), \theta_j(w_c)$ are the amplitudes and phases of the traveling waves at the ith and jth bands, δ is the inter sensor phase difference caused by wave propogation between the two sensors. In practical systems, M is determined by array geometry as well as the frequency resolution of the cochlear filter bank. For ACIDS recording system, experiment shows M=3, therefore, only very small disparity is to be considered in ITD estimation. With such small disparity, we can assume linear phase difference between neighboring bands, i.e.,

$$\theta_{i+c} = i\Delta + \theta_c, \qquad i = -M, -M+1, \cdots, 0, \cdots, M-1, M \tag{5.3}$$

where θ_c is the phase delay for subband c at frequency w_c, and Δ is the phase difference between two neighbouring bands. Also, we can assume $A_i(w_c) \approx A_j(w_c) \approx A_c$ since these

bands are close and highly overlapped. The correlation operation $C(y_i, y_j)$ defined in the stereausis system becomes

$$C_{ij} = \int_T A_c^{\,2} \cos(wt + \theta_i)\cos(wt + \theta_j + \delta)dt$$

$$= \int_T \frac{1}{2} A_c^{\,2} \cos(\theta_i - \theta_j - \delta)dt + \int_T \frac{1}{2} A_c^{\,2} \cos(2wt + \theta_i + \theta_j + \delta)dt$$

$$= \int_T \frac{1}{2} A_c^{\,2} \cos(\theta_i - \theta_j - \delta)dt = \frac{T}{2} A_c^{\,2} \cos(\theta_i - \theta_j - \delta) \qquad (5.4)$$

For discrete system, the correlation function becomes:

$$C_{ij} = \sum_{n=1}^{L} \frac{1}{2} A_c^{\,2} \cos((i-j)\Delta - \delta)$$

$$= \frac{L}{2} A_c^{\,2} \cos((i-j)\Delta - \delta) \qquad (5.5)$$

where L is the frame size.

The disparity at band c is calculated along the cross-section bar, where

$$(i, j) \in \{i = c + k,\ j = c - k, k \in [-M, M]\}$$

define the disparity sequence:

$$D_k = C_{c+k,c-k} = \frac{L}{2} A_c^{\,2} \cos(2 * k\Delta - \delta)$$

$$= \frac{L}{4} A_c^{\,2} \{\exp[\, j(2 * k\Delta - \delta)] + \exp[-j(2 * k\Delta - \delta)]\} \qquad k \in [-M, M] \qquad (5.6)$$

To extract phase delay δ from $\{D_k\}$, we perform correlation operation on $\{D_k\}$,

$$G = \sum_{k=0}^{2M} D_k \exp(-j\frac{2\pi}{2M}k) \qquad (5.7)$$

Since disparity at k= $\pm M$ corresponding to $\pm\pi$ phase delay, we have

$$2M\Delta = \pi \qquad (5.8)$$

Now, put (5.8) into (5.7),

$$G = \sum_{k=0}^{2M} D_k \exp(-j2\Delta k)$$

$$= \sum_{k=0}^{2M} \frac{L}{4} A_c^2 [\exp(-j\delta) + \exp(-j(\frac{2\pi k}{M} - \delta))]$$

$$= \frac{L(2M+1)}{4} A_c^2 \exp(j(-\delta)) \quad (5.9)$$

Therefore, for a complex number G, we have

$$\text{Angle(G)} = (-\delta) \quad (5.10)$$

$$\text{Amplitude(G)} = = \frac{L(2M+1)}{4} A_c^2 \quad (5.11)$$

From (5.10) and (5.11), it is obvious that the complex number G provides enough information for DOA estimation on bands, i.e., the amplitude G is proportional to signal power at w_c and the angle of G is proportional to the phase delay δ. From δ, the DOA estimation is given by:

$$\theta = \arcsin(\frac{s\delta}{w_c D}) \quad (5.12)$$

where D is the distance between 2 microphones, s is the sound propagation speed in the air, θ is the estimated angle of arrival.

5.2 Experiments on Vehicle DOA estimation

The above scheme is tested against the battlefield acoustic data from ACIDS database. In order to examine the multi-vehicle DOA performance of this algorithm, we used a mixed signal in this experiment. First, two recordings from ACIDS database are normalized into equal energy, and then the data from each microphone is mixed with the

data from the corresponding microphone of another vehicle. Since the vehicle is fast moving object, its impact angle changes within second, therefore, DOA must be carried out on a short time window. In this case, we use quarter second as processing window. After segmentation, each framed data is fed into the Stereausis network, and from which we obtain the corresponding disparity curves on each band. Finally, we obtain an angle and power estimation on each of the characteristic frequencies using the proposed algorithm in 5.1.

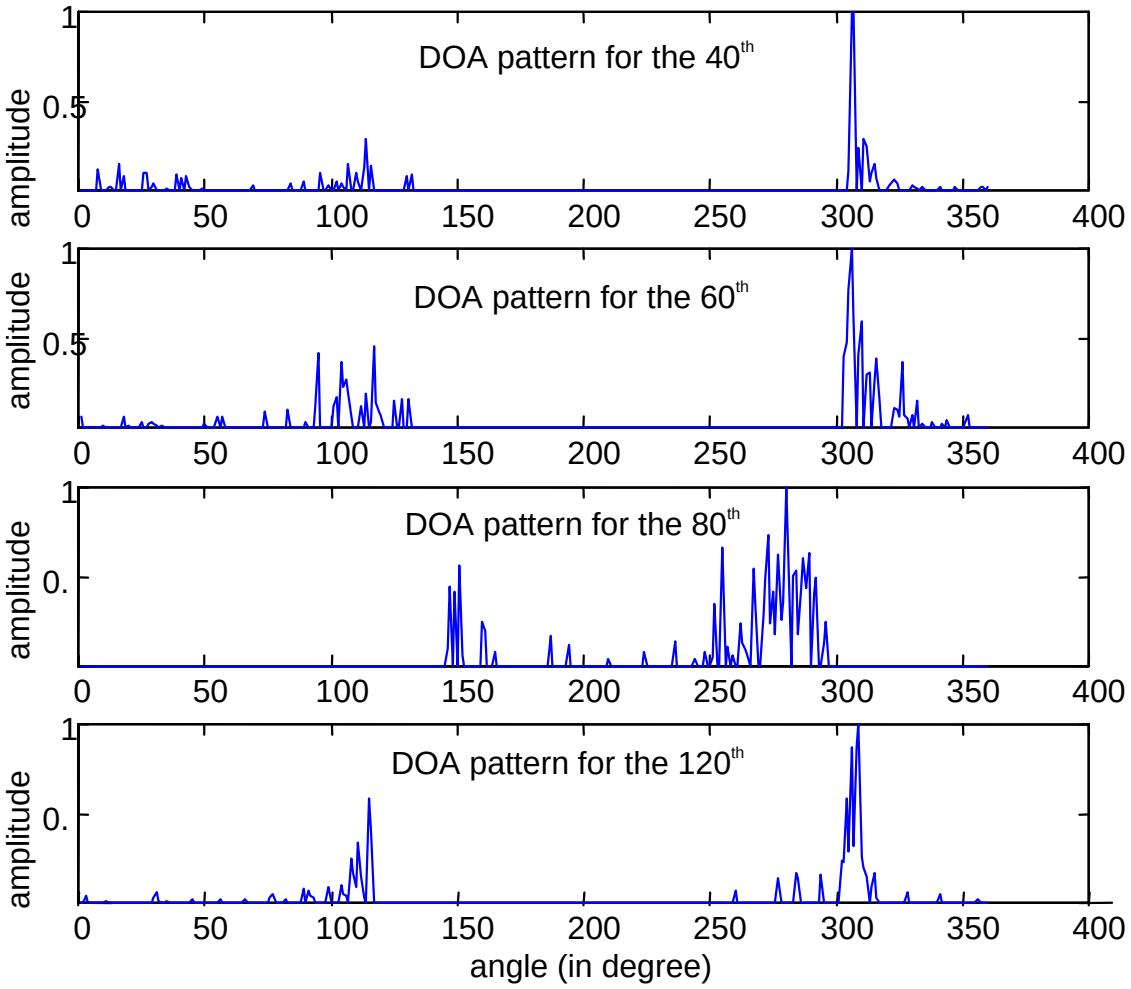


Figure 5.4 DOA estimation at different frames

Fig. 5.4 shows the DOA estimation at several different frames. On each frame, the estimator gives a series of peaks, each peak corresponds to DOA estimation from one w_c. The position of the peak is the angle estimation result, and the amplitude of the peaks represents the energy of this band. Since there are only 2 sets of peaks, two vehicles can be clearly distinguished. Fig. 5.5 shows the 2-D plot of DOA pattern for all frames. From these two figures, we can draw the following conclusion:

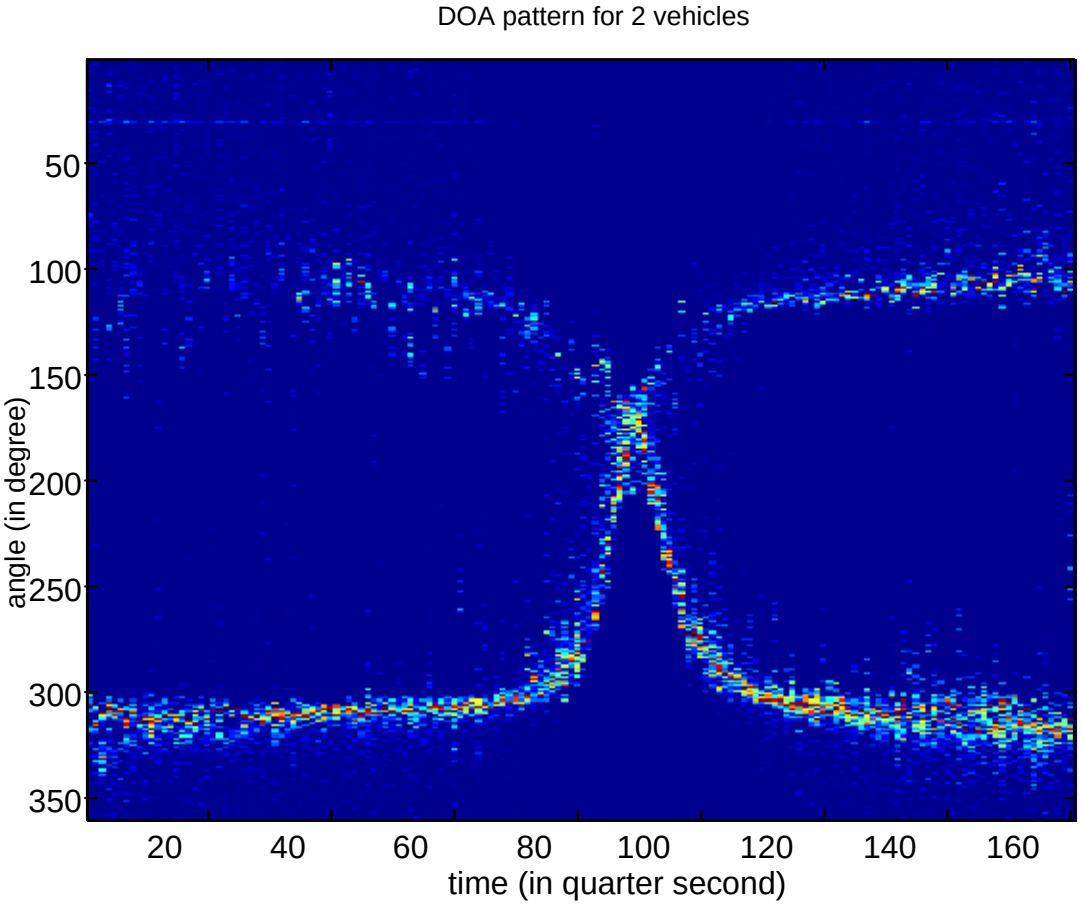


Figure 5.5 DOA pattern for mixed vehicle signal

1. The SNR on different subbands is different from each other, the estimation on high SNR bands is more reliable than other bands, i.e., the height of each peak is a clear indication of reliability of its estimation.

2. Since the two vehicles are spatially separated, the estimation peaks in the DOA pattern can be clustered into 2 groups, each group centers on the true impact angle of one vehicle. This natural clustering mechanism is the theoretical foundation for following signal separation algorithm.

3. If Vehicle A is dominant on the nth band, this band will give correct DOA estimation for Vehicle A. If both vehicles have strong signals in the nth band, its peak will be somewhere between the true impact angles of the two vehicles.

4. In some of the frames, signal from one vehicle is stronger than that from another vehicle, therefore, the DOA pattern for the weak signal is corrupted by the strong signal. The degree of degradation will depend on the energy ratio of the two signal as well as spectrum similarity of the two vehicles.

5.3 DOA aided vehicle ID

Signal separation is an indispensable step before multi-vehicle classification. From the DOA pattern in Fig.5.5, it is obvious that Stereausis network can provide robust DOA even with only 3 sensors. Based on this result, a straightforward scheme for the mixed auditory spectrum separation for small array is possible, which is described below:

1. Pattern smoothing

As shown in Fig.5.6, a hamming window of length 100 is applied to the DOA pattern in Fig.5.4. After the smoothing, only two peaks remain. From these two peaks, we obtain two angle estimations: θv_1 and θv_2, these two results will be used in the following steps to cluster the cochlear filter banks and construct spectral separation template.

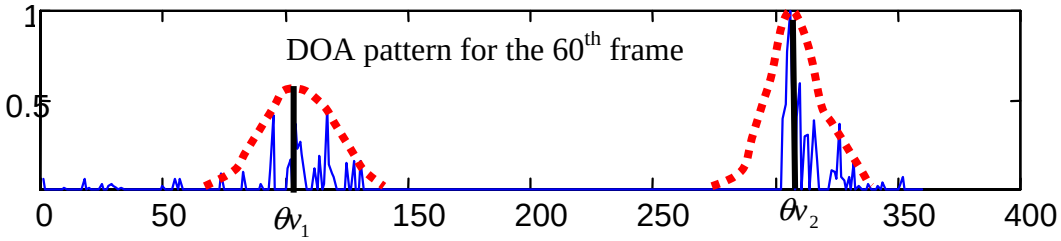


Figure 5.6 Smoothed DOA pattern using Hamming window

2. Band grouping

The 128 cochlear filter banks are grouped into two sets according to DOA estimation on each band. For example, if θ_c is the DOA result on band c, then band c should be assigned to vehicle 1 if $d_{ang}(\theta v_1, \theta_c) < d_{ang}(\theta v_2, \theta_c)$. Here $d_{ang}(\bullet,\bullet)$ is the angular distance measure, it is defined by the following equation:

$$d_{ang}(\theta_i, \theta_c) = \min[\mathrm{mod}(\theta_i - \theta_c + 360, 360), \quad \mathrm{mod}(\theta_c - \theta_i + 360, 360)] \quad i = 1, \quad 2 \qquad (5.13)$$

3. Separation template

The function of the template is to emphasize the component from one vehicle in the mixed spectrum while suppress the component of another vehicle. Therefore, the value of the template should be proportional to the ratio of energy between these two vehicles on the interested band.

Not all cochlear bands are considered in the construction of the template. Energy of vehicle signal concentrates in bands between 40^{th} ~120^{th}, therefore, only these 81 bands will be considered. Furthermore, the signals in some bands are so weak that DOA on these bands is highly unreliable, therefore, if the energy in one band is lower than certain threshold, its associated template value will be fixed to 1. In our scheme, this threshold is set to be 10 percent of the energy of the strongest band.

For bands whose energy is above the threshold, if its associated DOA is exactly the same as θv_1, which implies that Vehicle 1 dominates in this band, then the template value at this frequency will be set to 2 (amplify). If its DOA equals θv_2, alternatively, the template value will be set to 0 (suppress).

If both vehicles have strong energy in the same band, as shown below:

$$y_i(t) = A_1 \cos(w_c t + \theta_i) + A_2 \cos(w_c t + \theta_{12} + \theta_i) \quad (5.14)$$

$$y_j(t) = A_1 \cos(w_c t + \theta_j + \delta_1) + A_2 \cos(w_c t + \theta_{12} + \theta_j + \delta_2) \quad (5.15)$$

here, A_1 and A_2 are the power spectral density of the 1st and 2nd vehicle signal on frequency w_c, θ_i and θ_j are the phase responses of the ith and jth cochlear filter, θ_{12} is the phase difference between the two sources, δ_1 and δ_2 are the phase differences originated from inter-sensor wave prorogation delay. Obviously, if signal from different sources mixed up in the same band, the value of angle(G) and amplitude(G) will depend on all variables including A_1, A_2, δ_1, δ_2, θ_{12} and θ_i. From two known values, angle(G) and amplitude(G), we need to find out 5 unknowns, it is a standard ill-posed problem. However, in order to build a template whose value is proportional to the real signal energy, we need to know the exact value of A_1 / A_2. Here, we adopted a simplified assumption to speed up the processing, i.e., for a band with mixed signal from both vehicles, its DOA estimation, θ_{w_c}, will be somewhere in the middle between the true impact angle of the two vehicle. The angular distance between θ_{w_c} and θ_1 and θ_{w_c} and θ_2 will satisfy the following equation:

$$\frac{A_1}{A_2} = \frac{d_{ang}(\theta_{w_c}, \theta_1)}{d_{ang}(\theta_{w_c}, \theta_2)} \tag{5.16}$$

after A_1 / A_2 is obtained through (5.16), the template is defined on this mixed band using the following heuristic equation:

$$Template(w_c) = \begin{cases} 1 + 2 * \left| 0.5 - \min(\frac{A_1}{A_2}, \frac{A_2}{A_1}) \right| & if \quad \min(\frac{A_1}{A_2}, \frac{A_2}{A_1}) < 0.5 \\ 1 & otherwise \end{cases} \tag{5.17}$$

The above template is only for one of the two vehicles. For the other vehicle, the following formula is used to generate a complementary template:

$$Complementary\_template(w_c) = 2 - Template(w_c) \tag{5.18}$$

Obviously, for all w_c, c=1,2, …, 128, the template value will be a real number between 0(suppression) to +2 (enhancement).

5.4 Simulation of DOA aided classification.

After we obtain the two templates, we apply them to the mixed auditory spectrum:

$$Fsep1(w_c) = Fmixed(w_c) * Template(w_c) \quad w_c = 1,2,...,128 \tag{5.19}$$

$$Fsep2(w_c) = Fmixed(w_c) * Complementary\_template(w_c) \quad w_c = 1,2,...,128 \tag{5.20}$$

here, $Fmixed(w_c)$ is the mixed auditory spectral density function at w_c, $Fsep1(w_c)$ and $Fsep2(w_c)$ are the separated spectral density function at w_c.

The separated spectrum is presented to the previously trained classifier as described in Chapter 4. If a PTSVQ classifier is to be used, the separated spectrum should also go through the cortical filter to provide the necessary multi-resolution representation. In Fig. 5.7, the original spectrum from each vehicle, the mixed spectrum, the template and the separated spectrum are plotted.

The classification experiment is based on synthetic data. Here the mixed vehicle acoustic data is created by adding real acoustic data from two vehicles. The first real acoustic data is selected from one of the class 4 recordings in the ACIDS database, while the 2nd data is from a class 6 recording. For each recording, only the strongest 40 seconds are kept, each 40 seconds data are normalized to unit energy and added up together to build a mixed signal. Then, the mixed signal is segmented into quarter second frames, and feed into the proposed Stereausis network.

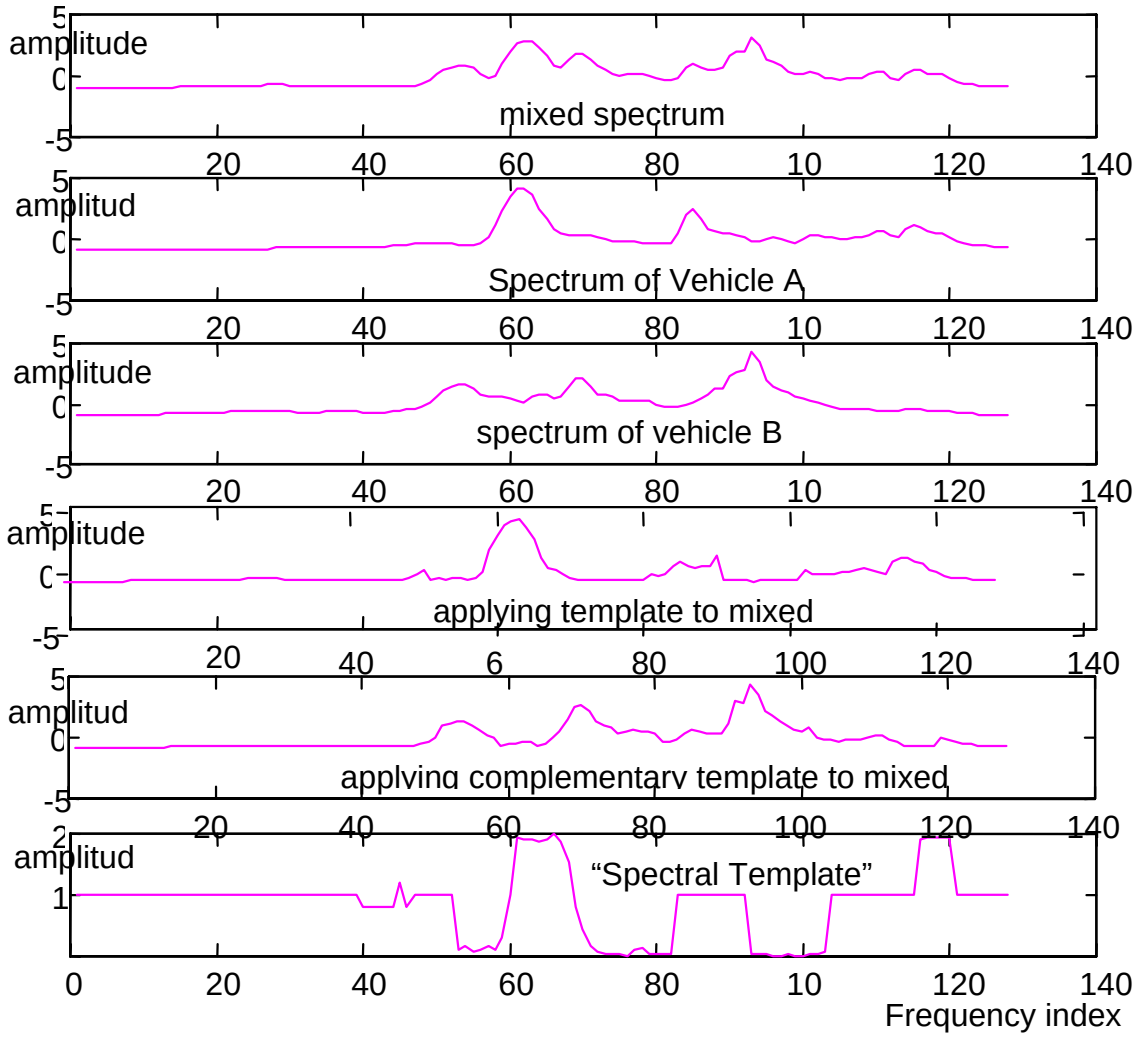


Figure 5.7 Signal separation based on spectral template

Classification Result:

1.(Best case) Separated spectrum is presented to a LVQ 137-cell classifier (the same one as in Chapter 4). Among all 320 decisions (160 frames, each frame provides 2 separated spectra for classification), 262 (82%) are correct.

2. (Worst case) Apply the mixed spectrum directly to a LVQ 137-cell classifier, and find the two best matches in the LVQ centroid set (no template is used here). Two classification decisions are made for each frame. Among 320 decisions, 99 (31%) are correct.

3. (PTSVQ with template) The mixed auditory spectrum is presented to the Cortical filter bank, then templates are applied to the multi-resolution representation from the cortex module. A PTSVQ (137-cell) classifier is used to perform the classification. Among 320 decisions, 170 (53 %) are correct.

4. (PTSVQ with weighted error) Mixed spectrum is presented to the Cortical filter bank to obtain the multi-resolution representation. No templates are involved yet. Then a PTSVQ (137-cell) classifier is adopted. Inside the classifier, a weighted distance is computed using the template as weighting vector. Among 320 decisions, 109(34 %) are correct.

Conclusion:

1. The LVQ classifier achieves 82% correct classification using the separated spectrum. There is 51% classification gain compared to the no template case. This result suggests that Stereausis based DOA estimation greatly improve the performance for multiple-vehicle ID system.

2. In Fig.5.7, the separated spectrum is highly similar to the original spectrum. This result suggests that DOA estimation based signal separation performs well and behaves quite similarly to the traditional beamforming.

3. Classification experiments suggest that tree structure classifier suffers from the introduction of spectral templates. This is reasonable because at higher layer of the tree, small error in the template may direct the search to the wrong branch. To solve this problem, we need to allow more early decisions to propagate to lower layer or devise a full search scheme.

4. Besides using template to separate the spectrum, another scheme is to use template weighted distance in the VQ search stage, as the case in simulation 4. The motivation of this scheme is that: when template value is high, it implies that one vehicle is dominating on this band, so considering only the error on these high SNR bands may be better than considering the whole spectrum. However, the assumption above is not a sound one because when the template is low on some bands, error on those bands is mistakenly neglected. The simulation result also confirms that matching the spectrum only on high SNR bands may lead to serious degradation on vehicle ID performance.

5. When the two vehicles are too close in their direction, or the spectrums of the two vehicle are similar, DOA estimation based on Stereausis network is no longer reliable. As an example, Fig. 5.8 shows the DOA pattern when two vehicles are very close to each other. In the first 80 frames, peaks in the DOA pattern from the two vehicles merged into a single ridge, therefore, signal separation based on DOA is totally impossible. In general, most array processing algorithms suffer from spatial and spectral similarity. To

solve these problems, either larger arrays or more advanced signal separation methods should be employed.

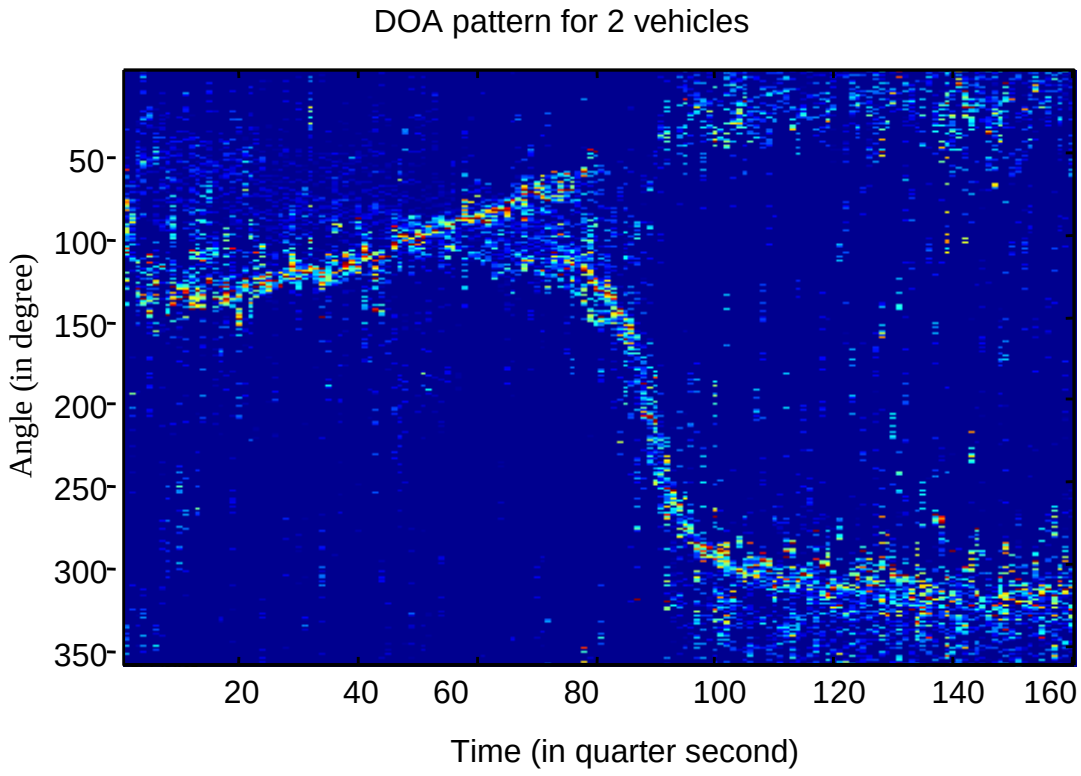


Figure 5.8 DOA pattern for two closely spaced vehicles.

5.5 Future work and open problems

To sum up, our biologically based DOA system demonstrates great potential in multiple-vehicle ID problem. It not only introduces a new view point in accounting for biologically based sound localization and separation, but also proposes an efficient array processing method for small arrays. However, to develop a complete multi-class, dynamic, multi-scale combined localization and classification algorithm for acoustic vehicle data, the following open problems should be answered.

- For bands that both vehicles have significant energy, current Stereausis based DOA system can not provide precise angle estimation. In this case, we need advanced signal separation method, biologically or non-biologically based. For example, Independent Component Analysis (ICA) [34] has shown great potential in separating linear mixed acoustic signals. However, whether it can be used as preprocessing unit for multiple vehicle classification system is still a open problem.
- So far, our approach has been concentrated on DOA driven classification algorithm, which is quite similar to classical beamforming. However, physiology experiment shows that human beings can recognize a specific talker in a multi-talker environment, and then use the talker ID to aid into the DOA tracking of the current talker. In order to develop ID aided DOA system, we should link deeper with auditory physiology, psycho-acoustic measurements, cortical and cognitive models to introduce more understanding and other cues into the framework
- Develop algorithms for vehicle ID based on partial, DOA tagged spectra. Although the template weighted distance failed to provide improvement in classification, bands that has higher template value should be more important than the other bands, because these bands has only one vehicle dominant and can provide higher SNR signal for DOA estimation. How and where the spectral template should be used remains an open problem.

Chapter 6 Conclusions and further research

In this thesis, a prototype vehicle signal classification system is implemented and tested. Using PTSVQ clustering algorithm followed by a decision fusion unit, it achieves more than 90 percent correct classification. The system uses a logarithmic search time, enables on-line training, and the associated training time is negligible compared to LVQ. Furthermore, in the sense of either feature extraction or aggressive topological classification, it is one of the first practical classification systems totally based on biological hearing models. Its success not only confirms the validity of these models, but also encourages more extensive usage of them in other areas such as speech recognition system or mechanic fault detection system.

Another major contribution of this thesis is our approach for the combined DOA and vehicle classification problem. The multi-vehicle DOA system, which is also based on biological localization model, provides robust multiple-vehicle DOA estimation. From this DOA estimation, a new signal separation scheme based on spectral template is proposed, and it shows great potential in improving the performance of multi-vehicle ID system. Through further research, we hope this sound separation algorithm may help to explain some anatomical problems in physiology science, or replace the traditional beamforming technique on small array acoustic signal processing.

The most promising improvement for the Vehicle ID system may come from the integration of current system with other Vehicle ID system, i.e., Combining our biological feature extraction models with other classification algorithm, or test other

feature extraction models on our VQ based classifier. Generally, every feature extraction model and classification algorithm has its advantages and shortcomings. In the long run, a decision fusion mechanism that can adaptively wake up the most appropriate function block under specific environment might be the most promising way to improve the overall system performance.

Another immediate future research is mathematical analysis of the PTSVQ algorithm. We need to establish a theoretical foundation on how much classification gain PTSVQ can provide over GTSVQ, and in what way. Our long-term goal is to develop a tree structured LVQ algorithm that can directly use the multi-resolution representation from cortical model. In this approach, we hope to combine LVQ's optimal classification with TSVQ's fast search and high compression ability, thus we will be able to design classification system that has advantages from both systems.

Finally, for combined DOA and Vehicle ID system, we need to Link deeper with auditory physiology, psycho-acoustic measurements, cortical and cognitive models to introduce more understanding and other cues into the framework. Through this research, we may achieve simultaneous DOA driven ID and ID driven DOA in a reciprocal manner, which might be the exact mechanism as in biological binaural hearing system.

Bibliography

[1] John Baras, and A. Lavigna, “Convergence of a neural network classifier,” Proc. of 29th IEEE conf. on Decision and Control, pp. 1735-1740, Dec. 1990

[2] A. Lavigna, Nonparametric Classification Using Learning Vector Quantization. PhD thesis, Dept. of Electr. Eng., University of Maryland, College Park, Maryland 20742, 1989, ISR Technical Report PhD 90-1

[3] John Baras, Sheldon Wolk, “Hierarchical wavelet representations of ship radar returns,” ISR Technical Report 93-100

[4] Xiaowei Yang, Kuansan Wang, and Shihab Shamma, “Auditory representations of Acoustic Signals,” IEEE Trans. Inform. Theory, vol. 38, pp824-839, 1992

[5] Kuansan Wang, and Shihab Shamma, “ Spectral shape analysis in the central auditory system,” IEEE trans. Speech and Audio Processing, vol 3, no.5, pp382-395, 1995

[6] John.S. Baras, Shedon Wolk, “Wavelet based progressive classification of high range resolution radar returns,” Proceedings of SPIE International Symposium on ntelligent Information Systems, vol 2242, pp967-977, 1994

[7] Karen L. Oehler, and Robert Gray, “Combining image compression and classification using vector quantization,” IEEE trans. Pattern Analysis and Machine Intelligence, vol. 17, No.5, pp 461-472, 1995

[8] Johnn S.Baras, and S. Dey, “Combined Compression and Classification with learning vector quantization,” submitted to IEEE Trans. Inform. Theory, Jan 1999

[9] John S. Baras, and Sheldon I. Wolk, “Wavelet based progressive classfication with learning: applications to radar signals,” Proceedings of SPIE 1995 International

Symposium on Aerospace/defense sensing and Dual-use Photonics, vol. 2491, pp339-351, 1995

[10] Shihab Shamma, Naiming Shen, P. Gopalaswamy, " Stereausis: binaural processing without neural delays," Journal of the Acoustical Society of America, vol. 86, no.3, p.989-1006.

[11] C.A.Mead, X.Arreguit, and J.Lazzaro, "Analog VLSI model of binaural hearing", in IEEE Transaction on Neural Networks, vol.2, no.2, p.230-6

[12] M.P.DeSimio, T.R. Anderson, "Phoneme recognition with binaural cochlear models and the stereausis representation", in ICASSP-93, 1993 IEEE International Conference on Acoustics, Speech, and Signal Processing, pp512-4 vol.1.

[13] A. Gerso, and R.M Gray, Vector Quantization and Signal Compression. Kluwer Academic Publishers, Boston, 1991.

[14] Somkiat Sampan, "Neural fuzzy techniques in vehicle acoustic signal classification," PhD thesis, Dept. of Electr. Eng., Virginia Polytechnic Institute and State University, Blacksburg VA, April 1997

[15] Robert Karlsen, Grant Gerhart, Thomas Meitzler, Richard Goetz, Howard Choe "Wavelet analysis of ground vehicle acoustic signatures," SPIE Proc. vol. 2491, pp560-570, 1995

[16] Zbigniew Korona and Mieczyslaw Kokar, "Model based fusion for multi-sensor target recognition," SPIE Proc. vol. 2755, pp178-189, 1996

[17] W. Dress, and S. Kercel, “Wavelet-based acoustic recognition of aircraft,” SPIE proc. Wavelet Applications. vol. 2242, pp778-791, 1994

[18] Howard Choe, Robert karlsen, Grant Gerhart, Thomas Meitzler, “Wavelet based ground vehicle recognition using acoustic signals,” SPIE proc. vol. 2762, pp434-445, 1996

[19] John Baras, Sheldon I. Wolk, “Efficient organization of large ship radar databases using wavelet and structured vector quantization,” IEEE Twenty-seventh Asilomar Conference on Signal, System & Computers, pp491-498, Nov, 1993

[20] John Baras, S. Wolk, “Wavelet-based hierarchical organization of large image databases: ISAR and face recognition,” Proceedings of SPIE 12th International symposium on aerospace, defense sensing, simulation and control, vol. 3391, pp.546-558,April 1998

[21] T.Kohonen, Self-Organizing Maps. Heidelberg, Germany: Springer-verlag, 1995.

[22] A. Baldon, “Using auditory models for speaker normalization in speech recognition,” Proc. Symp. Speech Recog., Montreal, Canada, 1986

[23] R.Lyon, “A computational model of binaural localization and separation,” presented at IEEE proc. ICASSP, Boston, MA, 1983

[24] J.Cohen, “Application of an auditory model to speech recognition,” J. Acoust. Soc. Am., vol.85, pp2623-2629, 1989

[25] S. Seneff, “ A joint synchrony/mean-rate model of auditory processing,” J. Phonet., vol.16, no.1, pp55-76, 1988

[26] M.M.Merzenich, P.L.Knight, and G.L.Roth, “Representation of cochea within the primary auditory cortex in cat,” J. Neurophysiol., vol. 28, pp231-249, 1975

[27] T. Kohonen, LVQ-PAK The Learning Vector Quantization Program Packet, version 2.1, Oct 9, 1992, URL: http://hpux.petech.ac.za/ftp/hpux/NeuralNets/lvq_pak-2.1/lvq_pak-2.1-ss-9.01.tar.gz

[28] E.A.Scott, “Recognition of aerospace acoustic sources using advanced pattern recognition techniques,” Master Thesis, Virginia Polytechnic Institute and State Univ. 1991

[29] N. Kumar et al. “An analog VLSI chip with asynchronous interface for auditory feature extraction,” IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing, vol.45, no.5, p. 600-6, 1998

[30] N. Kumar et al. “An analog VLSI architecture for auditory based feature extraction,” IEEE International Conference on Acoustics, Speech, and Signal Processing, p. 5 vol5. 1997

[31] H.S. Colburn, “Theory of binaural interaction based on auditory nerve data. I. General strategy and preliminary results on interaural discrimination,” J. Acoust. Soc. Am. 75, pp879-886, 1973

[32] H.S. Colburn, N.I.Durlach, “Models of binaural interactions,” Handbook of perception,” Edited by E.C.Carterette and M.P. Friedman, Vol.IV. 1978

[33] F. Bilsen, “Pitch of noise signals: Evidence for a central spectrum,” J. Acoust. Soc. Am. 61, pp150-161, 1977

[34] Anthony J. Bell, Terrence J. Sejnowski, “An information-maximisation approach to blind separation and blind deconvolution,” Technical Report no. INC-9501, Institute for Neural Computation, UCSD, Feb, 1995