1-1-2002

Optimally weighted local discriminant bases : theory and applications in statistical signal and image processing

Kamyar Hazaveh Hesarmaskan
Ryerson University

Recommended Citation

Hazaveh Hesarmaskan, Kamyar, "Optimally weighted local discriminant bases : theory and applications in statistical signal and image processing" (2002). *Theses and dissertations.* Paper 26.

Optimally Weighted Local Discriminant Bases - Theory and Applications in Statistical Signal and Image Processing

by

Kamyar Hazaveh Hesarmaskan

A thesis
presented to Ryerson University
in partial fulfillment of the
requirement for the degree of

Master of Applied Science

in the Program of
Electrical and Computer Engineering

Toronto, Ontario, Canada, 2002

Your file Votre référence
ISBN: 0-612-87157-6
Our file Notre référence
ISBN: 0-612-87157-6

I hereby declare that I am the sole author of this thesis.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Signature

I further authorize Ryerson University to lend this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Signature

Ryerson University requires the signature of all persons using or photocopying this thesis. Please sign below and give address and date.

Optimally Weighted Local Discriminant Bases – Theory and Applications in Statistical Signal and Image Processing

Master of Applied Science
2002

Kamyar Hazaveh Hesarmaskan
Electrical and Computer Engineering Program
Ryerson University

Abstract

This thesis is concerned with Local Discriminant Basis (LDB) algorithm, its properties, optimization and applications in feature extraction and classification. LDB algorithm targets features extraction from redundant dictionaries such as wavelet packets or local trigonometric bases at low computational complexity.

As the main contribution of this thesis, an optimization process is introduced to further improve the accuracy of the overall scheme in applications when a region of interest can be specified by the experts in the field of application (based on LDB selected features) to further characterize signal classes in smaller regions. Audio signal and textured image classifications are practical applications that are studied in this thesis to test the efficiency of optimally weighted local discriminant basis algorithm (OLDB) as a feature extraction scheme. Various properties of the algorithm such as noise behavior and stability analysis are studied from an engineering perspective. The implementation aspects of the algorithm in one dimension are reviewed as well as in two dimensions that serve as implementation guidelines.

Acknowledgements

I would like to express my thanks and gratitude to my supervisor, Prof. Kaamran Raahemifar. This work would not have been possible without his continuous support and encouragement.

I am thankful to Prof. Sridhar Krishnan for introducing me to the field of statistical signal processing and for helpful discussions during the course of my studies at Ryerson University. The database for audio signal classification application was provided for this thesis by Prof. Sridhar Krishnan research group at Ryerson University. I would like to thank Prof. Naoki Saito from Mathematics Department at University of California – Davis and Prof. Fred Warner from Mathematics Department at Yale University for helpful discussions and support in the implementation of the algorithms.

I am grateful to Prof. Michael Kolios, the external examiner of my defense committee, for his valuable comments. During the course of this research, I have gratefully benefited from valuable comments from Prof. Sridhar Krishnan and Prof. Javad Alirezaie as my thesis examining committee members. I am grateful to Prof. Javad Alirezaie for helpful discussions on classifiers and image processing.

Dedications

To my parents, Bita, my sister, and Babak, my brother, that I owe my life and success to.

Contents

List of Tables

List of Figures

Chapter One

Introduction

Two of the most important problems studied in the context of signal and image processing are signal/image compression and classification. Compression in general stands for a process most commonly known as a transformation, linear or nonlinear, in which signal coordinates are mapped onto another set of coordinates in a way that signal storage space is optimized. Compression is based on disposal of less important information in the signal to reduce signal storage space. The same process leads to de-noising the input signal simultaneously considering noise to be a less important component of a signal. With the advent of state-of-the-art sensors and data acquisition machinery in all fields of science and engineering from the sophisticated magnetic resonance or ultrasound imagers to the huge amount of multimedia signal being transmitted through next generation communication systems, there is a need for effective compression methods for optimal utilization of limited technological, computational and natural resources.

The main idea of compression algorithms is to identify the most relevant features in a signal and disposal of irrelevant information for loss-less or in some cases tolerable lossy compression. The way one defines important and unimportant or relevant and irrelevant components depends on the particular application. An ideal automated scheme adapts itself so that the most relevant information is extracted tailored to the particular application. This explains why adaptive compression algorithms have gained such popularity.

The problem of pattern recognition and classification has been around for centuries. The need for machine object recognition in environments not safe for human operator or applications requiring high accuracy or speed, has led to tremendous amount of today's knowledge about different machine recognition schemes. A similar methodology as in compression algorithms is employed in signal classification problems although the goal is totally different. The ideal classifier is known as *Bayes* classifier [1] that is impractical to implement in most real signal and image processing applications due to our limited knowledge of the nature of statistical process under study. Even with collection of enough data from the process, the computational complexity to analyze such data is too high making it difficult to implement [1]. Therefore the first step in a classification application is to extract the most important features of signal space for classification. In general, the process of automatic recognition can be categorized into supervised and unsupervised methods. Adaptive methods are attractive because they allow for fast adaptation of the classification system to the particular statistical process that the input signals originate from. In either scheme one needs to extract most important features of the signal space before classification or recognition is established. This is primarily due to the ideal Bayes classifier implementation being impractical [1, 2]. In an unsupervised method such as Principal Component Analysis (PCA) the most important features of the statistical process at hand are extracted first. Then the selected features are directly fed into a classifier for classification. However, computational complexity of PCA, which is $O(N^3)$, makes it an impractical tool for the analysis of signal spaces of high dimensionality. However, supervised methods such as the one studied in this thesis can extract features from the signal space through an adaptive process with lower computational complexity.

Another important consideration in signal/image processing tools is the signal space being stationary or non-stationary. Although signal/image processors are equipped with advanced tools for stationary signal analysis, non-stationary signal processing is still in its infancy. In all the well-known analysis tools in signal processing, communication system design, control system design, etc., it is assumed that signals originate from stationary or at least wide sense stationary sources. This condition is assumed to hold in a finite or infinite interval. Therefore the signals encountered in most applications are at least assumed to be wide sense stationary in a finite interval. However, signals are in general non-stationary. The majority of concepts and ideas in time-frequency analysis of signals have been developed for non-stationary signal analysis. This is a more realistic approach.

In 1992, Coifman and Wickerhauser [3, 4] first proposed a supervised scheme, *i.e.* best basis algorithm, for extracting the most important features from a signal that proved to be an excellent adaptive method for compression and de-noising of non-stationary signals [5]. Best basis algorithm looks for time-frequency concentration of the signal in a wavelet packet or local trigonometric basis dictionary. These collections of basis functions are most suitable for adaptive non-stationary signal analysis. The reason is that they are shaped into a binary tree that naturally stems from the way they are constructed. From computer science point of view, this is an excellent data structure for various sort and optimization applications like the one looking for the best related features either for compression or classification. On the other hand, the time-frequency atoms in a wavelet packet dictionary or local trigonometric basis are suitably distributed in the time-frequency plane that allow appropriate time-frequency decomposition of the signals thus making them an excellent tool for non-stationary signal analysis [4].

In 1996, Coifman and Saito [6, 7] proposed a modification of best basis algorithm, namely Local Discriminant Basis (LDB) algorithm, for feature extraction and classification. The method proved to work well for non-stationary geological well-logging signal classification to infer petrophysical properties or the lithology of subsurface formations [7]. Wavelet packet and local trigonometric basis dictionaries are equipped with fast transformations, *i.e.* $O(N \log N)$. The search for most relevant features in tree structure of these dictionaries is also an $O(N \log N)$ computation that makes the LDB algorithm of superior performance compared to PCA in terms of computational complexity.

Here is the summary of contributions of this thesis to the original LDB algorithm:

1. Optimization of original LDB algorithm by the steepest decent algorithm
2. Application of the new optimized LDB algorithm in classifying audio signals
3. Application of the new optimized LDB algorithm in classifying textured images
4. Noise analysis of the algorithm in presence of white Gaussian noise, textured background and colored noise
5. Stability analysis of the algorithm with different number of training signals in different classes

An LDB feature extraction scheme as studied and improved in this thesis is depicted in block diagram format in Fig.1.1. A set of N-dimensional training signals form C different classes is collected from the statistical process under study in order to train the system. N_c denotes the number of training signals in class c. By an adaptive process in the original LDB algorithm and by using a set of training signals a new basis is selected such that the new basis vectors are ordered according to their discrimination power between different signal classes. Discrimination power of each basis function shows the importance of that basis function in discriminating between signals from different classes. Several discrimination measures are possible that are studied in detail in Chapter four. The result of LDB feature extraction module is the $N \times N$ orthonormal transformation matrix denoted by $L_{N\times N}$. Here N stands for input signal dimension that is the number of Cartesian coordinates in a one-dimensional signal or the number of pixels in a grayscale image. Input signal are transformed into the ordered most discriminating basis by $L_{N\times N}$. The columns of $L_{N\times N}$ are the actual LDB functions or vectors. Chapters two, three, four and five of this thesis are concerned with the construction of the transformation matrix $L_{N\times N}$. Another important module inside LDB algorithm is the dimension reduction block that keeps the topmost discriminating basis functions and discards the rest. The result of this stage is a new matrix $L'_{N\times N}$ that has zero columns except for the first few columns that are equal to that of $L_{N\times N}$. This way the LDB-based algorithms discard the unnecessary information. The effect of varying the number of selected features that is the number of non-zero columns of $L'_{N\times N}$ is studied in Chapters four, eight and nine by performing several experiments with original and optimized LDB as introduced in this thesis.

A major contribution of this thesis is toward finding a weight matrix $A_{N\times N}$ to help to decrease the statistical average distance between signals and the teacher signals of each class. Teacher signals are class representatives that can be identified by experts in the field of application based on LDB selected features. $A_{N\times N}$ is obtained by an optimization process to further improve the accuracy of the algorithm. The original LDB module plus optimization block will be referred to as Optimally Weighted (Optimized) Local Discriminant Basis (OLDB) algorithm. Experimental results verify the usefulness of the weight matrix in conjunction with Linear Discriminant Analysis (LDA) as the classifier. Chapter eight of the thesis discusses the process to obtain $A_{N\times N}$. Although original LDB algorithm selects the most discriminating features from the input signals but these features are confined to the dictionary (wavelet packet or local trigonometric basis) that the search for basis is performed in. On the other hand, LDA that is used as end point classifier in all the experiments in this thesis is the optimum classifier if input signal distributions are normal [1]. Optimization block as realized in OLDB helps to shape the distribution of selected features to look closer to normal distribution.

Triangular waveform classification problem as introduced in [34] first appears in Section 4.7 of this thesis and is used as a synthetic criterion statistical process to test the theory developed in the thesis and to perform experiments. This is a classical problem for which Bayes classifier error rate is known to be around 14%. This process is true non-stationary and has been used as a test bed for feature extraction/classification schemes by several authors [6], [34], [56]. The three classes as constructed in this synthetic example form a triangle in the high dimension space [34] and each edge of the triangle forms a class; in other words, data distribution is nontrivial, yet computationally easy to generate this database.

In order to perform experiments, both one-dimensional (1D) and two-dimensional (2D) OLDB algorithms are implemented in computational environment of Matlab® using WaveLab version 8.02 libraries [47]. Detailed implementation instructions and guidelines are given in Chapter seven of the thesis for 1D and 2D implementations of OLDB.

Another contribution of the thesis is studying the performance of optimally weighted (optimized) local discriminant bases (OLDB) in classifying audio signals. The goal is to classify pieces of classical, rock and popular music. Speech and music are known as true non-stationary processes that warrant a true time-frequency tool. Furthermore, classification into three or more signal classes as studied in this application is a challenging problem and OLDB results in 70% accuracy for classifying classical, rock and popular music segments. The audio classification studied in Chapter eight of this thesis has applications in MPEG-7 standardization that is underway by International Organization for Standardization (ISO) as an information retrieval based scheme [56].

There are a lot of potential applications for LDB algorithm in science and engineering as an effective adaptive statistical feature extractor/classifier. However, the various properties of LDB algorithm need to be studied from an engineering perspective. Another important contribution of this thesis is the study of engineering aspects of LDB algorithm such as noise analysis and behavior (in presence of both white Gaussian noise and colored noise), dependence upon the number of training signals, the response to variable number of training signals in different signal classes, etc.

Here is how the thesis is organized. Chapter two is on historical review of transform domain beginning from the Short Time Fourier Transform (STFT) and the notion of spectrograms to wavelet, wavelet packets and more advanced basis functions such as chirplets and brushlets. Chapter two serves as the mathematical foundation for the rest of the thesis. Chapter three explains the best basis algorithm in detail. Local discriminant bases, original and improved versions, are introduced in Chapters four and five. Chapter six discusses the applications of LDB that are the result of a literature review in search for LDB application areas in signal and image processing. Chapters seven, eight and nine contain the main contribution of this thesis: the development and application of OLDB algorithm. Chapter seven outlines specific guidelines for the implementation of LDB algorithm in one dimension and two dimensions respectively. Chapter eight is devoted to introducing the concepts and ideas of teacher signals and regions of interest to the optimization of LDB and theoretical foundations of OLDB. Two practical application of OLDB in audio signal classification (1D signal processing application) and texture classification (2D image processing application) are given in Chapter eight. OLDB performance in presence of white Gaussian noise, textured background and colored noise are studied in Chapter nine. The stability of OLDB in case of variable number of training signals in different classes is also covered in Chapter nine. A discussion containing future work follows in Chapter ten.

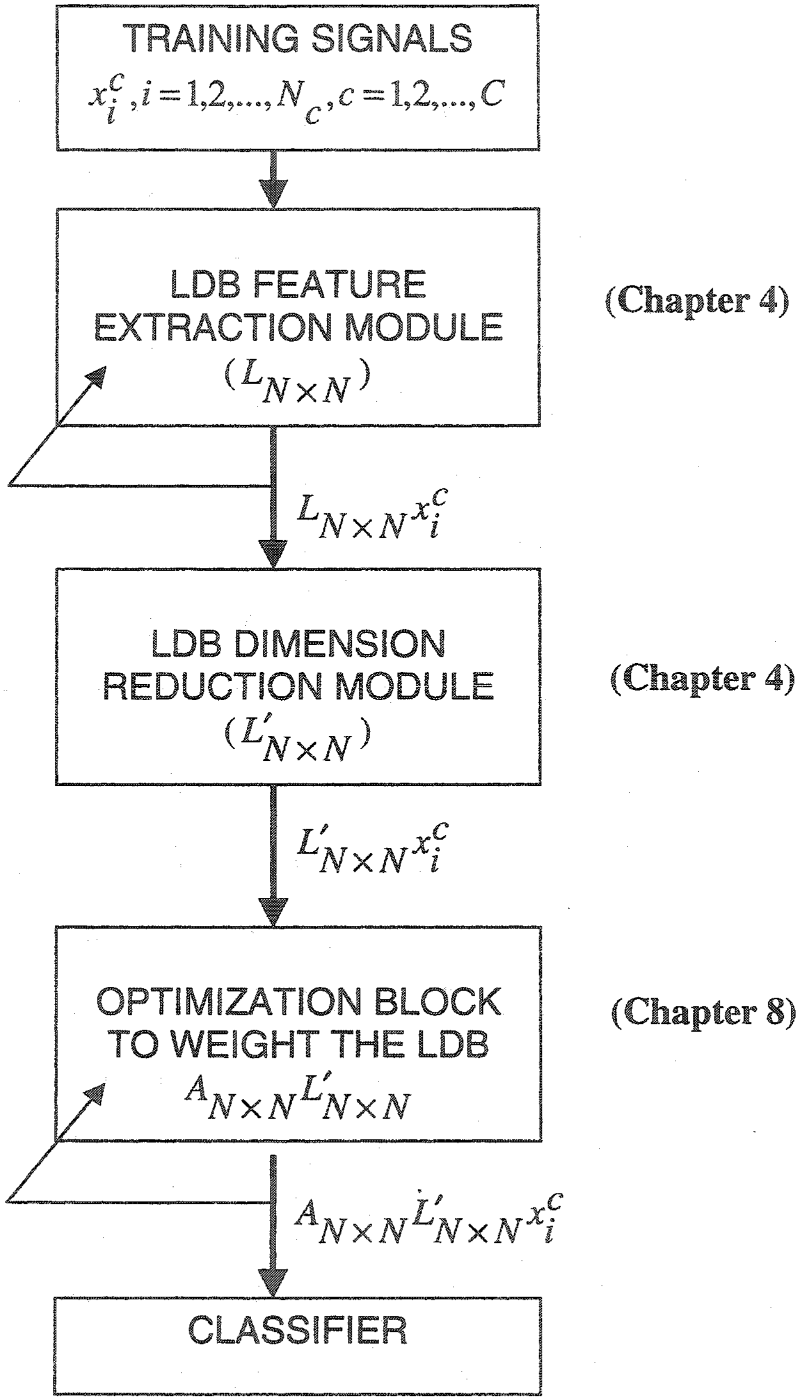

TRAINING SIGNALS
$x_i^c, i = 1,2,\ldots,N_c, c = 1,2,\ldots,C$
LDB FEATURE EXTRACTION MODULE
$(L_{N\times N})$
(Chapter 4)
$L_{N\times N}x_i^c$
LDB DIMENSION REDUCTION MODULE
$(L'_{N\times N})$
(Chapter 4)
$L'_{N\times N}x_i^c$
OPTIMIZATION BLOCK TO WEIGHT THE LDB
$A_{N\times N}L'_{N\times N}$
(Chapter 8)
$A_{N\times N}L'_{N\times N}x_i^c$
CLASSIFIER

Fig.1.1 The role of optimization block in an optimally weighted local discriminant basis feature extraction scheme (the thesis chapters discussing individual blocks given at adjacent right)

Chapter Two

Transform Domain

The mathematical theory and foundations of local discriminant basis algorithm lie in time-frequency redundant transforms that are studied in this chapter. These are complete sets of basis functions for the class of square-integrable functions. The concepts of *normed* space and *Banach* space are introduced. The class of square-integrable .functions, $l^2(R)$, is presented as a complete Banach space that is called a *Hilbert* space. Any transform is considered as a complete orthornormal sequence for $l^2(R)$ as a Hilbert space. The underlying ideas of different transformations and their usefulness are studied in detail. Beginning from the applications of transform domain in signal processing and discussing the limitations of classical *Fourier* transform in signal compression and classification, *wavelet*, *wavelet packet*, *chirplet* and *brushlet* transforms are introduced as non-stationary signal analysis tools in later Sections.

Frequency domain representation as obtained by Fourier transform not only helps to study the various properties of signals and systems but more importantly it has led to concepts and ideas that would have never emerged had they not been thought of in frequency domain. The majority of novel ideas in communication systems like frequency allocation, modulation, multiplexing, etc., stem from frequency domain signal representations. The same is true for the majority of applications in signal/image processing and control systems.

Fourier transform is not a useful tool if the time localization of events in a signal is required. This is because Fourier transform is a global transform in the sense that it is in the form of integration of the signal multiplied by an exponential function over the entire time axis. In order to capture the local phenomena in a signal one solution is to limit the Fourier transform to a finite length closed interval in time domain. This very simple idea led to the introduction of *Gabor* general family of transforms and in particular *Short Time Fourier Transform* (STFT) [8]. Yet another solution is the use of transforms of variable resolution nature like wavelet transform. This category, better known as multi resolution analysis in image processing context, has played a major role in understanding signals and systems in the last two decades [9]. However, in all linear time-frequency transforms such as wavelet transform there is a fixed relationship between the resolution in time domain and resolution in frequency domain at any single time-frequency basis function or atom. This is best explained by the *Heisenberg uncertainty principle* originally introduced in quantum mechanics [9]. If the signals at hand originate from a non-stationary source, it is desirable to have more flexibility in terms of time and frequency resolution at different time and frequency locations. The ultimate goal is to capture frequency characteristics of the signals at any single point in time. A family of nonlinear transforms initiated by Wigner-Ville distribution in 1948 was further expanded to the general family of Cohen transforms in the hope of capturing instantaneous frequency in the signal [10]. The need to analyze non-stationary systems has led to the introduction of a number of adaptive algorithms for the selection of the most appropriate time-frequency basis functions also known as time-frequency atoms (from a redundant set of basis functions) to capture the local characteristics of non-stationary signals. Best basis [3], local discriminant basis (LDB) [7], matching pursuit [11] and projection pursuit [12] are examples of the existing techniques for adaptive time-frequency analysis.

This is how this chapter is organized. The basic concepts and underlying definitions of Hilbert space are given in Section 2.1 which serves as the basis for all future discussions on transform domain and reveals the mathematical foundations of transform domain signal processing. The core ideas of compression and classification in transform domain are studied in Section 2.2. Since the goal of time-frequency analysis is the study of non-stationary phenomena, the non-stationary statistical process is mathematically formulated and characterized in Section 2.3. Different time-frequency transforms as used in this thesis along with their construction, properties and event capturing resolution in time-frequency plane are studied in Section 2.4 through Section 2.16. Classical transforms such as Fourier and wavelet are reviewed. The state-of-the-art redundant or over-complete transforms such as wavelet packet and local trigonometric bases are studied. Basis functions such as chirplets [13] and brushlets [14] are also covered briefly. These categories of basis functions can be used in local discriminant basis algorithm of Chapter four and eight although to our knowledge currently no implementation of such adaptive schemes exists. In next chapters local discriminant basis algorithm is presented as an adaptive method for the selection of best basis from wavelet packets as redundant sets of basis functions for signal classification applications.

2.1 Hilbert Spaces and Orthonormal Systems

The mathematical foundations of all time-frequency representations lie in complete orthonormal sequences in separable Hilbert spaces. The preliminaries and definitions in order to understand Hilbert space bases are given in this section. In signal processing applications one mostly deals with a certain class of functions called $l^2(R)$ functions. The function space $l^2(R)$ is the space of all complex-valued Lebesgue-integrable functions [15] defined on R with the $l^2(R)$ norm [16] defined by (1),

$$\|f\|_2 = [\int_{-\infty}^{+\infty} |f(x)|^2 dx]^{1/2}, \tag{1}$$

where $\|f\|_2$ stands for the norm in $l^2(R)$ as a *normed* vector space [17] and $|f(x)|$ stands for absolute value of a real variable or the magnitude of a complex function. Elements of $l^2(R)$ are called square-integrable functions. Many functions in physics and engineering, such as wave amplitude in classical or quantum mechanics, all finite energy voltage or current waveforms in electrical engineering, etc., are square-integrable and that contributes to the importance of $l^2(R)$ in science and engineering.

Example 2.1. The function $|x|^{-1/2} e^{-|x|} \notin l^2(R)$. On the other hand, $(1+|x|)^{-1} \in l^2(R)$. Both statements are verified with the existence and non-existence of Lebesgue integral [8] for the particular function.

Definition 2.1. A normed space X is called *complete* if every *Cauchy* sequence [15] in X converges to an element of X. A complete normed space is called a *Banach* space [17].

Example 2.2. R^n, the space of real n-dimesional vectors and the space of complex n-dimensional vectors, C^n, both constitute Banach spaces with the common Euclidean distance as norm [17].

It can be verified that $l^2(R)$ constitutes a complete normed space or Banach space with the norm defined in (1) [8]. To see properties of a complete normed space the reader is referred to [17]. $l^2(R)$ is equipped with an inner product given in (2),

$$< f, g >= \int_{-\infty}^{+\infty} f(x)\overline{g(x)}dx \ , \tag{2}$$

where "<,>" shows the inner product of its entries and $\overline{g}$ denotes the complex conjugate of g. To each inner product space there is a norm assigned that is directly induced by its inner product [17]. The norm introduced in (1) is the same as the norm induced with inner product (2) [8].

Definition 2.2. A complete inner product space is called a *Hilbert* space.

It can also be shown that $l^2(R)$ is a complete inner product space and thus a Hilbert space [8].

Definition 2.3. Let X be an inner product space. A family S of nonzero vectors in X is called an *orthogonal system* if the inner product of any two disjoint elements of S is zero, *i.e.* they are orthogonal. If in addition, the norm of each element of S is 1, then S is called an orthonormal system.

Definition 2.4. A finite or infinite sequence of vectors, which form an orthonormal system is called an orthonormal sequence.

Definition 2.5. An orthonormal sequence $\{x_n\}$ in an inner product space is said to be complete if for every $x \in X$ one has,

$$x = \sum_{n=1}^{\infty} < x, x_n > x_n \ . \tag{3}$$

In other words, an orthonormal sequence is considered complete if any signal x can be represented by its projections on this sequence.

Definition 2.6. An orthonormal system S in an inner product space X is called an orthonormal basis if every $x \in X$ has a unique representation

$$x = \sum_{n=1}^{\infty} \alpha_n x_n \ , \tag{4}$$

where $\alpha_n \in C$ and x_n's are distinct elements of S. Any orthonormal system for $l^2(R)$ results in signal representation that can prove useful in a particular application.

Example 2.3. The orthonormal system,

$$\phi_n(x) = \frac{e^{inx}}{\sqrt{2\pi}} \ , \tag{5}$$

is complete in the space $l^2(R)$. The proof of completeness is not simple. The reader is referred to [8] for a detailed treatment of this problem. Note that (5) represents the usual Fourier basis for $l^2(R)$.

Definition 2.7. A Hilbert space is called separable if it contains a complete orthonormal sequence. Finite dimensional Hilbert spaces are considered separable.

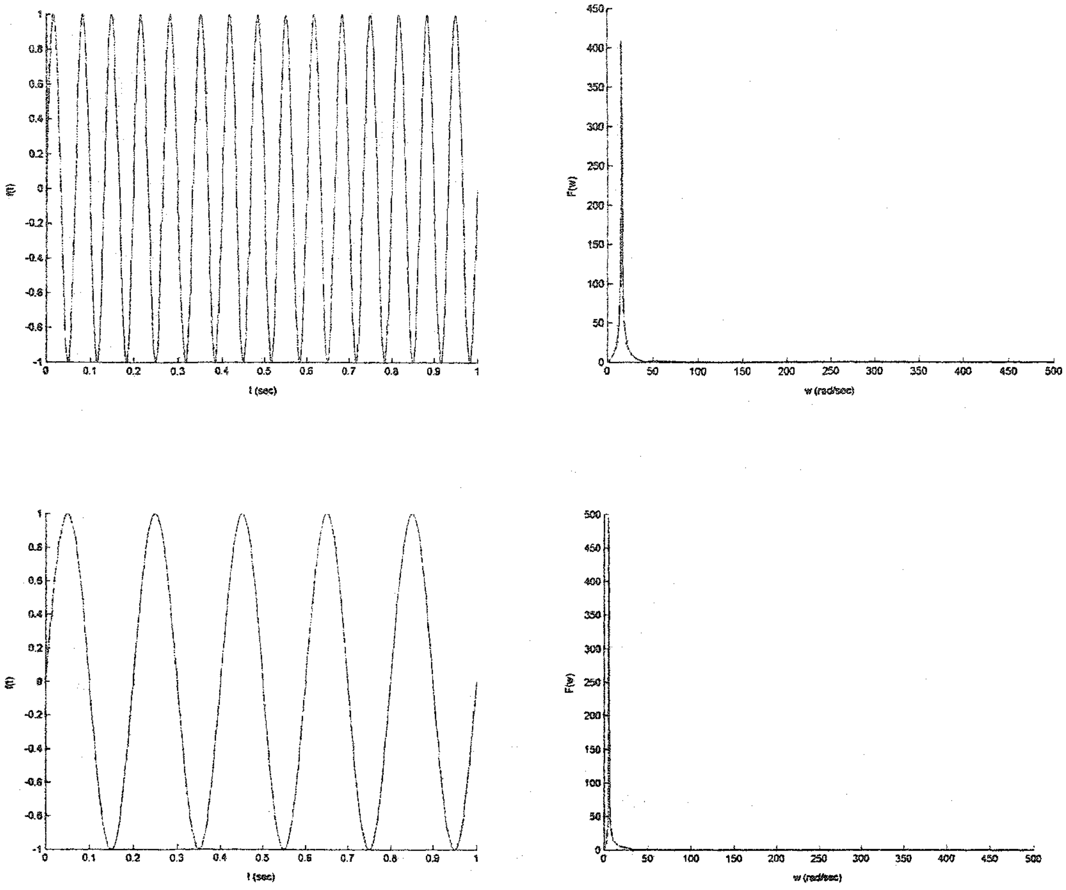
Example 2.4. Based on Example 2.3 the Hilbert space $l^2(R)$ is separable.

All the efforts toward finding a suitable signal representation are equivalent to finding a complete orthogonal sequence in $l^2(R)$. The number of such sequences is infinite. A number of complete orthogonal sequences are studied in later sections of this chapter.

2.2 Compression and Classification in Transform Domain

Fourier transform is best known for its extensive application over centuries in understanding the behavior and frequency content of stationary signals and systems that are encountered in different areas. The Fourier transform introduces a one-to-one map between signal space and its transform domain both to be $l^2(R)$ functions. Equality and uniqueness in all the following discussions are considered to hold in $l^2(R)$ norm as a Hilbert space. This means that whenever two signals are considered equal, their difference is of zero energy. This is a useful definition in the context of electrical engineering since a voltage signal with zero energy cannot for instance activate an electrical motor or produce audible acoustic signal. This is one of the main reasons why $l^2(R)$ with its usual inner product and induced norm has gained much importance in science and engineering.

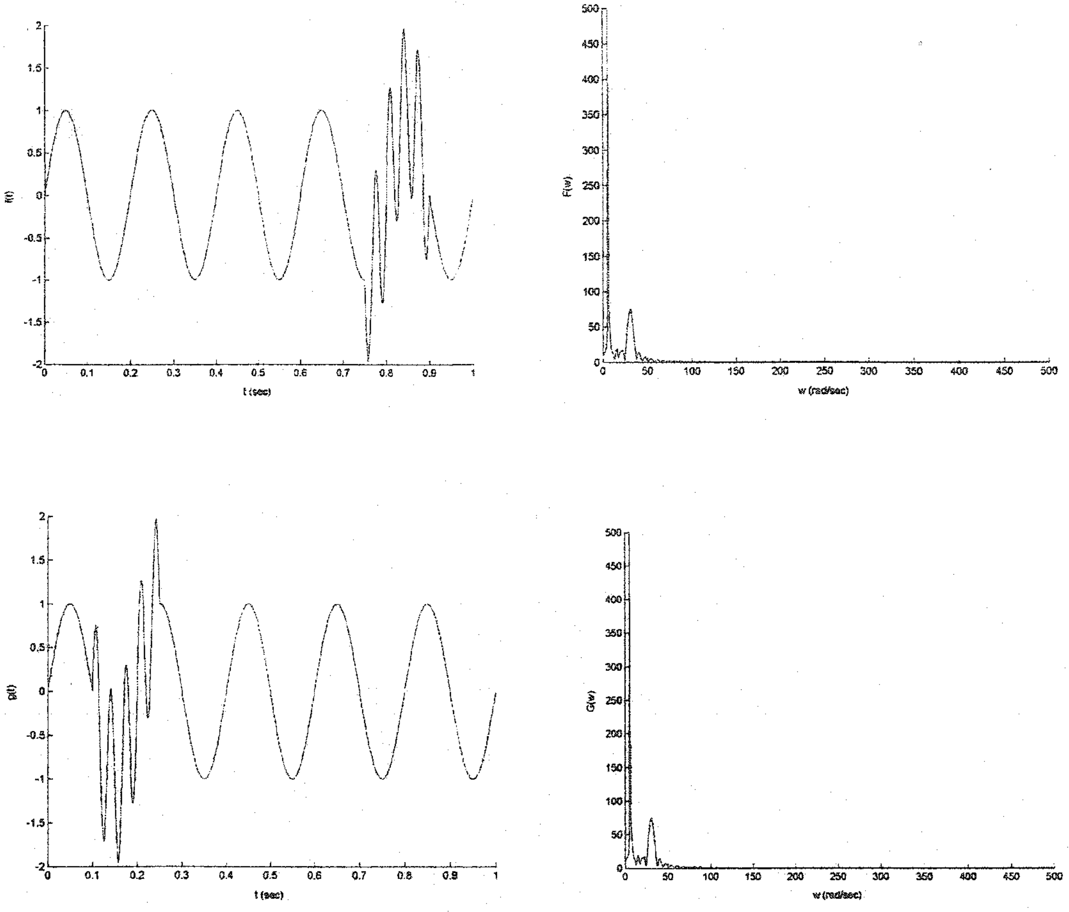
Of all different applications of Fourier transform in science and engineering we are only interested in its properties and significance in signal compression and signal classification context. Fig.2.1 explains how the Fourier transform is useful in a compression or classification problem. The signals in Fig.2.1 (a) are from two different sources with two different frequencies. Fig.2.1 (b) is the Fourier transform of the signals depicted in Fig.2.1 (a). Apparently it's much easier to classify signals in the transform domain than the original time domain. As far as classification is the only concern, the transform domain is preferred since the two signals are represented by the locations of the impulses in frequency domain that serve as an unambiguous and efficient criteria for classifying the signals. Similarly if signal compression is the major concern, the Fourier transform is again a valuable tool since it is clearly reducing the signal dimension, basically from infinity to one, *i.e.* the position of the impulse. Besides the tones in Fig.2.1 (a) can be unambiguously reconstructed from their Fourier transforms. However, it's easy to think of examples in which the Fourier transform is not able to differentiate between two different signals. Fig.2.2 (a) is an example of two different signals with the same Fourier transform as depicted in Fig.2.2 (b).



(a) (b)

Fig.2.1 (a) Two signals originating from sources with different frequencies (b) the Fourier transform of the signals with different peak locations suitable for compression and classification

The Fourier transform cannot tell the difference between signals in Fig.2.2 (a) simply because it's a global transform or integration in time domain. Due to the very first definition of Fourier transform as a global integration operator, it is impossible to locate local phenomena in the signal. In other words, one can identify the presence of a particular frequency in the signal but there is absolutely no way to tell when the event corresponding to this particular frequency has happened.



(a) (b)

Fig.2.2 (a) Two signals originating from a non-stationary process to be classified (b) the Fourier transform of the signals in (a) look the same due to inability of the Fourier transform to track local events in the signals, thus making it not suitable for non-stationary signal analysis

2.3 Stationary and Non-stationary Signals

Signals are in general non-stationary. A complete representation of non-stationary signals requires frequency analysis that is local in time, resulting in time-frequency analysis of signals and the notion of instantaneous frequency [10]. The Fourier transform analysis of signals has long been recognized as a great tool for the study of stationary signals and processes where the statistical properties are invariant over time. By statistical properties we mean the expectation of the process as a random variable and its integer powers at any particular point in time. Of particular importance are the first and second moments [19]. In recent years several time-frequency analysis tools have been introduced that are particularly suitable for the analysis of signals originating from or produced by non-stationary sources. Decomposition of a signal into a small number of elementary functions that are well localized in time and frequency plays an important role in signal processing. Such decompositions reveal important structures for non-stationary signal analysis such as speech and music. A number of these techniques are reviewed in next sections. In Chapter eight the adaptive time-frequency analysis tool developed in this thesis is applied in classification of music as a non-stationary signal space. While testifying the stationary of a process is not easy unless in trivial cases, the proof of non-stationarity of a process can be as easy as observing changing statistical properties over time.

2.3.1 Deterministic versus Random Signals

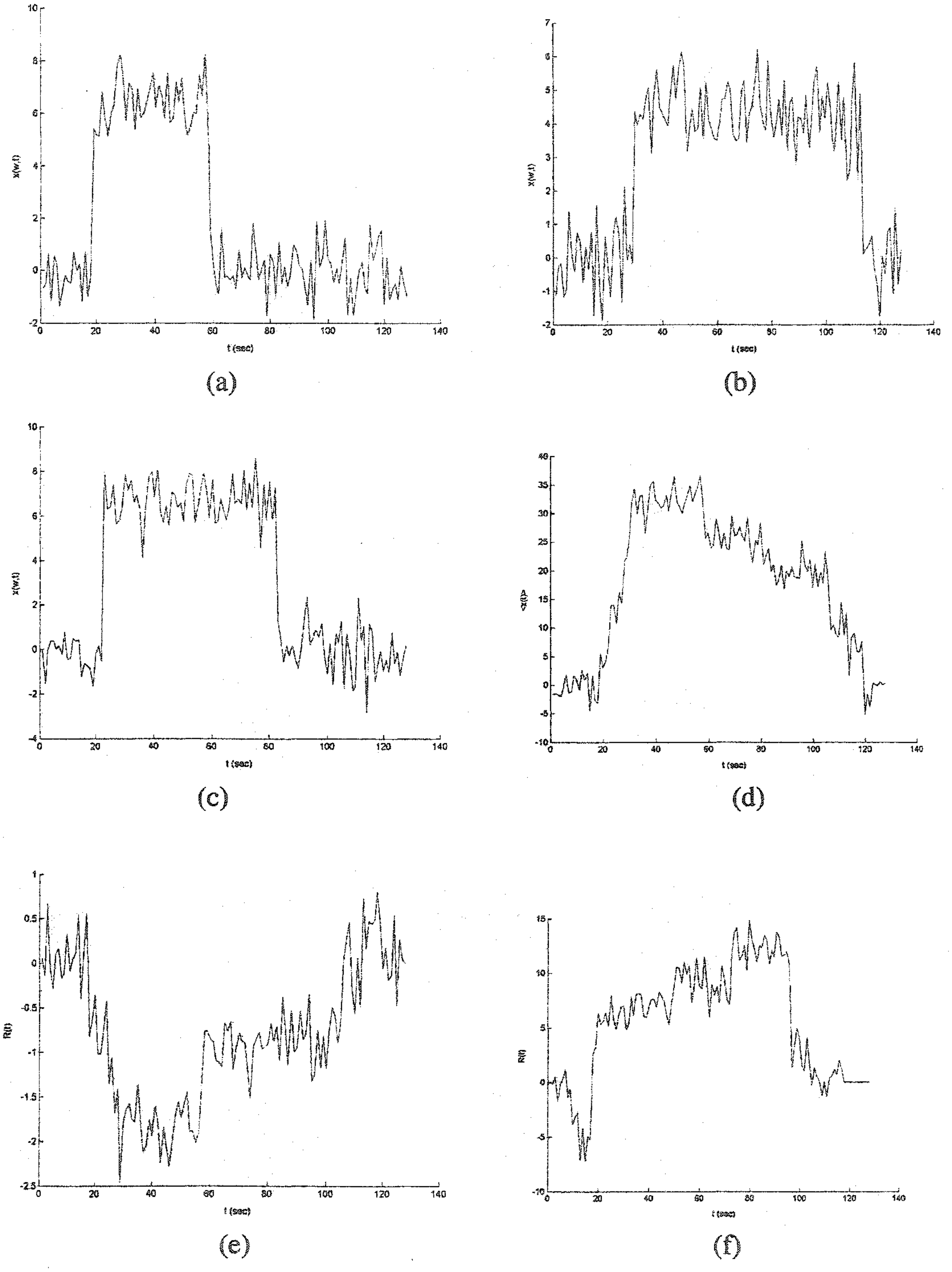
In general there are two main category of signals: (i) deterministic and (ii) random or stochastic. A signal is called deterministic if it can be determined or regenerated explicitly, under identical conditions, in terms of a mathematical expression. On the other hand, signals are in general random or stochastic in the sense that they can not be determined precisely at any given instant of time even under identical conditions by exact mathematical expressions. Obviously for description of random signals, probabilistic and statistical information are required. The collection of the data generated by a random source at n different times constitutes an *ensemble* of n separate records (Fig.2.3). The average value at time t over the ensemble x is defined by (6),

$$< x(t) >= \lim_{n\to\infty}\frac{1}{n}\sum_{k=1}^{n} x_k(t), \tag{6}$$

where x takes any one of the set of values x_k and $k = 1,2,3,...,n$ (Fig.2.3 (d)). The average value of the product of two samples taken at two separate times t_1 and t_2 is called the autocorrelation function R that is defined for each separate record and is given by (7),

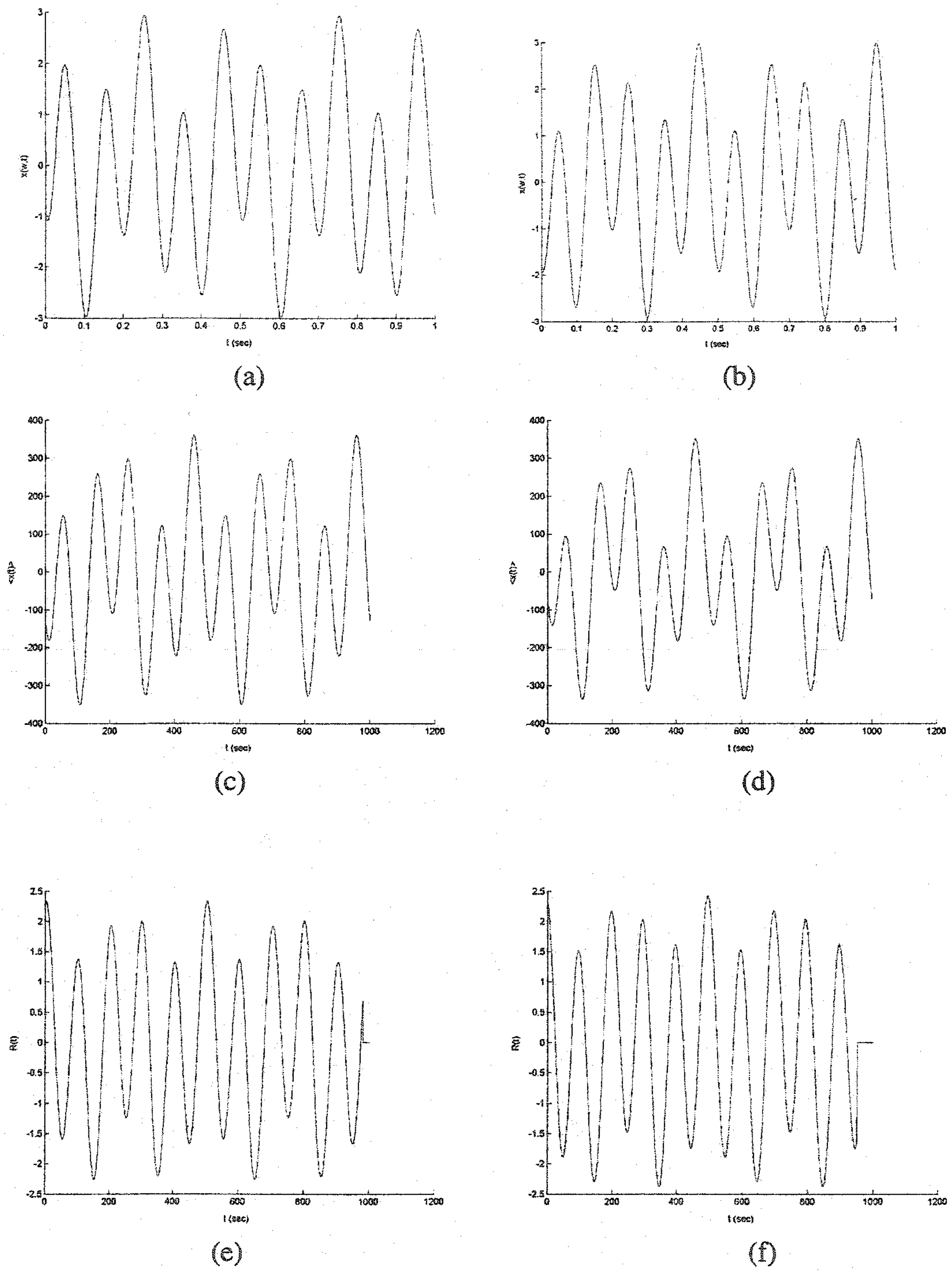
$$R(t_1,t_2) = \lim_{n\to\infty}\frac{1}{n}\sum_{k=1}^{n} x_k(t_1)x_k(t_2). \tag{7}$$

In the above expression x_k's are records of the process. The process of finding R values at different time instants is referred to as ensemble averaging and may be done for each individual record and continued over the entire record length to provide statistical information on the complex set of records (Fig.2.3 (e) and (f)). A random process is called *stationary* in the *strict sense* if its statistics are not affected by a shift in the time origin. More specifically this means that $x(t)$ and $x(t+\tau)$ have the same statistical properties for arbitrary τ. In other words, the idea of stationarity is that a time translation of a sample function results in another sample function having the same probability.



(a) (b) (c) (d) (e) (f)

Fig.2.3 (a) (b) (c) Ensemble functions of a non-stationary random process (that will be described in Section 4.7) (d) the ensemble average (e) autocorrelation function calculated using $t_1 = 1$ and t_2 varying (f) the same experiment with $t_1 = 10$



(a)

(b)

(c)

(d)

(e)

(f)

Fig.2.4 (a) (b) Ensemble functions of a sinusoidal process with random phase (c) ensemble average using 20 samples (d) ensemble average using 50 sample (e) autocorrelation function calculated with $t_1 = 20$ fixed and t_2 varying (f) the same experiment with $t_1 = 50$

A signal is called *stationary* in the *wide sense* if the values of $< x(t) >$ and $R(t_1, t_2)$ remain constant for all records and all possible values of t_1 and t_2 provided that $\tau = t_1 - t_2$ is kept constant. Therefore R can be written as $R(\tau)$. In most practical situations, a signal is called stationary if the above conditions are satisfied over a finite time interval as in Fig.2.4. A signal is non-stationary if the values of $< x(t) >$ and $R(t_1, t_2)$ depend on the particular record or vary with time instants t_1 and t_2. Fig.2.3 is an example of signal recorded from such a process. Whether a process is stationary or not can be verified easily if an analytical description of the process is available. Lacking such a description, one has to collect data and analyze it to check whether the process is stationary. For details of procedures to be used for collecting, analyzing and testing data for determining the stationarity of a process, the interested reader is referred to [20]. However in applications where the changes are slow, the signal is regarded as stationary. In a number of practical purposes in communications and signal processing, the signals are assumed to be stationary in the wide sense. However, a more realistic approach is to consider them as non-stationary processes.

2.3.2 Ergodic Random Processes

Ergodicity deals with the problem of determining the properties of a random process, such as the mean and autocorrelation function, from a single record. Under certain conditions, the statistical characteristics of a signal can be obtained from one of its long records. To examine the ergodicity of a random process, one should compare the statistical moments of the process with that of a single record [19]. By definition, an ergodic process is necessarily a stationary one but the inverse may not be true. In practice it is very hard to decide on the basis of data if a random process is ergodic. One has to decide about ergodicity based on reasoning about the physical phenomenon involved. In order to be ergodic, a random process must be stationary and randomness must be evident in time variation as well as in the selection of a record. In addition, time averages should not depend on the particular record that is selected. The processes that are the subject of this thesis are non-stationary and non-ergodic. Therefore always an ensemble of statistical process is used for the purpose of illustration.

2.4 Short Time Fourier Transform (STFT)

It was shown in Fig.2.2 that Fourier transform is unable to classify signals whenever a time localization of discriminating features is required. In order to treat problems as the one shown in Fig.2.2, it is possible to take the Fourier transform over closed intervals in time domain. This procedure results in several possible treatments of Fig.2.2 classification problem and time-frequency domain invention.

The idea is to multiply the signal by a window function such as the Gaussian window function in Fig.2.5 (b). Then one computes the Fourier transform of the portion of the signal that is confined to the window function. This results in the transformed signal in the frequency domain as shown in Fig.2.5 (d). Fig.2.5 (f) is the result of shifting the window over the time axis and calculating the Fourier transform each time. One can place the frequency content of each interval vertical to that interval to get the complete *Short Time Fourier Transform* (STFT) as shown in Fig.2.6 that is known as *spectrogram* of the signal. The mathematical expression for STFT is given in (8). As far as the window function is known, this expression is a 1-1 correspondence between the signal and its transform. The inverse STFT formula is given in (9).

$$F(u,\xi)=\int_{-\infty}^{+\infty} f(t)g(t-u)e^{-i\xi t}dt \tag{8}$$

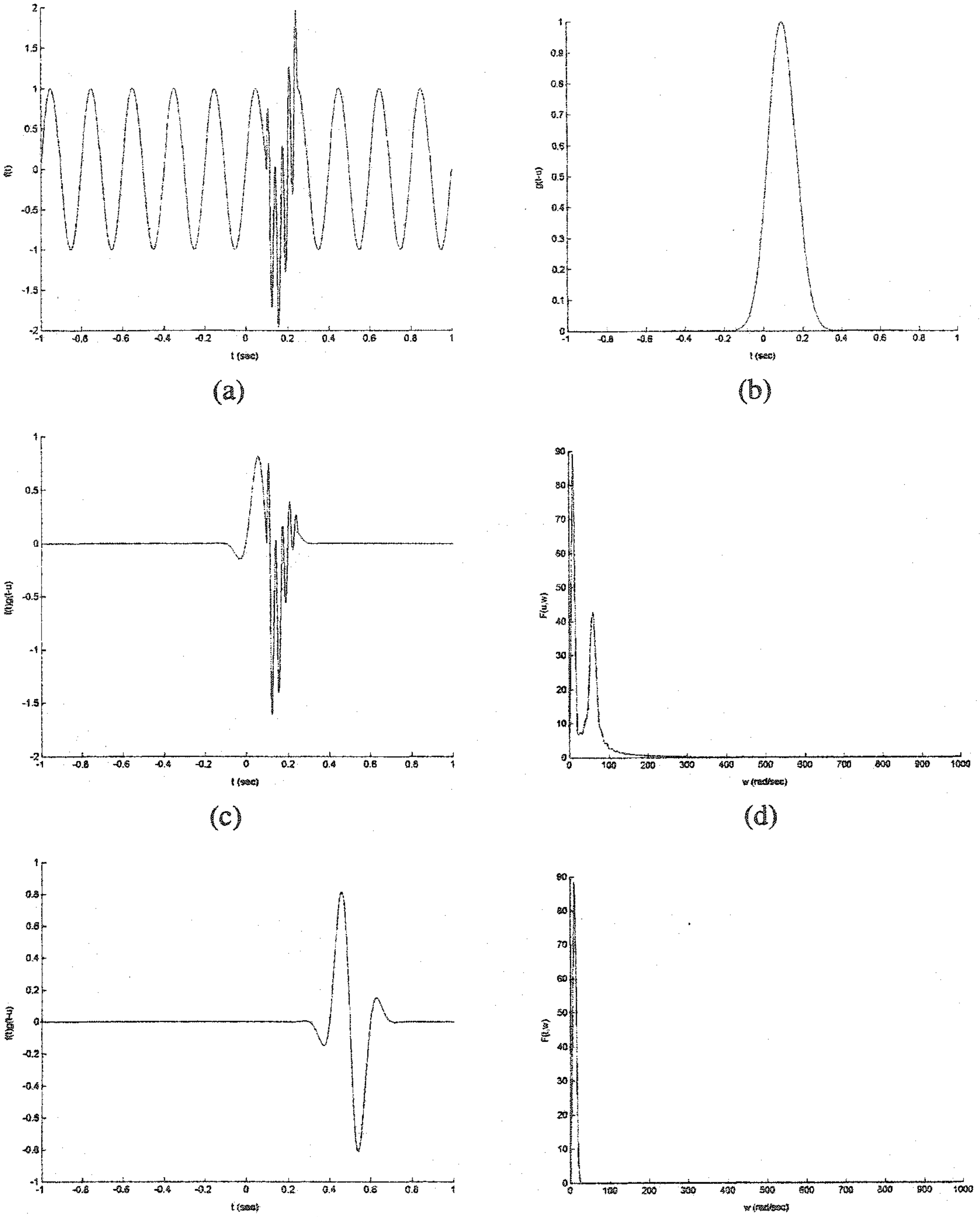
$$f(t)=\frac{1}{2\pi}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} F(u,\xi)g(t-u)e^{i\xi t}d\xi du \tag{9}$$

In (8) and (9), $g(t)$ is the window function. u is the translation parameter and ξ is representative of frequency modulation of the window function as in continuous Fourier transform.

Fig.2.6 is a convenient representation of Short Time Fourier Transform of two signals that apparently can tell the difference in time-frequency domain. The spectrograms of Fig.2.6 are obtained by multiplication of the input signal by a succession of adjacent windows. The summation of window functions used in obtaining spectrograms is the unity function [9]. The spectrogram provides a means to study the frequency content of signals in different time intervals. The spectrograms in Fig.2.6 can be obtained by several computer tools such as the one used to produce figures in this thesis namely FAWAV [54]. What is shown in Fig.2.6 is the beginning of our discussion of time-frequency plane. Time-frequency transforms such as the one in Fig.2.6 contain the information about the frequency content of the signal at different times. However, there is always a limitation in how accurately one can locate an event simultaneously in time and frequency.

2.5 The Heisenberg Uncertainty Principle

It is a well-known fact in quantum mechanics that if one needs to measure the position and momentum of a particle at the same time, there is a limit on how accurately this measurement can be done. The *Heisenberg uncertainty principle* states that the product of the uncertainty associated with the position and the momentum measurements of a particle cannot be less than a threshold [9, 10]. A similar principle governs the measurement of instantaneous frequency [9].



(a)

(b)

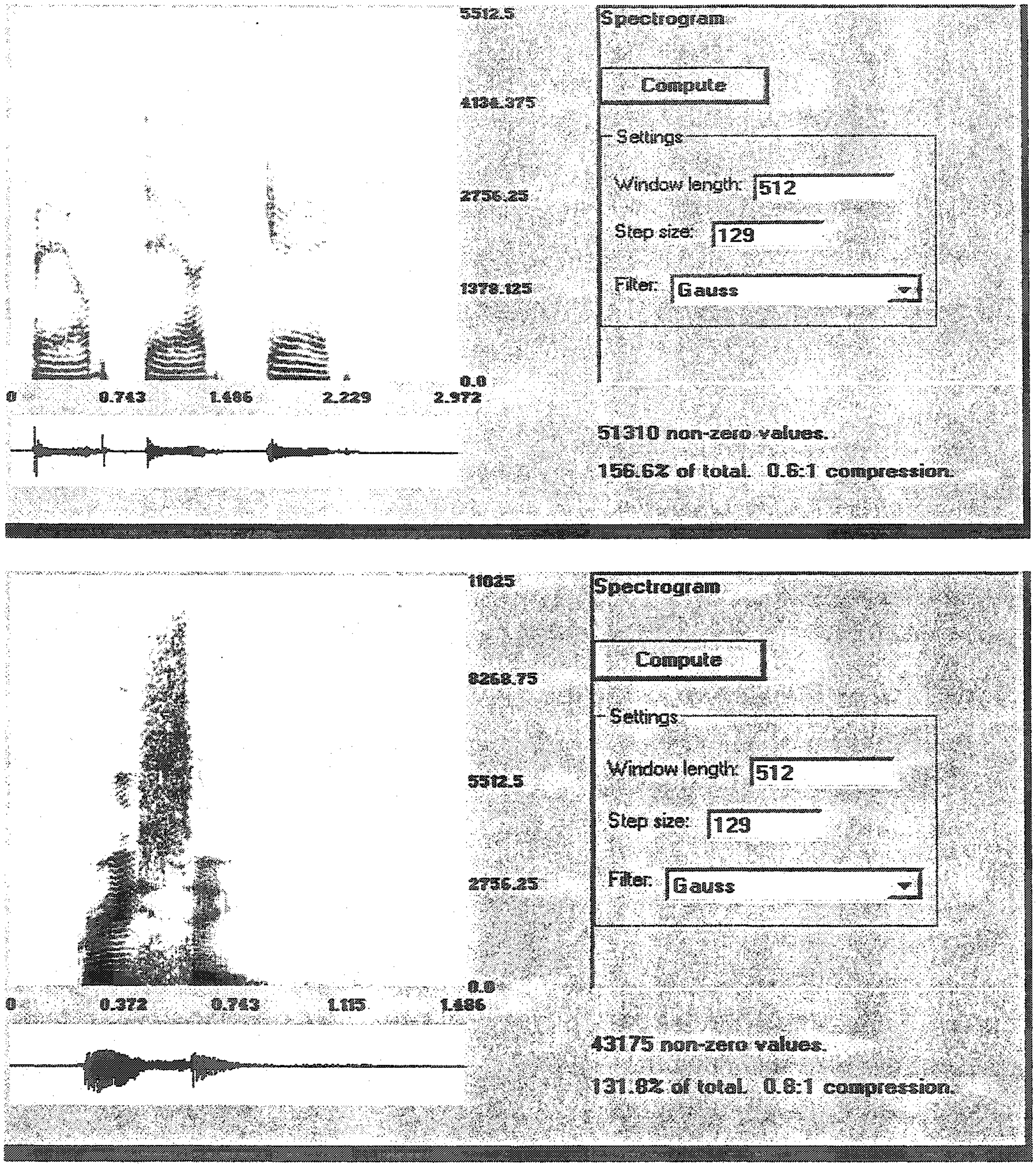
(c)

(d)

(e)

(f)

Fig.2.5 (a) A non-stationary signal (b) window function to analyze the signal (c) windowed function (d) the Fourier transform of the windowed function (e) the windowed function with a shifted window (f) its Fourier transform



Window length: 512

Step size: 129

Filter: Gauss

51310 non-zero values.

156.6% of total. 0.6:1 compression.

Window length: 512

Step size: 129

Filter: Gauss

43175 non-zero values.

131.8% of total. 0.8:1 compression.

Fig.2.6 Two spectrograms showing STFT computed for evaluating non-stationary speech signals: a word recording (top) and a whistle recording (bottom)

In section 2.4 it was asserted that STFT is a method to capture local phenomena in a non-stationary signal. This ideally means that it is a means of specifying the frequency content of a signal at any single point of time with zero uncertainty or ultimate accuracy. Nonetheless, this is not achievable due to the Heisenberg uncertainty principle. Fig.2.7 (a) shows this fact clearly in the measurement of the frequency content of a single tone. If the integration is done on entire time axis (usual Fourier transform) or equivalently if one can observe the signal over a very large time window then one can identify it as a tone and show its frequency content as a single impulse exactly at the frequency of the tone. However, this is not possible if one takes a part of the tone or a windowed tone as shown in Fig.2.7 (b). Here the frequency content of the signal cannot be identified with any desired high accuracy. The reason is better shown in Fig.2.8 when one can easily tell the difference between the two signals in Fig.2.7 and Fig.2.8 outside the time window. Therefore, by only looking at a windowed portion of the signal it is not possible to identify the frequency content of a signal unless some uncertainty is taken into account. This is the very basic limitation of time-frequency transforms.

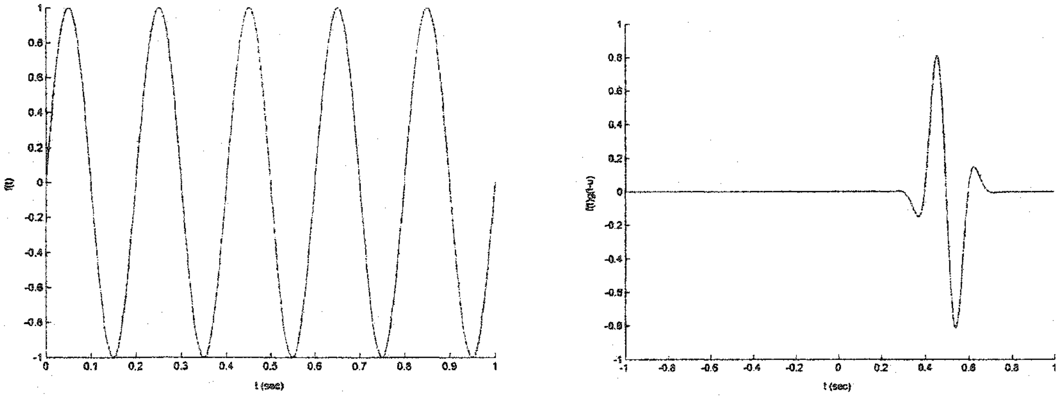


Fig.2.7 (a) A stationary sinusoidal signal (b) its windowed version

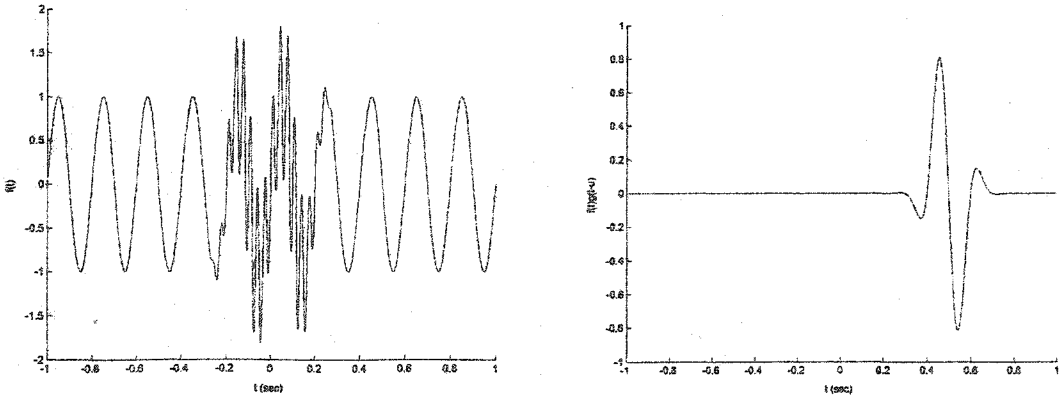


Fig.2.8 (a) A non-stationary signal (b) its windowed version that looks like the one given in Fig.2.7 (b) although the signals are totally different

Heisenberg formulated the uncertainty principle in the context of quantum mechanics, which states that the position and momentum of a particle described by a wave function cannot be determined exactly and simultaneously [8, 9]. In signal processing, time and frequency concentrations of energy of a signal $f(t)$ are also governed by the Heisenberg uncertainty principle. The expected value of time and frequency are given by (10) and (11). The variances of these quantities are given by (12) and (13),

$$<t>=\frac{1}{\|f\|_2^2}\int_{-\infty}^{+\infty} t|f(t)|^2 dt \quad , \tag{10}$$

$$<\omega>=\frac{1}{\|F\|_2^2}\int_{-\infty}^{+\infty} \omega|F(\omega)|^2 d\omega, \tag{11}$$

$$\sigma_t^2=\frac{1}{\|f\|_2^2}\int_{-\infty}^{+\infty}(t-<t>)^2|f(t)|^2 dt \quad , \tag{12}$$

$$\sigma_\omega^2=\frac{1}{2\pi\|F\|_2^2}\int_{-\infty}^{+\infty}(\omega-<\omega>)^2|F(\omega)|^2 d\omega \quad , \tag{13}$$

where "< >" denotes expectation or averaging process. $f(t)$ is the time or space domain signal and $F(\omega)$ is its Fourier transform. $\|f\|_2$ and $\|F\|_2$ stand for $f(t)$ and $F(\omega)$ magnitude in $l^2(R)$ as functions of t and ω respectively. σ_t^2 and σ_ω^2 are representative of the spread of $f(t)$ in time and frequency respectively.

Proposition 2.1. If $f(t)$ and $tf(t)$ and $\omega F(\omega)$ belong to $l^2(R)$ and $\sqrt{t}|f(t)|\to 0$ as $|t|\to 0$, then

$$\sigma_\omega^2\sigma_t^2\geq\frac{1}{4}, \tag{14}$$

where σ_t is a measure of duration of signal $f(t)$ and σ_w is a measure of frequency dispersion or bandwidth of its Fourier transform $F(\omega)$ as defined in (12) and (13) respectively [9]. Equality in (14) holds if and only if $f(t)$ is a Gaussian distribution as given in (15).

$$f(t)=Ce^{-bt},b>0 \quad , \tag{15}$$

where C is an arbitrary real constant [9]. In a time-frequency analysis of signals, the measure of resolution of a signal $f(t)$ in time or frequency domain is given by σ_t and σ_w.

Then the joint resolution is given by the product $\sigma_w\sigma_t$ and the 1/4 lower bound of this product is known as minimality condition. In other words, the product cannot be arbitrary small and is always greater than the minimum value 1/4 which is attained only in case of Gaussian distribution. In many applications in science and engineering, signals with a high concentration of energy in the time and frequency domains are of special interest. The uncertainty principal can also be interpreted as a measure of this concentration of the second moment of *f*(*t*) squared and its energy spectrum $F(\omega)$ squared.

2.6 Gabor Transform

Signals can be described in time domain or frequency domain by the traditional Fourier transform or STFT method. It was recognized long ago that Fourier transform or spectral analysis of a signal with a long duration is of little practical value in analyzing the behavior of signals produced by real physical processes. Transient signals such as speech or ECG signals require the idea of frequency analysis that is local in time. In general the frequency content of a signal varies with time. Therefore there is a need for joint time-frequency analysis of the signal in order to fully describe the characteristics of the signal as discussed in previous sections. Motivated by the Heisenberg principle, Gabor, a Hungarian-British physicist and engineer introduced the joint time-frequency representation of a signal in 1946 [10]. At the same time Ville introduced the Wigner distribution to unfold the signal in time-frequency domain [10].

Gabor first identified a signal with a family of waveforms that are well concentrated in time and in frequency. He called these elementary waveforms the time-frequency atoms that have a minimal spread in a time-frequency plane [9]. In fact, he formulated a fundamental method for decomposition of signals in terms of elementary signals or atomic waveforms. In order to incorporate both time and frequency localization properties, he first introduced the windowed Fourier transform (STFT or Gabor transform) by using a Gaussian distribution function as a window function. The major idea was the use of a time-localization window function like $g_a(t-b)$ for extracting local information from the Fourier transform of a signal, where parameter a measures the width of the window or the scale and the parameter *b* is used to translate the window in order to cover the whole time domain. In fact, Gabor used $g_{t,\omega}(\tau)=\overline{g}(\tau-t)e^{i\omega t}$ as the window function by translating and modulating the complex conjugate of the function *g*, where $g(\tau)=\pi^{-1/4}e^{-\tau^2/2}$, which is the so-called canonical coherent states in quantum physics and this transform was previously presented as STFT. The Gabor transform of *f*(t) with respect to $g_a(t-b)$ denoted by $F_{g_a}(t,\omega)$, is defined in (16),

$$F_{g_a}(t,\omega)=\int_{-\infty}^{+\infty} f(\tau)g_a(\tau-t)e^{-i\omega t}d\tau \ , \tag{16}$$

where *f*, *g* belong to $l^2(R)$ with the usual inner product . In (16), $g_a(t-b)$ is the scaled and shifted version of $g(t)$. In engineering applications *f* and *g* are finite energy signals. In quantum mechanics, $F_{g_a}(t,\omega)$ is referred to as the canonical coherent state representation of *f*. The inverse Gabor transform is given by (17),

$$f(\tau)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} F_{g_a}(t,\omega)\overline{g_a}(\tau-t)e^{i\omega t}dtd\omega \ , \tag{17}$$

where $\overline{g}_a(t-b)$ stands for complex conjugate of $g_a(t-b)$ and $F_{g_a}(t,\omega)$ is the corresponding Gabor transform.

2.6.1 Discrete Gabor Transform

In terms of sampling points defined by $t=mt_0$ and $\omega=n\omega_0$, where m and n are integers and t_0 and ω_0 are positive quantities, the discrete Gabor functions are defined by $g_{m,n}(t)=\overline{g_a}(t-mt_0)e^{-in\omega_0 t}$ atoms [8]. These functions are called the Weyl-Heisenberg coherent states, which arise from translations and modulations of the Gabor window function. The set of sample points $\{(mt_0,n\omega_0)\}$ is called the Gabor lattice [10]. The discrete Gabor transform is defined in (18) and (19) is referred to as Gabor series of *f*(*t*),

$$F(m,n)=\int_{-\infty}^{+\infty} f(t)\overline{g}_{m,n}(t)dt\,, \tag{18}$$

$$f(t)=\sum_{m,n=-\infty}^{+\infty} F(m,n)g_{m,n}(t). \tag{19}$$

If the functions { $g_{m,n}(t)$ } form an orthogonal basis then *f* can be reconstructed by (21) provided that transform is taken by using (20). In the following expressions { $g^*_{m,n}(t)$ } is the dual frame of { $g_{m,n}(t)$ }. For a complete discussion of dual sequences and their properties refer to [9].

$$F(mt_0,n\omega_0)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty} f(\tau)g_{m,n}(\tau)d\tau\,. \tag{20}$$

$$f(t)=\sum_{m,n=-\infty}^{+\infty} F(mt_0,n\omega_0)g^*_{m,n}(t)=\sum_{m,n=-\infty}^{+\infty} < f, g_{m,n} > g^*_{m,n}(t). \tag{21}$$

2.7 Time-Frequency Atoms

The properties of time-frequency atoms introduced in Sections 2.6 are further studied in this section. A linear time-frequency transform correlates the signal with a family of waveforms that are well concentrated in time and in frequency. These waveforms are called time-frequency atoms. Let us consider a general family of time-frequency atoms $\{\phi_\gamma\}_{\gamma\in\Gamma}$ where γ can be a multi-index parameter from the index collection Γ. In this study it is supposed that $\phi_\gamma \in l^2(R)$ and that $\|\phi_\gamma\| = 1$. This family of time-frequency atoms would be the collection of $g_a(t-b)$ in case of Gabor transform. The corresponding linear time-frequency operator T associates to any $f \in l^2(R)$ the energy concentration by (22),

$$Tf(\gamma) = \int_{-\infty}^{+\infty} f(t)\bar{\phi}_\gamma(t)dt\,, \tag{22}$$

where $Tf(\gamma)$ is a measure of the energy of *f*(*t*) located at time-frequency location specified by $\phi_\gamma(t)$ and $\bar{\phi}_\gamma(t)$ stands for the complex conjugate of $\phi_\gamma(t)$. The Parseval formula (23) shows how the energy is concentrated in a single time-frequency atom. In (23), *F* and Φ_γ are Fourier transforms of *f*(t) and $\phi_\gamma(t)$ respectively.

$$Tf(\gamma) = \int_{-\infty}^{+\infty} f(t)\bar{\phi}_\gamma(t)dt = \frac{1}{2\pi}\int_{-\infty}^{+\infty} F(\omega)\Phi_\gamma(\omega)d\omega\ . \tag{23}$$

If $\phi_\gamma(t) \in l^2(R)$ is nearly zero when *t* is outside a neighborhood, then the projection of *f*(t) on this atom depends only on its values in this neighborhood. Similarly when $\Phi_\gamma(\omega) \in l^2(R)$ is nearly zero outside a center frequency, the projection of *f*(t) on the atom depends only on frequency components of the signal in this neighborhood. This fact results in time-frequency concentration that is needed to capture local phenomena in the signal. Fig.2.9 is a graphical representation of a time-frequency atom concentration in time and frequency.

2.8 Time-Frequency Plane

Here an idealized plane in which time and frequency are measured along the horizontal and vertical axes respectively known as time-frequency plane is introduced. A waveform is represented by a rectangle in this plane with its sides parallel to the time and frequency axes as seen in Fig.2.9 Such a rectangle is called *information cell.* More general information cell shapes are possible. For an introductory discussion on possible different cell shapes see [9]. The time and frequency of a cell can be read from its center of time and frequency or the lower left corner of the cell.

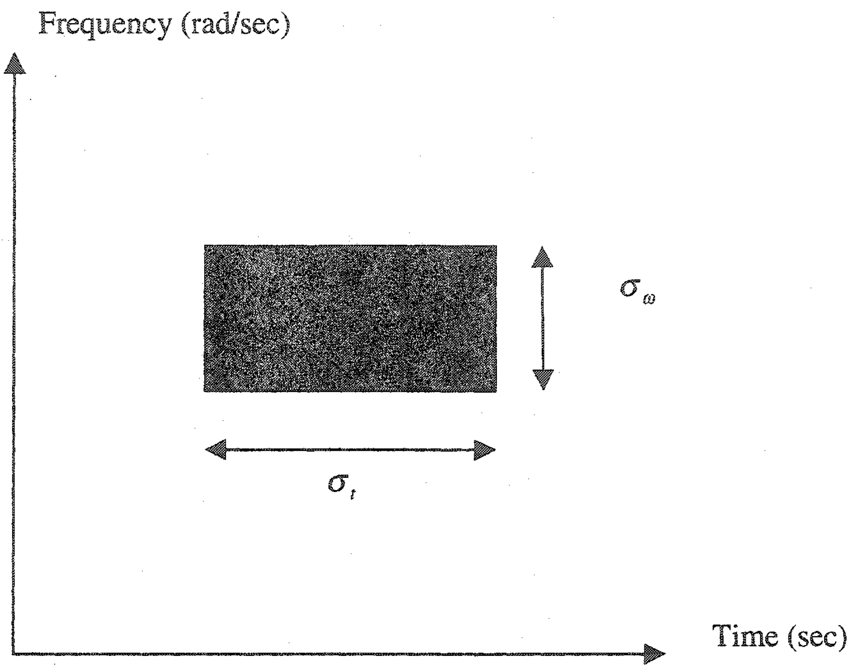


Fig.2.9 An information cell

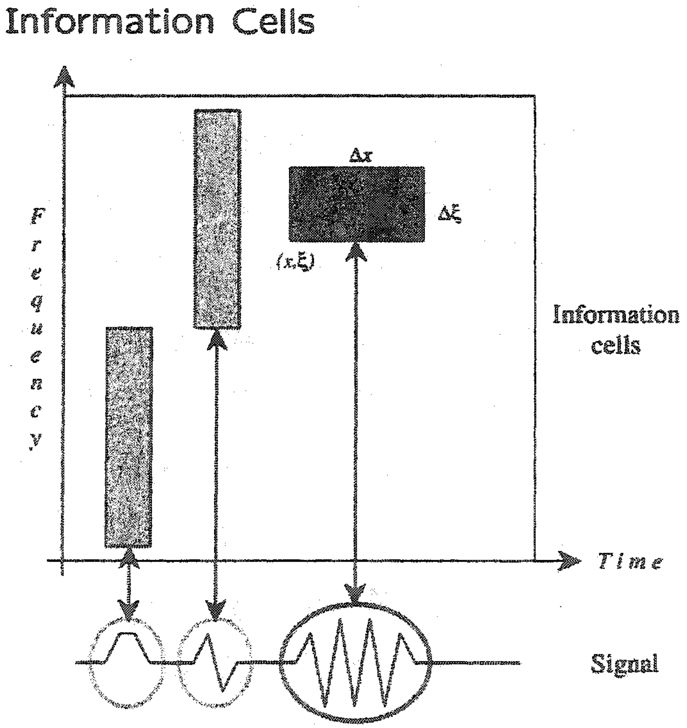


Fig.2.10 Different information cells with variable spread in time and frequency and with different energy concentrations: the darker the cell, the more energy it carries

Since there is an uncertainty in the time and frequency location of the waveform, either choice will work as long as the resolutions are specified. The uncertainties in time and frequency of the cell are represented by the width and height of the cell respectively. The product of the uncertainties, which is the area of the information cell, cannot be smaller than 1/4, the Heisenberg lower bound. Fig.2.10 shows how the spread of time-frequency atoms leads to information cells having variable width and height. The width and height of the cells are representative of signal variance in time and frequency domains around the central time and frequency of the signal respectively. These quantities are measured by equations (10) to (13) for a representative time-frequency atom $\phi(t)$. There are three representative cells shown in Fig.2.10. The two left cells have small time uncertainty but large frequency uncertainty. The fact that the cells are disjoint indicates that the respective waveforms are orthogonal [9]. The darker the information cell is, the more energy it carries. As seen in Fig.2.10, the right information cell carries more of the signal energy. The energy of the signal contained in a single time-frequency information cell is measured by (22) or equivalently from (23).

Time-frequency plane is an excellent tool in describing how the energy of the signal is distributed among different frequency ranges at different time locations. The Heisenberg uncertainty principal indicates that for continuous waveforms the area of the information cell cannot be less than 1/4. Only the Gaussian function that is suitably dilated, modulated and translated has the minimal information cell area. Other time-frequency atoms or waveforms are not too far off if we relax the minimality condition in the Heisenberg uncertainty principal described above [8, 9].

2.9 Time-Frequency Plane Tiling

A family of orthonormal time-frequency atoms with uniformly bounded Heisenberg product can be represented by information cells of equal area. A basis of such atoms results in a cover of time-frequency plane by rectangles. An orthogonal basis results in a disjoint cover, which is usually hard to attain if negligible overlap between adjacent cells is not ignored. Different specific time-frequency dictionaries have their own cover of time-frequency plane with their specific cell shapes. Fig.2.11 (a) and (b) show the Dirac and the Fourier bases tiles in the time-frequency plane. The Dirac basis consists of the usual Cartesian coordinates as basis functions. On the other hand the Fourier basis consists of pure monotone exponentials that describe the frequency content. The Dirac basis has optimal time localization and Fourier has optimal frequency localization. Between the two extremes there are many other transforms with different width and height for information cells, which often prove to be more useful for a particular application. Windowed Fourier or trigonometric transforms lead to a cover of time-frequency plane similar to the ones depicted in Fig.2.12. The ratio of frequency uncertainty to that of time uncertainty is the aspect ratio of the cell. The wavelet basis is an octave-band decomposition of the time-frequency plane as shown in Fig.2.13 (a). A wavelet packet basis gives more freedom in choosing the individual information cells as seen in Fig.2.13 (b). The tiling given in Fig.2.13 (b) is particularly suitable for a signal that contains two almost pure tones near 1/3 and 2/3 of the Nyquist frequency [18] and one intends to identify these for the classification purpose for instance. The various wavelet packet set of basis functions cover the frequency axis first whereas in case of local trigonometric basis this coverage happens for time axis first. Fig.2.14 (a) shows a time-frequency tiling as given by a local trigonometric basis. This category of redundant dictionaries is not considered in this study but knowing the functionality of local trigonometric transform in tiling the time-frequency plane is still helpful. For a complete discussion on construction and properties of local trigonometric transforms the reader is referred to [4]. The tiling would be ideal for analyzing a signal that its amplitude is to be exactly known at certain times as shown by narrow and very tall information cells. Again these values might prove to be the key aspect in a certain classification task. The more generalized tiling of Fig.2.14 (b) is attainable if one combines the wavelet packet and local trigonometric bases and the selection of appropriate basis functions is done from this huge collection of basis functions. The simultaneous selection of time-frequency atoms from different dictionaries is not included in the traditional implementations of local discriminant basis and is remained as a research problem.

2.10 Wavelet Transform

The mathematical expression for the STFT results in time-frequency plane as shown in Fig.2.12. Its apparent disadvantage is the fixed time resolution and hence fixed frequency resolution. According to Heisenberg uncertainty principal, once a desired time resolution is chosen, the frequency resolution is dictated by (24),

$$\sigma_\omega^2 = \frac{K}{\sigma_t^2}\ , \qquad K \geq \frac{1}{4}\ , \tag{24}$$

where σ_ω^2 and σ_t^2 are the STFT basis time-frequency atom spread in time and frequency as obtained by equations (10) through (13). The wavelet transform is a way of overcoming this deficiency. Faced with fixed resolution characteristics of STFT there is a need for transforms with different time and frequency resolution patterns. We approach wavelet transform from time-frequency viewpoint, however, the origins of wavelet transform go back to different area in applied science and several approaches toward definition of wavelets are possible. For more theoretical discussions the interested reader in referred to [8], [9], [21]. The classical continuous wavelet transform is given by (25),

$$F_g(t,s) = \int_{-\infty}^{+\infty} f(\tau)\overline{g}_{t,s}(\tau)d\tau = \int_{-\infty}^{+\infty} f(\tau)\frac{1}{\sqrt{s}}\overline{g}(\frac{\tau - t}{s})d\tau\ . \tag{25}$$

In this expression $g(t,s)$ is known as the mother wavelet function at position t and scale s. The scale is a concept interpreted best as the inverse of frequency.

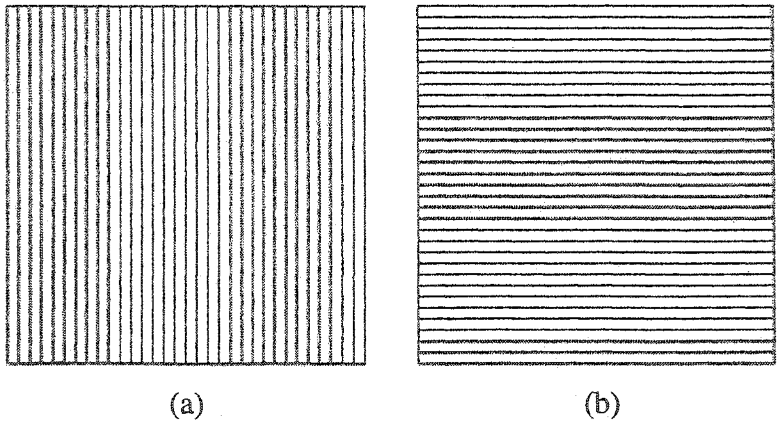


Fig.2.11 Tiling patterns in time-frequency plane with horizontal axis representing time or space variable and vertical axis representing frequency for (a) the Dirac basis (no frequency localization) (b) the Fourier basis (no time localization)

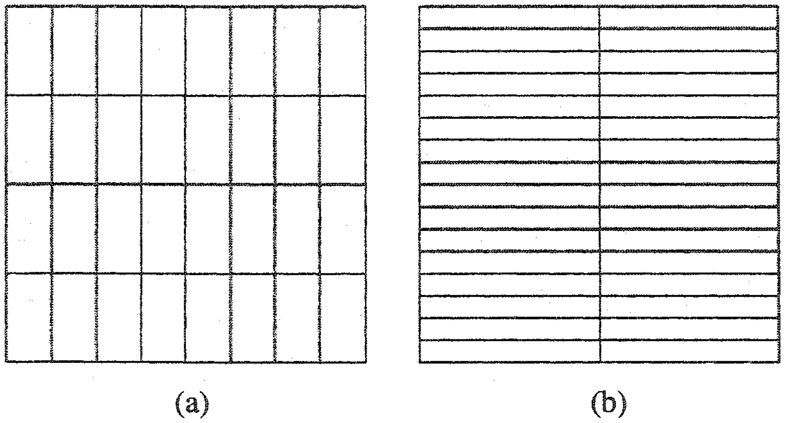


Fig.2.12 Two representative tilings of the STFT with (a) narrow (b) wide window

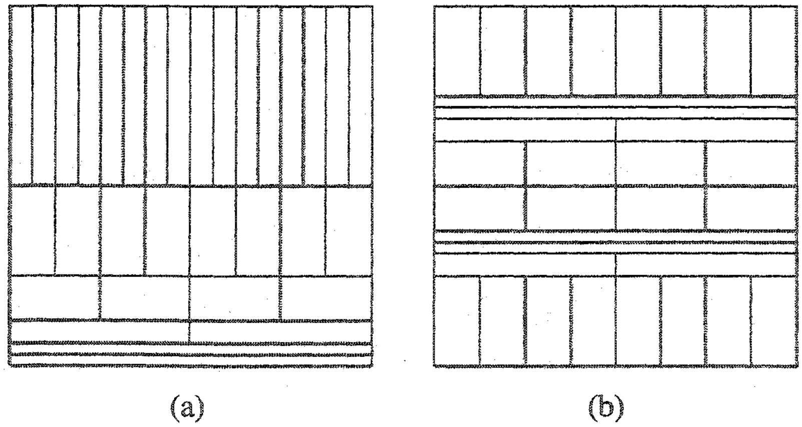


Fig.2.13 (a) A tiling pattern representative of a wavelet transform (b) a tiling pattern representative of a wavelet packet basis

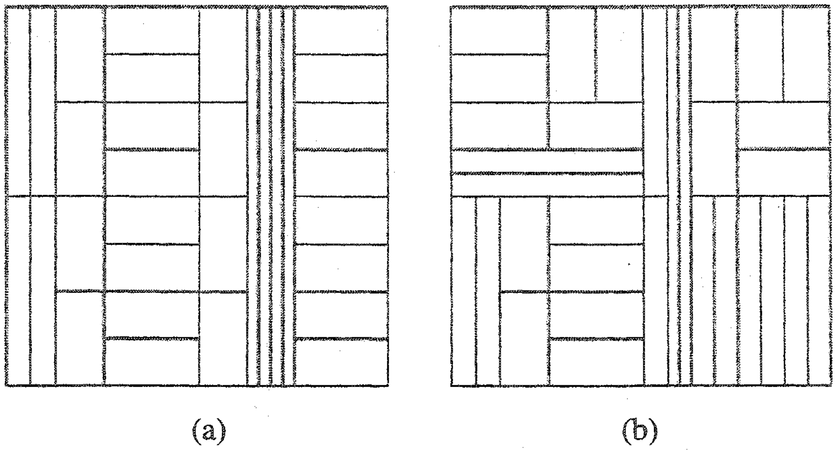


Fig.2.14 (a) A tiling pattern representative of a local trigonometric basis (b) a more general tiling pattern desirable in non-stationary signal analysis obtained by simultaneous basis selection from wavelet packet and local trigonometric bases

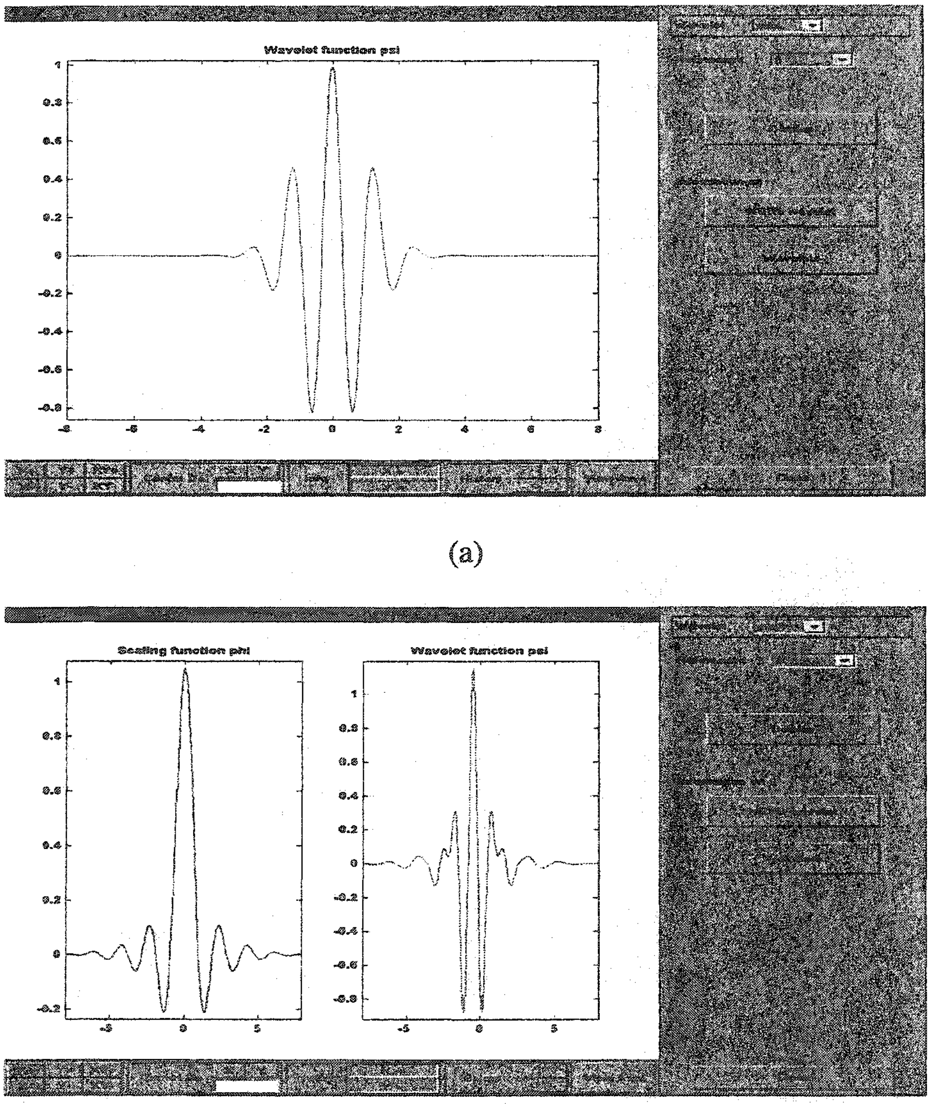
As seen in (25) each *g*(*t*,*s*) is the time shifted and scaled version of *g*(*t*) (the mother wavelet). Note the similarity between $\overline{g}_{t,s}(\tau) = \frac{1}{\sqrt{s}}\overline{g}(\frac{\tau - t}{s})$ and Gabor time-frequency atom $g_a(t-\tau)$. As a matter of fact the wavelet transform is a special case of Gabor transform with a pre-defined pattern for change of time and frequency resolution. As long as *g*(*t*) satisfies a number of smoothness criteria [21], it can be used as a mother wavelet function in wavelet transform. Fig.2.15 shows different mother wavelets used in the continuous wavelet transform. The inverse wavelet transform is given in (26) and (27). It is understood that equality holds in $l^2(R)$ sense, hence, making wavelet transform a one-to-one mapping.

$$f(t) = \frac{1}{C_g}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} F_g(t,s)g(\frac{\tau - t}{s})s^{-2}dsdt\,, \tag{26}$$

$$C_g = 2\pi\int_{-\infty}^{+\infty}\frac{|g(\omega)|^2}{|\omega|}d\omega < \infty\,, \tag{27}$$

where $F_g(t,s)$ is the wavelet transform as obtained by (25). $C_g < \infty$ is a quantity that if finite, guarantees the convergence of (25) and (26). The usefulness of wavelet transform lies in the fact that the high frequency content of many real world phenomena has a transient short duration. This is true in many real world signals in which high frequencies are transient events and for the most part, the lower frequencies govern the steady state and overall behavior of the system.

Therefore if a time-frequency transform such as the wavelet transform can locate high frequencies with higher time resolution and low frequencies with relatively lower but still acceptable resolution, it is a good choice for the analysis of a large class of signals and systems. A careful examination of (8) helps us to understand that the STFT is a function of time and frequency. Now if time and frequency are related in a manner as dictated by (25) one is able to achieve variable time and frequency resolution. As explained above, it is ideal for many cases if the time resolution and frequency resolution are related by an inverse equality. This is exactly what is happening in the wavelet transform variance formula as given in (28),



(a)

(b)

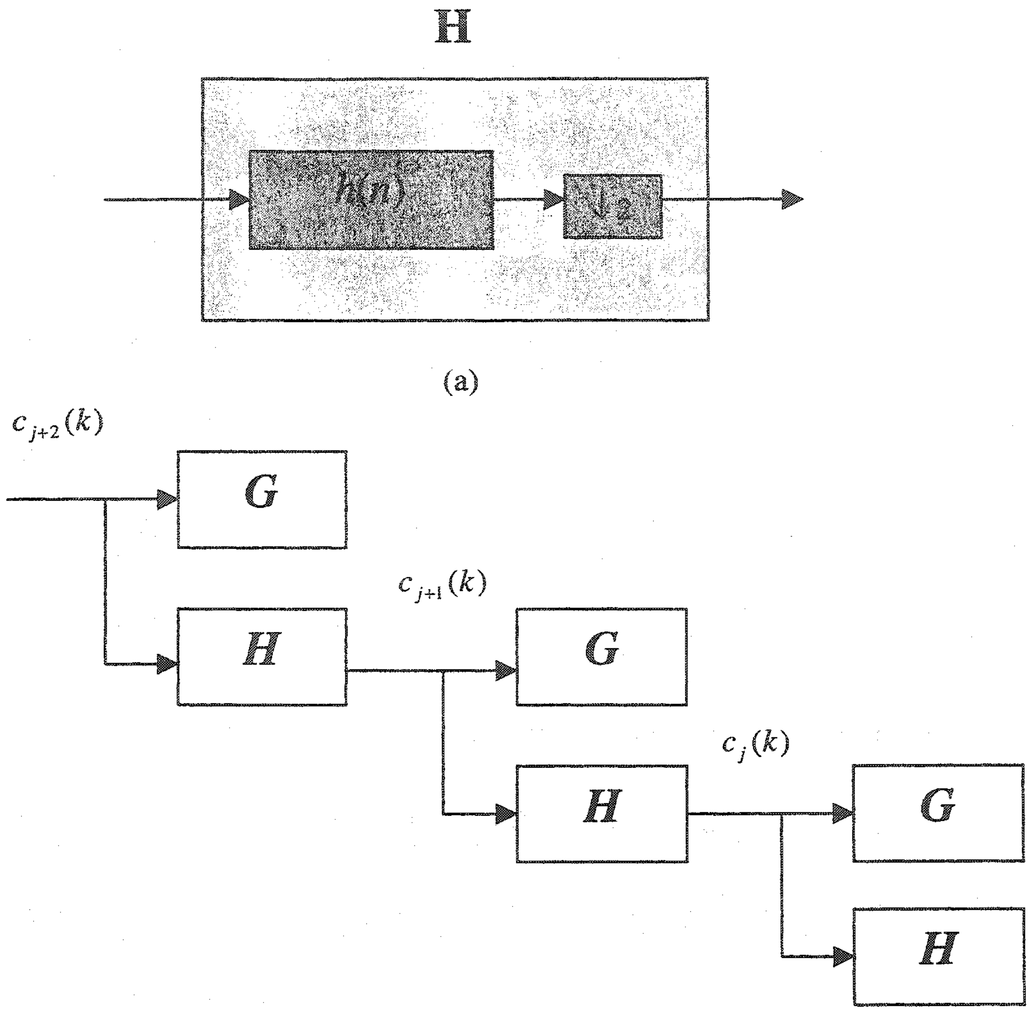
Fig.2.15 (a) Morlet mother wavelet (see [21]) (b) Meyer mother wavelet and scaling function (see [21])

$$\sigma_t = s\sigma_{t_0}, \sigma_\omega = \frac{\sigma_{\omega_0}}{s}, <t>=<t>_0, <\omega>=\frac{<\omega>_0}{s} . \qquad (28)$$

Here s is a parameter called scale and it is simply defined as the inverse of frequency. "< >" shows the average value or expectation and σ's show the spread. Zero indices indicate the corresponding quantities for $g(t)$ (the mother wavelet). One can easily see the relationship between $\sigma_\omega = \frac{\sigma_{\omega_0}}{s}$ and $\sigma_t = s\sigma_{t_0}$ and their fixed product in wavelet transform. Fig.2.13 (a) shows time-frequency plane as the result of wavelet transform. High frequencies are located accurately in time and low frequencies are located accurately in frequency. This helps in a lot of applications such as noise removal. In Fig.2.16 (a) one is able to locate high frequency noise exactly in time and filter it out and keep and process the original signal as it is accurately identified by its low frequency components in frequency domain. A more detailed description of how wavelet structures are used in de-noising follows in Chapter three.

2.10.1 Quadrature Filters

The term quadrature filter is used to denote an operator, which convolves and decimates. Decimation means down sampling. Fig.2.16 shows this operation pictorially. The filter operation is defined on sequences in the digital signal processing. Two main categories of quadrature filters are finite impulse response (FIR) and infinite impulse response (IIR) filters. An individual quadrature filter is not generally invertible; it loses information during the decimation step. However, it is possible to find two complementary filters with each preserving the information lost by the other; then the pair can be combined into an invertible operator. Each of these complementary operators has an adjoint operator: when filters are used in pairs to decompose digital input into pieces, it is the pair of adjoint operators, which puts these pieces back together. The operation is reversible and restores the original signal if the so-called exact reconstruction filters are used. The pieces will be orthogonal if orthogonal filters are used [4]. One way to guarantee exact reconstruction is to have "mirror symmetry" of the Fourier transform of each filter about $\xi = \frac{f}{f_s} = \frac{1}{2}$; which is half the sampling frequency f_s [18]; this leads to the so-called quadrature mirror filters (QMF). Unfortunately there is no orthogonal exact reconstruction FIR QMFs [4]. A different symmetry assumption from the one given above results in exact reconstruction FIR filters and the resulting filters are called conjugate quadrature filters or CQFs [4]. By relaxing the orthogonality condition, a large family of biorthogonal exact reconstruction filters is available [9], [18]. With an orthogonal pair of QFs like H and G, the orthogonal discrete wavelet transform, can be realized. If H and G are biorthogonal with duals not equal to them then, the biorthogonal discrete wavelet transform can be constructed. To each mother wavelet there is associated a pair of quadratic mirror filters H and G [9]. Fig.2.17 shows some examples.



(a)

(b)

Fig.2.16 (a) A quadrature filter consisting of a digital filter and a decimation component in series (b) wavelet transform decomposition using quadrature filters in the digital domain

2.10.2 Discrete Wavelet Transform

As part of digitally processing the signals, one needs to know how to obtain the transforms in the digital domain. In the digital domain, a pair of quadratic filters is used to obtain the wavelet transform. As discussed in Section 2.10.1, to each wavelet function there is assigned a pair of quadrature mirror filters that helps to compute the discrete wavelet transform. The expression for obtaining discrete wavelet transform digitally is given in (29) and (30) where low-pass filter $H=h(n)$ and high-pass filter $G=g(n)$ are called quadrature mirror filters discussed in Section 2.10.1. Fig.2.16 (b) shows the process to obtain the wavelet transform by repeatedly filtering the signal into its high pass and low pass portions by H and G filters. In (29) and (30) $c_j(k)$'s are the coefficients obtained at low pass portion and $d_j(k)$'s are the ones obtained at high pass,

$$c_j(k) = \sum_m h(m-2k)c_{j+1}(m) \ , \tag{29}$$

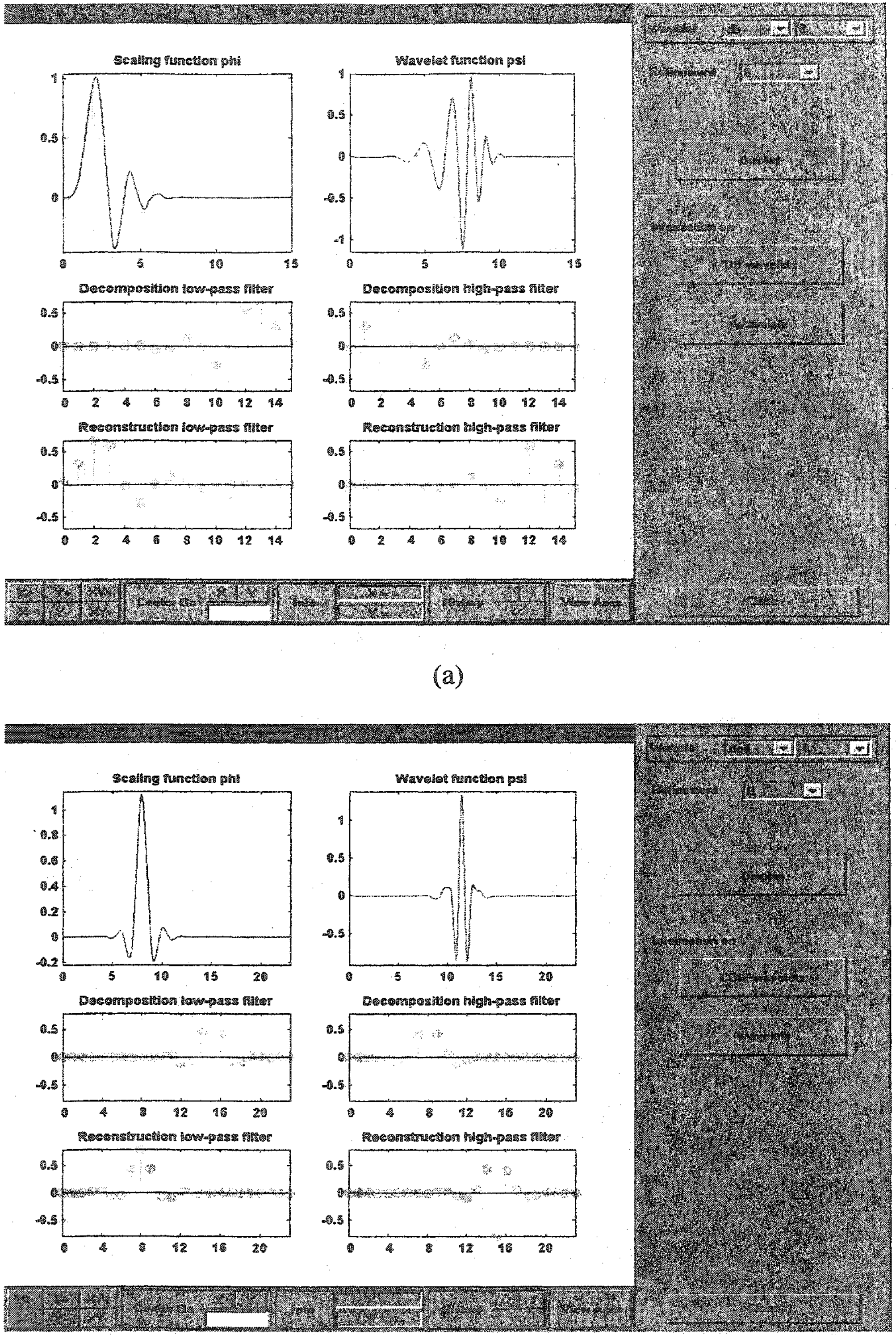
$$d_j(k) = \sum_m g(m-2k)c_{j+1}(m) \ . \tag{30}$$

This process is better explained by introducing the wavelet basis functions. Fig.2.18 is the wavelet basis that results in a time-frequency plane such as the one in Fig.2.13 (a). Fig.2.16 describes the digital process to decompose the signal to these basis functions. This is equivalent to determining the coefficients in (25) digitally. In Fig.2.16 signal components projected to the wavelet basis are obtained digitally as the output of the high pass and low pass filtering at each step.

2.11 Wavelet Packets

The wavelet transform works well for a large category of real world signals that have transient high frequency and longer low frequency content. Although this is the case in many applications, there are cases that more flexibility in terms of frequency and time resolution is desired. Examples are speech and music classification. Fig.2.19 shows a signal with a permanent high frequency portion and a transient low frequency portion. In such situations variable time and frequency resolutions are desired. Furthermore, in order to analyze non-stationary signals variable time and frequency resolution is needed. In this section the underlying concepts of wavelet packet are explained, which are going to be one of the main tools in local discriminant basis algorithm (LDB).

Wavelet packets are particular linear combinations or superposition of wavelets as explained in this section. They produce a collection of basis functions that retain many of the orthogonality, smoothness, and localization properties of the generating wavelet [4]. The coefficients in wavelet packet structures are computed by recursive algorithms that make the related computations of lower complexity. From a basis viewpoint, a discrete wavelet packet analysis includes projecting the signal into a collection of possible wavelet packet bases whereas the discrete wavelet transform is the projection into one fixed set of basis functions obtained by scaling and shifting of the mother wavelet (Fig.2.18). Discrete wavelet packet analysis will reveal no information about the specific signal unless one specifies a particular set of basis function from the wavelet packet redundant basis set. This way the signal can be perfectly reconstructed from its coefficients in the selected basis. Therefore at synthesis level, one should specify the wavelet packet coefficients and the chosen basis functions. The redundant nature of discrete wavelet transform results in having a choice for improving a specified cost function that helps to compress or classify the signal (Chapters three and four).



(a)

(b)

Fig.2.17 (a) Daubechies 8 mother wavelet and its associated QMF (b) Coiflet 4 mother wavelet and its associated QMF obtained by Matlab® wavelet toolbox

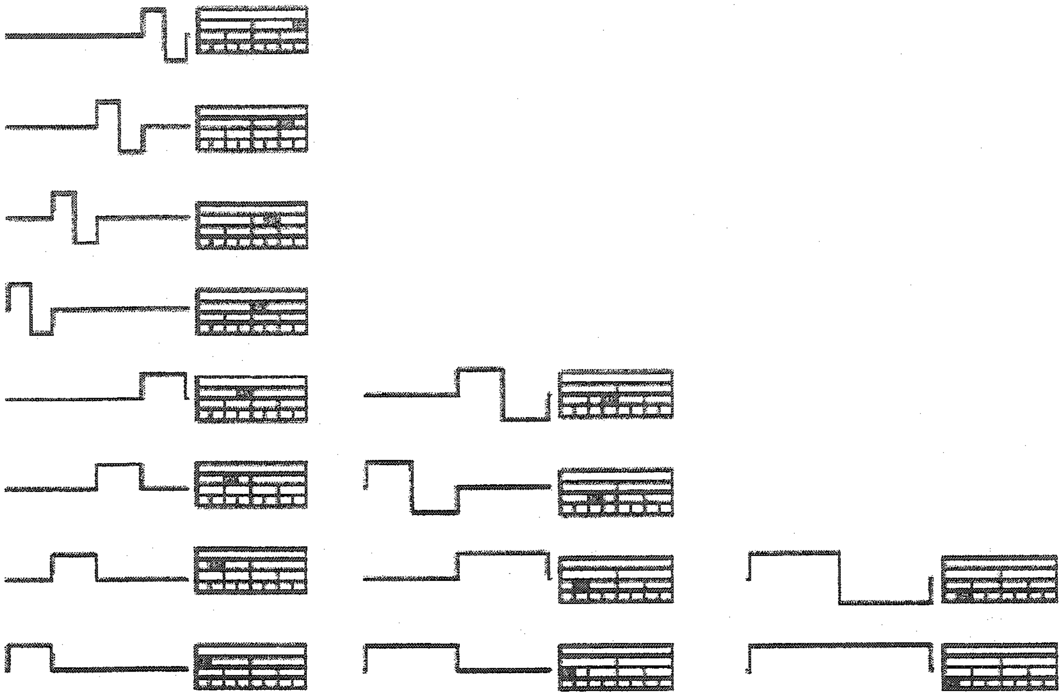


Fig.2.18 All Haar-Walsh basis functions that result in Haar-Walsh wavelet transform and their corresponding position in time-frequency plain given at adjacent right

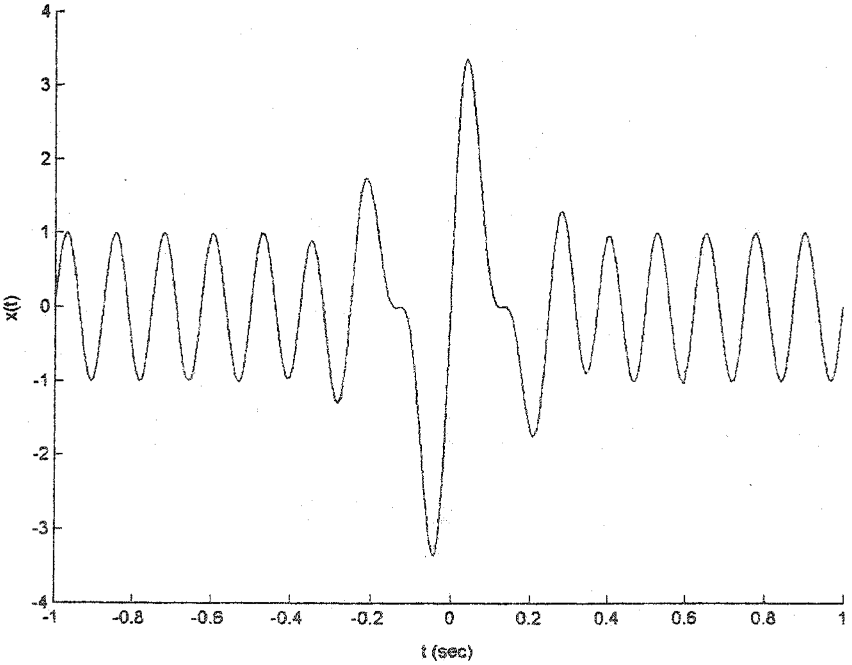


Fig.2.19 A synthetic signal with permanent high frequency and transient low frequency portion that wavelet transform is not suitable for its analysis

2.11.1 Definitions and Construction of Wavelet Packets

Let H and G be a conjugate pair of quadrature filters (QF) from an orthogonal or biorthogonal set. There are two other QF's H' and G' possibly equal to H and G, for which the maps H^*H' and G^*G' are projections on $l^2(R)$ and $H^*H'+G^*G'=I$, where I is the identical operator [4]. Fixed-scale wavelet packets on R are constructed using the QF's H and G. The following sequence of functions is defined recursively,

$$\psi_0 = H\psi_0 \ ; \ \int_{-\infty}^{+\infty}\psi_0(t)dt = 1, \tag{31}$$

$$\psi_{2n} = H\psi_n \ ;\psi_{2n} = \sqrt{2}\sum_{-\infty}^{+\infty} h(j)\psi_n(2t-j), \tag{32}$$

$$\psi_{2n+1} = G\psi_n \ ;\psi_{2n+1} = \sqrt{2}\sum_{-\infty}^{+\infty} g(j)\psi_n(2t-j). \tag{33}$$

The ψ_1 function is the mother wavelet associated to the filters H and G. The collection of the functions produced by the above procedure is called fixed-scale wavelet packet dictionary associated to H and G. The recursive structure of (31), (32) and (33) provides a natural arrangement in the form of a binary tree (Fig.2.20). The dual wavelet packets can also be defined using the dual QF's H' and G'. Let $a(j)=\sqrt{2}\psi_n(2t+j)$ be a sequence in j for (t,n) fixed. The equations (32) and (33) can be rewritten as in (34) and (35),

$$\psi_{2n}(t+i) = \sqrt{2}\sum_{-\infty}^{+\infty} h(2i-j)\psi_n(2t+j) = Ha(i) \ , \tag{34}$$

$$\psi_{2n+1}(t+i) = \sqrt{2}\sum_{-\infty}^{+\infty} g(2i-j)\psi_n(2t+j) = Ga(i) \ . \tag{35}$$

Multi-scale wavelet packets can also be constructed on R. All the functions in the preceding section have a fixed scale. Alternatively a multi-scale decomposition of $l^2(R)$ is possible. The translated, dilated and normalized function in (36) is a wavelet packet of scale index s, frequency index f and position index p,

$$\psi_{sfp} = 2^{-s/2}\psi_f(2^{-s}t-p) \ . \tag{36}$$

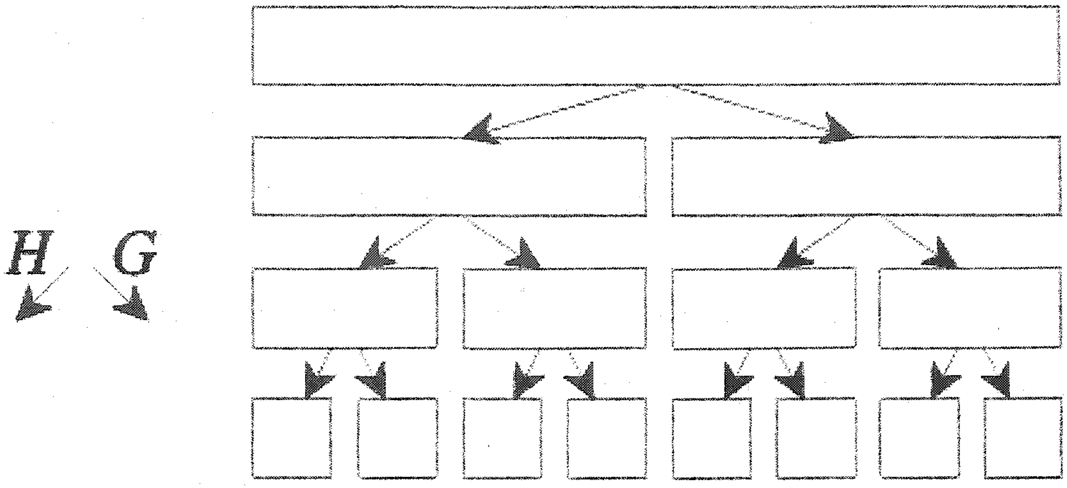


Fig.2.20 The recursive filtering on high pass and low pass portions of the signal resulting in wavelet packet dictionaries

The wavelet packets $\{\psi_{sfp}\}$ include several basis function sets for $l^2(R)$. If H and G are orthogonal QF's, then they are called orthogonal wavelet packets. The scale index gives a relative estimate of the extent in time or position domain. For the scaled wavelet packet of (36), the position uncertainty $\sigma_t(\psi_{sfp})$ of ψ_{sfp} is 2^s times the position uncertainty $\sigma_t(\psi_f)$ of ψ_f. When H and G are FIR filters supported in [0,R], then ψ_f will be supported in the interval [0,R] and ψ_{sfp} will be supported in the interval $[2^s p, 2^s(p+R)]$. For a complete and detailed description of wavelet packet construction, the interested reader is referred to [3], [4]. The following propositions reveal the tree structure of wavelet packets.

Proposition 2.2. If I is a disjoint dyadic cover [4] of R^+, then the well-behaved wavelet packets $\{\psi_{sfp} : I_{sf} \in I, p \in Z\}$ form a basis for $l^2(R)$. Furthermore if H and G are orthogonal QFs and I is a disjoint dyadic cover of R, then $\{\psi_{sfp} : I_{sf} \in I, p \in Z\}$ is an orthonormal basis for $l^2(R)$.

Remark. Note that any cover of R^+ gives rise to a basis, which makes this category of basis functions redundant.

Proposition 2.3. If I is a disjoint dyadic cover [4] of [0,1[, then the well-behaved [4] wavelet packets $\{\psi_{sfp} : I_{sf} \in I, p \in Z\}$ form a basis for $l^2(R)$. Furthermore if H and G are orthogonal QFs and I is a disjoint dyadic cover of R, then $\{\psi_{sfp} : I_{sf} \in I, p \in Z\}$ is an orthonormal basis for $l^2(R)$.

2.11.2 Numerical Calculation of Wavelet Packet Coefficients

If $\{\lambda_{sf}(p) : p \in Z\}$ are the projections of $x(t)$ in $l^2(R)$ with the "backward" basis functions as defined in (38) then the coefficient sequence $\{\lambda_{sf}(p) : p \in Z\}$ is obtained by (37) and satisfies the recursive relations (39) and (40),

$$\lambda_{sf}(p) =< x(t), \psi^{<}_{sfp} >= \int_{-\infty}^{+\infty} \overline{x}(t) 2^{-s/2} \psi_f(p - 2^{-s}t)dt, \tag{37}$$

$$\psi^{<}_{sfp} = 2^{-s/2} \psi_f(p - 2^{-s}t), \tag{38}$$

$$\lambda_{s+1,2f}(p) = H\lambda_{sf}(p) \ , \tag{39}$$

$$\lambda_{s+1,2f+1}(p) = G\lambda_{sf}(p) \ . \tag{40}$$

One can form the multi-scale wavelet packet functions into a binary tree whose nodes are the subspaces generated by $\{\lambda_{sf}(p) : p \in Z\}$. Each root is the sum of its two immediate descendants or children nodes. This is an orthogonal direct sum in the case that QFs are orthogonal. In order to compute wavelet packet coefficients first its coefficient sequence in the root subspace is found, and then the branches of the wavelet packet tree are followed in order to find the coefficients in the descendant subspaces. Following this reasoning one can compute λ by sequential filtering as given in (41) and (42). In the following expressions $\lambda(p) = \lambda_{00}(p)$ gives individual signal coordinates as the finest resolution,

$$\lambda_{sf}(p) = F_0 \ldots F_{s-1}\lambda(p), \tag{41}$$

$$\lambda = \lambda_{00} \text{ and } F_i = H \text{ if } f_i = 0 \text{ and } F_i = G \text{ otherwise.} \tag{42}$$

The synthesis counterpart formula for wavelet packets can be obtained by applying H' and G', the dual of H and G, to the sequence of $\{\lambda_{sf}(p) : p \in Z\}$. The coefficient sequence as given by (41) and (42) will enable us to reconstruct the signal by expression (43),

$$x = \sum \overline{\lambda}_{sf}(p) \psi'^{<}_{sfp} \ , \tag{43}$$

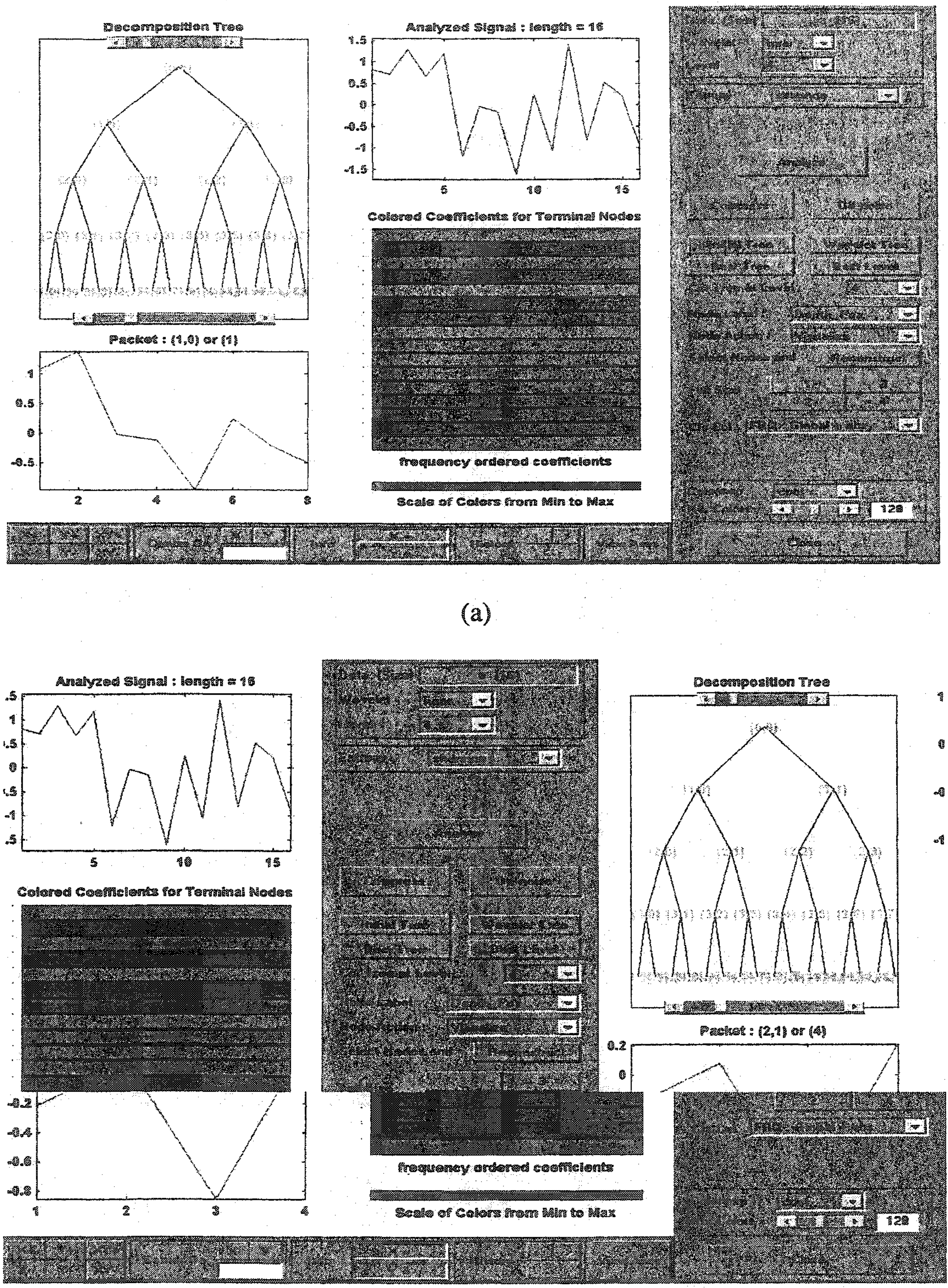
where $\psi'^{<}_{sfp}$ is the dual of $\psi^{<}_{sfp}$ as defined in (38). $\overline{\lambda}_{sf}(p)$ is the complex conjugate of coefficient $\lambda_{sf}(p)$ as obtained by (41) and (42). In practice one computes the sequence $\{\lambda_{sf}(p) : p \in Z\}$. Fig.2.21 is a pictorial explanation of what happens in a relatively small case. This is an example of a sixteen point signal analyzed up to three levels. Each row of coefficients is computed from the row above it by one application of either H or G.

Let's take a graphical intuitive approach toward definition of wavelet packets to see what happens if one continues recursive filtering on high frequency part as well as low frequency part in the discrete wavelet transform. This is exactly what is taking place in Fig.2.20 and Fig.2.21. As a result one gets new components or projections for the given signal. Fourier transform and wavelet transform are complete transforms in the sense that by $l^2(R)$-measure approximation, a signal can be perfectly constructed by its Fourier or wavelet coefficients as formulated in (25) and (26) for the wavelet transform. The transformation that is taking place in Fig.2.21 is clearly giving us more components (or coefficients) as expected. So basically this transform and the basis originating from it are redundant. This is the basic characteristic of wavelet packets that are exploited in our adaptive approach toward choosing a suitable basis. The wavelet packet transform has another very useful characteristic in terms of how its components are structured as a binary tree as given by *Proposition 2.2* and *Proposition 2.3.*

From the basis functions viewpoint wavelet packet transform is equivalent to decomposing the signal into the basis functions given in Fig.2.22. Again its seen that basis functions in this case contain the wavelet transform basis functions plus additional basis functions that are created as a result of recursive filtering on high pass portion of the signal as well as the low pass portion. The goal is to search among this redundant set of basis function for the basis that is as close to the ideal basis as possible for either compression or pattern recognition (classification). The usefulness of wavelet packet transform lies in the manner it decomposes time-frequency plane. However based on the basis selection strategy that one is going to adopt, different tilings of time-frequency plane are possible. Fig.2.14 shows possible time-frequency tiling as a result of different basis function selections from redundant basis function set in Fig.2.22. The freedom in assigning desirable time-frequency resolution to approximately any range of frequencies (in expense of loosing resolution in time) is the major characteristic that makes wavelet packets suitable for non-stationary signal analysis. The corresponding decomposition and reconstruction formulas can be obtained when the appropriate basis functions are selected.

2.11.3 Orthonormal Bases of Wavelet Packets

The redundant nature of wavelet packets and local trigonometric basis means the existence of several sets of basis functions for $l^2(R)$ inside the tree decomposition of these dictionaries. Starting with a sequence of $N = 2^L$ nonzero coefficients, one can decimate by 2 at least L times, which will provide more than 2^N bases. If the QFs are orthogonal, these will be orthogonal basis. For a detailed proof and discussion for counting the number of available sets of basis functions the reader is referred to [4].


Decomposition Tree
Analyzed Signal : length = 16
Packet : (1,0) or (1)
Colored Coefficients for Terminal Nodes
frequency ordered coefficients
Scale of Colors from Min to Max
128
Packet : (2,1) or (4)

(a)

(b)

Fig.2.21 (a) (b) Examples of decompositions into Coiflet 4 wavelet packet at different nodes of binary tree computed by Matlab® wavelet toolbox

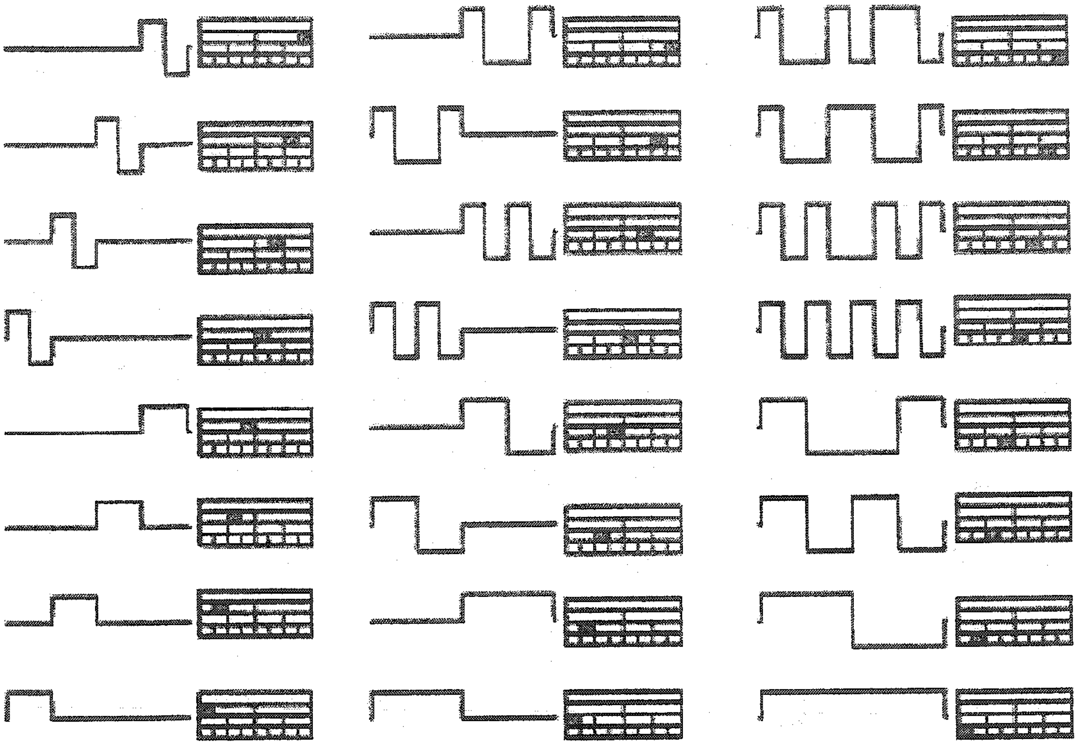


Fig.2.22 All Haar-Walsh basis functions used in evaluating wavelet packet decomposition and their corresponding time-frequency position

2.12 Complete and Over-complete Bases

Discrete wavelet transform as obtained by quadrature filters builds up a complete basis for $l^2(R)$ that is the collection of signals with finite energy. This complete set of basis functions is shown in Fig.2.18 for the Haar-Walsh continuous wavelet transform [9]. Theoretically this list consists of infinitely many but still countable number of basis functions. Nevertheless, the depth of refinement in the frequency domain is usually upper-bounded as one reaches a certain level of resolution because finer resolution reveals little information. As a matter of fact, if the input signal has N coordinates, the depth of the table is upper bounded by $L = \log_2 N$ in discrete wavelet transform (DWT). At this level of refinement one gets the individual signal samples where no more refinement is important in the digital domain. In contrast to continuous wavelet transform where the threshold on the level of resolution can vary, the level of refinement in digital domain is finite and equals $\log_2 N$. The collection of basis functions obtained by quadrature filters constitutes a complete basis for describing $l^2(R)$ signals (Fig.2.18). This means that the basis has the capability of approximating any $l^2(R)$ signal. The approximation in this context means that the energy of the difference between original signal and its approximation is zero. This can easily be induced from the definition of norm in $l^2(R)$ as a Hilbert space with the familiar functional inner product (Section 2.1). In the process of obtaining the wavelet packet transform, the high-pass and low-pass filtering is recursively done on the high pass portion as well as low pass portion of the signal. Since the wavelet basis, which is obtained as a result of high-pass and low-pass filtering on the low pass part is already complete, the wavelet packet set of basis function, which is often referred to as a *dictionary* contains more than enough basis functions to express $l^2(R)$ functions. This is why the wavelet packet dictionary is considered as over-complete or redundant set of basis functions. As a result one has the opportunity to select a suitable basis among different complete sets of basis functions available within a *wavelet packet dictionary*. The redundancy and the tree structure of wavelet packets and local discrimiant bases dictionaries enable a fast search for the best basis of Coifman and Wickerhauser [3] or the local discriminant basis of Coifman and Saito as discussed before [4]. In this thesis, only wavelet packets are used to look for basis functions therein. For a complete discussion on local trigonometric transforms that constitute another important category of redundant dictionaries the reader is referred to [22], [23].

2.13 Tree Structured Dictionaries

A set of basis functions is called a *binary tree* if it can be organized in a fashion similar to wavelet packet dictionaries or local trigonometric dictionaries [4] as depicted in Fig.2.20 and Fig.2.21. This is equivalent to the following definition.

Definition 2.8. A dictionary of orthonormal bases for $l^2(R)$ is a binary tree if it satisfies:

(a) Subsets of basis vectors can be identified with subintervals of $I = [0,1[$ of the form $I_{j,k} = [2^{-j} k, 2^{-j}(k+1)[$, for $j=0,1,\dots,J$, $k=0,1,\dots,2^j -1$, where $J \le n_0$.
(b) Each basis in the dictionary corresponds to a disjoint cover of I by interval $I_{j,k}$.
(c) If $\Omega_{j,k}$ is the subspace identified with $I_{j,k}$, then $\Omega_{j,k} = \Omega_{j+1,2k+1} \oplus \Omega_{j+1,2k}$.

Each subspace $\Omega_{j,k}$ is spanned by 2^{n_0-j} basis vectors $\{\omega_{j,k,l}\}$. In a wavelet packet dictionary (Section 2.11), the parameters k and l roughly indicate frequency bands and the location of the center of wiggles, respectively: the vector $\omega_{j,k,l}$ is roughly centered at 2^j l, has length of support 2^j and oscillates k times. Discrete wavelet packet coefficients at this location are given by $\{\lambda_{jk}(l)\}$. The one-to-one correspondence of the above definition is best seen in the wavelet packets as constructed by the machinery in Fig.2.20 that allows organizing the wavelet packet basis functions (components) in a binary or quadratic tree fashion depending upon the dimensional of input signals whether one dimensional or two dimensional. The binary tree nature of wavelet packet allows efficient and fast algorithmic search of best basis compared to more general approaches such as matching pursuit, which is an adaptive algorithm or Principal Component Analysis (PCA), which is a totally unsupervised algorithm.

2.14 Library of Bases

Each individual redundant set of basis functions that are obtained from wavelet packet decomposition or a local trigonometric basis is called a *dictionary of basis*. For example, the wavelet packet basis functions obtained from the process in Fig.2.20 with Haar-Walsh discrete wavelet [21], as the mother wavelet constitute a dictionary of basis as depicted in Fig.2.22. By changing the mother wavelet in this case several wavelet packet dictionaries can be created.

The collection of all these dictionaries that are all in tree format leads to a huge set of dictionaries, which is called *library of bases*. A general search for best basis starts in this library. Although the library contains many dictionaries due to different possible mother wavelet selections and different local trigonometric basis collections, the search can be confined to only a few dictionaries to find the best basis therein. However, one might still be able to accurately chose a best basis if the physical nature of the signals to be classified is partially known so that a careful initial mother wavelet selection can be done. The choice of the best dictionary to start the search with is equivalent to asking about the best wavelet for signal analysis. This is an interesting question and the general answer to this problem is still to be explored as research.

2.15 Chirplets

There are some signals that cannot be represented efficiently by the wavelet packets, such as *chirp* signals [26]. Analysis of signals with chirp components is very important in radar signal processing and Whistler analysis. Boashash *et al.* studied the absorption and dispersion effects in the earth by analyzing the chirp content of the seismic signals [24], [25]. Chirp functions exist in electroencephalography (EEG) signals, in bird voices, and in bat sonar signals. From these examples, one can see that a parameter that indicates the chirp content of a signal can be very useful as an additional parameter in a multi-parameter transform. The transforms that include the chirp functions have recently been introduced to the signal processing community. Mann and Haykin have used chirp multiplication (frequency shear) and chirp convolution (time shear) together with time-shift, frequency-shift, and scale parameters to define the continuous chirplet transform [26], [27]. Related ideas were presented by Mihovilovic and Bracewell as well [28]. They named the resulting analyzing functions *chirplets*. Baraniuk and Jones studied the two-dimensional (2-D) chirplet transform sub-spaces together with their discretization issues and Wigner-based formulation of the chirplet transform [29]. Recently, however, researchers have started using a generalization of the Fourier transform (FT), known as the fractional Fourier transform (FRFT) [30], that provides a measure for the angular distribution of energy in the time-frequency plane. The FRFT operator rotates the Wigner distribution (WD) of a signal in the time-frequency plane. Rotation is a special combination of the chirp convolution and chirp multiplication and results in an orthogonal transformation of the time-frequency coordinates.

Scaling, rotation, time-shift, and frequency-shift operators are applied to the unit Gaussian successively to obtain the four-parameter time-frequency atom. The set of these atoms constitute a dictionary of chirplet functions. This decomposition provides an efficient representation of chirp-like signals. Since the redundancy is increased, more compact representation of complex signals with a few high-energy components is possible, as compared with the commonly used decompositions with two or three parameters. In addition, a related time-frequency distribution (TFD) is defined for clear visualization of the signal components. Developing an adaptive classification scheme incorporating chirplet dictionaries for classifying signals will be an interesting continuation of the research in this thesis.

2.16 Brushlets

Brushlets were invented for directional texture analysis by Meyer and Coifman [14]. Edges and textures in an image can exist at all possible locations, orientations, and scales. The ability to efficiently analyze and describe textured patterns is thus of fundamental importance for image analysis and image compression. Wavelet packets make it possible to adaptively construct an optimal tiling of the Fourier plane, and they have been used for image compression as in best basis algorithm [4] and for classification as in LDB [6].

However, the tensor product of two real valued wavelet packets is always associated with four symmetric peaks in the frequency plane that originate from the manner in which wavelet packets are constructed using QMF. This product arises in computing wavelet packet transform of images [14].

In order to obtain a better angular resolution than the standard wavelet packets Meyer and Coifman expanded the Fourier plane into windowed Fourier bases [31]. The method results in an expansion of the image into a set of brushlets. A brushlet is a function reasonably well localized with only one peak in frequency. Furthermore, the brushlet is a complex valued function with a phase. The phase of the bi-dimensional brushlet provides valuable information about the orientation of the brushlet. One can adaptively select the size and locations of the brushlets in order to obtain the most concise and precise representation of an image in terms of oriented textures with all possible directions, frequencies, and locations. This new basis can be used for directional image analysis and to efficiently compress richly textured images. Incorporating brushlet basis into LDB computational engine for texture classification tasks will be an interesting extension of the studies performed in this thesis.

Chapter Three

Best Basis Algorithm

The best basis algorithm belongs to a class of entropy-based algorithms for efficient signal representation and compression. Since local discriminant basis algorithm (LDB) is the feature extraction counterpart of best basis algorithm, understanding the underlying principles of the best basis algorithm prepares the necessary foundation for introducing local discrimiannt basis algorithm in Chapter four. Coifman and Wickerhauser developed classical entropy-based algorithms for best basis selection in 1992 [3, 4]. Best basis algorithm is primarily designed to reduce the storage space needed for a signal and also it can be used for de-noising [5]. This is achieved by selecting a so-called best basis such that signal coordinates reach a minimum level of information content when projected onto the new basis [32]. When the information content of the signal coordinates in the new basis is low it means that the distribution of coefficients is such that the energy of the signal is concentrated in a few coordinates. Another natural application of best basis algorithm is in de-noising [5]. This is achieved simply by finding a representation with a few significant terms (less entropy representation) so that one can neglect the coefficients less than a threshold value. If signal to noise ratio is not too low one is able to extract the desired signal from the noisy one by thresholding operation hence getting rid of unimportant coordinates. In this chapter the classical best basis algorithm is reviewed. Its theory and applications are studied in detail.

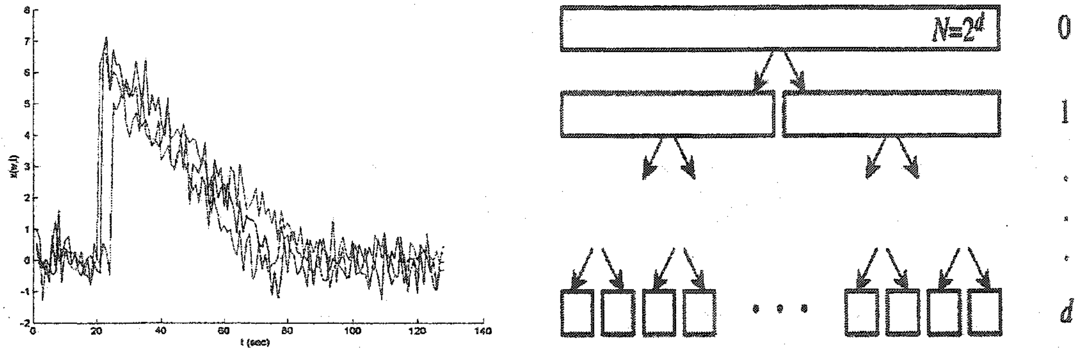
3.1 Theoretical Foundations of Best Basis Algorithm

Adapted waveform analysis as performed by the best basis algorithm exploits the tree structure of wavelet packet and local trigonometric basis dictionaries to allow for the selection of a basis with lowest information content level. It permits efficient compression of a variety of signals such as speech and image. Here we consider an adaptive scheme to select an efficient representation for signals originating from a non-stationary statistical process. The method is adaptive in the sense that it needs to be trained by training signals. In what follows it is assumed that the goal is to find an efficient representation for a class of signals originating from a synthetic non-stationary statistical process.

3.1.1 Dictionary Tree Pruning

The first step is to decompose the normalized training signal space into the time-frequency dictionaries such as wavelet packets (Section 2.11) or local trigonometric basis [22], [23]. Then the energy of each coordinate is computed simply by squaring the coefficient. For a class of training signals the summation of signals energy at a particular location in the dictionary gives a measure of importance of that single basis function in representing the training signals. Fig.3.1 (b) shows typical time-frequency energy map structure obtained from a synthetic statistical process depicted in Fig.3.1 (a). The key idea is to use a cost function to prune the wavelet packet or local trigonometric dictionary in search for an efficient basis. If the cost function is additive the search for best basis will be considerably faster. Both wavelet packets and local trigonometric basis yield to a binary or quadratic tree structure (based on 1D or 2D applications) as depicted in Fig.3.1 for one-dimensional signals. A similar but quadratic tree exists for two-dimensional images as shown in Fig.3.2 (a) and (b). The best basis selection strategy is to look from the most refined partition upward. In case of wavelet packets or local discriminant basis used to analyze digital signals the most refined partition is the signal usual Cartesian coordinates. In case of one dimensional signal space, the binary tree has a maximum depth of $L = \log_2 N$, where N is the original signal dimension. Fig.3.2 (b) is an example of complete two-dimensional wavelet packet structure by incorporating quadrature filters (Section 2.10). As apparently seen in the quadratic structure of these dictionaries, the maximum depth is $L = \log_4 N$, where N is the number of pixels in a gray scale image. The search starts from the most refined portions. The information cost that can be the *Shannon entropy* (see below) in this case is calculated at the children nodes at the bottom of the tree in Fig.3.1 (b) or Fig.3.2 (b). The information costs of each pair of adjacent children nodes are compared to that of their parent. In case of additive information cost this is simply accomplished by comparing the cost function at the parent node with the sum of cost function at children nodes. Either the parent node or the union of children nodes whichever has less information cost is selected and the search is continued to the top branch in the tree. The proposed search algorithm reveals the minimum entropy expansion for the signal. The above explained selection process is pictorially depicted in Fig.3.3.

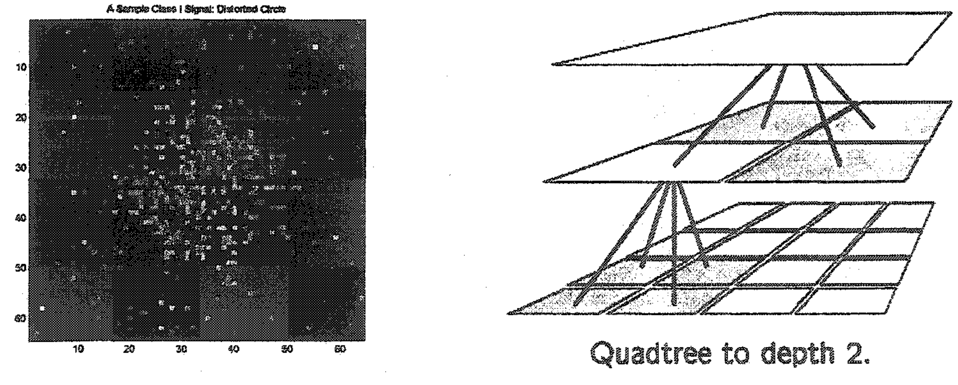
The top figure shows the binary tree structure of one dimensional wavelet packet with information cost assigned to its nodes. Starting from the bottom level the information cost of the children nodes together are compared to their corresponding parent node. A higher summation of information costs for children nodes means that their parent node is a better choice. In that case this summation goes inside the parenthesis to reflect the fact that the parent node is the winner (middle figure). The final selection process in the bottom figure specifies the desired basis, *i.e.* the best basis. The result of the above procedure might look like the one shown in Fig.3.4 or Fig.3.5. It is important to note that the selection may end up choosing the classical wavelet basis functions as has happened in Fig.3.4.



(a)

(b)

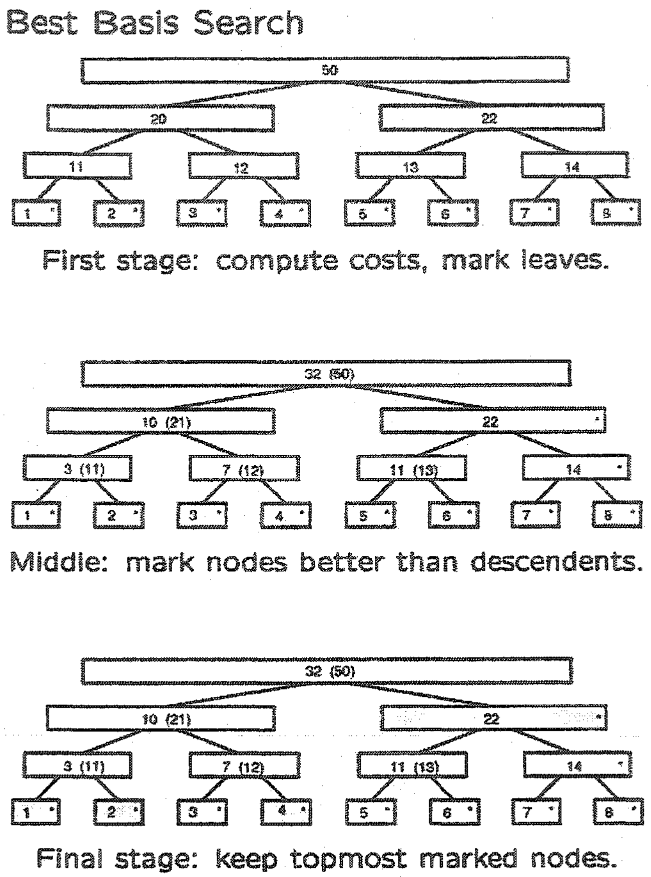
Fig.3.1 (a) Sample functions of a synthetic non-stationary statistical process (as will be described in Section 4.7) (b) the time-frequency energy map structure for one dimensional signal analysis



(a)

(b)

Fig.3.2 (a) A sample 64×64 grayscale image obtained from a synthetic non-stationary statistical process (as will be described in Section 4.8) (b) structure of time-frequency energy map for image processing applications



First stage: compute costs, mark leaves.

Middle: mark nodes better than descendents.

Final stage: keep topmost marked nodes.

Fig.3.3 Step-by-step graphical representation of best basis selection process

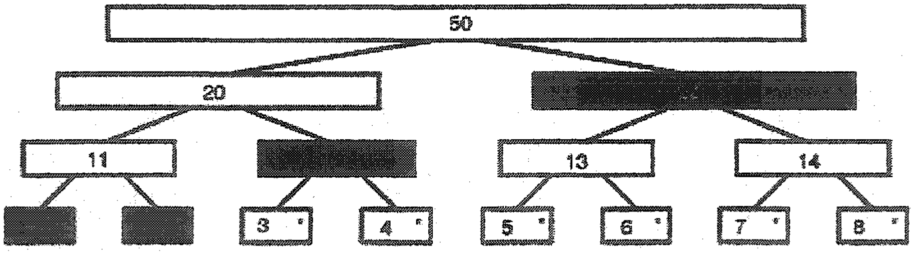


Fig.3.4 A possible best basis (the wavelet transform)

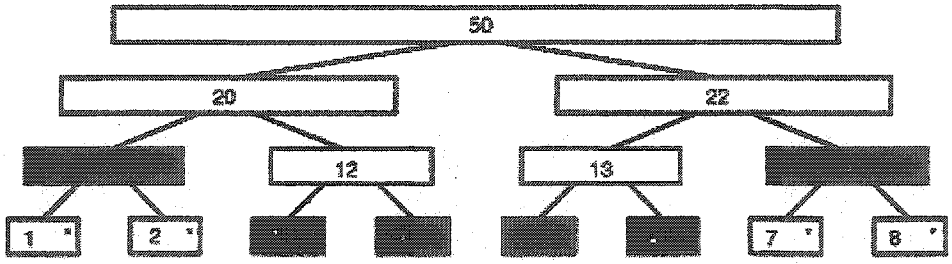


Fig.3.5 Another possible best basis different from the wavelet transform

3.1.2 Cost Function in Tree Pruning Process (Information Cost)

The information cost used in the algorithm for selection of best basis may vary for different applications. A real valued functional mapping each basis function in a wavelet packet or local trigonometric dictionary to its power of discrimination between disjoint classes of signals, *i.e.* $H:\{\psi_{sfp}\}\rightarrow R$, will help to determine the best basis for signal representation. $\{\psi_{sfp}\}$ is the wavelet packet basis as described in Section 2.11. Such functional should describe concentration or the number of coefficients required to accurately describe the signal in new coordinates. The value of the functional should be large when the coefficients are roughly of the same size and small when all but a few coefficients are negligible and thus need to be stored. Besides if the information cost is additive, it splits across Cartesian coordinates and makes the search for the best basis a fast divide and conquer algorithm [4]. Additive measure is formally defined in below.

Definition 3.1. A map H from sequence {xi} to R is called an additive information cost if $H(0)=0$ and $H(\{x_i\})=\sum H(x_i)$.

Definition 3.2. In a signal compression application the best basis relative to H for a vector x in dictionary D of bases is the one for which $H(Dx)$ is minimal. Here Dx stands for the representation of x in D as the new basis.

Definition 3.3. Theoretical dimension of {xi} is defined by (44) where p_n's are normalized signal coordinates given by (45),

$$\text{Theoretical dimension} = e^{-\sum p_n \log p_n}, \tag{44}$$

$$p_n = \frac{x_n^2}{\|x\|^2}, \tag{45}$$

where $\|x\|$ stands for magnitude of x in usual Euclidean measure. Information cost function in the best basis algorithm is the logarithm of theoretical dimension that is the well-known *Shannon entropy.*

Proposition 3.1. If $x_n = 0$ for all but finitely many values of n such as N, then inequality (46) holds,

$$1 \leq \text{Theoretical dimension} \leq N. \tag{46}$$

Proposition 3.2. If $\{x_n\}$ and $\{y_n\}$ are rearranged so that both $\{p_n\}$ and $\{q_n\}$ that are signal coordinates in the dictionary D are monotone decreasing, and if (47) holds for all m, then $H(p_n) \leq H(q_n)$,

$$\sum_{0<n<m} p_n \geq \sum_{0<n<m} q_n. \tag{47}$$

The best-known measure of information particularly suitable for compression and de-noising applications is the information measure developed by Shannon, *i.e.* Shannon entropy. In this case the Shannon entropy has a geometric interpretation as well [4]. The applicability of Shannon entropy in selecting the efficient signal representation is explained in the following example.

Example 3.1. The information content of $\{x_n\}=\{1,2,3,4,5\}$ with that of $\{y_n\}=\{2,2,4,4,5\}$ are compared using (58) and (59). Therefore $\{y_n\}=\{2,2,4,4,5\}$ has more information content according to what follows.

$$p_n = \frac{x_n^2}{\|x\|^2} \text{ and } q_n = \frac{y_n^2}{\|y\|^2}$$

$$H(p_n) = -\sum p_n \log p_n = \frac{1}{55}\log\frac{1}{55} + \frac{4}{55}\log\frac{4}{55} + \frac{9}{55}\log\frac{9}{55} + \frac{16}{55}\log\frac{16}{55} + \frac{25}{55}\log\frac{25}{55} = 1.2773$$

$$H(q_n) = -\sum q_n \log q_n = \frac{4}{65}\log\frac{4}{65} + \frac{4}{65}\log\frac{4}{65} + \frac{16}{65}\log\frac{16}{65} + \frac{16}{65}\log\frac{16}{65} + \frac{25}{65}\log\frac{25}{65} = 1.4008$$

Example 3.2. The theoretical dimensions of the sequences in *Example 3.1* are calculated as follows.

Theoretical dimension $(\{x_n\}) = e^{H(p_n)} = 3.5869$

Theoretical dimension $(\{y_n\}) = e^{H(q_n)} = 4.0584$

As seen in the above examples when the information content of the representation is less, signal energy is focused in fewer coordinates that one can keep and dispose the rest but yet loose negligible information. The cost table of the process in Fig.3.1 is given in Fig.3.3 for each individual set of basis functions. The selected minimum entropy pass shown in Fig.3.3 will result in the most efficient representation of the process. Note that the usual wavelet basis is one of the possible options when looking for the best basis.

3.1.3 Best Basis Algorithm

Here is a formal statement of best basis algorithm [4].

Step 0. Choose a dictionary of orthonormal bases *D*. Compute the depth of the tree, *J* and select an information cost function *H*.
Step 1. Expand the signal into the dictionary *D* and obtain coefficients $\{B_{j,k}x\}$. Here $\{B_{j,k}x\}$ stands for expansion of x in $B_{j,k}$ as a basis function in the dictionary dictionary.

Step 2. Set $A_{J,k} = B_{J,k}$ for $k = 1,2,\ldots,2^J - 1$.

Step 3. Determine the best subspace $A_{j,k}$ for $j = J-1, J-2, \ldots, 0$ and $k = 1,2,\ldots,2^j - 1$ by

$$
\begin{array}{ll}
A_{j,k} = B_{j,k} & ;\text{if } H(B_{j,k}x) \le H(A_{j+1,2k}x \cup A_{j+1,2k+1}x), \\
A_{j,k} = A_{j+1,2k} \oplus A_{j+1,} & ;\text{otherwise}
\end{array}
$$

Remark- By subspaces $A_{j,k}$ or $B_{j,k}$ we mean the subspaces generated by the corresponding basis functions at the particular tree dictionary position.

Proposition 3.3. Algorithm in 3.1.3 yields the best basis relative to H if H is additive (Section 3.1.2). The reader is referred to [4] for a proof of this proposition.

Fig.3.6 is a graphical representation of best basis algorithm mathematically stated above. In this figure the features are to be extracted from a set of training images. Training signal samples are decomposed into quadratic tree wavelet packet structures and the energy of projections are averaged to yield the average importance of each node in the wavelet packet or local trigonometric bases dictionaries. The best basis algorithm as described in Section 3.1.3 yields the best basis using the average decomposition wavelet packet quadratic tree structure.

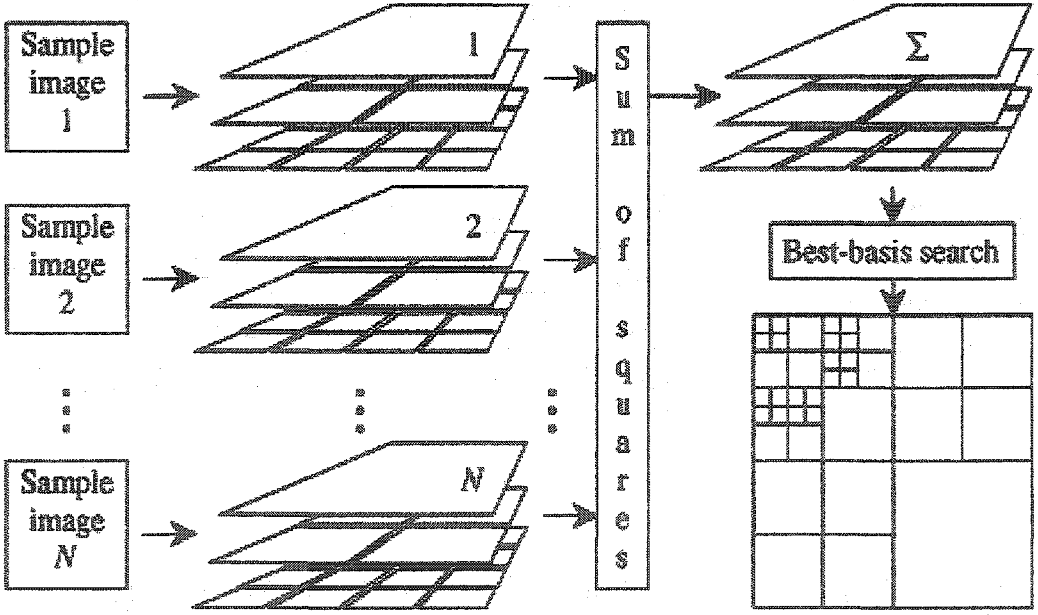


Fig.3.6 Best basis algorithm in graphical representation for two-dimensional applications

3.1.4 Computational Complexity of Best Basis Algorithm

The wavelet packet and local trigonometric bases dictionaries are equipped with fast *O*(*N logN*) transformation algorithms. Furthermore they are shaped into a tree structure which makes the search for best basis that of *O*(*N logN*) complexity. Here *N* is the signal dimension. The tree structure of the wavelet packet and local trigonometric basis dictionaries originate naturally from the way they are constructed (Chapter two).

3.2 1D Best Basis Algorithm - A Complete Example

Fig.3.7 is the wavelet packet decomposition as obtained by using Matlab® wavelet toolbox. In this example the data depicted in Fig.3.1 was decomposed into the wavelet packet generated by Coiflet 4 mother wavelet. Fig.3.8 is the best basis selected by Matlab® wavelet toolbox. By exploiting the thresholding operation one can compute a de-noised version of the signal in the same window as given in Fig.3.9.

3.3 Remarks

The process to obtain the best basis for a signal is obviously non-linear. Nevertheless, as soon as the basis is chosen, compression ratio via best basis is not drastically affected by noise since the noise energy will not change in the new basis and in case of reasonable signal to noise ratio, this will not affect the optimality of the basis. In this chapter an adaptive scheme for selecting the best basis for a class of signals originating from a statistical process was studied. However, the best basis algorithm was originally intended to perform the compression or de-noising operations on single input signals. So if signals are originating from different sources, an unsupervised best basis selection can help to compress signal or remove noise as the best basis algorithm implementation in Matlab® does.

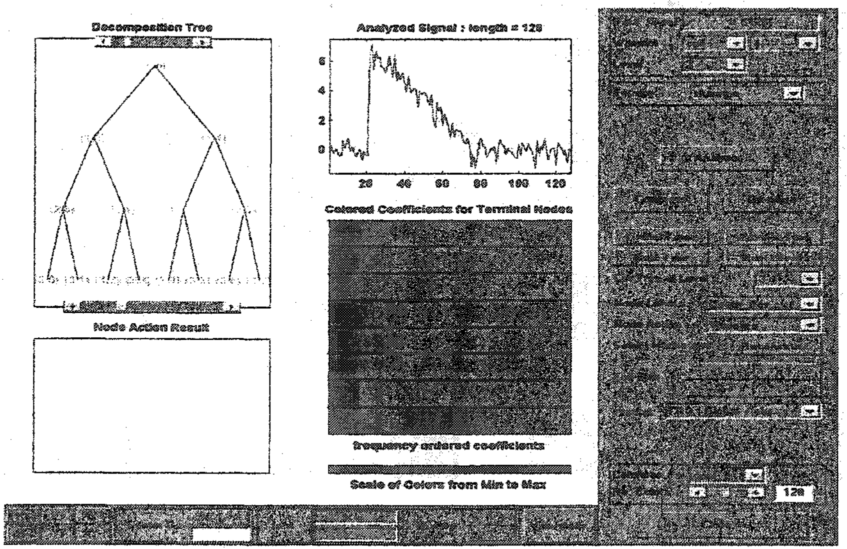

Decomposition Tree
Analyzed Signal : length = 128
6
4
2
0
20
40
60
80
100
120
Colored Coefficients for Terminal Nodes
Node Action Result
frequency ordered coefficients
Scale of Colors from Min to Max
128

Fig.3.7 Complete wavelet packet decomposition using Matlab® wavelet toolbox

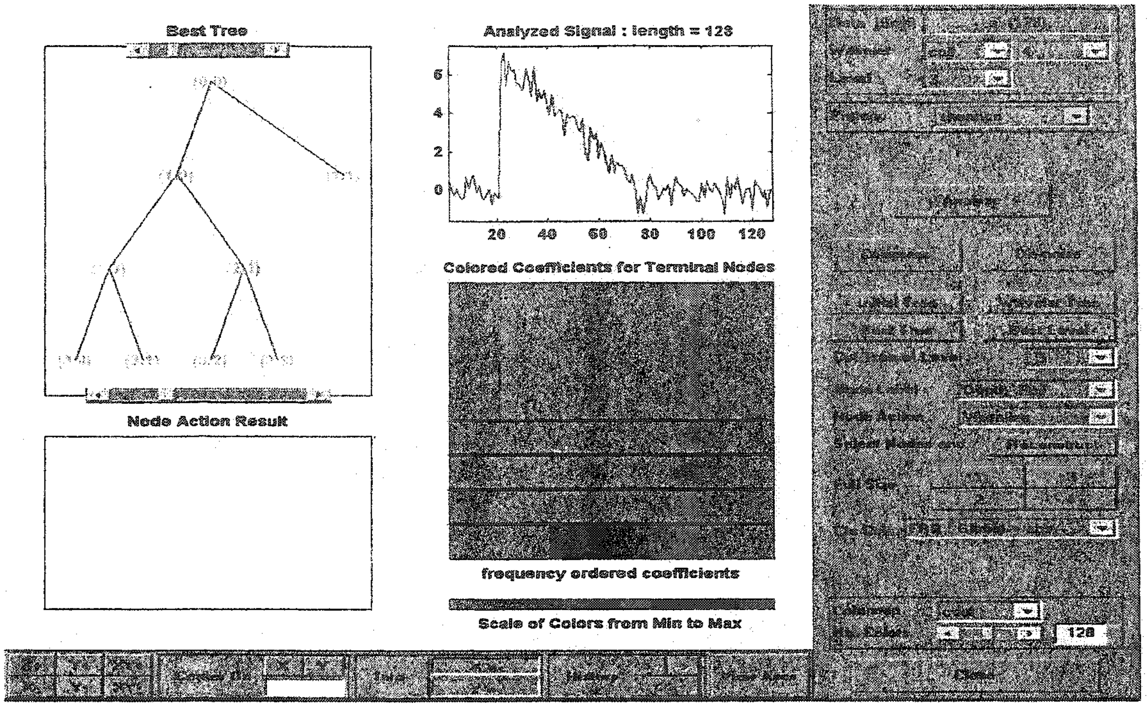

Best Tree
Node Action Result
Analyzed Signal : length = 128
6
4
2
0
20
40
60
80
100
120
Colored Coefficients for Terminal Nodes
frequency ordered coefficients
Scale of Colors from Min to Max
128

Fig.3.8 Best basis selected from the binary tree using Matlab® wavelet toolbox

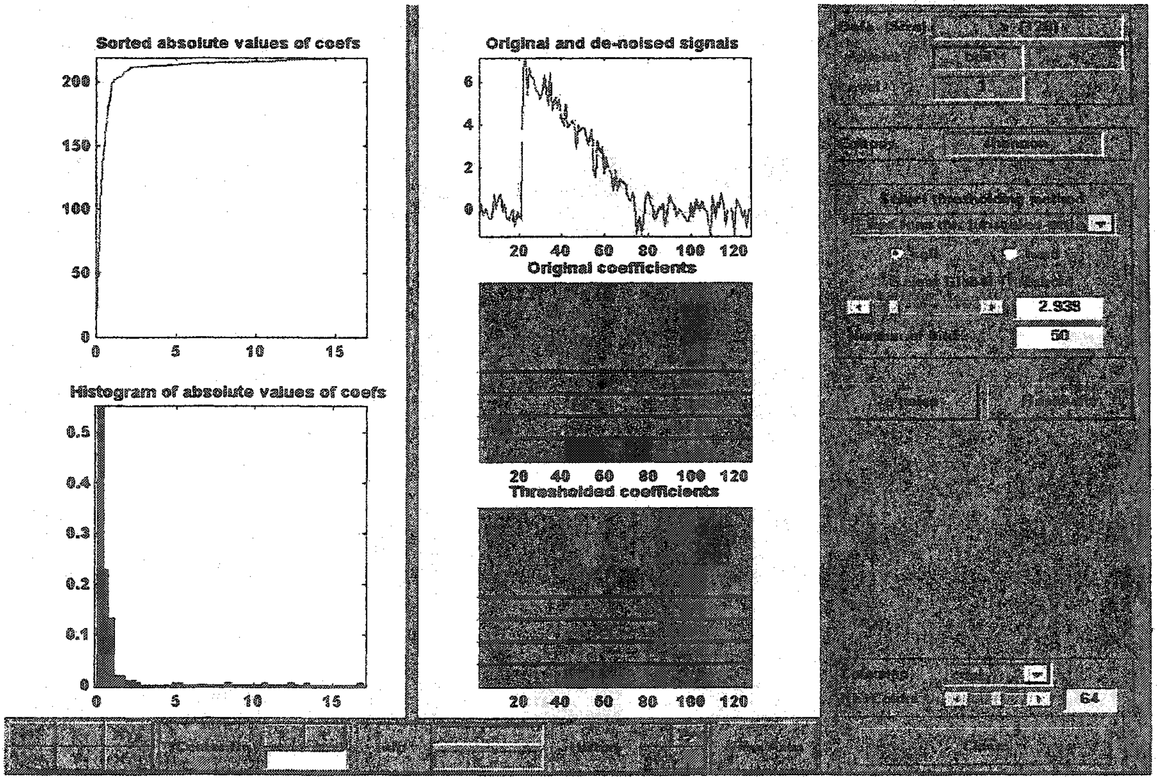


Fig.3.9 Tresholded and de-noised signal obtained by Matlab® wavelet toolbox

Chapter Four

Feature Extraction, Classification and Local Discriminant Bases

The best basis algorithm was introduced for signal compression and de-noising in Chapter three. The theory of the local discriminant basis (LDB) algorithm is explained in this chapter as a modification of best basis algorithm for feature extraction and classification applications. The key idea of LDB algorithm is to decompose signal energy into structured dictionaries for each class and then compare the efficiency of each basis function in the dictionary in discriminating between signals from different classes. After measuring the discrimination power of each basis function, a complete set of basis functions will be chosen as LDB.

The most important part of the algorithm is a redundant set of basis functions such as wavelet packet basis functions. LDB searches for a best selection of basis function for the classification of signals at hand. This process is equivalent to the extraction of the most discriminating features. LDB it is a modification of the best basis algorithm developed by Coifman and Wickerhauser [4] for signal classification/feature extraction applications (Chapter three).

4.1 The Problem of Object Recognition

Human beings recognize and classify objects with little or no conscious effort. In spite of this, a number of applications exist for machines capable of identifying and classifying objects with little or no operator assistance. In a number of environments, people have difficulty identifying objects. When ambient light is low or the atmosphere is murky, human vision is poor. On the other hand there are a number of difficult or hazardous work environments where it is preferable to use automated systems. Some of the applications may require speed or accuracy beyond human capabilities. Exploration of unknown or dangerous terrain, unexploded ordnance location, manufacturing, battlefield monitoring and optical character recognition are some of the applications of machine recognition. This is why object recognition has received a lot of attention in the research literature. Object recognition has many specific names such as target recognition, pattern recognition and is a part of scene analysis. Task specific names include face recognition, optical character recognition, and image analysis and speech recognition. The task of recognition can be divided into object localization and object identification. For both localization and identification tasks, an understanding of the objects being sought is necessary. Often a model base is used to describe the objects. Sometimes in a supervised process, the model is obtained as the training signals pass through the system.

Fig.4.1 shows typical components of a recognition system. The preprocessing stage is for noise removal or data normalization. In the feature extraction step, only semantically important information is kept and therefore signals are represented in a more compact format. The recognition algorithm, which matches the features to the models, is the last step in an object recognition process. In this particular stage, several methods of statistical analysis are employed to infer the probability of an object belonging to each individual class. Different parts of Fig.4.1 that is the general classification scheme are studied in this work. The best basis algorithm described in Chapter three performs as the preprocessing stage to reduce the storage and remove noise. Feature extraction is the topic of the current chapter. For a complete review of recognition algorithms and classifiers the reader is referred to [1].

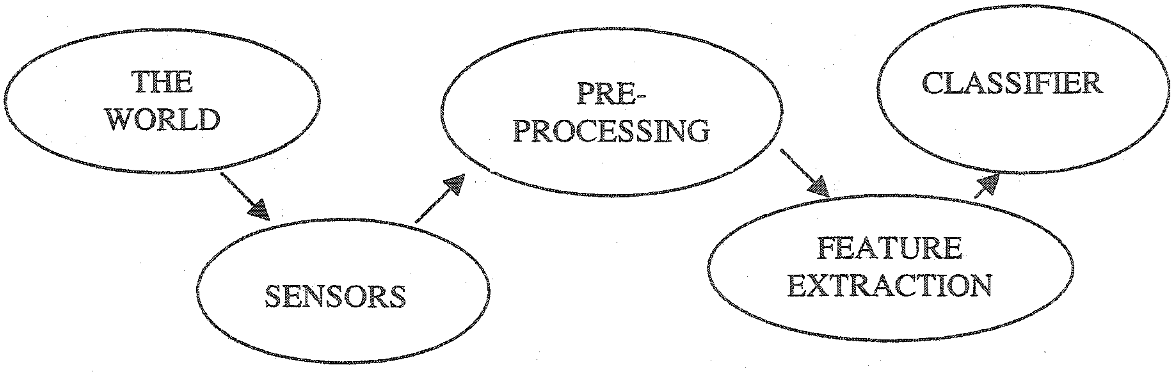


Fig.4.1 A complete machine classification scheme

4.2 The Problem of Feature Extraction and Dimension Reduction

Most signals that are encountered in digital signal processing are of very high dimension. For example an audio signal as a band limited signal with 20 kHz bandwidth will require about 40,000 samples a second for loss-less reconstruction according to Nyquist sampling theorem [18]. Therefore if the goal is to classify the voice of a male and female over a 25 msec period, one has to deal with a classification problem in 1000 coordinates. This is true for any other form of signal processing task that is to be done on this portion of the signal such as signal compression, classification, etc.

There has been tremendous effort toward reducing the signal dimension while keeping the relevant information in the reduced selected coordinates. It is necessary to note that discarding signal less important coordinates and dimension reduction is highly application-oriented. The method that works for signal compression application is not suitable for signal classification and vice versa.

In this section the general problem of feature extraction is reviewed. Suppose that we have a space $X \subseteq R^n$ of input signals and a space Y of class labels. The goal is to construct a classifier d: $X \rightarrow Y$ that assigns the correct class label to each input signal. The optimal classifier is known to be the so-called *Bayes* classifier [1]. However, Bayes classifier is impossible to construct due to high dimensionality of the real signals [1]. Examples of high dimensional signals are medical X-ray tomography (n=512^2), seismic signals (n=4000) and a speech segment (n=1024). Faced with the dimensionality and having such difficulty in constructing the Bayes classifier, the extraction of important features becomes essential. As Scott mentions in [32, Chapter 7] multivariate data in R^n are almost never n-dimensional and there often exists lower dimension structures of data. So based on the application whether compression or classification, the problem always has a lower intrinsic or theoretical dimension (Section 3.1.2). It is important to note that intrinsic dimension is an application-oriented quantity. Coifman and Wickerhauser basis selection scheme [4] followed by a simple thresholding on the amplitude of the coefficients results in significant signal dimension reduction. Saito's LDB algorithm [6] helps to reduce the dimensionality of the problem for the feature extraction and classification tasks. The feature extractor in LDB is formulated as $d = g \circ \Theta_m \circ \Psi$, where Ψ is an orthogonal transformation, Θ_m is a projection operator into m most important coordinates and g is a standard classifier. By a dimension reducer engine such as LDB, *i.e.* $\Theta_m \circ \Psi$, the most important basis vectors are selected and kept according to the classification task and the nonessential coordinates are discarded. The classifier, g, can be *Linear Discriminant Analysis* (LDA) [33], *Classification and Regression Trees* (CART) [34], *k-nearest neighbour* (k-NN) [35] or *artificial neural networks* (ANN) [36]. A complete discussion on different classifiers is given in [1].

4.3 The Role of Local Discriminant Basis Algorithm in Classification

Local discriminant basis is a preprocessing step to reduce signal dimension by extracting most efficient features in signal classification applications. The role of LDB machinery is best illustrated in Fig.4.2 where the task is *X*-ray image classification. Typically the input data is a 512×512 pixel image signal. Processing this high dimension signal is neither desirable nor practical if a computationally expensive algorithm is to follow next. LDB Algorithm is typically capable of dimension reduction to that of one tenth of the original signal [6], [40]. Considering the one step training phase, this is a valuable investment since it results in reduced computational load significantly after a fast, $O(N \log N)$, training phase.

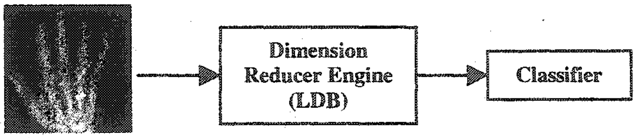


Fig.4.2 The role of LDB in a classification process

4.4 Overview of Local Discriminant Basis Algorithm

An overview of the LDB algorithm is given in flowchart format in Fig.4.3 for extracting features from a set of training images. The LDB method first decomposes normalized by class energy training signals from different classes in a time-frequency dictionary, which is a large collection of bases functions, *i.e.* wavelet packets or local trigonometric basis (Chapter two). Then signal energies are accumulated for each class separately to form a time-frequency energy distribution per class. Let us assume that ω is a typical time-frequency basis function (atom) in a time-frequency dictionary. Then the time-frequency distribution energy of class y along ω is defined in (48),

$$\Gamma_{\omega}^{y} = (\Sigma_{N_y} \| \omega . x_i^y \|^2)/(\Sigma_{N_y} \| x_i^y \|^2), \qquad (48)$$

where $\{ x_i^y \}$ is the collection of class y signals and N_y is the number of class y training signals. Note that the class normalization is realized in (48) that is vital to the correct performance of the algorithm if different classes differ significantly in average energy content. For a complete set of basis functions $B=\{\omega_1, \omega_2, \ldots, \omega_m\}$ the time-frequency energy map is given by (49),

$$\Gamma_B^y = (\Gamma\omega_1^y, \Gamma\omega_2^y, \ldots, \Gamma\omega_m^y). \qquad (49)$$

Based on the energy maps obtained, the discrepancy measure (discriminant) will determine the discrimination measure of each basis function. This measure shows how the specific basis function is successful in discriminating between different classes of input signals. These ideas are mathematically established in later sections of this chapter. After that a search algorithm will look for the LDB. Fig.4.4 shows the search mechanism in a single dictionary of basis functions. It can be extended to the search in a library if one decides to search among all available time-frequency dictionaries. As shown in Fig.4.4, LDB starts the search from the bottom of the binary tree shown on the top that consists of the children nodes. Starting from the bottom children nodes the discrimination measure (as the cost function of an optimization process) determines whether to keep the children nodes or their parent. If the summation of discrimination measures of children nodes is higher than their parents, the children nodes are more discriminating and the summation of their discrimination power will be shown on the parent node. The original discrimination measure of the parent node in this case is shown inside parenthesis (the middle figure). Finally the most discriminating basis covering the horizontal line is chosen as LDB (the bottom figure). The result of this process might look like Fig.4.5 and the corresponding time-frequency tiling might look like Fig.2.13 (b). It is important to note that the basis function selection process is highly dependent upon the cost function introduced in Fig.4.3 flowchart diagram as discriminant. In signal compression application the cost function tends to minimize the entropy of signal projections on new coordinate system (Chapter three) whereas in a signal classification problem the cost function tends to maximize the difference between different signal classes. These different approaches have led to different algorithms for different applications. Best basis algorithm is basically a compression algorithm whereas LDB is a classification/feature extraction tool.

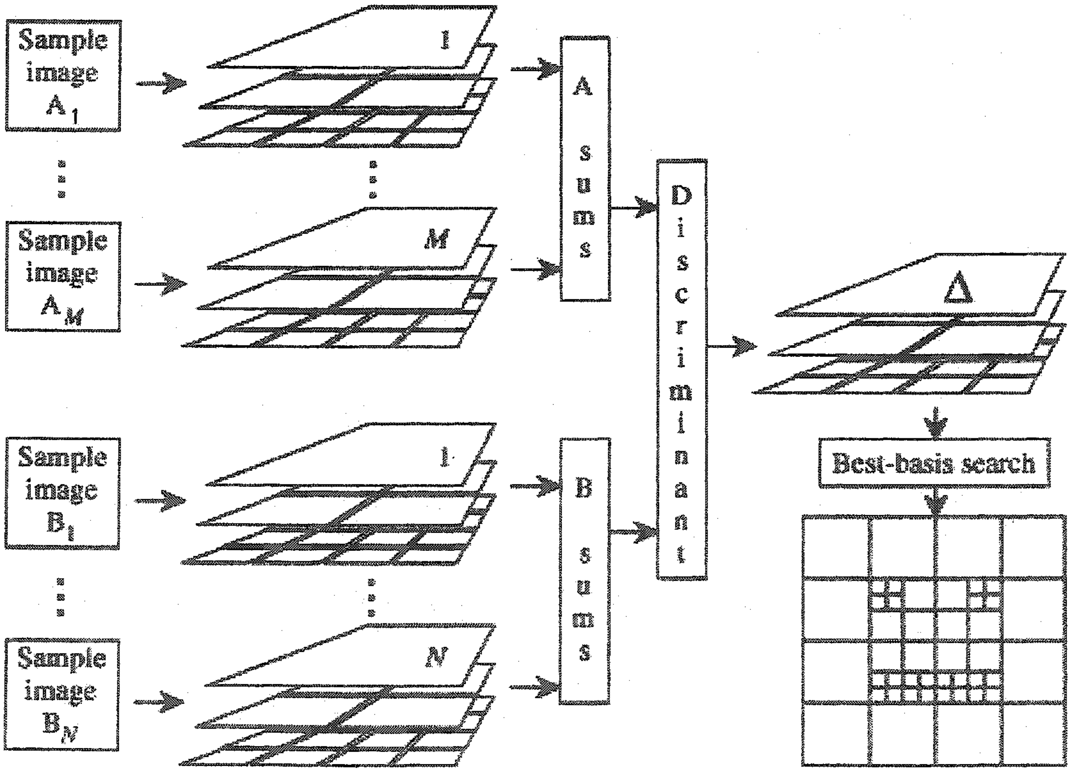
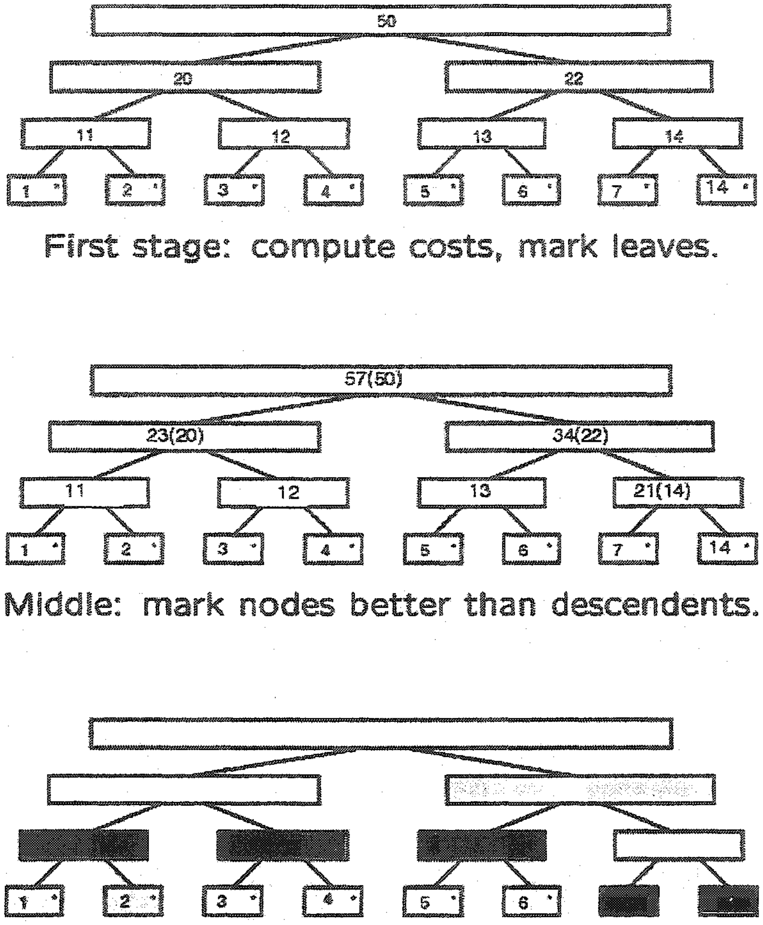


Fig.4.3 Graphical view of the local discriminant basis algorithm for a two-dimensional image processing application



First stage: compute costs, mark leaves.

Middle: mark nodes better than descendents.

Final stage: keep topmost marked nodes.

Fig.4.4 The steps toward calculation of LDB

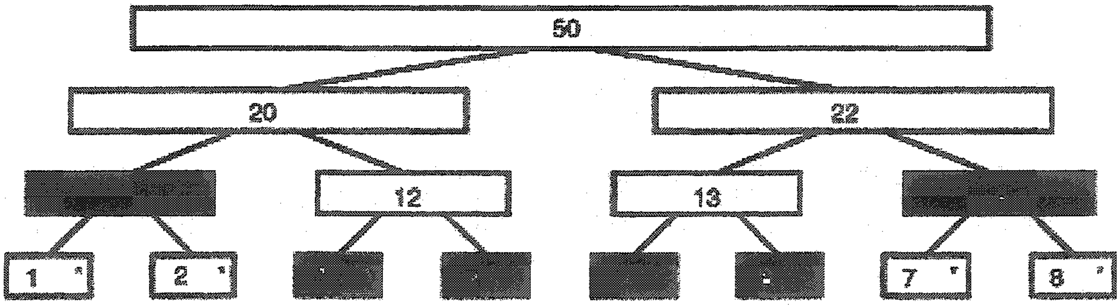


Fig.4.5 A possible LDB

4.5 Dictionary Tree Pruning with a Cost Function (Discrepancy Measure)

As pointed out in the previous section, the cost function for the algorithm will determine the functionality of the algorithm. Best basis algorithm and LDB are both statistical signal-processing tools as cost functions are based on statistical averaging concept [7]. Other non-entropy discrepancy measures are possible like the simple Euclidean distance. For signal compression applications a complete set of basis functions is considered better if the coordinates in this basis are less scattered and so concentrated in a few coordinates that one can keep and discard the rest. In information theory this is expressed as having less information content. So a coordinate system in which signal representations have less information content is considered to be the best basis. The well-known measure of information is the Shannon entropy. If $\{p_i\}=\{p_1, p_2, ..., p_n\}$ is signal representation in a new coordinate system, the information content of the representation can be measured by (50),

$$H(\{p_i\}) = -\sum p_i \log p_i , \tag{50}$$

where *log* is the normal logarithm. The situation is significantly different in signal classification applications. In feature extraction applications one is interested in basis functions that represent signals from different classes as different as possible so making the classification task much easier. Assume that $\{p_i\}=\{p_1, p_2, ..., p_n\}$ and $\{q_i\}=\{q_1, q_2, ..., q_n\}$ are signal x and y representation in a new coordinate system. For measuring the difference between x and y in this system, one can simply measure their Euclidean distance by (51),

$$H(\{p_i\},\{q_i\}) = \sum (p_i - q_i)^2 \ . \tag{51}$$

Although (51) is an easy calculation, but certainly it is a deterministic operation ignoring the information content. In a supervised process such as the one that is going to be implemented in LDB, there is a need for statistical measurements in order to make sound decisions. Experiments show that the results obtained by using relative entropy concept [6, 7] are quite satisfactory since this approach deals with the information content of a basis for a classification task. There are a number of choices for discrepancy measure. For classification problems one needs a measure to evaluate the power of discrimination of each basis function or the subspace associated with it. Once the discrepancy measure is selected, LDB algorithm searches for the best basis for classification problems the same way Wickerhauser's best basis algorithm does [4]. The goal in this section is to assign a discrimination measure to each basis function in a dictionary. The discrimination measure should assign a higher value to a basis function if the projections on this basis function differ more for signals from different classes. The following example introduces such a measure.

Example 4.1. The discrimination power of the representation of $\{x_n\}=\{1,2,3,4,5\}$ and $\{y_n\}=\{2,2,4,4,5\}$ can be compared with that of $\{z_n\}=\{2,-2,3,-4,5\}$using Euclidean measure. One needs to normalize the representations first. Then by using (62) one gets:

$$H(\{x_n\},\{y_n\})=(\frac{1}{55}-\frac{4}{65})^2+(\frac{4}{55}-\frac{4}{65})^2+(\frac{9}{55}-\frac{16}{65})^2+(\frac{16}{55}-\frac{16}{65})^2+(\frac{25}{55}-\frac{25}{65})^2=157\times10^{-4}$$

$$H(\{x_n\},\{z_n\})=(\frac{1}{55}-\frac{4}{58})^2+(\frac{4}{55}-\frac{4}{58})^2+(\frac{9}{55}-\frac{9}{58})^2+(\frac{16}{55}-\frac{16}{58})^2+(\frac{25}{55}-\frac{25}{58})^2=34\times10^{-4}$$

So the discrepancy measure is higher between $\{x_n\},\{y_n\}$ compared to that of between $\{x_n\},\{z_n\}$.

Example 4.2. *Example 4.1* can be equivalently studied using measure in (52) as discrimination measure known as *symmetric relative entropy* or *J*-divergence. $\{x_n\}$ and $\{y_n\}$are considered normalized in (52),

$$H(\{x_n\},\{y_n\})=\sum x_i \log\frac{x_i}{y_i}+\sum y_i \log\frac{y_i}{x_i}\ . \qquad (52)$$

The quantitative comparison is done in the following:

$$H(\{x_n\},\{y_n\})=(\frac{1}{55}-\frac{4}{65})\log(\frac{1/55}{4/65})+(\frac{4}{55}-\frac{4}{65})\log(\frac{4/55}{4/65})+(\frac{9}{55}-\frac{16}{65})\log(\frac{9/55}{16/65})+$$
$$+(\frac{16}{55}-\frac{16}{65})\log(\frac{16/55}{16/65})+(\frac{25}{55}-\frac{25}{65})\log(\frac{25/55}{25/65})=1745\times10^{-4}$$

$$H(\{x_n\},\{z_n\})=(\frac{1}{55}-\frac{4}{58})\log(\frac{1/55}{4/58})+(\frac{4}{55}-\frac{4}{58})\log(\frac{4/55}{4/58})+(\frac{9}{55}-\frac{9}{58})\log(\frac{9/55}{9/58})+$$
$$+(\frac{16}{55}-\frac{16}{58})\log(\frac{16/55}{16/58})+(\frac{25}{55}-\frac{25}{58})\log(\frac{25/55}{25/58})=704\times10^{-4}$$

Again the result obtained in *Example 4.1* with Euclidean discrepancy measure is verified. As seen in the above examples, the function introduced by (51) is a measure of discrimination. Also the function in (52) is a good measure of discrimination. In Chapter five we pose a problem which both (51) and (52) fail to discriminate in. The particular choice of discrepancy measure depends on the application at hand. In the original LDB algorithm, the binary-tree time-frequency dictionary is pruned by using a discrimination measure. Assume that there are only two classes of signals. The generalization of the method to more signal classes is straightforward. Examples are,

- *l^2-distance:*

$$W(\Gamma_B^{\,1}, \Gamma_B^{\,2}) = \Sigma_{\omega\in B} \|\Gamma_\omega^{\,1} - \Gamma_\omega^{\,2}\|^2, \tag{53}$$

- *Relative entropy (or Kullback-Leibler divergence):*

$$D(\Gamma_B^{\,1}, \Gamma_B^{\,2}) = \Sigma_{\omega\in B} \Gamma_\omega^{\,1} \log(\Gamma_\omega^{\,1}/\Gamma_\omega^{\,2}), \tag{54}$$

- *Symmetric relative entropy (or J-divergence):*

$$J(\Gamma_B^{\,1}, \Gamma_B^{\,2}) = D(\Gamma_\omega^{\,1}, \Gamma_\omega^{\,2}) + D(\Gamma_\omega^{\,2}, \Gamma_\omega^{\,1}), \tag{55}$$

- *Hellinger distance:*

$$H(\Gamma_B^{\,1}, \Gamma_B^{\,2}) = \Sigma_{\omega\in B} (sqrt(\Gamma_\omega^{\,1}) - sqrt(\Gamma_\omega^{\,2}))^2, \tag{56}$$

where *sqrt* stands for square root and $\Gamma_\omega^{\,i}$ is the energy of class i on basis function ω. In the first step LDB is the children nodes at the lowest part of the tree in Fig.4.4. Then the discrimination measures of each two children nodes are compared to their parent's. If the sum of the discrimination measures of the children nodes is higher than their parent's, the children nodes are kept. Otherwise, the parent node is chosen as the LDB (Fig.4.4). Once a complete basis (LDB) is selected, we further choose $m(< n)$ atoms from the LDB to reduce the dimensionality of the problem by discarding unnecessary coordinates for classification. A typical m would be around $n/10$ [6], [40]. The simplest way of choosing m atoms from a collection of n atoms is to sort them in the order of decreasing discrimination power and to retain the first m atoms. It is important to note that the functionality of LDB lies in the over-complete nature of binary tree wavelet packet and local discriminant basis dictionaries [2]. This redundancy enables the introduction of different search algorithms with different discriminant measures to prune the binary tree in an optimal manner that is tailored to the particular application.

4.6 Original LDB Algorithm (Type I)

Here is a formal statement of the algorithm. A training set of signals is needed for training the system and finding the best basis. Assume that a set of N training signals from L different classes are given and N_l is the number of signals in class l. Let $\{x_i^{(l)}\}$ denote the collection of training signals in class l, in which the superscript (l) indicates the class that the signal belongs to. Suppose that each x belongs to a unique class so that $N = N_1 + N_2 + ... + N_l$. LDB uses this training set of signals to search for a best basis in available libraries of bases with respect to its cost function.

Definition 4.1. Given a sequence of vectors $\{p^{(c)}\}^C_{c=1}$, their J-divergence is defined as in (57) where the summation is taken over all pairs of i and j that are not equal.

$$J(\{p^{(c)}\}^C_{c=1}) = \Sigma\Sigma J(p^{(i)}, p^{(j)}). \tag{57}$$

Definition 4.2. Let $\{x_i^{(c)}\}_{i=1}^{Nc}$ be a set of training signals belonging to class c. Then the time-frequency energy map of class c, denoted by Γ_c, is a table of real numbers specified by the triplet (j,k,l) as in (58),

$$\Gamma_c(j,k,l) = \frac{\sum_{i=1}^{N_c} (\omega_{j,k,l}^T x_i^{(c)})^2}{\sum_{i=1}^{N_c} \left\| x_i^{(c)} \right\|}, \tag{58}$$

for $j = 0,1,...,J$, $k = 0,1,...,2^j - 1$ and $l = 0,1,...,2^{n_0 - j} - 1$. In other words, Γ_c is computed by accumulating the squares of expansion coefficients of the signals at each position in the binary tree followed by the normalization of the total energy of the signals belonging to class c. This normalization may be important especially if there is significant difference in the number of samples among classes. The following notation is used in LDB algorithm stated below,

$$\Delta_{j,k} = J(\{\Gamma_c(j,k,.)\}_{c=1}^{C}) = \sum_l J(\Gamma_1(j,k,l),...,\Gamma_C(j,k,l)) \tag{59}$$

Here is an algorithm to select an orthonormal basis (from the library), which maximizes the discriminant measure on the time-frequency energy distributions of classes. This is called local discriminant basis (LDB) algorithm. Let $B_{j,k}$ denote a set of basis vectors at the subspace $\Omega_{j,k}$ arranged as a matrix,

$$B_{j,k} = \{\omega_{j,k,0},...,\omega_{j,k,2^{n_0-j}-1}\}. \tag{60}$$

Let $A_{j,k}$ represent the LDB (which the algorithm is searching for) restricted to $B_{j,k}$.

Algorithm (The Local Discriminant Basis Selection Algorithm). Given a training database of C classes of signals,

Step 0. Choose a library of orthonormal bases (*i.e.*, specify QMFs for a wavelet packet).
Step 1. Construct time-frequency energy maps Γ_c for $c = 1,...,C$.
Step 2. Set $A_{J,k} = B_{J,k}$ for $k = 0,1,...2^J-1$.
Step 3. Determine the best subspace $A_{j,k}$ for $j = J-1,...,0$, $k = 0,1,...,2^j-1$ by the following rule:

```
If    $\Delta_{j,k} \geq \Delta_{j+1,2k} + \Delta_{j+1,2k+1}$
      then    $A_{j,k} = B_{j,k}$
      else    $A_{j,k} = A_{j+1,2k} \oplus A_{j+1,2k+1}$
```

Step 4. Order the basis functions by their power of discrimination (see Step 5)
Step 5. Use $k(<n)$ most discriminant basis functions for constructing classifiers.

The LDB is selected in Step 3. However, to reduce the dimensionality of the problem, Step 4 and 5 are still necessary. In Step 4, there are several choices as a measure of discrimination power of an individual basis function. Let $\lambda = (j,k,l) \in Z^3$ be a triplet specifying one of the LDB functions selected in Step 3. The following is a candidate for the measure in Step 4,

$$J(\Gamma_1(\lambda),...,\Gamma_C(\lambda)). \tag{61}$$

Proposition 4.1. The discriminant measure J is additive in the sense that

$$J(\{p_i\}^n{}_{i=1}, \{q_i\}^n{}_{i=1}) = \sum J(p_i, q_i). \tag{62}$$

Proposition 4.2. The basis obtained by Step 3 of LDB Algorithm maximizes the additive discrepancy measure J on the time-frequency energy distributions among all the bases in the library obtainable by the divide-and-conquer algorithm.

Readers can refer to [6, 7] for proof. Step 5 reduces the dimensionality of the problem from n to k without loosing the important discriminant information in terms of time-frequency energy distributions among classes. Thus many interesting statistical techniques, which are usually computationally too expensive for n dimensional problems become feasible. How to select k is a challenging question [6], [40]. Also how to select the libraries that are fed into LDB algorithm remains to be an open research problem.

4.7 One Dimensional LDB Algorithm in Triangular Waveform Classification

For the purpose of illustration, example 5.2 of [6] is studied. In this classification problem, three classes of signals are generated by the process given in (63).

$$\begin{aligned} c(i) &= (6+\varsigma).\chi_{[a,b]}(i) + NoiseLevel.\varepsilon(i) \\ b(i) &= (6+\varsigma).\chi_{[a,b]}(i).\frac{i-a}{b-a} + NoiseLevel.\varepsilon(i) \\ f(i) &= (6+\varsigma).\chi_{[a,b]}(i).\frac{b-i}{b-a} + NoiseLevel.\varepsilon(i) \end{aligned} \tag{63}$$

"a" is an integer-valued uniform random variable on the interval [16,32]. "$b-a$" also obeys an integer-valued uniform distribution on [32,96]. "ς" and "ε" are the standard normal variates, and $\chi_{[a,b]}(i)$ is the characteristic function on $[a,b]$. $c(i)$ is called the 'cylinder' class whereas $b(i)$ and $f(i)$ are known as 'bell' and 'funnel' classes. $NoiseLevel \geq 0$ determines the SNR of the generated signal space.

The statistical process in (63) serves as a classical criteria to test the effectiveness of classification schemes. It is known as "triangular waveform classification" problem. It is a truly non-stationary statistical process with different classes having equal statistical averages. Several authors have used this example in testing the effectiveness of classification schemes [34]. The Bayes classifier error rate is known for this example and is about 14%. The classes form a triangle in higher dimensional space and each edge of the triangle forms a class; in other words, data distribution is nontrivial, yet computationally easy to generate this process. For these reasons, triangular waveform classification example is repeated in all the experiments in this thesis to study the effect of optimizing LDB vectors in Chapter eight and LDB algorithm characteristics in Chapter nine. For a complete description of the properties of the statistical process given in (63) the interested reader is referred to [34].

In this study i varied between 1 and 128. The statistical averages of the each class in the above process over a set of 360 training signals with $SNR \approx 25$ are depicted in Fig.4.6. The average LDB coefficients for different classes of generated signals for two different SNR levels as obtained from LDB implementation in Matlab® (Chapter seven) are depicted in Fig.4.7. In this experiment, a wavelet packet dictionary generated by Coiflet 4 mother wavelet is used to decompose the signals and symmetric differential entropy or J-divergence was used as discrimination measure. It is easily observed that the energy of signals from different classes are concentrated in the first most discriminating basis functions basically from the first basis function till the twentieth. Also it is understood that as the level of noise increases, the number of significant basis functions decreases. Besides the signal coefficients in the new local discriminant basis, the actual local discriminant vectors are given as the output and are depicted in Fig.4.8. The misclassification rate in case of SNR=25 by only classifying the first eleven LDB coordinates is around 2%. In case of SNR=9 and by classifying the first eight LDB coordinates, the misclassification rate is about 7.5%. Linear Discriminant Analysis (LDA) of R. A. Fisher [1], [33] is used in all experiments as the classifier to classify the LDB extracted features.

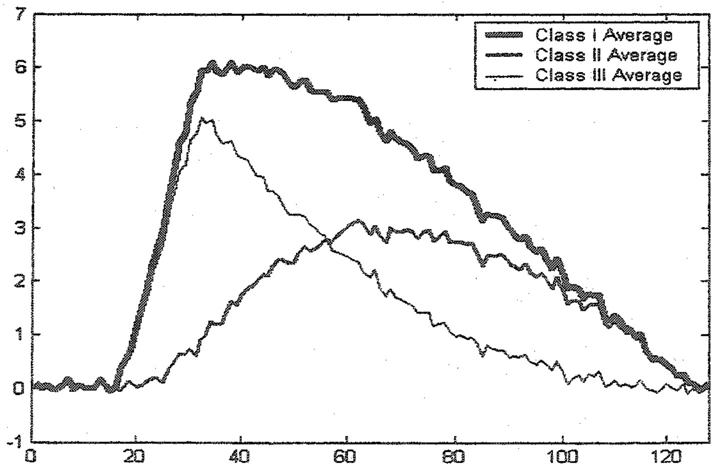


Fig.4.6 Statistical averages of different classes of signals produced by equations in (63)

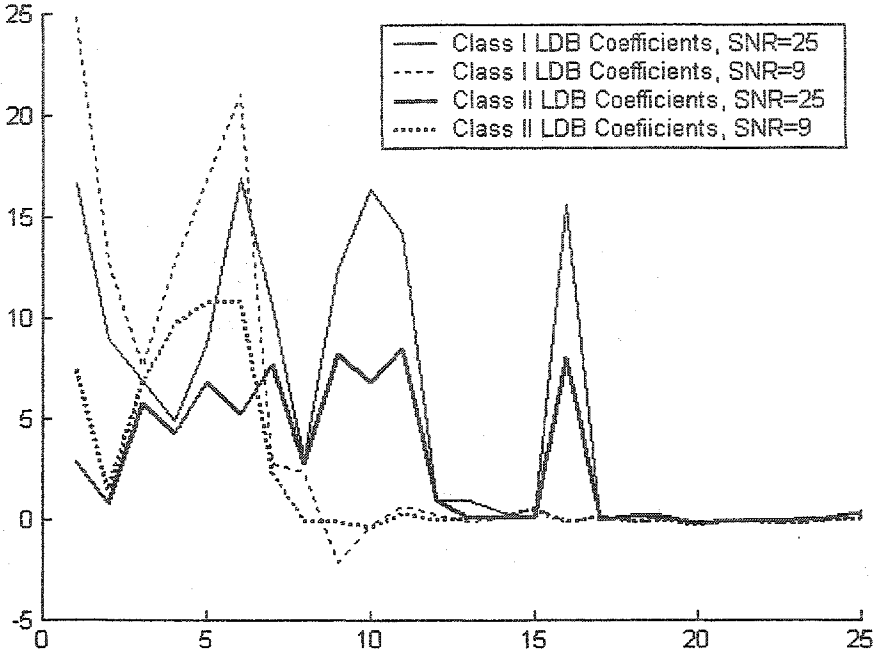


Fig.4.7 Average LDB coefficients of class I and class II of the signals in (63) at different SNR levels

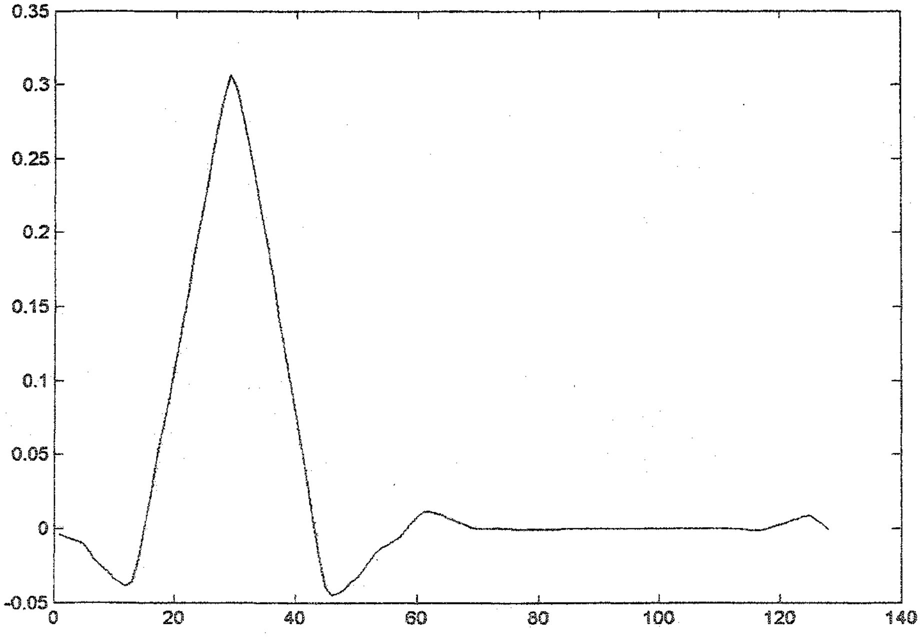
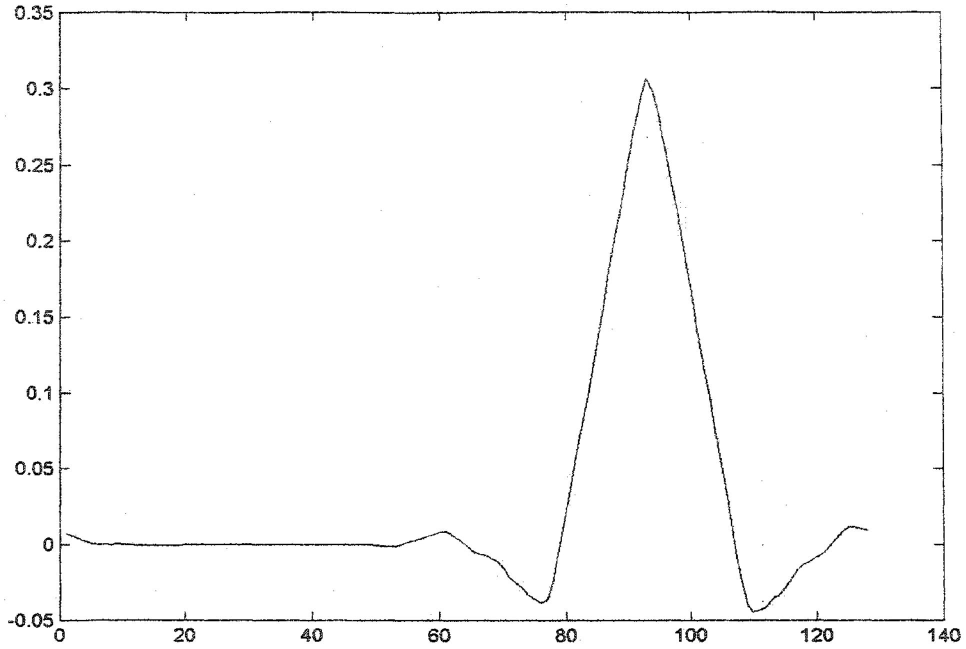


Fig.4.8 Selected LDB features



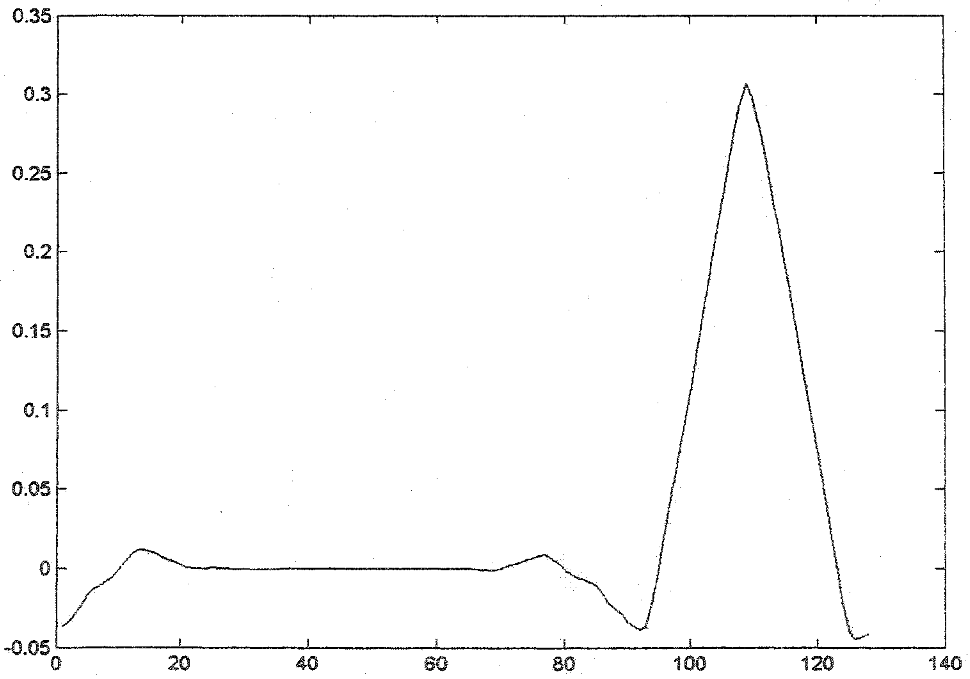


Fig.4.8 (continued)

4.8 Two Dimensional LDB Algorithm in Shape Classification

Herein the effectiveness of two-dimensional LDB algorithm in image processing applications is examined. Three classes of images are generated by the statistical processes described in (64) through (68) respectively. The size of each image is fixed to a 64×64 block. The edges of shapes are generated by *x*(*k*)'s and *y*(*k*)'s that specify the horizontal and vertical edges and are generated by the statistical processes given below in (64-67). The brightness of each pixel inside and outside the regions is given by (68) thus making the training set a grayscale 8-bit image database. The process in (64) is generating a class of distorted circles (Fig.4.9). The range of *k* is between 1 and 64. In all the expressions "ξ", "ξ'" and "ς" are uniform random variables on [-1,1] and [0,1] respectively. "ε" is a standard normal variate,

$$x_1(k) = 32 + 5\xi + \varsigma(5+\varepsilon) - \sqrt{\left|225-(32+5\xi-k)^2\right|}, \tag{64a}$$

$$x_2(k) = 32 + 5\xi + \varsigma(5+\varepsilon) + \sqrt{\left|225-(32+5\xi-k)^2\right|}, \tag{64b}$$

$$y_1(k) = 32 + 5\xi' + \varsigma(5+\varepsilon) - \sqrt{\left|225-(32+5\xi'-k)^2\right|}, \tag{64c}$$

$$y_2(k) = 32 + 5\xi' + \varsigma(5+\varepsilon) - \sqrt{\left|225-(32+5\xi'-k)^2\right|}. \tag{64d}$$

The second class as generated by the process in (65) is a collection of distorted rectangles (Fig.4.10). Again *k* ranges between 1 and 64,

$$x_1(k) = 15 + 5\xi + \varsigma(5+\varepsilon), \tag{65a}$$

$$x_2(k) = 39 + 5\xi + \varsigma(5+\varepsilon), \tag{65b}$$

$$y_1(k) = 15 + 5\xi' + \varsigma(5+\varepsilon), \tag{65c}$$

$$y_2(k) = 39 + 5\xi' + \varsigma(5+\varepsilon). \tag{65d}$$

The third class is a combination of the above-mentioned classes in the sense that it is the collection of distorted circles and distorted rectangles placed at upper left and lower right respectively (Fig.4.11). If $1 \le k \le \frac{N}{2}$ then the equations below produce a set of distorted circles in the upper left corner of the image,

$$x_1(k) = 20 + 5\xi + \varsigma(5+\varepsilon) - \sqrt{\left|100 - (20 + 5\xi - k)^2\right|}\,, \tag{66a}$$

$$x_2(k) = 20 + 5\xi + \varsigma(5+\varepsilon) + \sqrt{\left|100 - (20 + 5\xi - k)^2\right|}\,, \tag{66b}$$

$$y_1(k) = 20 + 5\xi' + \varsigma(5+\varepsilon) - \sqrt{\left|100 - (20 + 5\xi' - k)^2\right|}\,, \tag{66c}$$

$$y_2(k) = 20 + 5\xi' + \varsigma(5+\varepsilon) + \sqrt{\left|100 - (20 + 5\xi' - k)^2\right|}\,. \tag{66d}$$

However, when $1 + \frac{N}{2} \le k \le N$ the following expressions generate a set of distorted rectangles at the lower right of the image,

$$x_1(k) = 30 + 5\xi + \varsigma(5+\varepsilon)\,, \tag{67a}$$

$$x_2(k) = 50 + 5\xi + \varsigma(5+\varepsilon)\,, \tag{67b}$$

$$y_1(k) = 30 + 5\xi' + \varsigma(5+\varepsilon)\,, \tag{67c}$$

$$y_2(k) = 50 + 5\xi' + \varsigma(5+\varepsilon)\,. \tag{67d}$$

The brightness of each pixel is given below for regions inside the edges for all classes in (68a). However, the brightness of each pixel outside the edges is given by (68b). Here "ε'" is a standard normal variate,

$$\mathit{IntensityInside} = 225 \times \left|\sin(\varepsilon')\right|\,, \tag{68a}$$

$$\mathit{IntensityOutside} = 100 \times \left|\varepsilon'\right| \times \chi_{[1,2]}(\varepsilon') + 20 \times \left|\varepsilon'\right|\,. \tag{68b}$$

A sample image from each class is depicted in Fig.4.9 to Fig.4.11. The signals generated by statistical process in (64) represents a class of distorted circles. The second class as obtained by (65) is a collection of distorted rectangles. Statistical process in (66) and (67) produces a combination of a distorted circle and a distorted rectangle. Noise is added to all generated images as given in (69). "ε''" is a normal variate and *NoiseLevel* is a constant. Fig.4.12 to Fig.4.14 show the noisy images with $SNR = 4$.

$$\mathit{Noise} = \mathit{NoiseLevel} \times \varepsilon'' \tag{69}$$

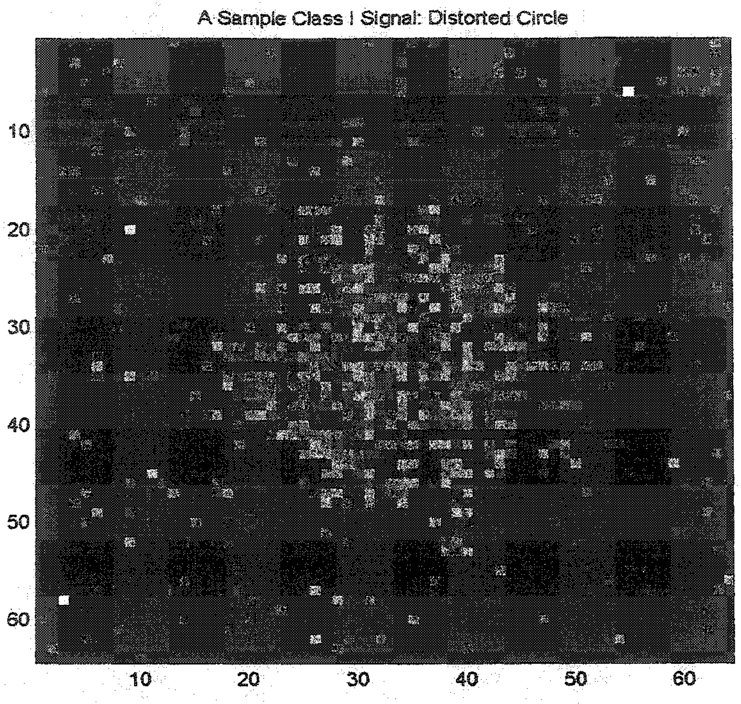


Fig.4.9 Class I: distorted circle

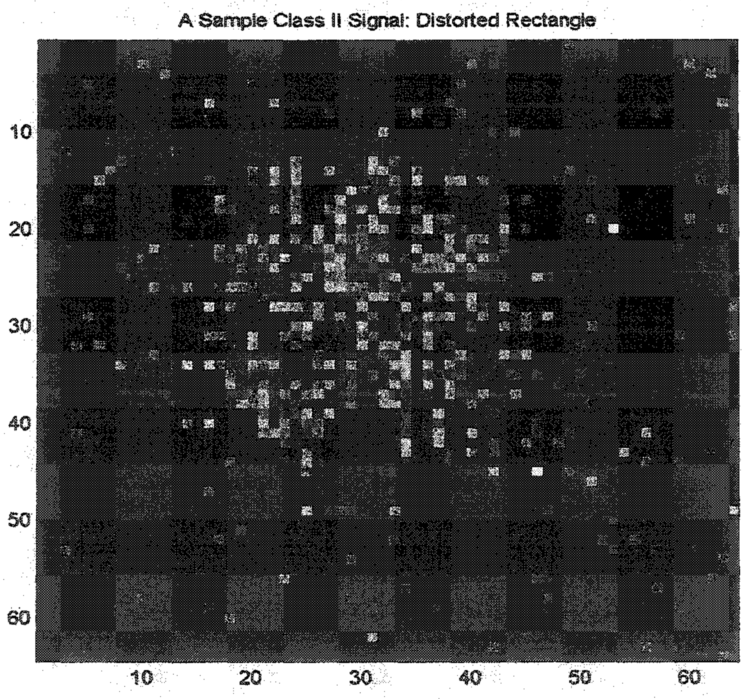


Fig.4.10 Class II: distorted rectangle

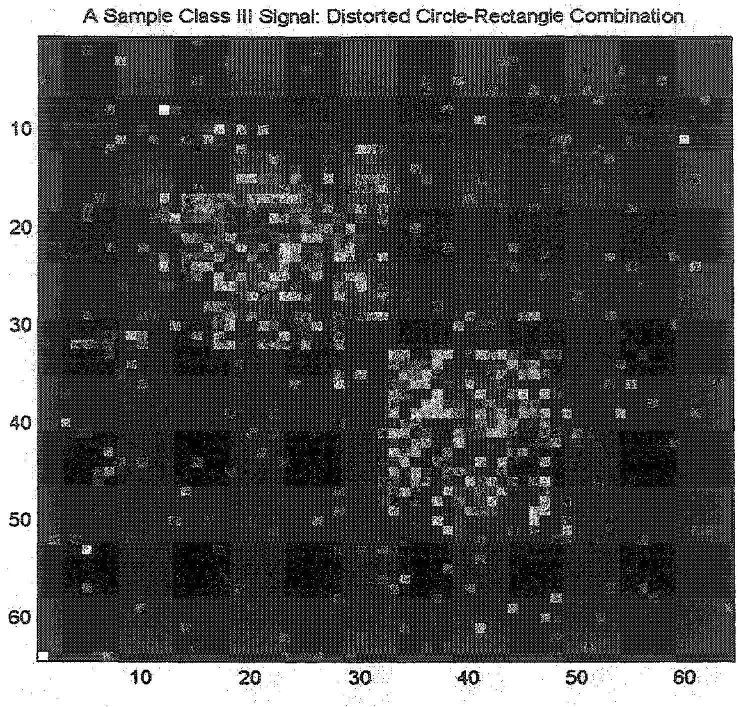


Fig.4.11 Class III: distorted circle and rectangle

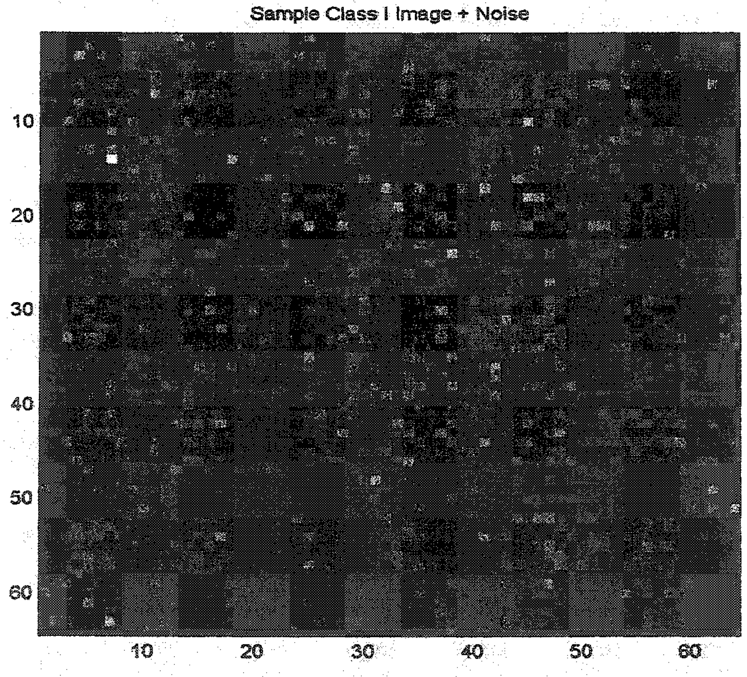


Fig.4.12 Noisy class I image

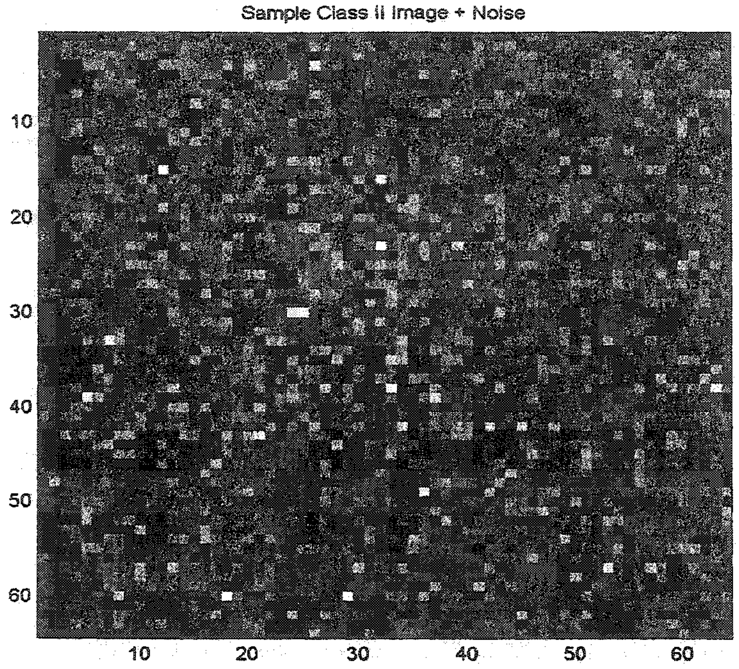


Fig.4.13 Noisy class II image

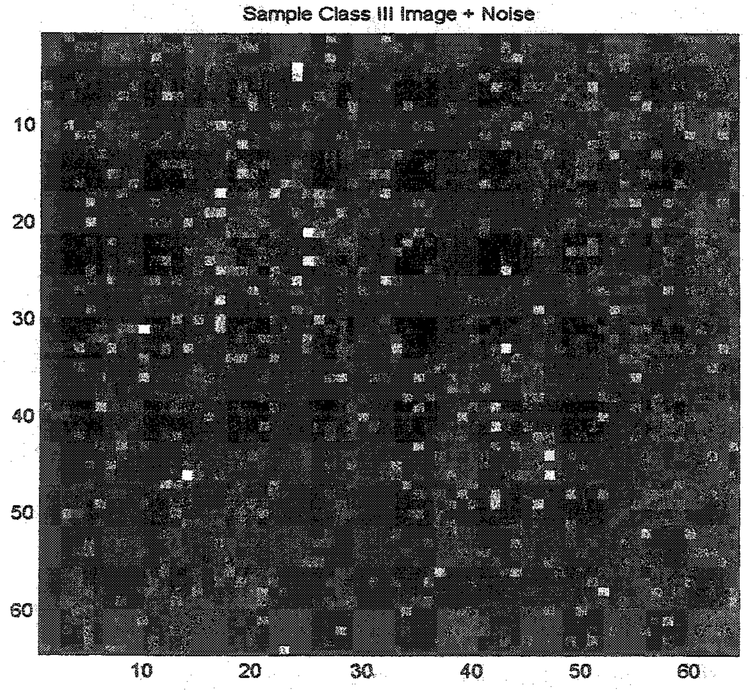


Fig.4.14 Noisy class III image

Here is how this experiment is performed. The training set consisted of 150 images containing equal number of samples from each class. Each image is a 64×64 pixel, 8-bit grayscale block. A Daubechies 20 QMF was used to analyze the problem using symmetric differential entropy as discrepancy measure. Fig.4.15 to Fig.4.17 show the average image of different classes in LDB domain obtained by averaging the brightness of each pixel. The misclassification rate of the overall process using 10 LDB features and LDA classifier is 0% with *SNR* = 4. Fig.4.18 to Fig.4.20 show most discriminating basis functions. Fig.4.21 to Fig.4.23 show the images filtered after only keeping the first 21 most discriminating LDB features and translation into ordinary Cartesian coordinates.

Note: This example is also carried out to Chapter eight and Chapter nine of the thesis to examine the efficiency of optimized local discriminant basis algorithm (OLDB) and studying the noise behavior and other properties of OLDB. The example in Section 4.8 is designed for this thesis and due to complex non-stationary statistical processes described in (64-68) generating image boundaries and brightness; it serves as an efficient non-stationary image classification criterion problem.

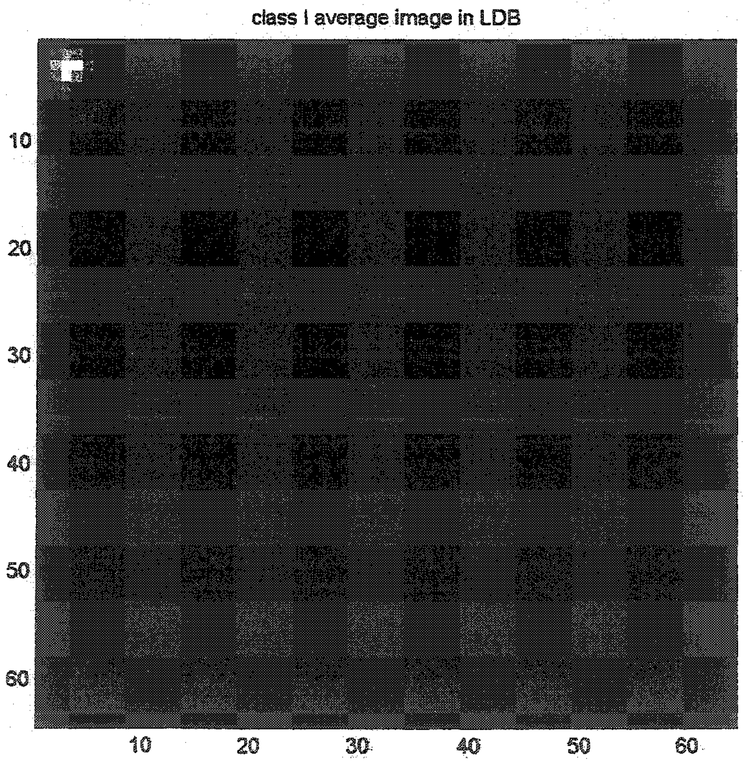


Fig.4.15 The average of class I in LDB domain

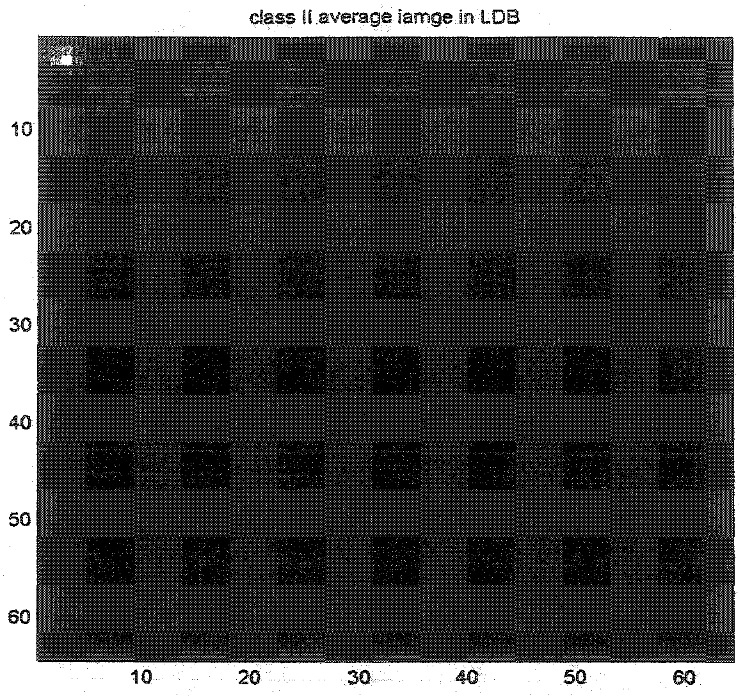


Fig.4.16 The average of class II in LDB domain

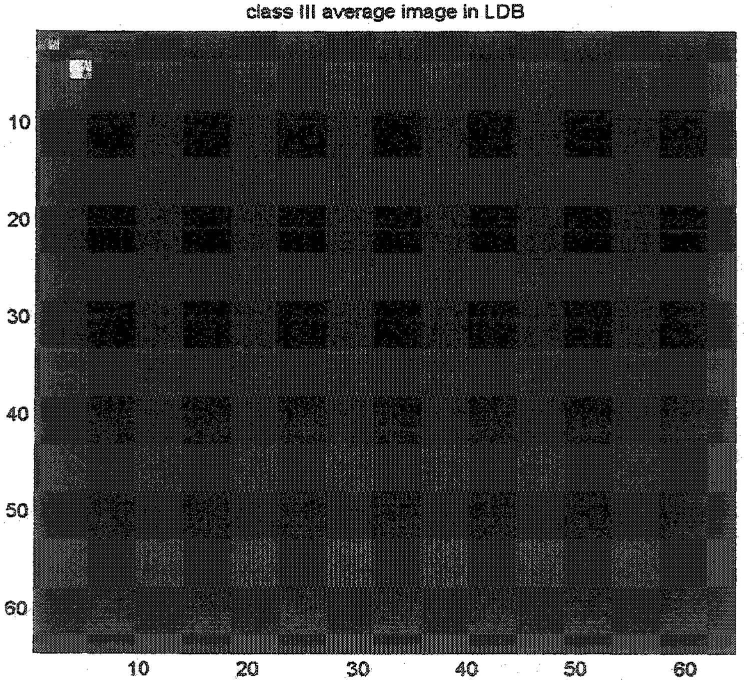


Fig.4.17 The average of class III in LDB domain

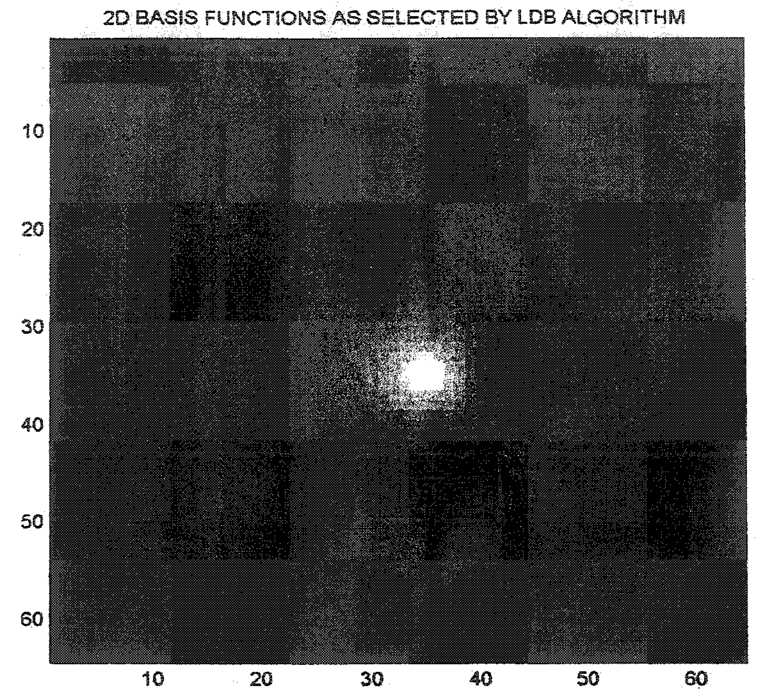


Fig.4.18 Most important basis selected by LDB algorithm

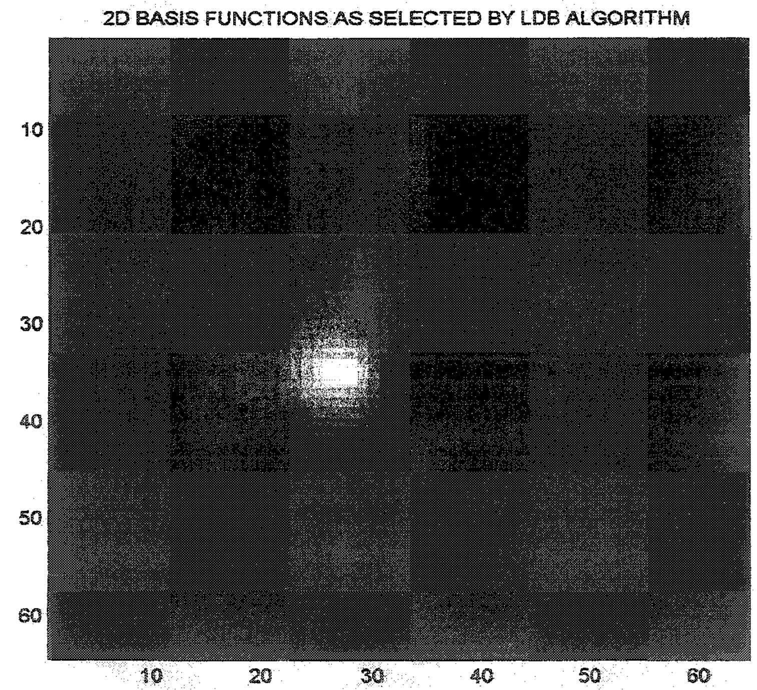


Fig.4.19 Second most important basis selected by LDB algorithm

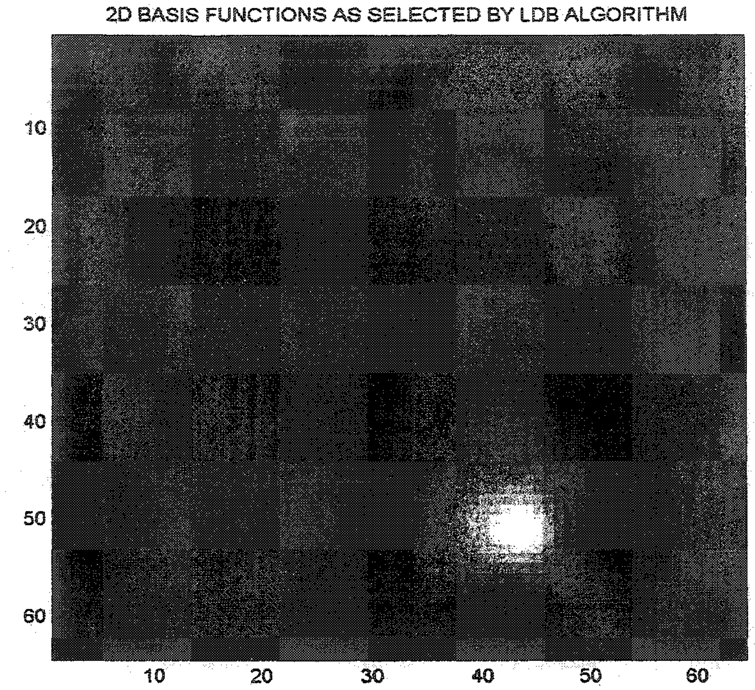


Fig.4.20 Third most important basis selected by LDB algorithm

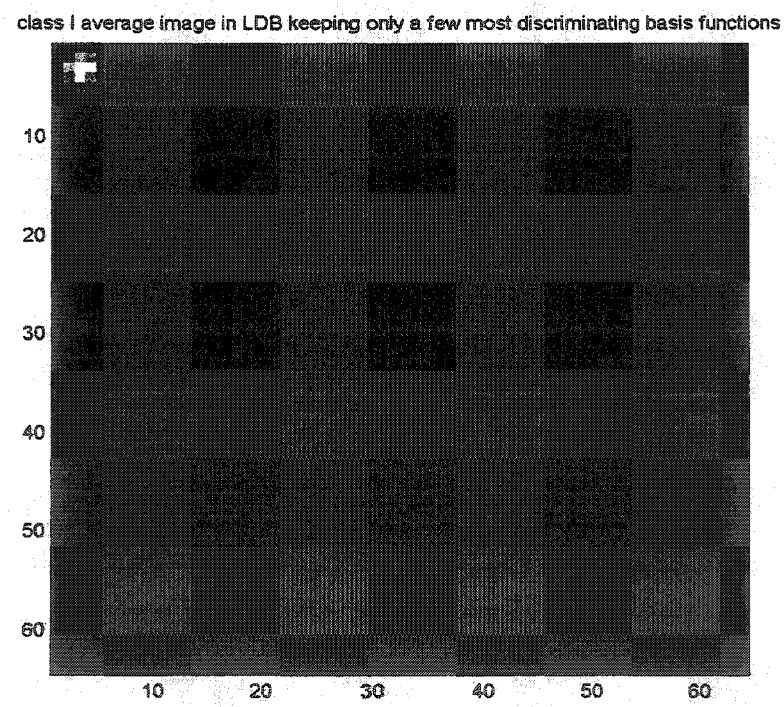


Fig.4.21 Class I average keeping only the first ten most discriminating features

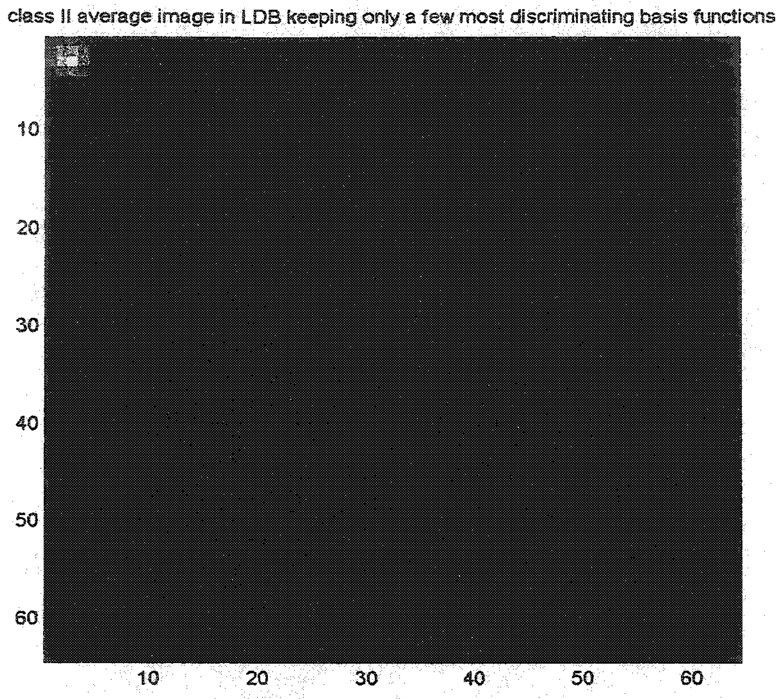


Fig.4.22 Class II average keeping only the first ten most discriminating features

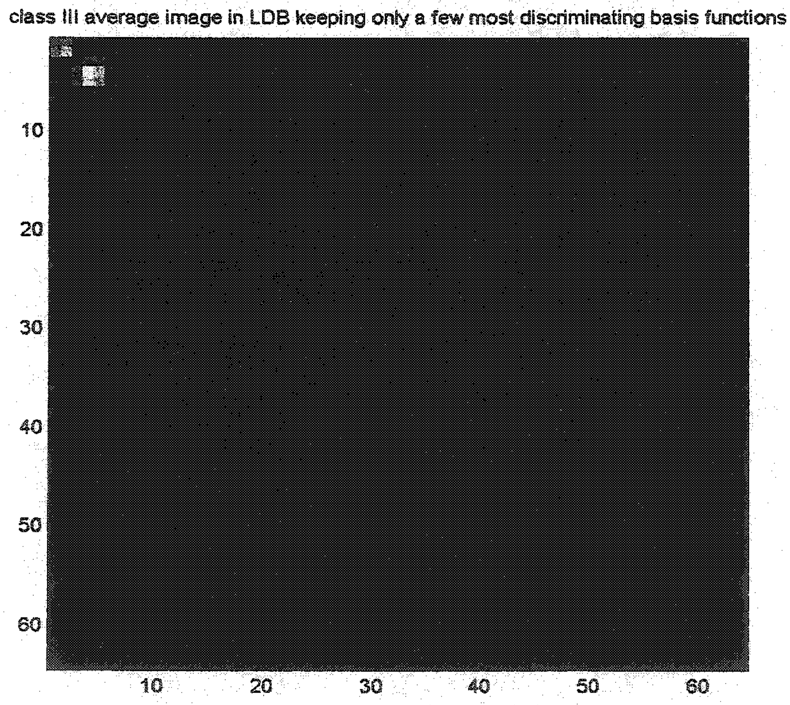


Fig.4.23 Class III average keeping only the first ten most discriminating features

Chapter Five

Improved Local Discriminant Basis Algorithm

In this chapter, two major challenges associated with local discriminant basis algorithm (LDB) are studied. The first challenge is due to intractability of signal distributions in the LDB algorithm when the particular coefficient distributions play the role of differentiating between classes rather than signal energies. In original LDB, which will be called LDB Type I in this chapter, only signal projection energies on basis functions of the dictionary under study are taken into account and compared later that makes the LDB method incapable of differentiating between certain distributions.

The second problem is due to translation invariance of wavelet packets and local trigonometric transforms [4] which remains to be the main drawback of techniques incorporating these dictionaries. As will be shown in this chapter, the decomposition into wavelet packet and local trigonometric dictionaries is not translation invariant. In case of Fourier transform, since the Fourier transform is a global integration, a translation in time or space domain does not affect the Fourier transform. However, wavelet packet and local trigonometric transforms are local transforms meaning that decomposition into these dictionaries will change if the input training and test signals are translated. This difficulty causes sever problems in LDB since it compares these decomposition maps across different signal classes. Therefore all training and test signals in different classes should be perfectly aligned, and that is not achievable in real applications. In this chapter solutions to overcome the translation invariance problem are presented [6], [41].

5.1 Pattern Probability Density Functions

It is possible to construct a simple classification problem that is intractable by original LDB algorithm. Suppose *Class 1* of input signals consists of a single basis function in a waveform dictionary with amplitude 10 buried in white Gaussian noise with zero mean and unit variance. *Class 2*, is the same as *Class 1*, except for the amplitude of waveform that is –10. Time-frequency energy distributions for the two classes as described in Chapter four are identical and LDB is unable to find the right discriminant coordinate. *Example 5.1* shows why this happens. As a matter of fact the problem is that all LDB Type I takes into account is the signal energy projected on a particular basis function in the dictionary not the actual coefficient distribution.

Example 5.1. The discrimination power of the basis function on which class coefficients are $\{p_i\}=\{1,2,3,4,5\}$and $\{q_i\}=\{-1,2,-3,4,-5\}$ can be calculated using Euclidean and symmetric divergence discrepancy measures (after normalization) as below.

$$H(\{p_i\},\{q_i\})=(\frac{1}{55}-\frac{1}{55})^2+\ldots+(\frac{25}{55}-\frac{25}{55})^2=0$$

$$J(\{p_i\},\{q_i\})=(\frac{1}{55}-\frac{1}{55})\log\frac{1/55}{1/55}+\ldots+(\frac{25}{55}-\frac{25}{55})\log\frac{25/55}{25/55}=0$$

Here the problem is that the LDB Type I algorithm described in Chapter four only takes the energy of projections into account not the actual distribution. This example together with the previous one taken from [37] suggests that it is sometimes necessary to consider the distribution of signal expansion coefficients for individual coordinates.

Here is a coordinate wise discrimination measure. Let's call the projection of input signal x onto a unit vector ω in the dictionary Z_ω. Then Z_ω^y is the projection of class y signals. The manner in which Z_ω is distributed for each signal class is important since it is used to quantify the discrimination measure of Z_ω. The original LDB algorithm measure is based on the differences of mean class energy of projection Z_ω. Other possibilities of such measures for discrimination are given next, which are based on distribution instead of energy.

- *Type II LDB*: A measure based on the differences among the probability distribution functions (pdf's) of Z_ω.

- *Type III LDB*: A measure based on the differences among the cumulative distribution functions (cdf's) of Z_ω.

- *Type IV LDB*: A measure based on the actual classification performance (*e.g.* a rate of correct classification) using the projection of the training signals.

Type II LDB has already been applied to real world applications [37]. The initial problem of Type II would be the estimation of pdf's from the available database. Suppose that p_ω^1 and p_ω^2 are the pdf's of *Class 1* and *Class 2* signals in ω direction respectively. Here are some possible discrimination measures proposed by Saito *et al.* [37], [56]:

- *Relative entropy* (or *Kullback-Leibler divergence*):

$$D(p_\omega^1, p_\omega^2) = \int p_\omega^1 \, log(p_\omega^1/p_\omega^2) \, dx, \tag{70}$$

- *Hellinger distance:*

$$H(p_\omega^1, p_\omega^2) = \int (sqrt(p_\omega^1)-sqrt(p_\omega^2))^2 dx, \tag{71}$$

where sqrt stands for square root and p_ω^i is the probability density function of class i projection on ω. Now assume that $B=\{\omega_1, \omega_2,\ldots, \omega_N\}$ is a basis. We use the summation of the discriminant measures of B individual vectors as the discriminant measure of basis B between *Class 1* and *Class 2* signals. Another possible approach is to sum only the discriminant measures that are larger than a certain threshold. The feasibility of Type III and Type IV LDB's is still under study [37].

The proposed methodology helps to better discriminate between classes of input signals. However, the initial estimation of probability density functions is not easy and its accuracy depends on the number of training signals. It is interesting to note that even in applications where the traditional projection energy based LDB (LDB Type I) has no difficulty discriminating between classes like classification of geophysical acoustic waveforms, incorporating probability density functions is a more realistic approach and helps to promote the efficiency of the algorithm [56]. For a recent study on the theory of LDB based on probability density functions and its application in classification of geophysical acoustic waveforms the reader is referred to [56].

5.2 The Problem of Translation Invariance

The lack of translation invariance is a major problem of the tree-structured dictionaries that are used for LDB divide-and-conquer algorithm. Suppose that the system is trained to classify the input signals into *Class 1* and *Class 2* with a synchronized training data. Now if the input to be classified is delayed about 10 samples, a wrong classification occurs, since the decomposition into wavelet packet and local trigonometric dictionaries is not translation invariant [4]. The problem is graphically shown in Fig.5.1 and Fig.5.2.

For some one-dimensional problems, that the signals are generally well synchronized, one can expand the training data in the following manner. Each training signal is translated over $-\tau,-\tau+1,\ldots,\tau-1,\tau$ periodically. This technique is known as *spin-cycle* method. This way the number of training data signals is multiplied by $2\tau+1$. The spin-cycle method is practical if τ is small. In general spin-cycle method is not practical for higher dimensions unless τ is small and a preferred direction for performing the translation operation is specified. An example is to translate training images along their diagonal. The following pre-processing step to align the training images has been proposed by Kronquist and Storm [39] for an image processing application.

- Train the algorithm on fixed position (time/space synchronized) data.
- When classifying, use some method (see below) to determine the position/localization of the interesting signal.
- When position determined, reposition the signal in the original (training) position and classify in the normal way.
- Finally translate the signal back.

This method tracks both the class and the synchronization of the signal. The method to determine the position of the signal is of course problem dependant. One possible method is to localize a specific feature in the class of signals, for example, the point with the maximum amplitude or zero crossing, or an easy to observe characteristic in a typical class. Then choose the correct translation by aligning the selected feature with the training data feature. If the noise is significant, and the selected LDB basis is highly dependent on small translations, two techniques can be combined by performing a $-\tau$, $-\tau + 1,\ldots,\tau - 1,\tau$ spin-cycle translation of the repositioned signals. Fig.5.3 is an example of this process for a one-dimensional signal depicted pictorially.

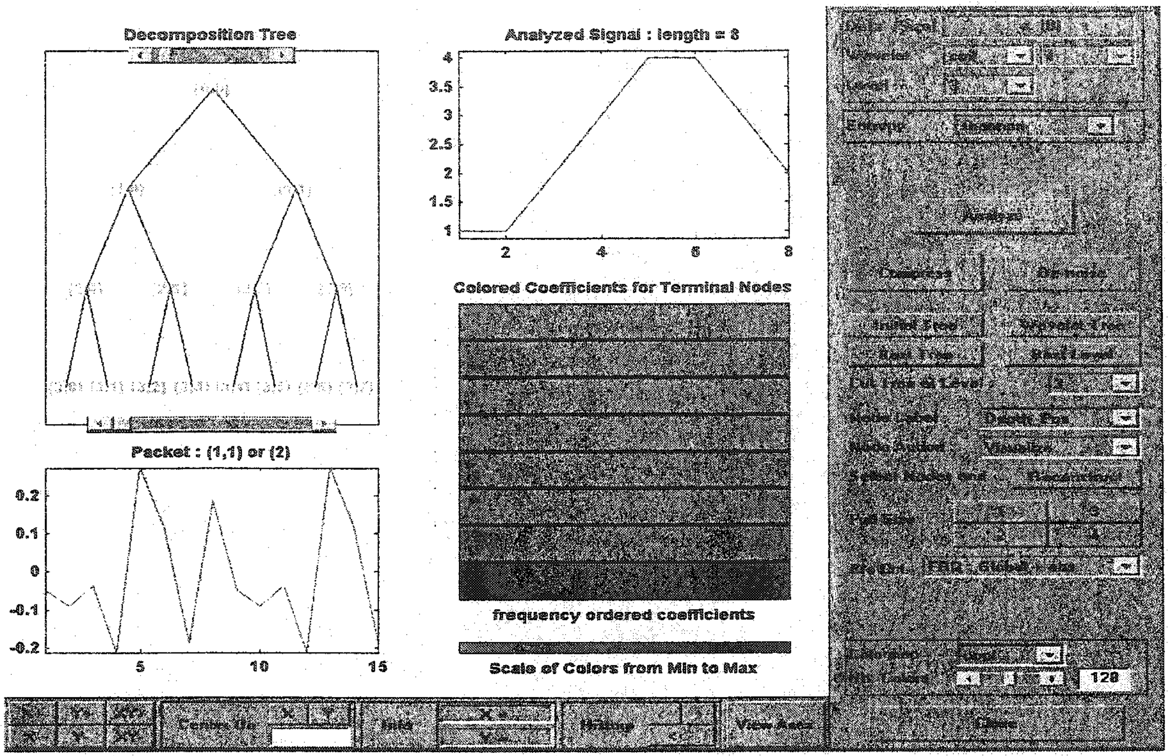


Fig.5.1 Projection into a basis function in the wavelet packet dictionary

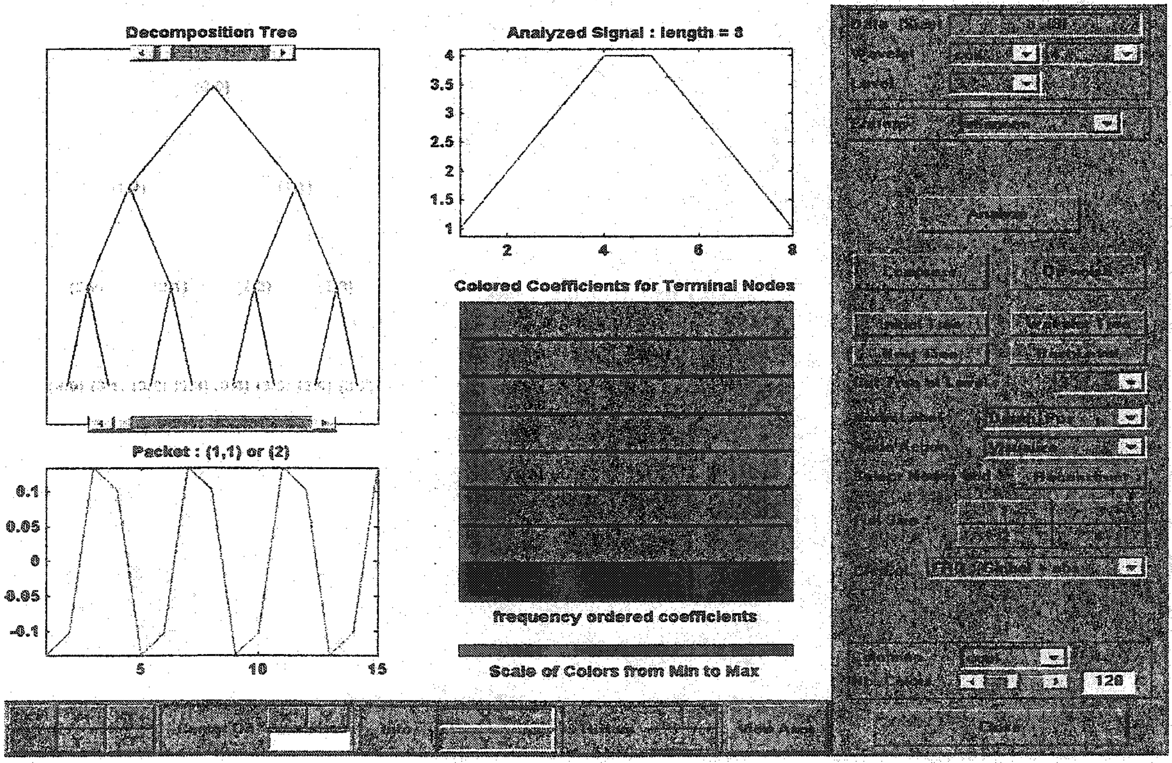

Decomposition Tree
Analyzed Signal : length = 8
4
3.5
3
2.5
2
1.5
1
2
4
6
8
Colored Coefficients for Terminal Nodes
frequency ordered coefficients
Scale of Colors from Min to Max
Packet : (1,1) or (2)
0.1
0.05
0
-0.05
-0.1
5
10
15
Analyze
Compress
De-noise
Close
128

Fig.5.2 The shifted version of the signal in Fig.5.1 projected on the same basis function

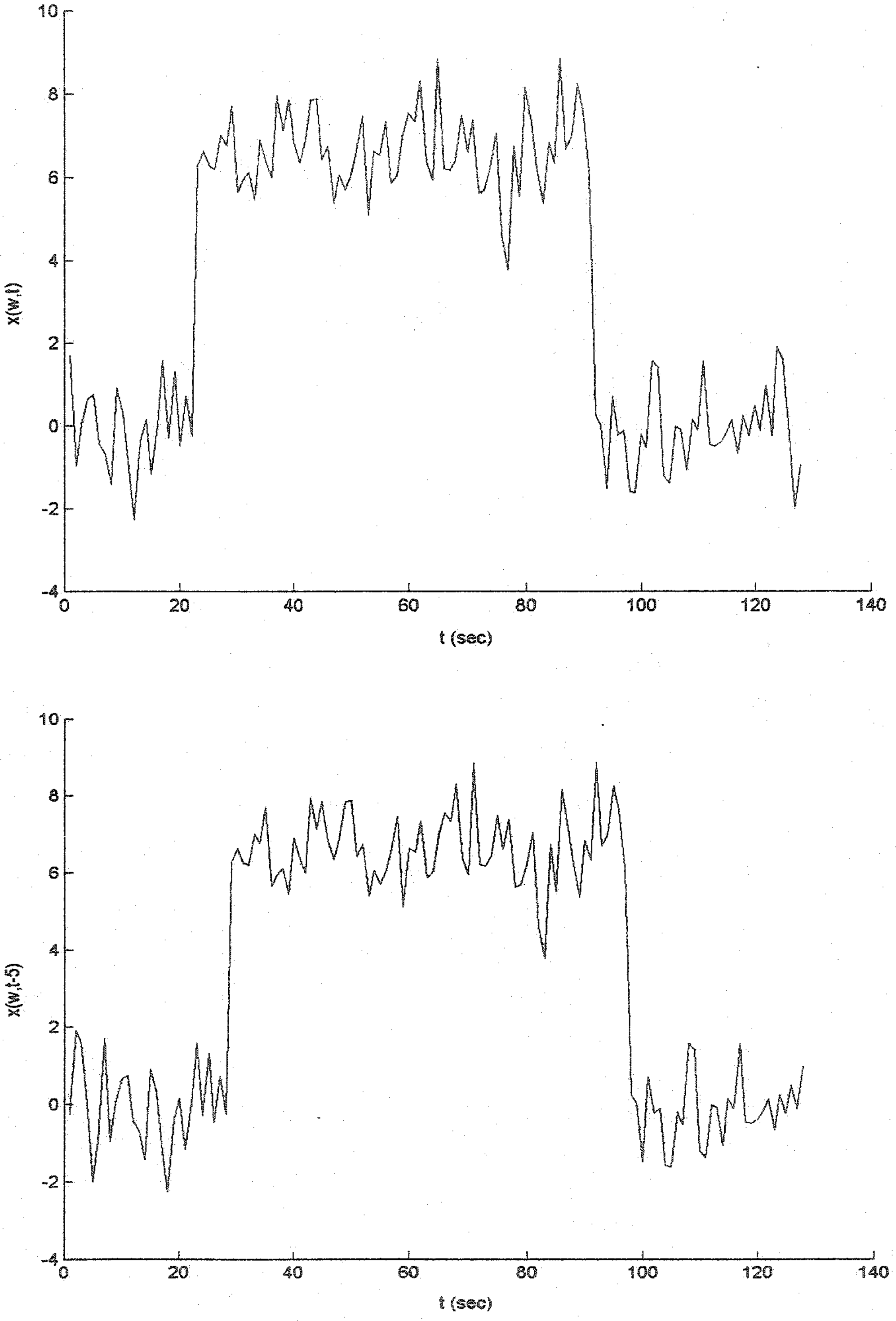


Fig.5.3 Spin-cycle method for overcoming the problem of translation invariance: the original training signal (above) and its periodically shifted version by five seconds (below)

Chapter Six

Applications of Local Discriminant Bases

In this chapter a number of applications reported in research literature using local discriminant basis (LDB) algorithm are presented. Each application area and the usage of LDB algorithm for signal classification are explained individually. There are many potential areas of research for application of LDB. The collection includes a number of applications for local discriminant basis as found in literature and some research reports and preprints. Local discriminant basis seems to work well for applications in which time and speed are the key factors, yet a high accuracy for classification problem is required. The LDB algorithm still in mathematical context needs to be studied and experienced in engineering applications. There seems to be a number of engineering applications for LDB algorithm. In particular, using the algorithm for determining the most discriminating features in biometrics and biomedical image processing is of high interest. Target detection in defense industry is one of the key applications of two-dimensional LDB algorithm. Reservoir detection in oil and gas industry has been the oldest area of application for one-dimensional LDB algorithm [6].

The origins of LDB method go back to the base basis algorithm that has proved useful in biomedical signal and image compression and specially a great tool for de-noising biomedical signals and images. The best results in terms of speed and accuracy belong to application of LDB method in ultrasound imaging and biometrics [44].

6.1 Target Detection in Sonar Images

Kronquist and Storm [39] have employed local discriminant basis for real-time detection of mines in sonar images. The technique introduced in Section 5.2 for pre-processing the images to align the training images is used to overcome the lack of translation invariance. Note that here the wavelet packet basis functions are two dimensional hence they constitute a quadratic tree.

Locating a target in a sonar image is a difficult problem. In a large digital image specifying the nature of the target and its position turns out to be a more difficult problem. The target like background in sonar images creates difficulties in the classification process. The classes to map the images to are targets and non-targets. The black tail and the target consist the features that separates target from background noise. The classifier takes into account both the structure of the mine and its tail.

In the training process Kronquist and Storm [39] used segments of the total image with size 64×64 pixels and all training targets are aligned at the same position. By running a local discriminant basis algorithm, the most separating vectors are chosen. The selected vectors contain the tail of the mine and the training is done with 33 images. If the training is done only with 15 images, the tail is not clearly selected that degrades the performance.

In a large sonar image, the image is decomposed into smaller 64×64 images and the classification is performed by first translating possible mines to the training position to handle the problem of translation invariance (Section 5.2). Kronquist and Storm [39] got Table.6.1 results by a set of 33 images as the training set. As the number of training signals is increased from 15 to 33, the tail of the target is selected in the LDB so it is expected that classification accuracy could further improve if more training signals were at hand.

Number of Training Targets	Missing a Target (%)	False Target Detection (%)
15	30	1.9
33	10	0.8

Table 6.1 LDB algorithm results used in target detection in sonar images

6.2 Classification of Geophysical Acoustic Waveforms

Acoustic measurements are used in geophysical well logging to infer petrophysical properties or the lithology of subsurface formations [7]. In sonic logging, an acoustic pulse is generated at the transmitter of a measurement tool lowered into a borehole. Then, the pulse propagates through the surrounding formations. Finally the pressure field is recorded at the receiver located in the upper portion of the same tool. A typical recorded waveform in digital format consists of three types of localized wave components: *P* wave (a refracted compressional wave), *S* wave (a refracted shear wave) and the *Stoneley* wave (a guided surface wave). The *P* and *S* waves follow paths that minimize the travel times between the transmitter and the receiver. The Stoneley wave, which is guided by the fluid-rock interface travels more slowly than the *P* and *S* waves and is the dominant event at later times in the waveform. Traditionally velocities of these three wave components (with or without their amplitudes) have been used to infer petrophysical/lithologic properties of the surrounding formations such as porosity, mineralogy, grain contacts, fluid saturation, volume percentage of various rocks such as sandstone, shale, and limestone, etc. see [7], [40] and references therein for the detailed physics behind these relationships. The waveforms contain more information than just the velocity and amplitude.

N. Saito [7] used 402 acoustic waveforms recorded in a certain well at various depth levels to train a local discriminant basis to classify the rock types of surfaces. The misclassification rate has been about about percent when using 20 LDB features in conjunction with LDA classifier [7]. For a recent treatment of this problem by LDB Type II algorithm, the reader is referred to [56].

6.3 Land use Classification of Synthetic Aperture Radar (SAR) Images

The basic problem in image processing arises here in determining land use from Synthetic Aperture Radar (SAR) images. Since the smallest 8×8 block in an image with 3 layers will result in a feature vector with 192 coordinates. Therefore a preprocessing method such as LDB is in place for such analysis. Rogers and Carolyn [41] used a Type II local discriminant basis for decreasing the dimensionality of their land-use determination problem. Sample image blocks with known land use are used for training phase. In the classification task however, the images are broken into blocks of the same size as the training blocks. The feature vector is formed from the top few most discriminant coordinates for each image plane. This way the dimensionality of the problem is reduced effectively. A maximum likelihood classifier was used for classification. Object-oriented software in C++ was used to compute the Type II local discriminant basis for 2D samples. The effects of changing different parameters such as the size of training blocks, the amount of overlap between extracted image blocks and the number of coordinates selected from the Type II LDB expansion was studied.

The land use classes correspond to buildings, grassy regions, tress and fields. After the training phase, they decomposed the images into 8×8 blocks overlapped by 6 pixels in both dimensions. The top one to three most discriminant coordinates were then fed into a maximum likelihood classifier for classification. Additional tests were run for block sizes of 4×4 and 16×16 with two and fourteen pixels overlaps respectively. The best results for 16×16 blocks were obtained when only one top coefficient was used from each plane. The classification results were satisfactory as reported by the authors [41].

6.4 Theater Missile Defense

Cassabaum *et al.* [42], [43] have examined the efficiency of LDB algorithm by using one-dimensional Infra Red (IR) sensor data. The objects consist of: a re-entry vehicle (RV), an associated object (AO), a booster, and burnt solid fuel (debris). The data had time-varying IR signature, which possesses unique spectral patterns. The fluctuations in signature intensity resulted from the angular motion of the objects and their projected area and/or temperature relative to the receiving sensor. Since the four objects underwent different impulsive forces according to their function and different inertial characteristics, the features were well separated among classes. Cassabaum *et al.* [42] used over 30,000 training signals. Test data set was 500 for each of the four classes. They classified the signals to two classes: RV and other. Wavelet packet dictionaries were constructed by using Daubechies 04 and Coifman 06 mother wavelets. Nearly 100% classification accuracy was obtained with either wavelet packet dictionary. In another experiment the performance of Fuzzy logic classifier together with LDB was studied [14].

6.5 Ultrasound Imaging of Apoptosis

C. Bejar and S. Krishnan [44] have studied the application of local discriminant basis algorithm in ultrasound imaging of apoptosis. Apoptosis or programmed cell death occurs during the morphogenesis, tissue renovation and in the regulation of immunity system [45], [46]. The cellular changes during apoptosis are monitored by ultrasound. Backscattered ultrasound can be used to characterize apoptosis [46].

C. Bejar and S. Krishnan [44] studied the performance of time-frequency distributions in classifying the tissues into normal and abnormal. Comparisons were performed with traditional autoregressive modeling and Cepstral analysis. LDB proved to be superior compared to traditional methods resulting in near 100% accuracy in determining the normal and abnormal tissue.

Chapter Seven

Implementation of Local Discriminant Basis Algorithm

In this chapter the implementation issues of local discriminant basis are studied. The local discriminant algorithm is implemented using the computational environment of Matlab®. A large number of subroutines and functions specific to Coifman and Wicherhauser best basis algorithm [4] are borrowed from WaveLab (version 8.02) libraries that are available as freeware from Stanford University [47]. However in order to implement the local discriminant basis algorithm a number of changes in best basis functions were necessary as explained in this chapter.

The best basis algorithm engine of Coifman and Wickerhauser can most easily be realized by using WaveLab (version 8.02) libraries. WaveLab is a library of Matlab® routines for wavelet analysis, wavelet packet analysis, cosine-packet analysis and matching pursuit. It contains over 1100 .m files, which are documented, indexed and cross-referenced in various ways. It is available free of charge through http://www-stat.stanford.edu/~wavelab and in UNIX, Macintosh and MS-Windows platforms. The Matlab® implementation of LDB algorithm is achieved in several steps and in a particular order that will be studied in detail in this chapter. A number of wavelet packet tables used inside the code are given in Appendix A as reference and their interpretation follows as soon as they appear in the implementation.

The very first one-dimensional LDB computational engine was provided for this thesis by computational mathematics research group at Yale University. Research performed in this thesis is based on the LDB algorithm implementation that Yale University has filed a patent application for as the basis for further software development.

7.1 Implementation of Local Discriminant Bases Algorithm in One Dimension

The step-by-step implementation of LDB algorithm in one dimension is studied in this section. Most of the functions used in this section are borrowed from 1D wavelet packet library of WaveLab.

7.1.1 Normalizing the Input Training Data

In this step, the input data is normalized to the total energy of the class it belongs to. This means that for each signal in a particular class, the total energy of the class is computed and the signal is divided by the total energy of the signals of the same class. By total energy of each class we mean the summation of the energies of individual training signals. This is an essential step since if signals in different classes are collected with different energies (as is the case in many practical situations); the difference in signal energy can impair the performance and accuracy of the algorithm. This will become clear when we explain how the average energy of class projections plays its role in LDB selection. Thus Class Norms are computed for each class and each input training data is divided by its corresponding Class Norm. Fig.7.1 related to the example in Section 4.7 shows why this is an essential step. The difference in class energies before normalization in Fig.7.1 can lead to wrong discrimination measure assignment as discussed in Section 4.5.

7.1.2 Calculating the Desired Quadrature Mirror Filter (QMF)

In the very beginning one should decide what type of wavelet packet or trigonometric transform to use. In this thesis, the focus is on wavelet packet implementation of LDB. In the first step the Quadrature Mirror Filter (QMF) corresponding to the selected mother wavelet was generated using "MakeONFilter" procedure, which is available through WaveLab libraries [47]. Search for LDB is also possible by looking among different wavelet packets and local trigonometric basis dictionaries. However, in this study and in all experiments performed in the thesis, the LDB selection scheme works within a pre-specified wavelet packet dictionary that one has to specify as the input to the Matlab® code.

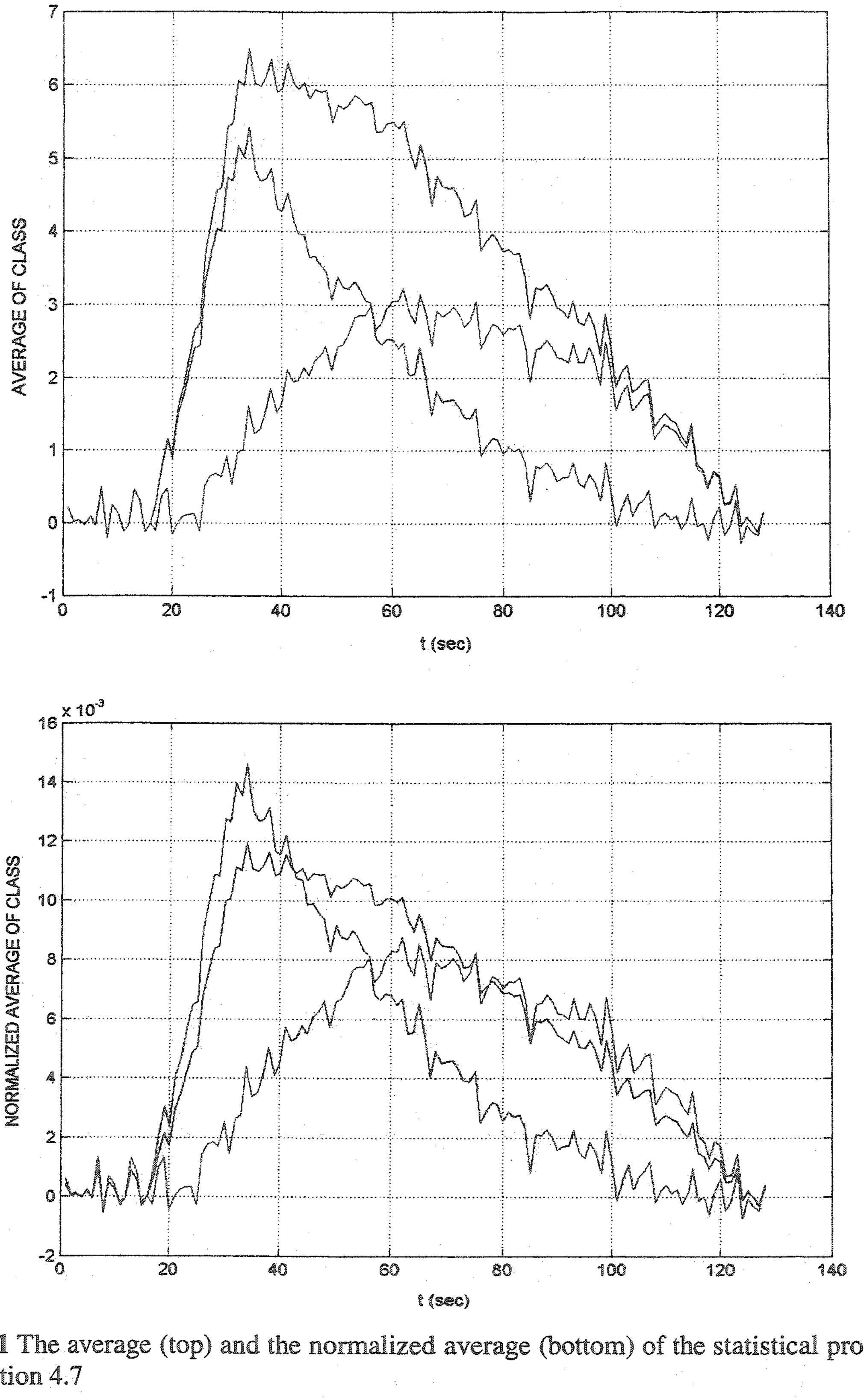


Fig.7.1 The average (top) and the normalized average (bottom) of the statistical process of Section 4.7

If other types of wavelets are needed that are not included in MakeONFilter routine, one can directly specify the QMF in a subroutine as done in this thesis. For other types of wavelets, the QMFs are produced by calling a subroutine directly specifying the coefficients of QMF as given in Wickerhauser's book [4].

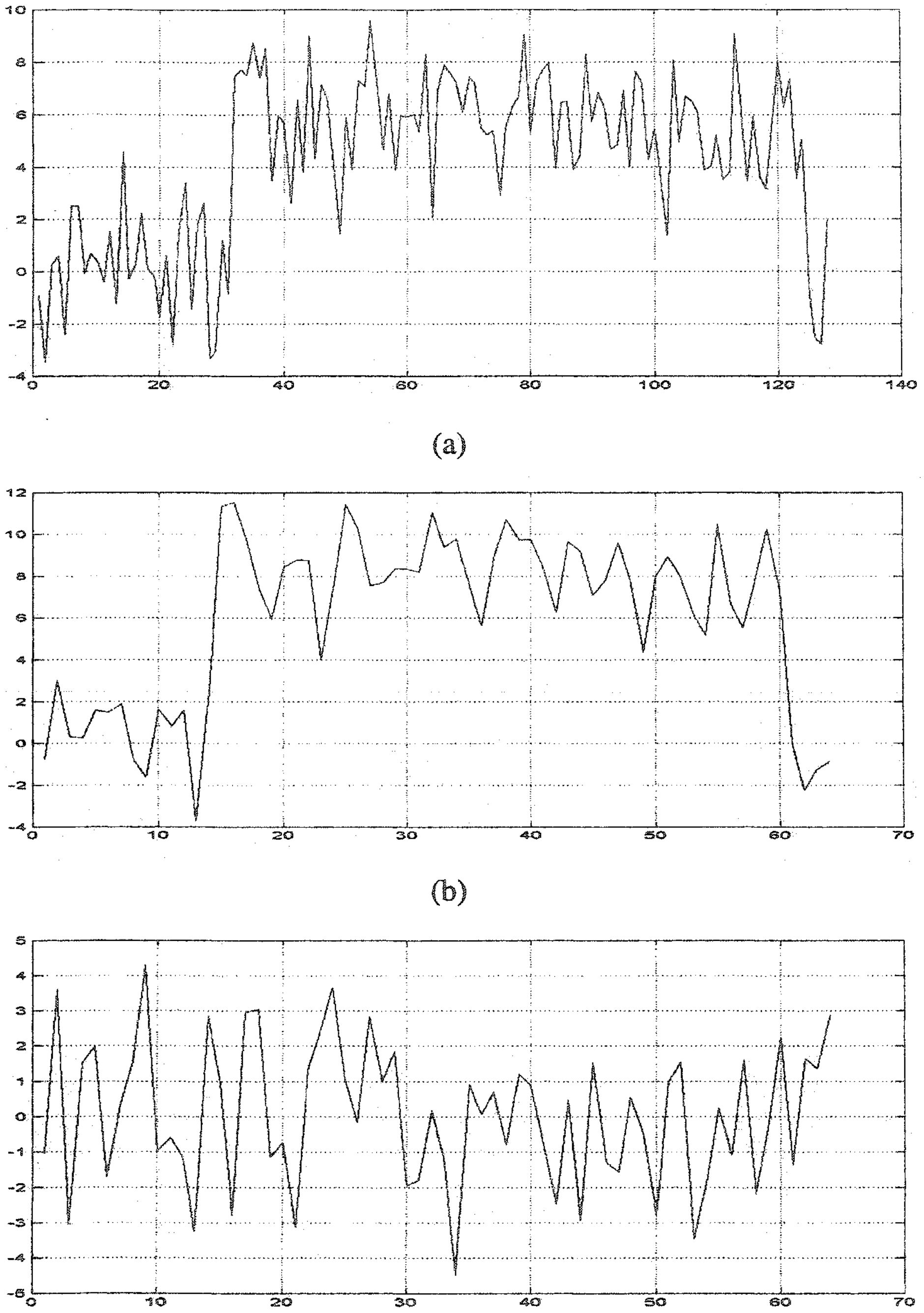
7.1.3. Computing Wavelet Packet Decomposition Maps

As soon as the normalized data and QMF are ready, another WaveLab function called "WPAnalysis" is used to compute the wavelet packet decomposition of input data [47]. Once the QMF is specified, "WPAnalysis" is capable of producing a complete decomposition of the input signal up to the specified level in the specified wavelet packet dictionary. Since at each level of decomposition the signal is decomposed into segments with half of the original size, the maximum meaningful level of decomposition for one-dimensional signal space would be $L = \log_2 N$ where N is the input signal length. At this level of refinement the individual signal coordinated are obtained as filtered signals. What one should expect as the output of the procedure is an $(L+1) \times N$ wavelet packet table.

Table A.1 in Appendix A, shows this decomposition for a signal. The correct interpretation of this table is important. The coefficients have different meaning at different levels. At lowest level (the leftmost column), one gets the input signal coordinates as the finest resolution. However the signal coefficients at each node of wavelet packet tree at depth *d* and at position *b* are given by "packet" function. Fig.7.2 explains the correct interpretation of coefficients in a $(L+1) \times N$ wavelet packet table that is vital to our understanding of how the Matlab® code works.

7.1.4 Computing Time-Frequency Energy Maps

Once the input data is decomposed into a wavelet packet dictionary, the projections onto individual wavelet packet basis functions are squared to give the energy of the decomposition on that particular basis function. Then the energy of the projections on a particular basis function for the signals in a specified class are summed up to give the time-frequency energy for that particular class and at that specific basis function. By repeating the operation for all the input signals and all the individual basis functions in the wavelet packet dictionary, one gets as many time-frequency energy maps as the number of classes. These are the objects that are further processed for the selection of the most discriminating basis functions. Table A.2 (Appendix A) shows these time-frequency energy maps for one class of signals generated in Section 4.7.

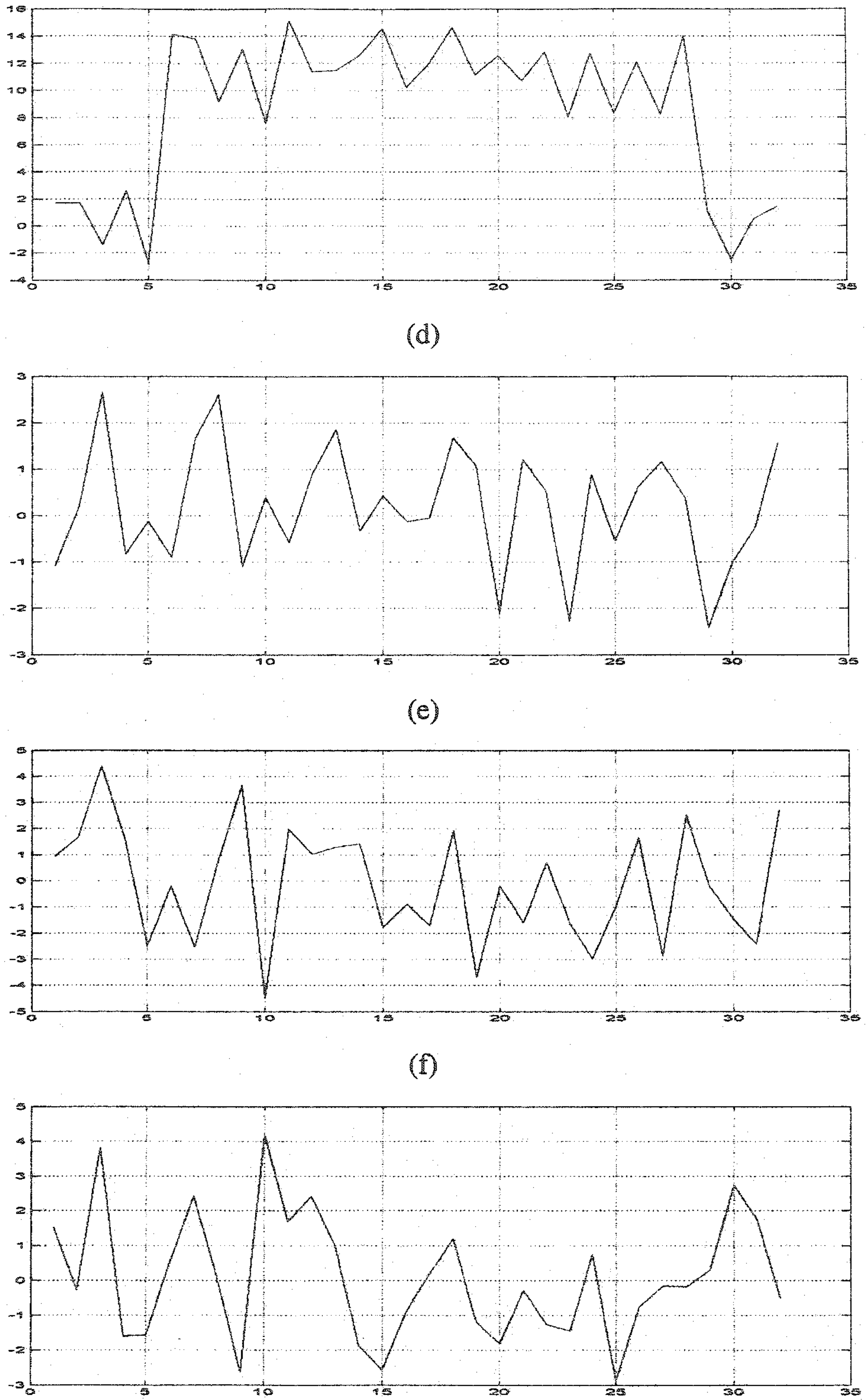


(a)

(b)

(c)

Fig.7.2 Interpretation of wavelet packet table for a sample class I function of statistical process in Section 4.7 (a) The first column of Table A.1 (Appendix A) is the signal itself (b), (c) the second column contains the coefficients of the signal as projected to basis functions on the second topmost level of binary tree (d), (e), (f), (g) third column contains the decomposition into third level basis functions in the wavelet packet dictionary



(d)

(e)

(f)

(g)

Fig.7.2 (continued)

7.1.5 Computing Cost Table

The time-frequency energy maps are representative of the strength or the level of energy of projections for each particular class. The energy maps are quantitative measures that help in transition from statistical signal space to some measure representative of the signal space. The *Cost Table* has a structure similar to wavelet packet dictionary. There is a number in the Cost Table corresponding to each individual position in a wavelet packet dictionary that relates to a specific basis function. This number is representative of the capability of the specified basis function in discriminating between different classes of input data.

In order to compute Cost Table tree one may use any of the discrepancy measures presented in Chapter Four. The number assigned to each individual position in the Cost Table is computed by plugging the time-frequency energies at the same position into the discrepancy measure. Table A.3 is a cost table computed for energy maps in Table.A.2 using symmetric relative entropy or *J*-divergence.

7.1.6 Selection of the Local Discriminant Basis

In this step Cost Table is the single necessary ingredient for choosing the most discriminating set of basis functions inside a wavelet packet dictionary. The children nodes at the lowest level of wavelet packet dictionary are compared to their parents. The Cost Table quantities corresponding to the children nodes are added and compared to that of their parent. This is the advantage of having an additive discrepancy measure that makes this comparison very fast with one addition and one comparison operation. This comparison is performed at each level of Cost Table (that correspond to levels in a wavelet packet dictionary). The locations that are selected correspond to the basis functions in the wavelet packet tree that have the best discrimination power. There is a build in function namely "BestBasis" in WaveLab that is used for detecting the best basis function of Coifman and Wicherhauser [4], [47]. This function is also available in Matlab® wavelet toolbox.

However, this function chooses the best basis for a compression problem. To be brief, it looks for minimum entropy whereas in a classification application, maximum discrimination power is sought. So the selection criteria in this function is reversed and is stored as a different function called "WorstBasis." Here are the "BestBasis" and the altered "WorstBasis" version used in our experiments. Fig.4.4 shows different steps in selecting the LDB and the final LDB is given in Fig.4.5 as examples of the LDB basis selection process.

BestBasis Function

```
function [basis,value] = BestBasis(tree,D)
basis = zeros(size(tree));
  value = tree;
  for d=D-1:-1:0,
    for b=0:(2^d-1),
                vparent = tree(node(d,b));
                vchild  = value(node(d+1,2*b)) + value(node(d+1,2*b+1));
                if(vparent <= vchild),
                        basis(node(d,b)) = 0;
                        value(node(d,b)) = vparent;
                else
                        basis(node(d,b)) = 1;
                        value(node(d,b)) = vchild;
                end
    end
   end
```

WorstBasis Function

```
function [basis,value] = WorstBasis(tree,D)
basis = zeros(size(tree));
  value = tree;
  for d=D-1:-1:0,
    for b=0:(2^d-1),
                vparent = tree(node(d,b));
                vchild  = value(node(d+1,2*b)) + value(node(d+1,2*b+1));
                if(vparent >= vchild),
                        basis(node(d,b)) = 0;
                        value(node(d,b)) = vparent;
                else
                        basis(node(d,b)) = 1;
                        value(node(d,b)) = vchild;
                end
    end
  end
```

7.1.7 Ordering the LDB Functions According to Their Discrimination Power

At this stage, the LDB functions are known. However, the main idea of LDB is to further reduce the dimensionality of the problem. To this end, the basis functions are sorted according to their discrimination power and only a small complete subset of the best discriminating basis functions are chosen as most discriminating features. This is done by decomposing the Cost Table into the selected basis by "WorstBasis" and then ordering the basis functions according to the projections of the Cost Table on this basis. Decomposition into the selected basis is done by "UnpackBasisCoeff" that is another Wavelab function [47].

7.1.8. Identifying the LDB Vectors

Having selected the LDB, its vectors are to be introduced next. These are functions corresponding to the selected locations in wavelet packet dictionary. One can compute the LDB coefficients of any signal by first getting its wavelet packet decomposition by "WPAnalysis" and then using "UnpackBasisCoeff" to get signal coefficients in LDB. In addition to this technique, one can directly compute the coefficients in a basis inside in a wavelet packet dictionary by using "IPT_WP" function. Having the LDB basis functions as vectors is more practical since the conversion into LDB domain can be done more straight forward by a matrix multiplication that performs a change of basis into LDB domain. This is simply a matrix whose columns are LDB vectors. LDB vectors corresponding to the LDB for the problem given in Section 4.7 are depicted in Fig.7.3.

One way to calculate LDB vectors is to convert their representation in LDB domain to usual Euclidean coordinates. An LDB function corresponds to an object either a vector (1D) or a matrix (2D) with a 1 at the location of the LDB function and zero elsewhere. This representation can be converted to Euclidean domain using "FPT_WP" function in WaveLab.

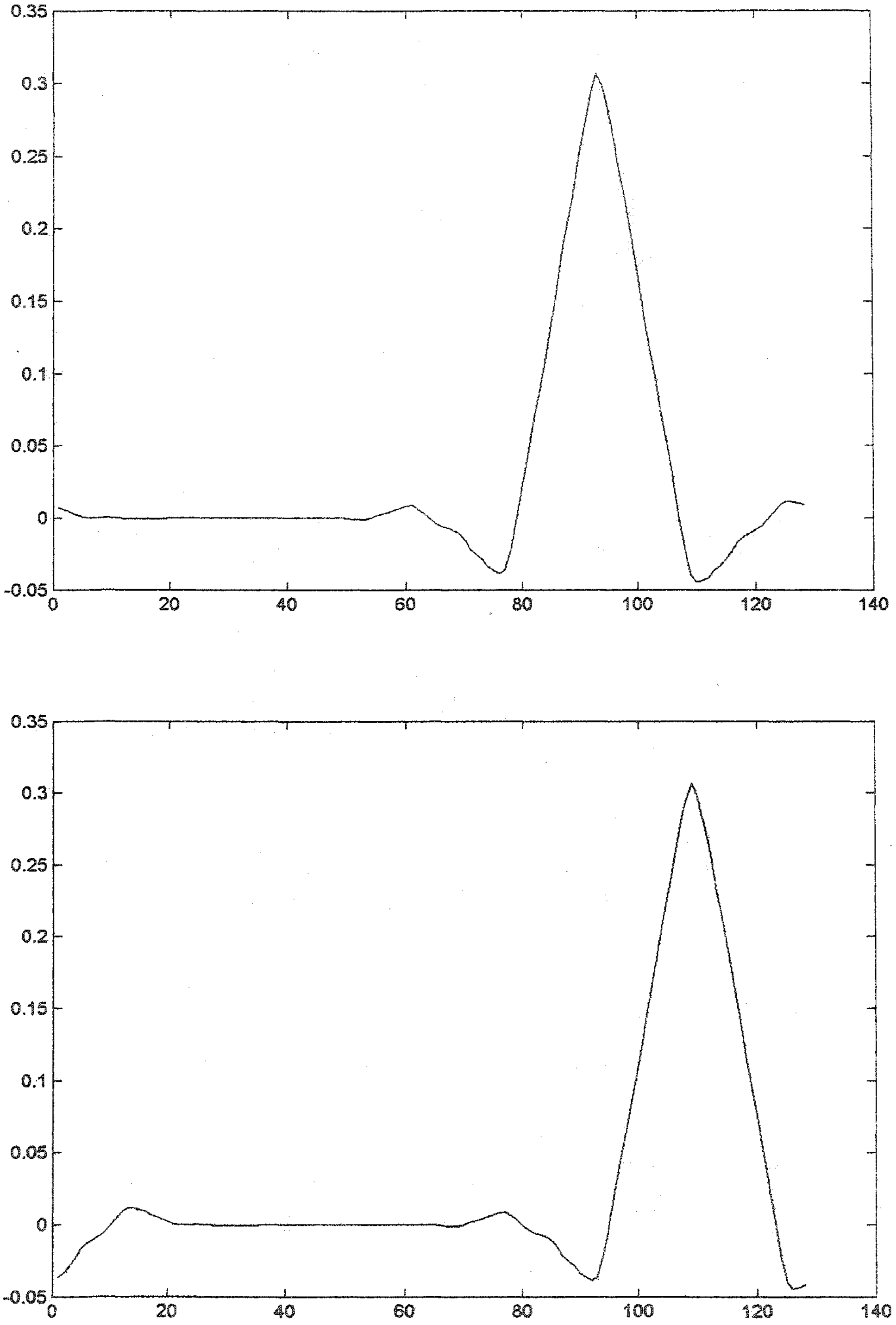


Fig.7.3 Four most discriminating features of statistical process in Section 4.7 as obtained by LDB algorithm; the features are ordered from top to bottom according to their discrimination power

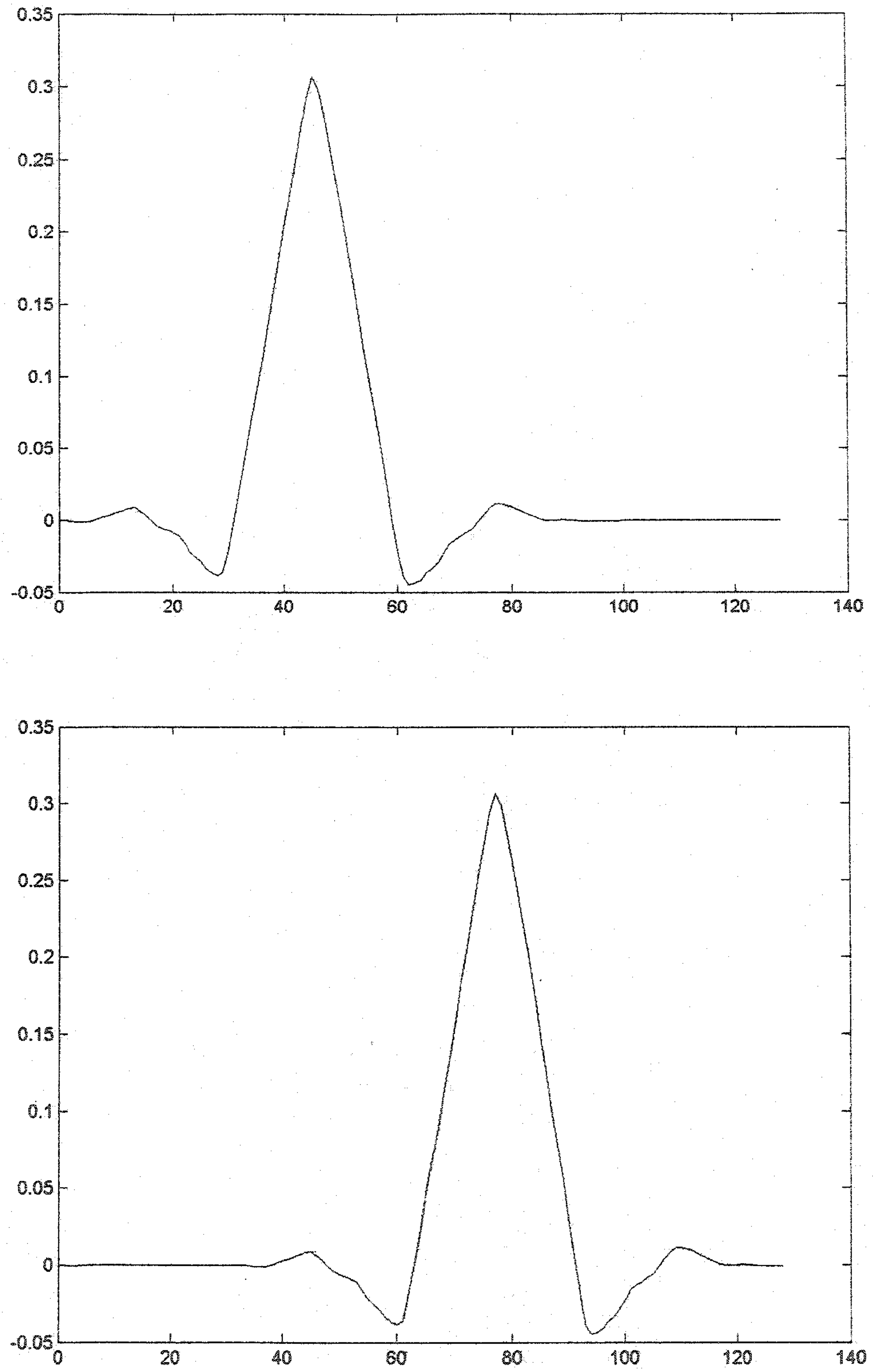


Fig.7.3 (continued)

7.2 Implementation of Local Discriminant Bases Algorithm in Two Dimensions

The implementation issues of a search for LDB in 2D images as compared to 1D signals are considered in this Section. The major difficulty is in locating the wavelet packet basis functions inside the huge wavelet packet decomposition that constitutes a quadratic tree this time. All the required steps to implement 2D LDB classifier are discussed in detail.

A 2D classification experiment is implemented in Matlab® by converting the previous 1D code to 2D time-frequency analysis. The translation into 2D time-frequency analysis is straightforward except for the computations involving quadratic tree structure. However, the built in routines inside WaveLab help for the major part of calculations. Approximately every function that is defined for 1D wavelet packet or local trigonometric transform has a 2D counterpart in WaveLab. In the following the same steps as in 1D LDB algorithm is followed and the differences are explained. Having understood the major steps in the method as described in Chapter four and Section 7.1 translating the Matlab® implementation from 1D to 2D case is straight forward.

7.2.1 Normalizing the Input Training Data

In this step, the input data is normalized to the total energy of the class it belongs to. Here the input images are in the form of matrices whose entries specify the brightness of the corresponding pixel in the image. The norm of each image class is computed by averaging the norm of its members. Norm of a matrix is defined to be the square root of the summation of all its entries squared.

7.2.2 Calculating the Desired Quadrature Mirror Filter

Initially one should decide on the type of wavelet packet or trigonometric transform to use. In this thesis, the focus is on wavelet packet implementation of LDB. In the first step the quadrature mirror filter (QMF) corresponding to the selected mother wavelet was generated using "MakeONFilter" procedure, as described in 7.1.2.

7.2.3 Computing Multi-Resolution Wavelet Packet Decomposition Images and Ordering

When the QMF generator of the wavelet packet is known, one can exploit the "Calc2dPktTable" function, which is a WaveLab function to produce wavelet packet tables [47]. This would be a two dimensional object that needs to be interpreted carefully. As seen in Chapter two, in 2D case, the decomposition into wavelet packet dictionary yields matrices at different levels of refinement. In image processing context this is known as multi resolution analysis. Every step toward the bottom of the wavelet quadratic tree results in a new image with a different level of resolution. However projections into individual wavelet packet basis functions are included within this refined image. Let's assume that the images to be classified are $N \times N$. If one reshapes every matrix that is obtained at different resolutions levels into a $N^2 \times 1$ vector and then sets the vectors side by side together, the wavelet packet quadratic tree coefficients will make a $N^2 \times (L+1)$ matrix, where $L = \log_4 N$ is depth of the tree. It is important to note that this is just an intermediate implementation technique.

The wavelet packet table as described above proves to be a very handy and useful object for wavelet packet computations basically reducing the three-dimensional array processing to that of 2D matrix calculations. Nevertheless, "Calc2dPktTable" as it originally comes with WaveLab library, normalizes input data before any further processing. The result is the wavelet packet corresponding to normalized input data. In LDB algorithms computations, one only needs to normalize data by its class energy and the normalizing feature in "Calc2dPktTable" causes some difficulties. For this reason, the initial normalization step in "Calc2dPktTable" is omitted, which explains why our wavelet packet tables in Appendix A are computed slightly differently.

Table A.4 shows this decomposition for images in class I of the example in Section 4.8, which has been truncated due to space limit. This table needs to be interpreted carefully. The coefficients have different meanings at different levels. At lowest level, one gets the input image coordinates reshaped into a $N^2 \times (L+1)$ vector as the finest resolution. However the position of the image coefficients at each node of wavelet packet tree for a basis function at depth *d* and at position *bx* and *by* are given by "quadbounds" function [47]. Fig.7.4 explains the correct interpretation of coefficients in a $N^2 \times (L+1)$ wavelet packet table that is of ultimate importance to our understanding of how the Matlab® code works.

7.2.4 Computing Time-Frequency Energy Maps

Once the input data is decomposed into a wavelet packet dictionary, the projections onto individual wavelet packet basis functions (quadlets [4], [47]) are squared to give the energy of the decomposition on that particular basis function. This means that the norm of the matrix obtained as the image entry at a particular position of the wavelet packet tree is going to be used as the energy of the signal at that position. Then the energy of the projections on a particular basis function for the signals in a specified class are summed up to give the time-frequency energy for that particular class and at that specific basis function. By repeating the operation for all the input signals and all the individual basis functions in our wavelet packet dictionary, one gets as many as time-frequency energy maps as the number of classes. Table A.5 presents a sample of time-frequency energy maps as a result of decomposing images from three different classes.

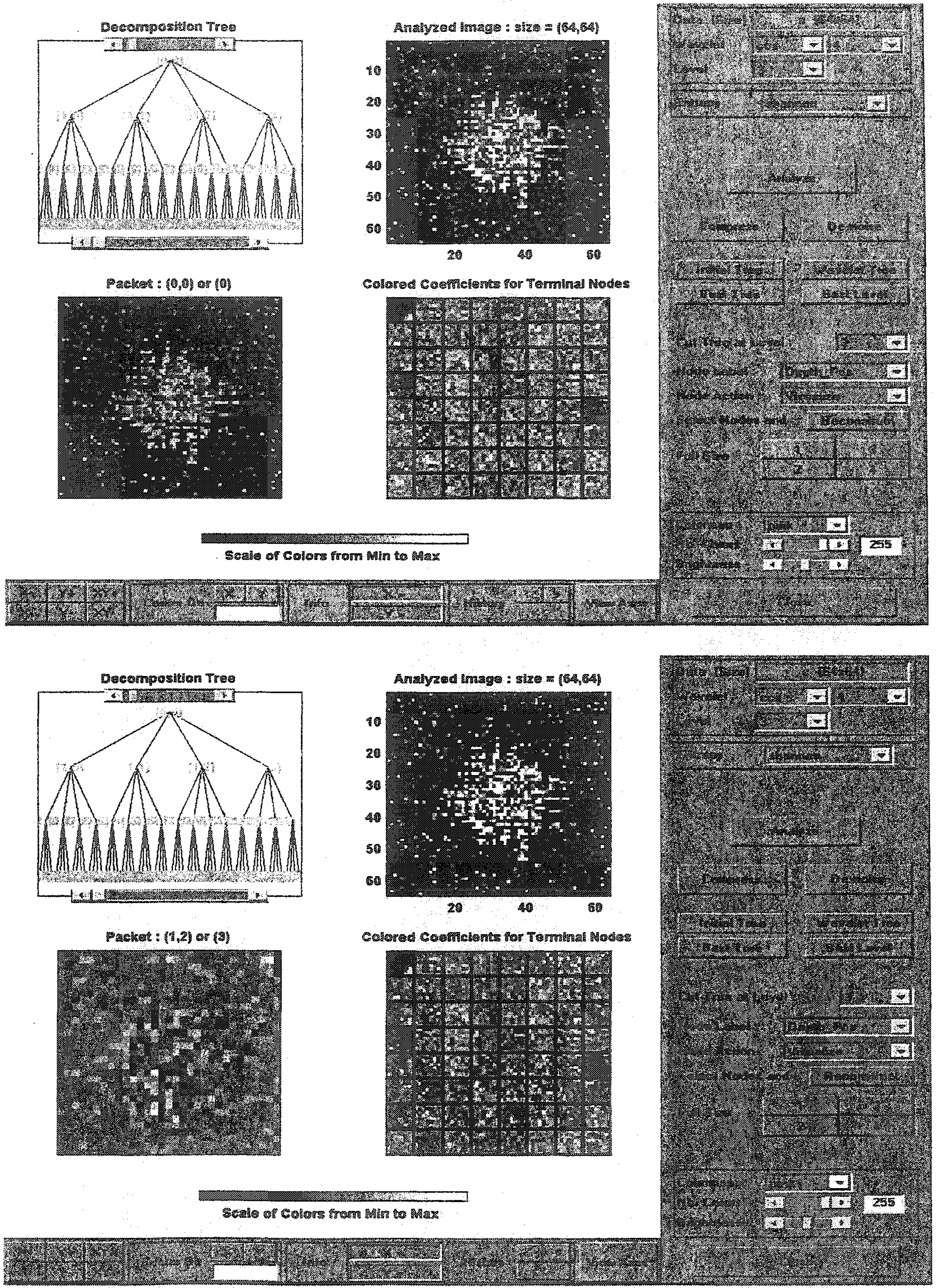
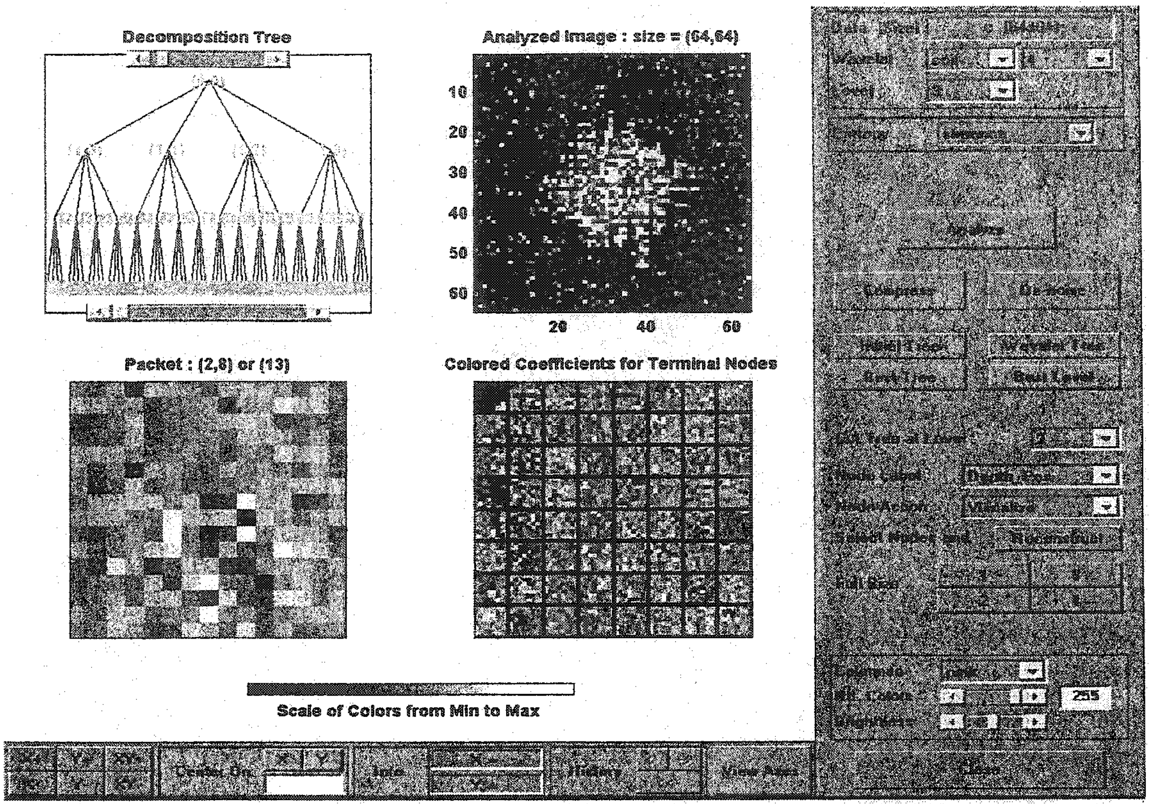

Decomposition Tree
Analyzed Image : size = (64,64)
10
20
30
40
50
60
20
40
60
Packet : (0,0) or (0)
Colored Coefficients for Terminal Nodes
Scale of Colors from Min to Max
255
Decomposition Tree
Analyzed Image : size = (64,64)
10
20
30
40
50
60
20
40
60
Packet : (1,2) or (3)
Colored Coefficients for Terminal Nodes
Scale of Colors from Min to Max
255

Fig.7.4 Image decomposition at different nodes of quadratic 2D wavelet packet tree



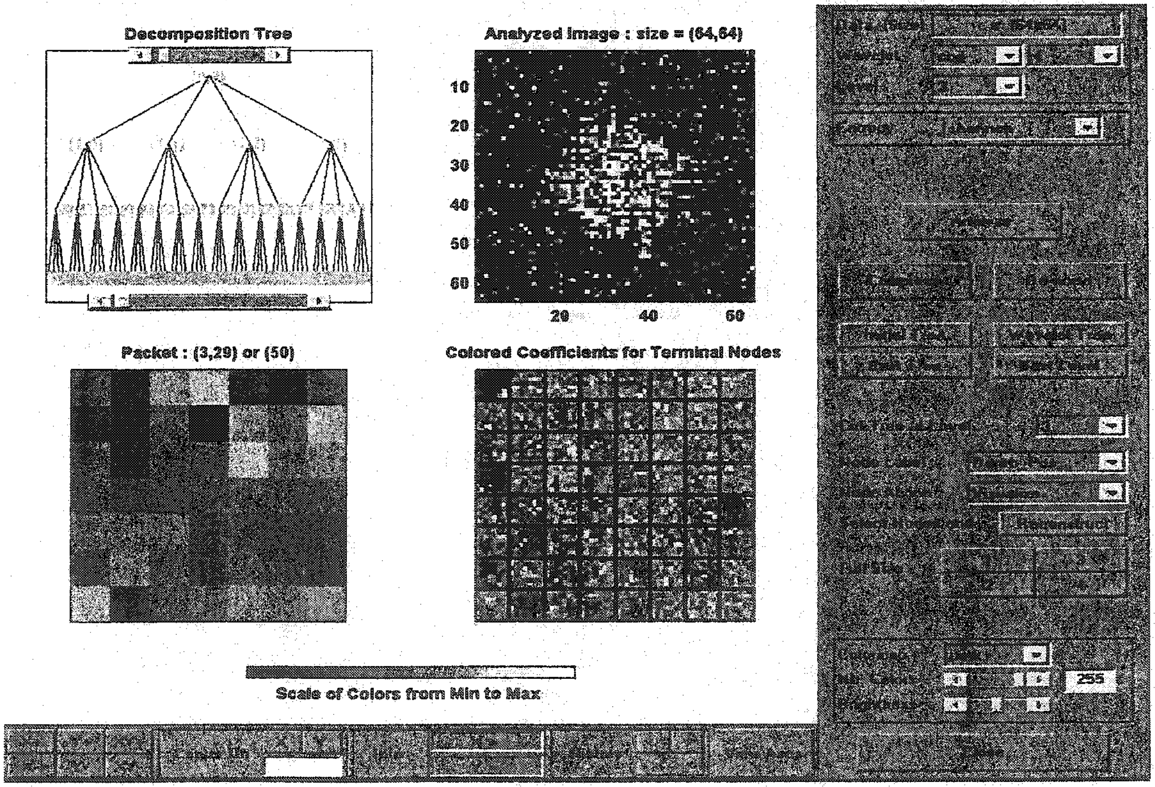

Decomposition Tree
Analyzed Image : size = (64,64)
10
20
30
40
50
60
20
40
60
Packet : (3,29) or (50)
Colored Coefficients for Terminal Nodes
Scale of Colors from Min to Max
Reconstruct
255

Fig.7.4 (continued)

7.2.5. Computing Cost Table

In this step there is nothing special about images compared to 1D functions. In a quadratic tree each parent has got four children nodes that one needs to compare their discrimination power to their parent's discrimination power. Table A.6 in Appendix A is a cost table computed for energy maps in Table A.5 using symmetric relative entropy or *J*-divergence (Chapter four).

7.2.6 Selection of the Local Discriminant Basis

This step is very much the same as 1D case. There is a build in function namely "Best2dBasis" in WaveLab [47] that is used for detecting the best basis function of Coifman and Wicherhauser [4].

However, this function chooses the best basis for a compression problem. To be brief, it looks for minimum entropy whereas in classification applications, maximum discrimination power is sought. Therefore one has to reverse the selection criteria in this function, which is stored as a different function called "Worst2dBasis." To see how this is done refer to section 7.1.6.

7.2.7. Ordering the LDB Functions According to Their Discrimination Power

In order to sort the basis functions according to their power in discriminating between images from different classes one has to sort the Cost Table. This is done by decomposing the Cost Table into the selected basis by "WorstBasis" and then ordering the basis functions according to the projections of the Cost Table on this basis. Decomposition into the selected basis is done by "Unpack2dBasisCoeff" that is another WaveLab function [47].

7.2.8. Identifying the LDB Matrices

Similar to one-dimensional case, we are interested in getting the 2D wavelet packet basis functions as matrices. LDB matrices corresponding to the LDB for the problem in Section 4.8 are depicted in Fig.7.5.

One solution to calculate LDB matrices is to convert their representation in LDB domain to usual Euclidean coordinates. An LDB function corresponds to an object either a vector (1D) or a matrix (2D) with a 1 at the location of the LDB function and zero elsewhere. This representation can be converted to Euclidean domain using "FPT2_WP" function in WaveLab [47].

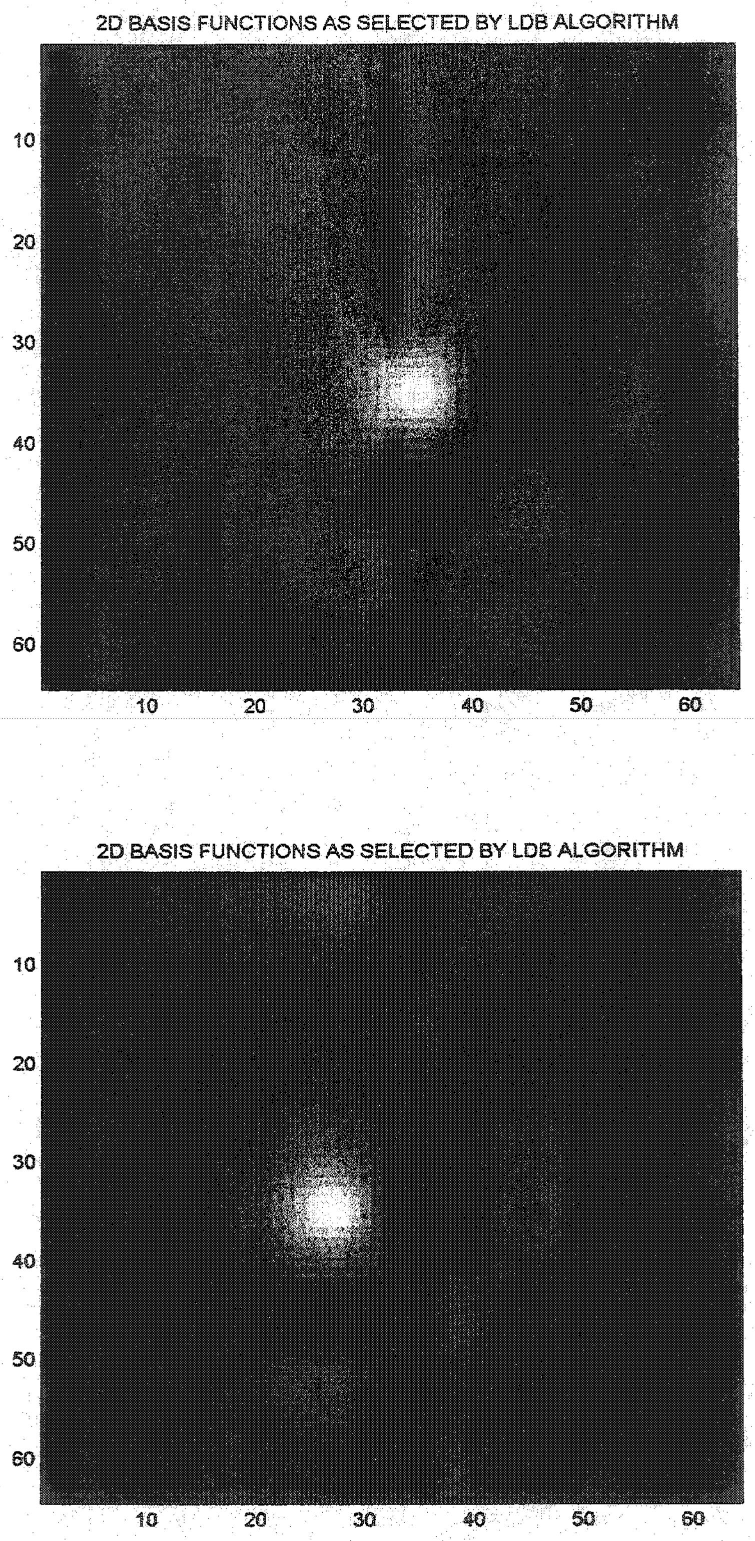


Fig.7.5 Most discriminating LDB basis functions ordered from top to bottom according to their discrimination power

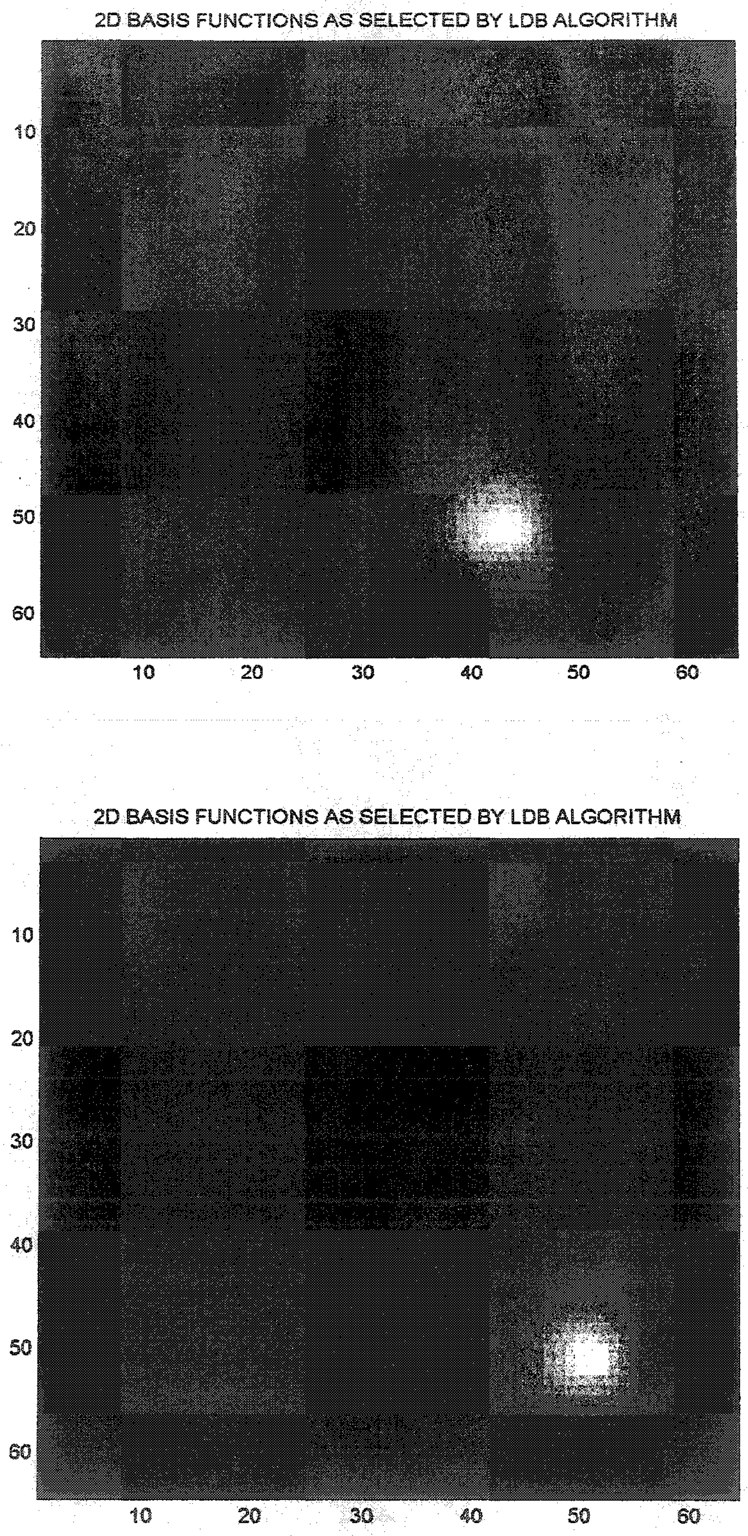


Fig.7.5 (continued)

Chapter Eight

Optimally Weighted Local Discriminant Basis Algorithm (OLDB)

In this chapter an optimization method is proposed to further improve the accuracy of the algorithm by adding an optimization block to promote the discrimination measure of the local discriminant basis (LDB) chosen. This strategy is useful specially when one intends to keep the number of selected features below a certain value. This feature boosting is necessary when background data has strong correlation with regions of interest in signal space under study. Yoshida [38] has employed the same technique for improving the performance of matching pursuit method [11] for extraction of microcalcifications from mammograms.

It was observed that when the number of selected features is below the minimum required number of features to stabilize the misclassification rate as obtained by LDB scheme; a further drop in misclassification rate is possible by altering or in a way optimizing the selected features. These observations encouraged us to further pursue the idea of optimizing local discriminant basis. The first optimization strategy undertaken and implemented is the steepest decent algorithm. The original idea of weighting the selected basis came from [38] where an optimal weight vector was incorporated to improve matching pursuit algorithm for classifying mammograms to benign and malignant cases. The weights are used to reduce the difference of the signals in every class with their corresponding class averages in the neighborhood of a small region of interest. The process proved to be useful in conjunction with Fisher's Linear Discriminant Analysis (LDA) [33]. Studying the distribution of features before and after optimization block justifies this approach. Note that the optimization block efficiency is highly dependant on the type of classifier used to classify LDB extracted features. The proposed scheme is particularly applicable in the context of image processing and in applications, where regions of interest can be specified, *i.e.* target detection or mammography. Some observations based on trial and error experiments are presented first which make the ground for further implementation and analysis of the proposed optimization block. The theory of optimized local discriminant basis algorithm (OLDB) is developed in Section 8.2 and Section 8.3. OLDB simulation results in synthetic signal and image processing examples of Section 4.7 and Section 4.8 are given in Section 8.4 and Section 8.5. In Section 8.6 the performance of OLDB is tested in audio signal classification that to our knowledge is the first application of LDB-based algorithms in this area. A practical texture classification problem is studied in Section 8.6 by using OLDB.

8.1 Preliminary Observations

There are examples where it can easily be seen that changing the LDB vectors manually in trial and error fashion results in improvements in misclassification rate. Some trial and error experimental observations follow.

Example 8.1. This example taken from [6] shows how altering the selected feature vectors would alter the misclassification rate. Three classes of signals are generated by the process given in (72) that was studied in Section 4.7. The reader is referred to Section 4.7 for more details about the architecture of the statistical process given in (72).

$$\begin{aligned}
c(i) &= (6+\varsigma).\chi_{[a,b]}(i) + NoiseLevel.\varepsilon(i)\\
b(i) &= (6+\varsigma).\chi_{[a,b]}(i).\frac{i-a}{b-a} + NoiseLevel.\varepsilon(i)\\
f(i) &= (6+\varsigma).\chi_{[a,b]}(i).\frac{b-i}{b-a} + NoiseLevel.\varepsilon(i)
\end{aligned} \tag{72}$$

"a" is an integer-valued uniform random variable on the interval [16,32]. "$b-a$" also obeys an integer-valued uniform distribution on [32,96]. "ς"and "ε" are the standard normal variates, and $\chi_{[a,b]}(i)$ is the characteristic function on $[a,b]$. $c(i)$ is called the 'cylinder' class whereas $b(i)$ and $f(i)$ are known as 'bell' and 'funnel' classes. In this study i varied between 1 and 128. Samples of signals generated in different classes are given in Fig.8.1 with SNR = 9. The statistical averages of each class generated in the process above over a set of 360 training signals are depicted in Fig.4.7. LDA [33] was fixed as the classifier in this study.

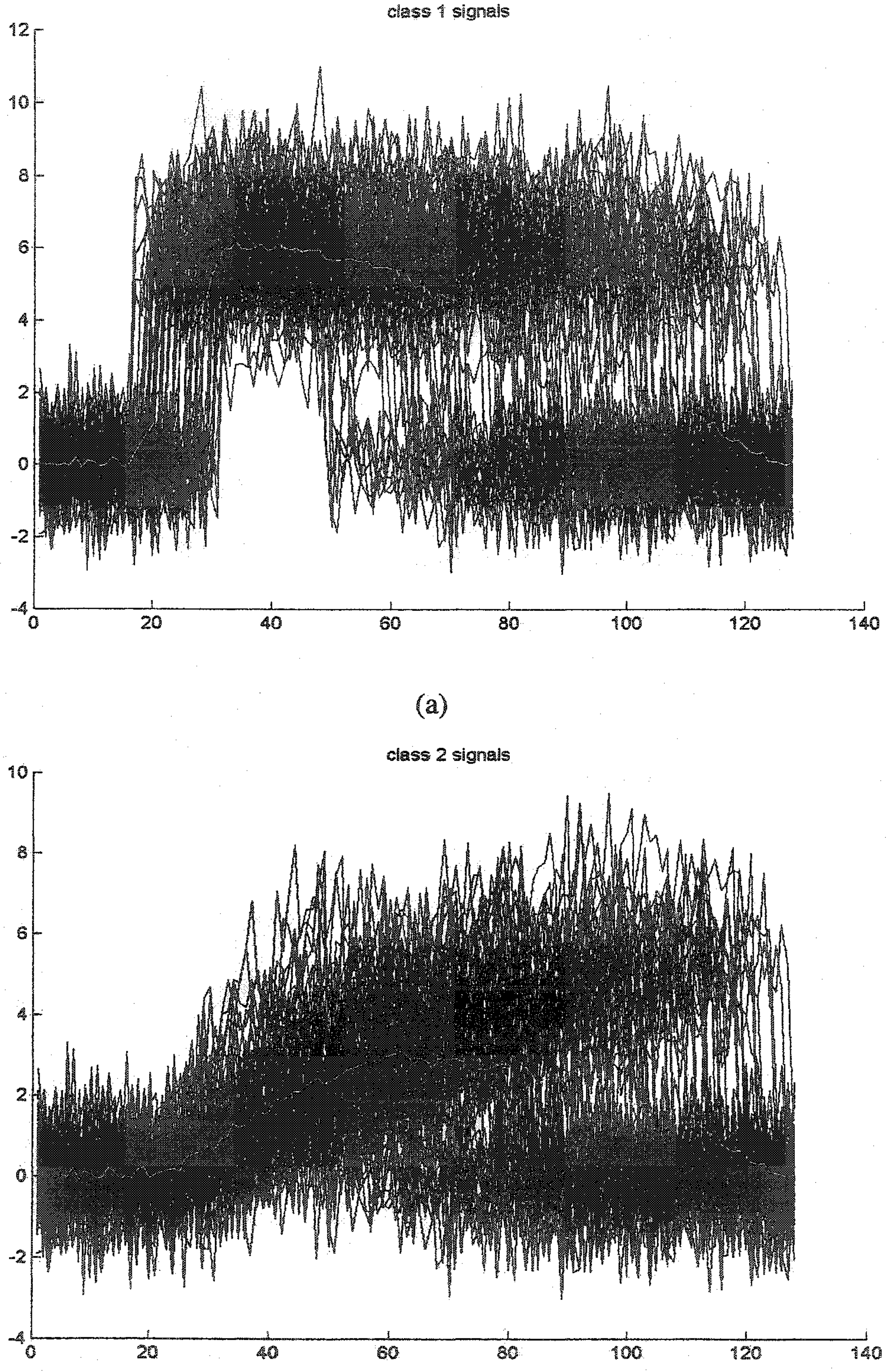
The QMF used was the one representing Coiflet 4 mother wavelet and symmetric differential entropy was used as discrepancy measure. Fig.4.8 shows the average LDB coefficients for different classes of generated signals as obtained from LDB software and for different noise levels. The actual distribution of LDB coordinates for different classes are given in Fig.8.2. Besides the signal coefficients in the new local discriminant basis, the local discriminant basis functions or vectors are given as the output and are depicted in Fig.4.9. We have tried to alter the selected local discriminant basis by manually changing the LDB vectors according to the linear expression given in (73),

$$\omega_{new} = a\omega_{old} + b, \tag{73}$$

where "a" and "b" are constants and "ω" is a basis function selected by LDB algorithm. Fig.4.9 shows that signal energies are mainly concentrated in the first 20 coordinates. For that reason, the trial and error experiment is performed on the most discriminant basis vectors from 1 to 20. The linear change is only applied to the first 20 coordinates. Table 8.1 summarizes the results of this experiment that motivated the author to investigate the possibility of optimizing LDB coefficients in a systematic manner. As the level of noise is increased, more LDB features are buried in noise. This can be inferred by looking at Fig.4.8 where LDB average representations are depicted at two different SNR levels. This explains the decrease in the number of selected LDB features for classification as reflected in the third column of Table 8.1. In the practical SNR range between 10 and 30 dB, the experiments inspire strong possibility of improvement. At very high or very low SNR levels, the improvement is not as evident as before due to perfect performance of LDB at high SNR levels and weak LDB discrimination power at low SNR levels. Fig.8.3 shows the altered LDB vectors.

SNR (dB)	Number of LDB Vectors Changed	Number of LDB Vectors Classified	Decrease in Misclassification Rate (%)
37	18	12	0
25	16	11	25
18	14	10	36
13	12	9	50
9	10	8	8

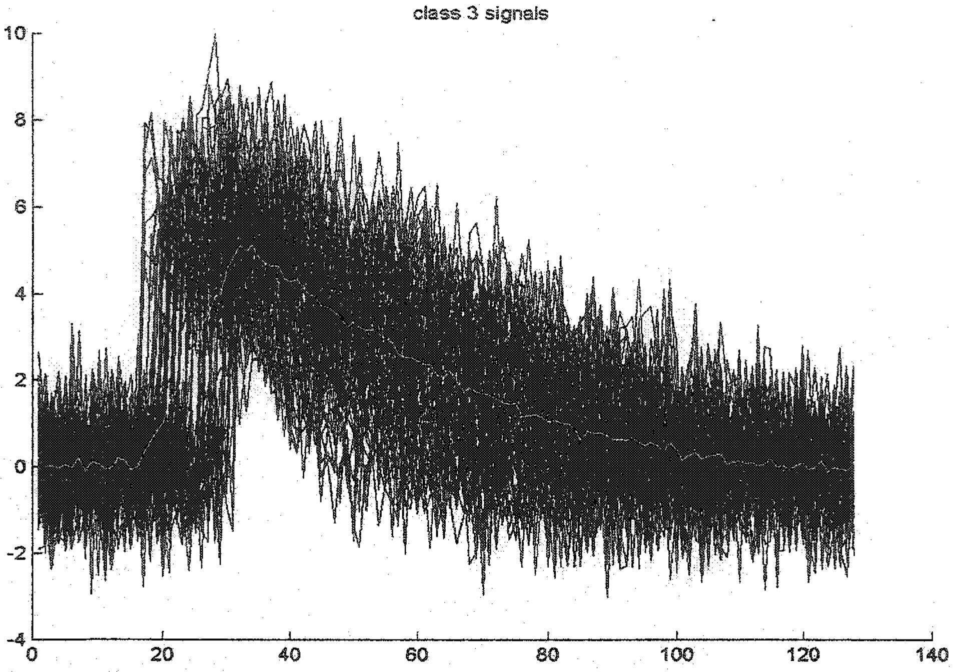
Table 8.1 Decrease in misclassification rate in a trial and error experiment



(a)

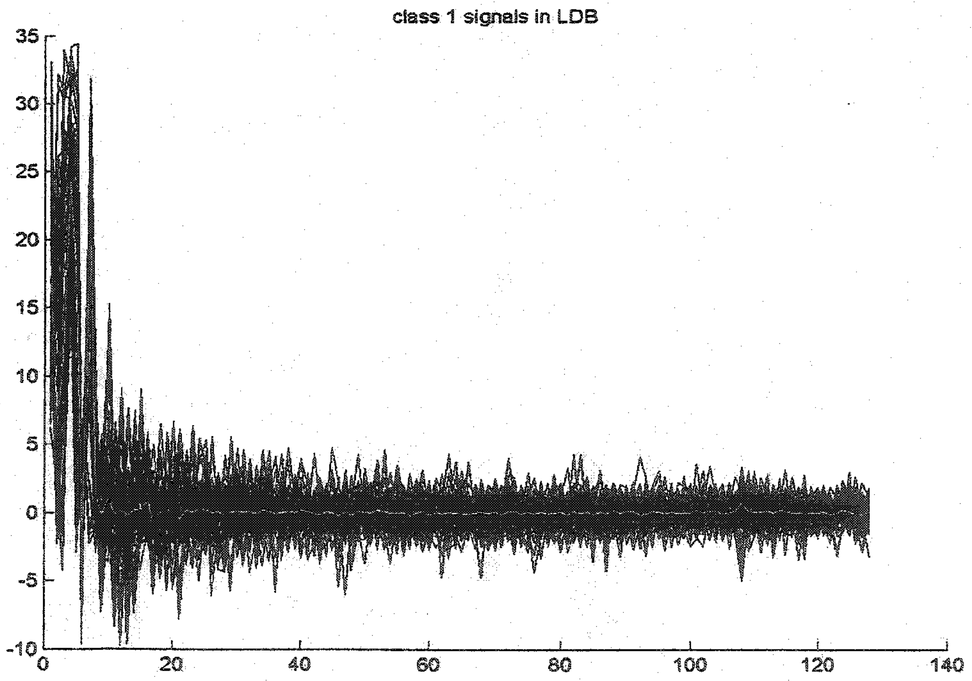
(b)

Fig.8.1 Three classes of signals as generated by (72) depicted in (a), (b) and (c)



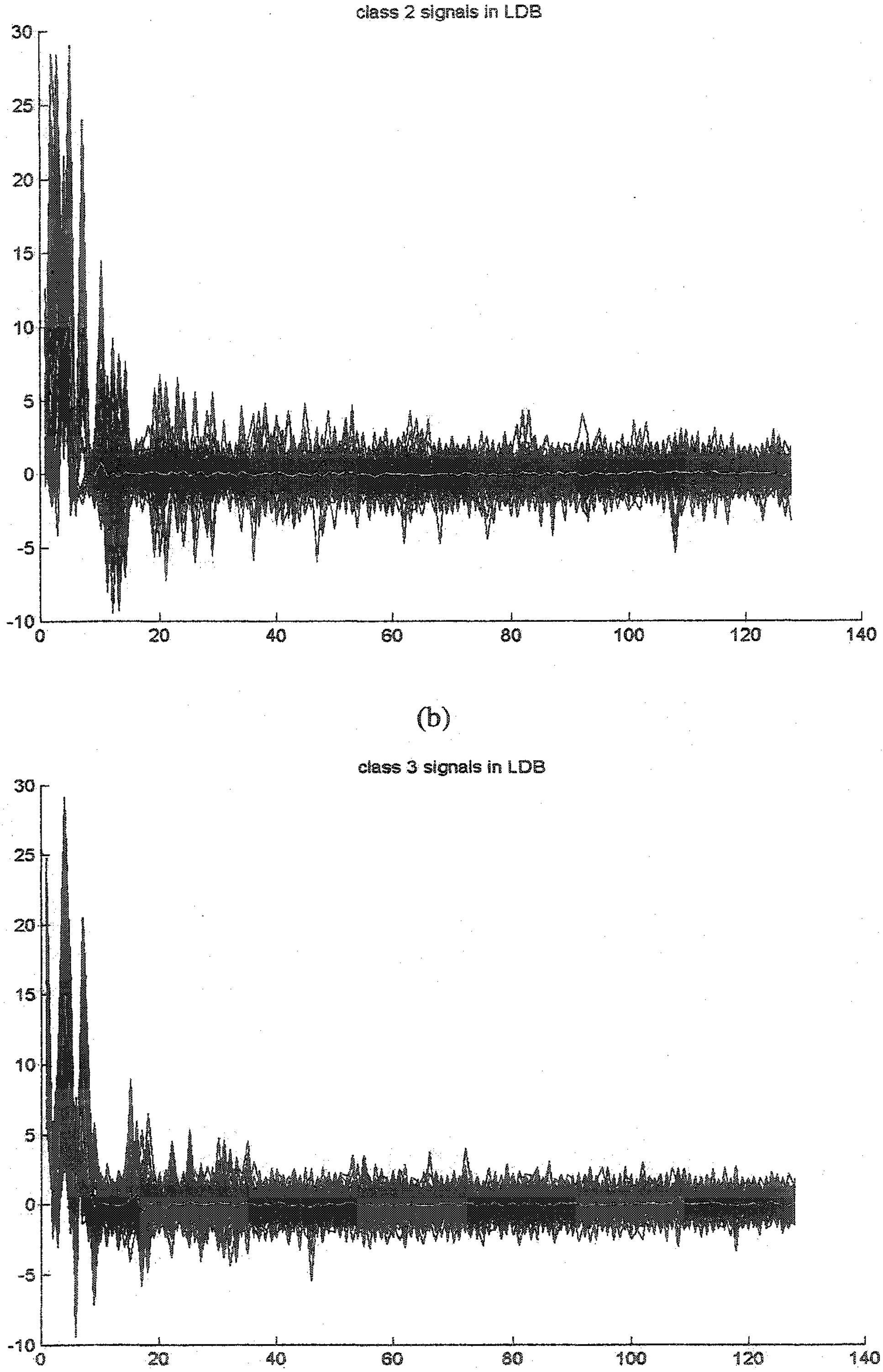
(c)

Fig.8.1 (continued)



(a)

Fig.8.2 Three classes of signals generated by (72) in LDB domain



(b)

(c)

Fig.8.2 (continued)

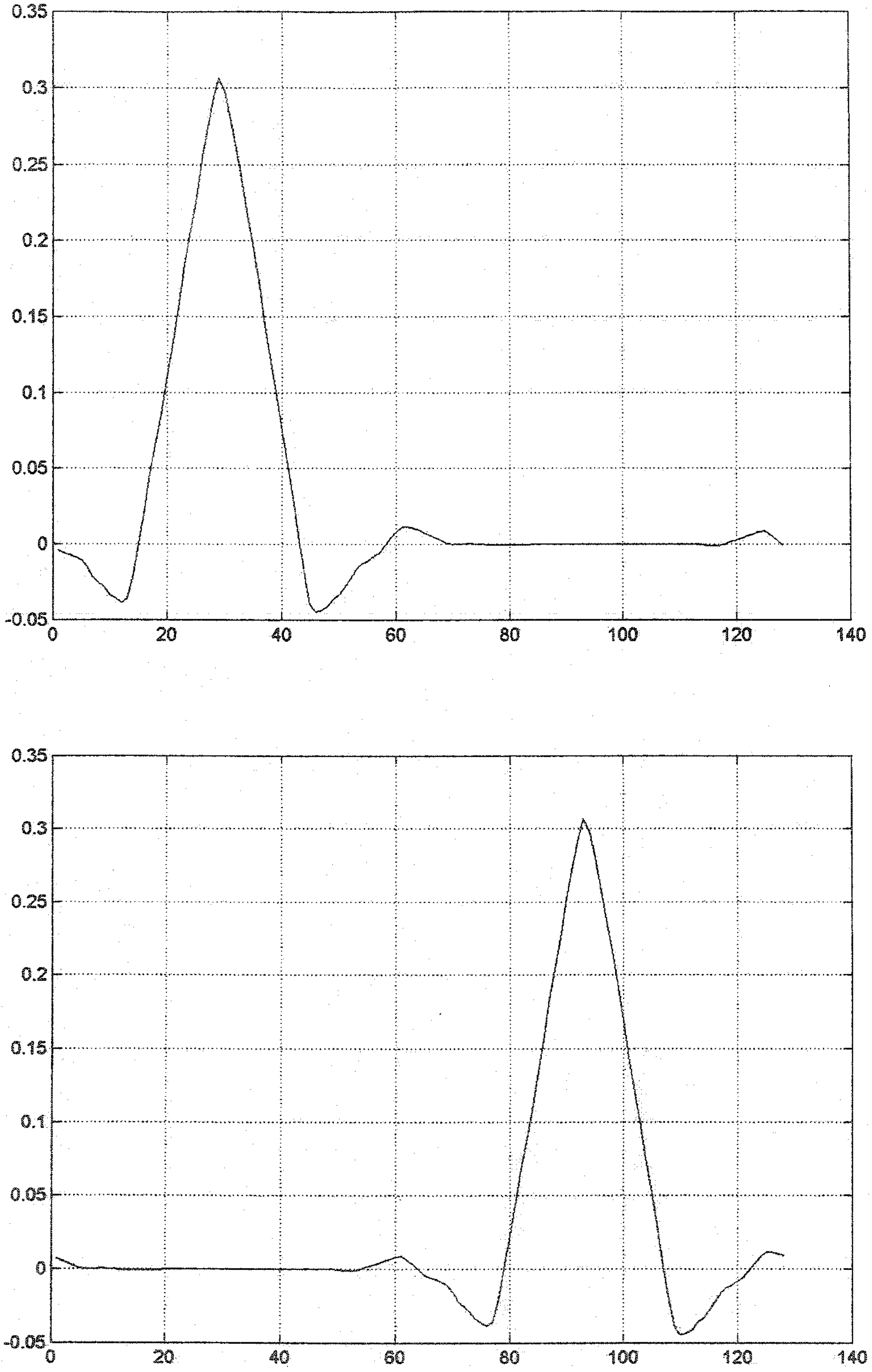


Fig.8.3 Two most discriminating features selected by LDB

8.2 Optimally Weighted Local Discriminant Bases (OLDB)

Herein, the theory of optimally weighting the basis function by the steepest decent algorithm is developed. The basis functions selected by LDB capture some of the common features or background components as well. This is particularly important when background structure is highly correlated to the features of interest. A particular example is microcalcifications in a mammogram [38]. In this section a method of boosting desired distinguishing features among different classes is obtained by optimally weighting the basis functions.

Let us assume that $\psi_\gamma, 1 \le \gamma \le N$ is the collection of basis functions that are obtained as the output of LDB algorithm. Here N is signal/image dimension. The effect of assigning a weight sequence such as $\omega_\gamma, 1 \le \gamma \le N$ to these basis functions is studied next. The goal is to promote the discrimination power of $\omega_\gamma \psi_\gamma, 1 \le \gamma \le N$ hence improving the classification accuracy. To this end, the difference between samples of each signal class with the so-called *teacher signals* [38] is minimized. Teacher signals are class representatives for each class of signals. They should be produced for any application of optimized LDB separately by looking at selected LDB features and if necessary with the aid of experts in that field of application. An example is specifying a few mammograms representative of benign and malignant breast cancer with the help of a radiologist. These class representatives are further processed to identify only regions of interest. Again in mammography context, only certain area of the mammogram are considered important as far as breast cancer detection is the only concern. Therefore, in order to reduce computational load of the overall classification scheme teacher signals focused to their regions of interest are used in optimization block.

Teacher signals for 1D and 2D classification problems discussed in Section 4.7 and Section 4.8 are represented in Fig.8.4 and Fig.8.5 respectively. The reduction in computations is huge in 2D case where windowing different classes around regions of interest, is performed.

However, the concept of regions of interest, although best known in image processing context for various kinds of image processing applications, is highly application oriented in a feature extraction/classification problem. This fact brings up controversial discussions as how specifying different regions of interest affect the overall proposed classification scheme. The main idea of specifying regions of interest for a classification problem is to reduce the signal dimension for LDB optimization right in time or space domain. In order to avoid such discussions; a guard region is produced by considering a safety factor for specifying regions of interest. This is best illustrated with the aid of Fig.8.6 and Fig.8.7. Therefore after regions of interest are identified, the signal/image-processing engine finds a guard region around these areas in order not to loose any valuable information. The new teacher signals obtained in this process are called guarded teacher signals.

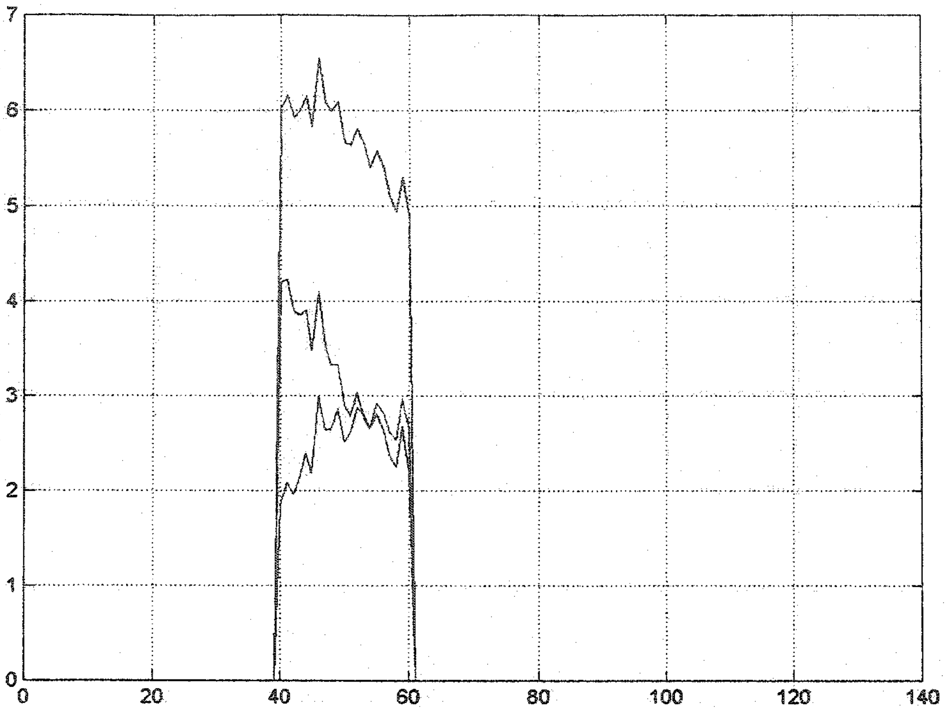


Fig.8.4 A set of teacher signals for the process in described in (72)

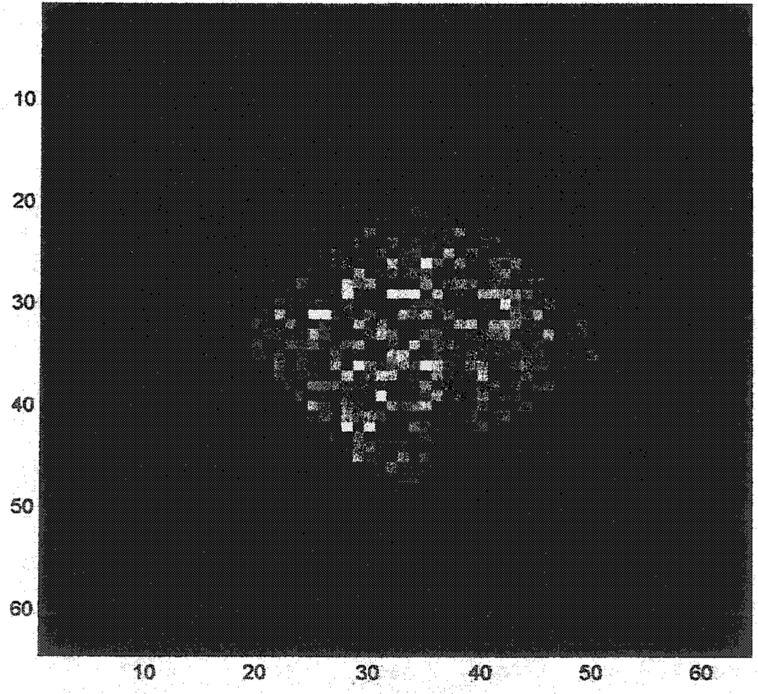


Fig.8.5 Class I teacher signal for the process described in Section 4.8

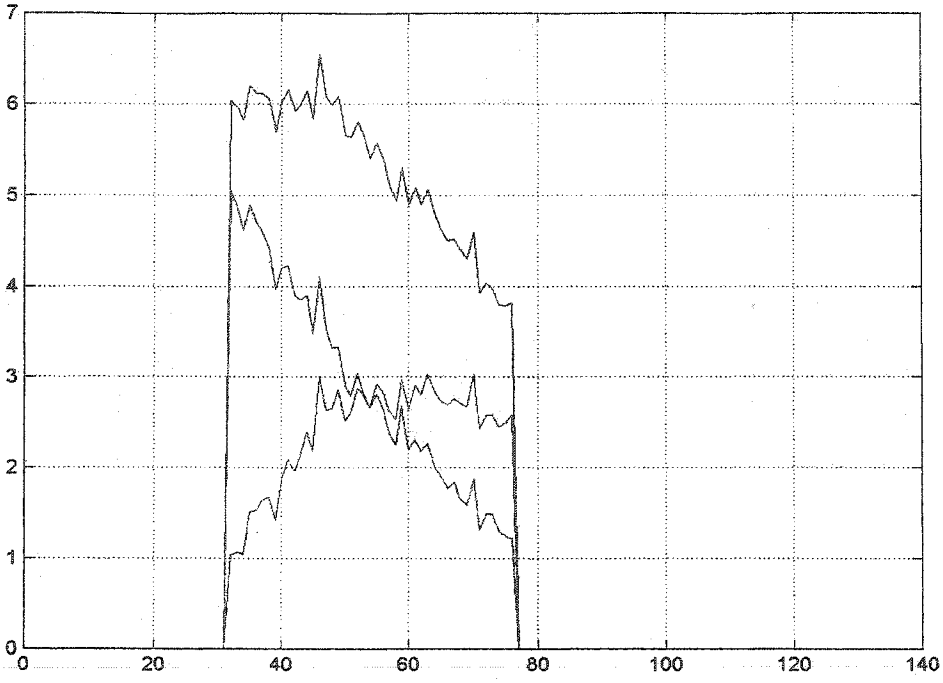


Fig.8.6 A set of guarded teacher signals for the process in described in (72)

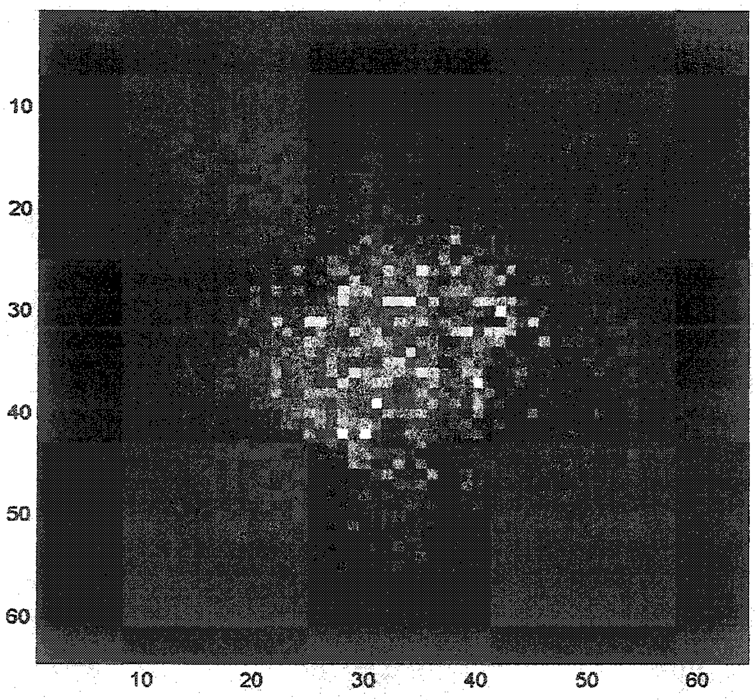


Fig.8.7 A sample guarded class I teacher signal for the process described in Section 4.8

In the process obtained by (1), teacher signals for each class are chosen to be the average of the signal class windowed at the regions of interest; therefore there are a limited number of them. The error function is formulated in (74). In this expression, α_γ^k's are the LDB coordinated of signal s^k. The corresponding teacher signal is designated by T^k.

$$E(\omega)=\frac{1}{K}\sum_k\sum_x(S^k(\omega,x)-T^k(x))^2 \tag{74}$$

$$S^k(\omega,x)=\sum_\gamma \alpha_\gamma^k \omega_\gamma \psi_\gamma(x) \tag{75}$$

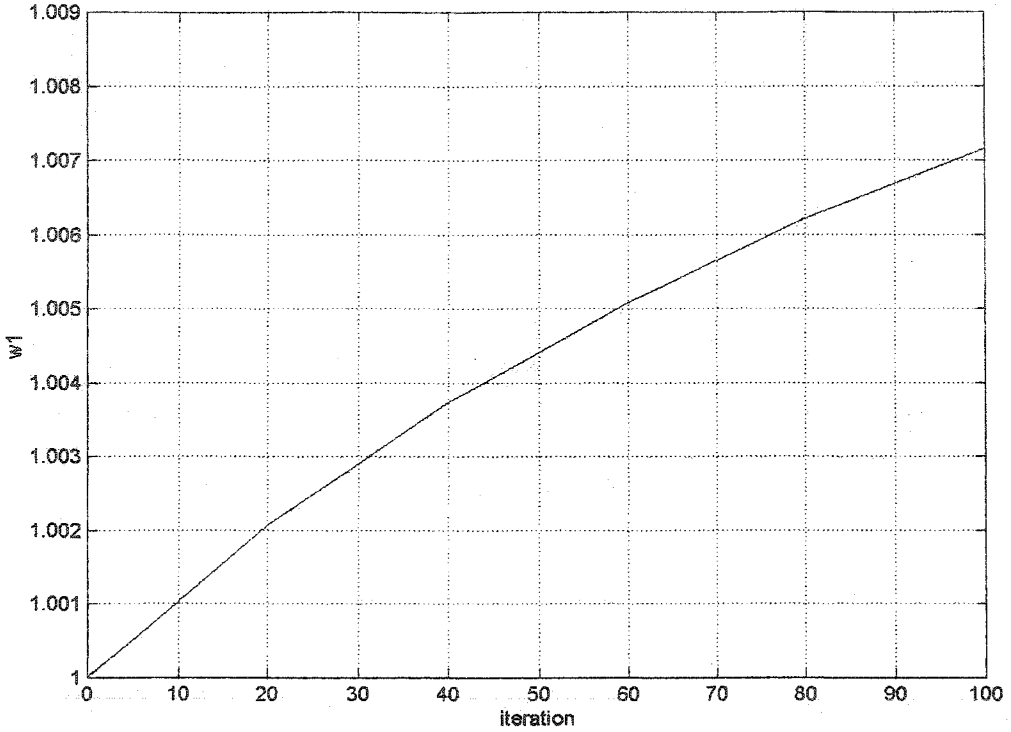
In (74) and (75) k ranges over the samples used in the optimization phase whereas x ranges over time or pixels in an image processing application and K is the number of samples used for optimization. Note that the optimization process is done with considerably less number of samples selected from training set and also the range of x is limited to region of interest. The minimization of error function formulated in (74) reduces the difference of the samples with the teacher signals focused at regions of interest. The samples for calculation of error are chosen randomly from the training signals set, nevertheless, they are fixed for all optimization calculations after the selection is done. The error function $E(\omega)$ is minimized by a gradient decent algorithm. The process produces an optimal set of weights that maximally separate different signal classes. The partial derivative of $E(\omega)$ in terms of a weight ω_γ is given in (76),

$$\frac{\partial E}{\partial \omega_\gamma}=\frac{2}{K}\sum_k\sum_x(S^k(\omega,x)-T^k(x))\alpha_\gamma^k\psi_\gamma(x) \tag{76}$$

Therefore the weight update formula for gradient decent algorithm will be as in (77),

$$\omega_\gamma \to \omega_\gamma - \eta\frac{\partial E}{\partial \omega_\gamma} \tag{77}$$

Here η is a user-defined learning rate. Fig.8.8 shows the change in first five weight factors assigned to five most discriminating LDB vectors versus iteration. Fig.8.9 shows the change in the first two most discriminating LDB feature distributions before and after optimization block.



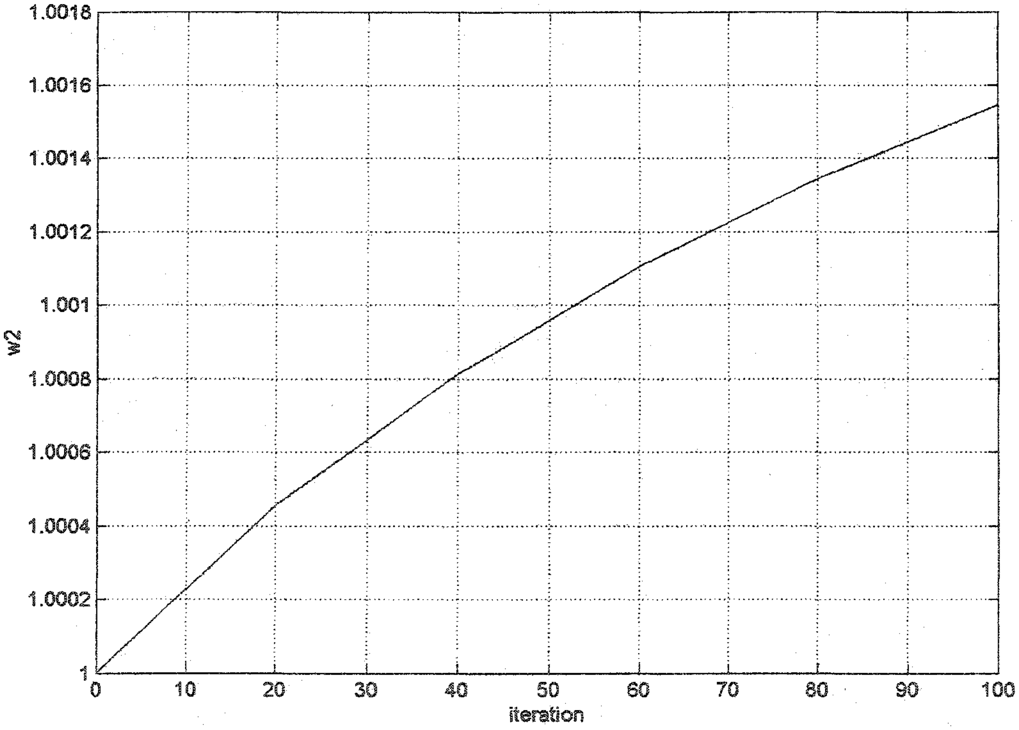
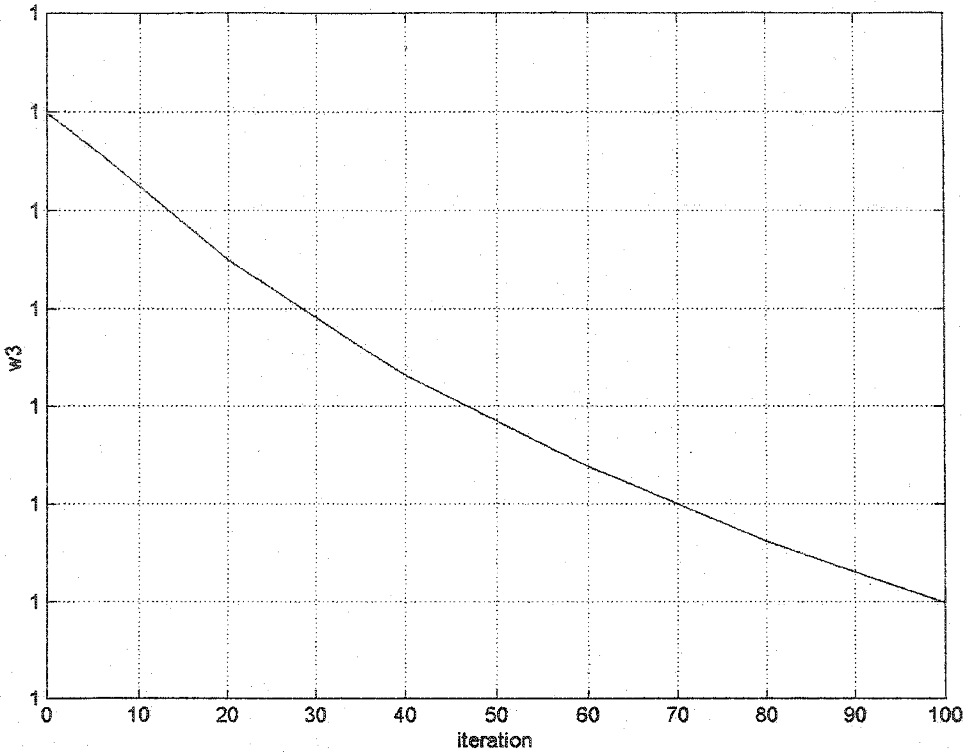


Fig.8.8 Changes in weight factors assigned to the five most discriminating LDB vectors; graphs are ordered from top to bottom according to discrimination powers



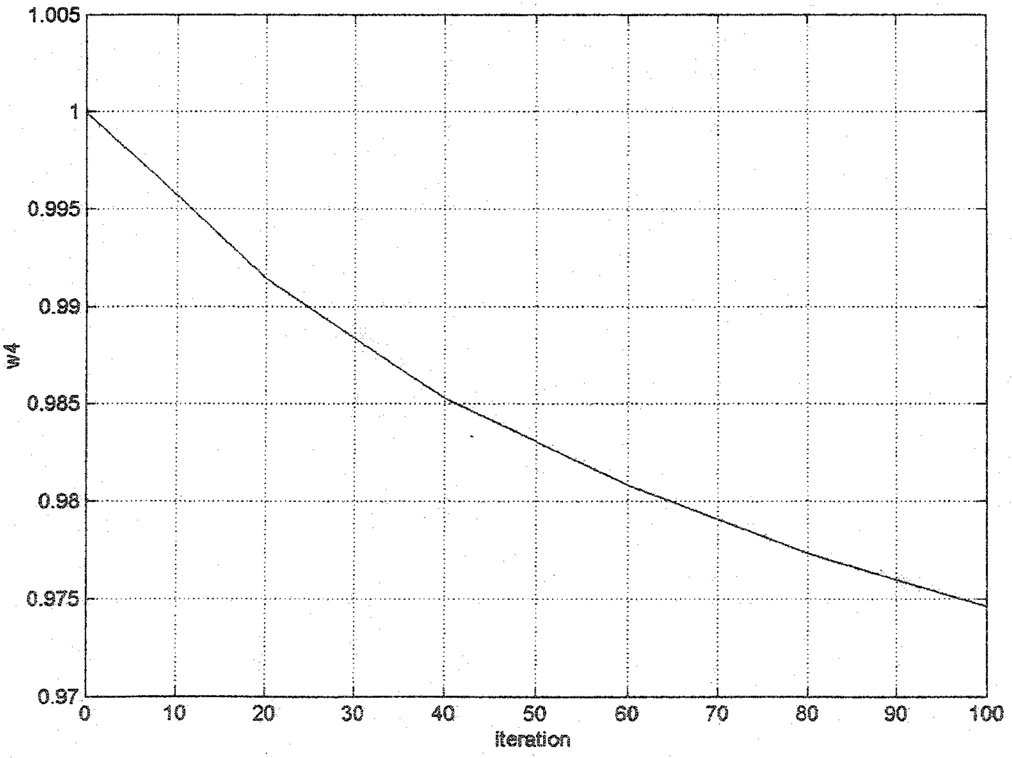


Fig.8.8 (continued)

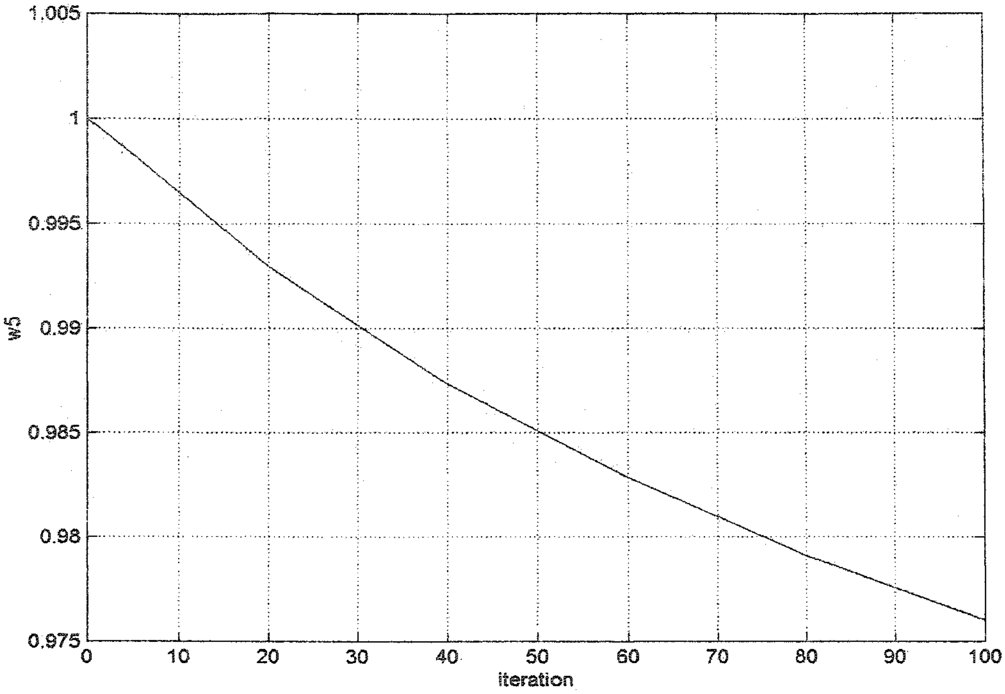
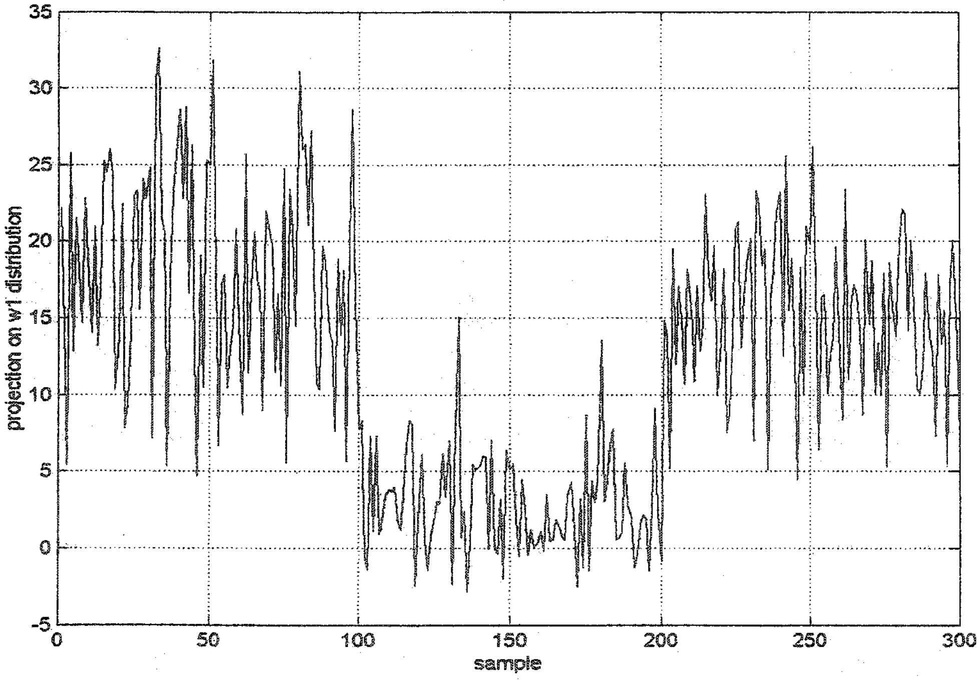
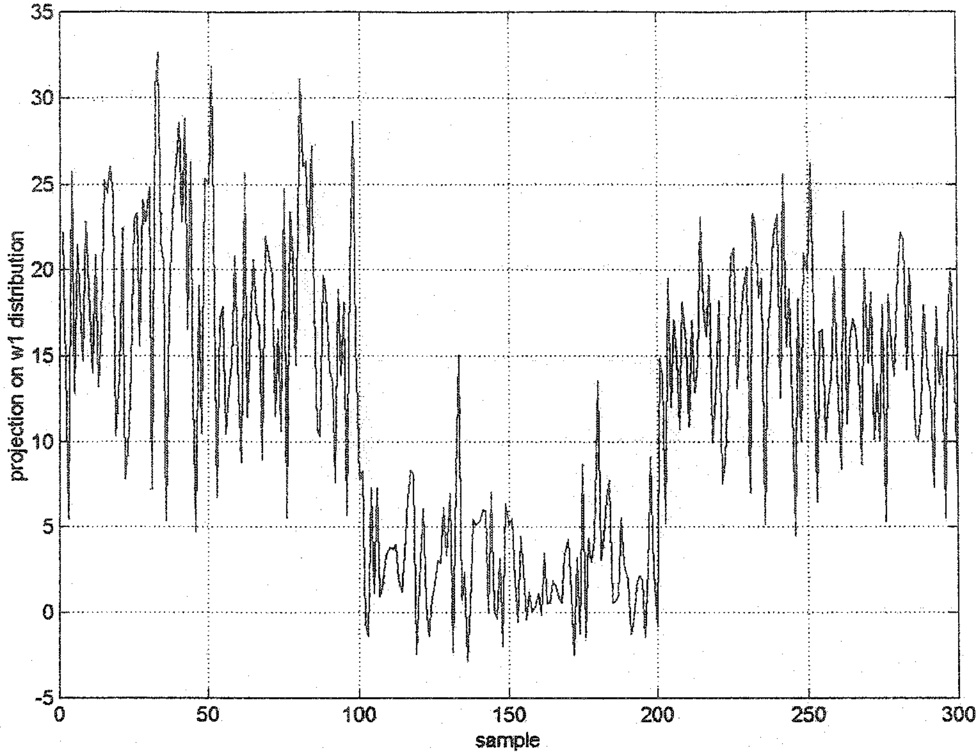


Fig.8.8 (continued)



(a)

Fig.8.9 (a) and (b) The projection into most discriminating LDB before and after optimization (c) and (d) the same data for the second most discriminating LDB



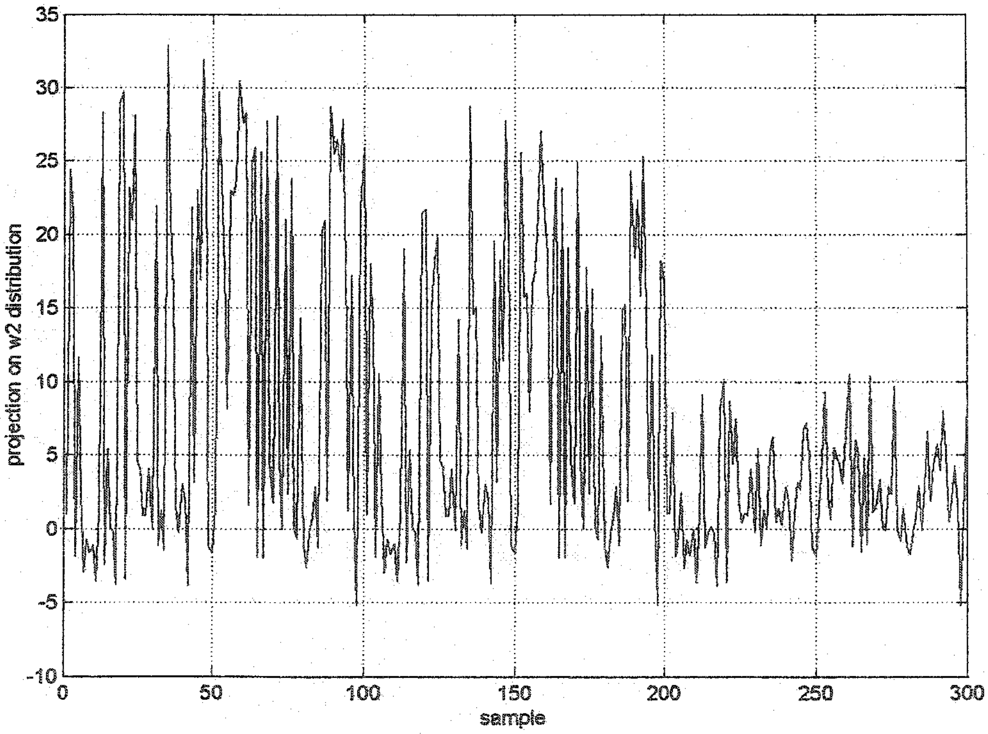


Fig.8.9 (continued)

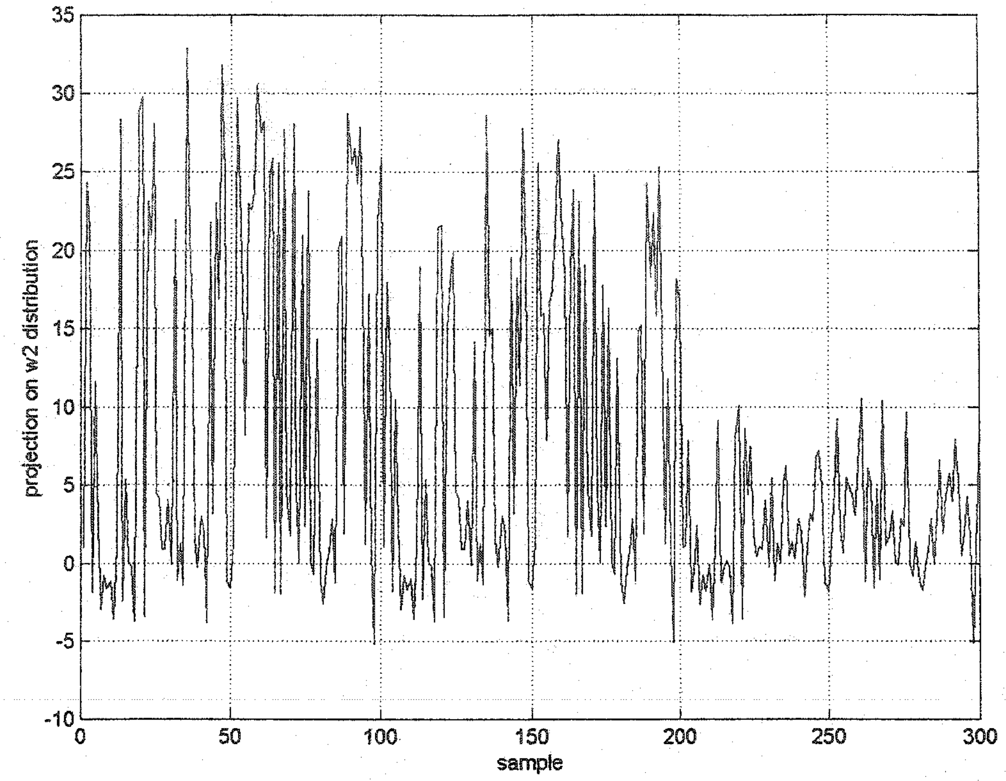
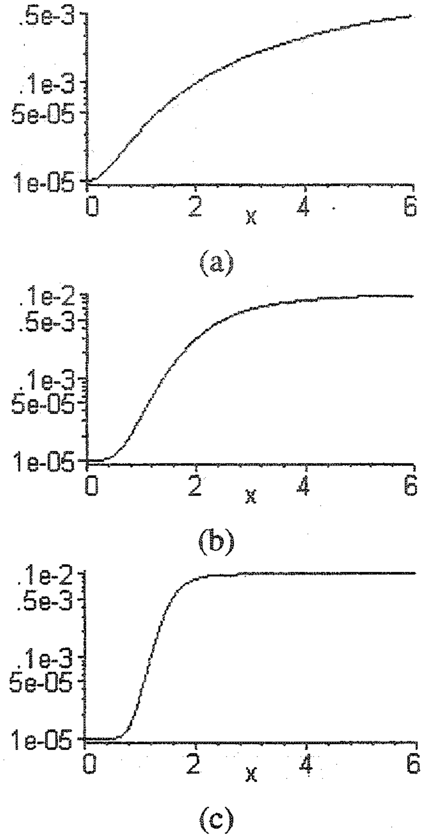


Fig.8.9 (continued)

8.3 The Effect of Learning Rate on Convergence of Optimized LDB

The role of learning factor, η, in convergence of the optimization block is evident. Smaller value of η should be selected to prevent bypassing the local minima, and to help the algorithm to eventually converge to it. η should not be too small for realistic execution time. Using a fixed η in the steepest decent algorithm does not seem to be the best choice. By adaptively adjusting the learning rate one can improve the convergence of the algorithm and at the same time make sure that the algorithm captures local minima.

Fig.8.10 shows some possible variable learning rate strategies that lead to adaptive adjustment of learning rate. The particular choice of the learning curve adaptation pattern is governed by problem specifications. However, note that higher order adaptation patterns result in sharper drop of learning rate around the desired error magnitude.



(a)

(b)

(c)

Fig.8.10 Different patterns for learning rate adaptation according to rational functions of the form $\frac{ax^n + b}{cx^n + d}$ depicted in (a), (b) and (c) for $n = 2,4,8$ respectively

8.4 Optimally Weighted LDB (OLDB) Simulation Results

In the first experiments with OLDB, local discriminant analysis was used to capture LDB for a set of 300 training signals described in Example 8.1. The wavelet packet used in this study was generated by a Coiflet 4 mother wavelet. Symmetric relative entropy (*J*-divergence) was used as discrepancy measure. Linear discriminant analysis was fixed as the classifier in this study. Two sets of teacher signals were generated by windowing the statistical averages in the intervals [40,55] and [40,60] to study the effect of window size, which is representative of the effect of regions of interest (Fig.8.11). These two sets of teacher signals are referred to as 15-point and 20-point windows in all the graphs. The number of samples used in optimization phase was fixed to 21 equally distributed between three classes (*K*=21). Another set of 300 signals was generated as a test bed for examining the effectiveness of optimization process. Fig.8.12 shows the results of using the first 12 most discriminative LDB vectors. The weight update process was performed on 15 and 20 first most discriminative LDB vectors as opposed to 128 to lower computational load. The experiment continued up to 100 iterations. Fig.8.13 and Fig.8.14 show similar results for classification based on 10 and 8 most discriminative LDB vectors respectively.

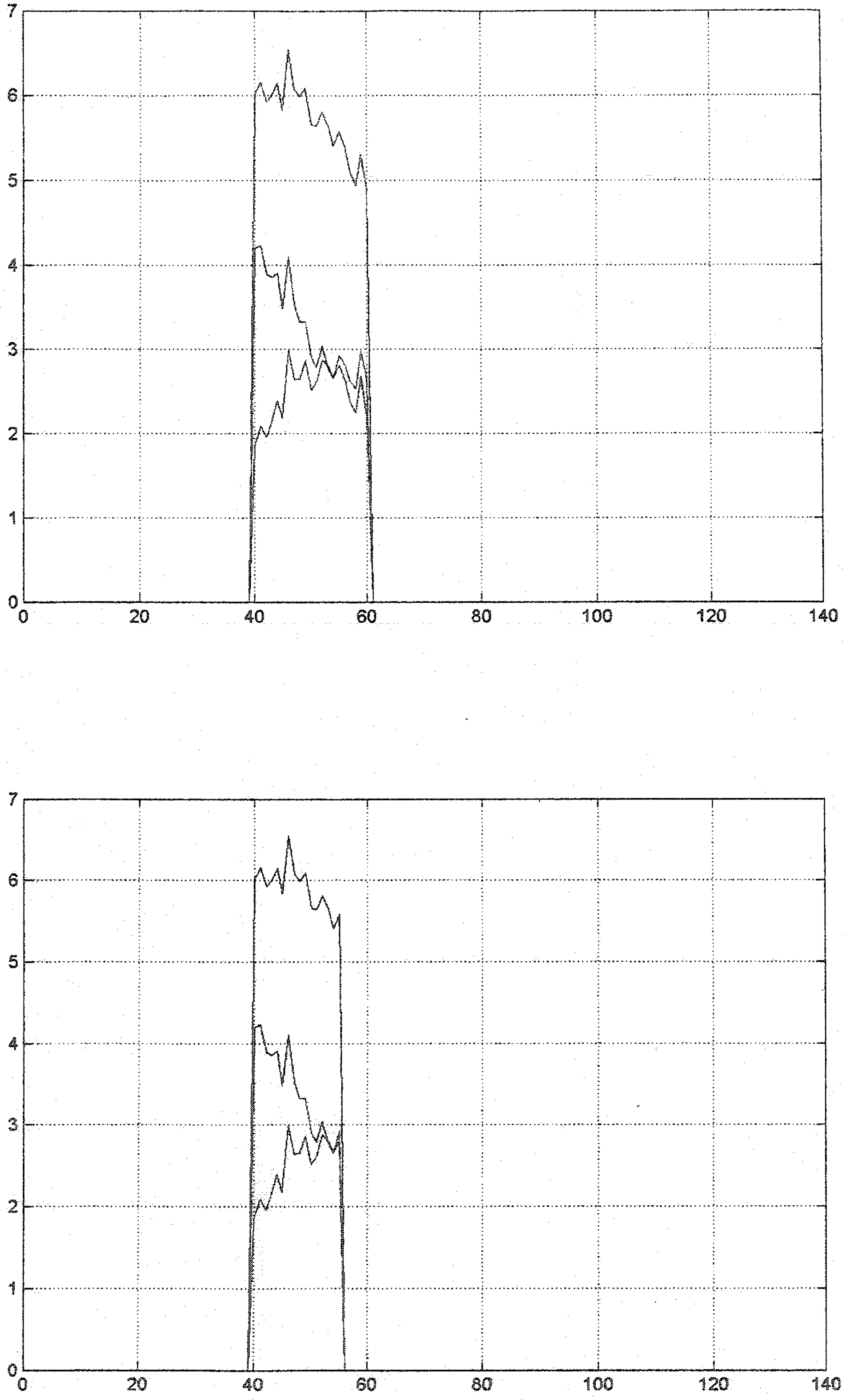


Fig.8.11 Two sets of teacher signals used to optimize LDB performance

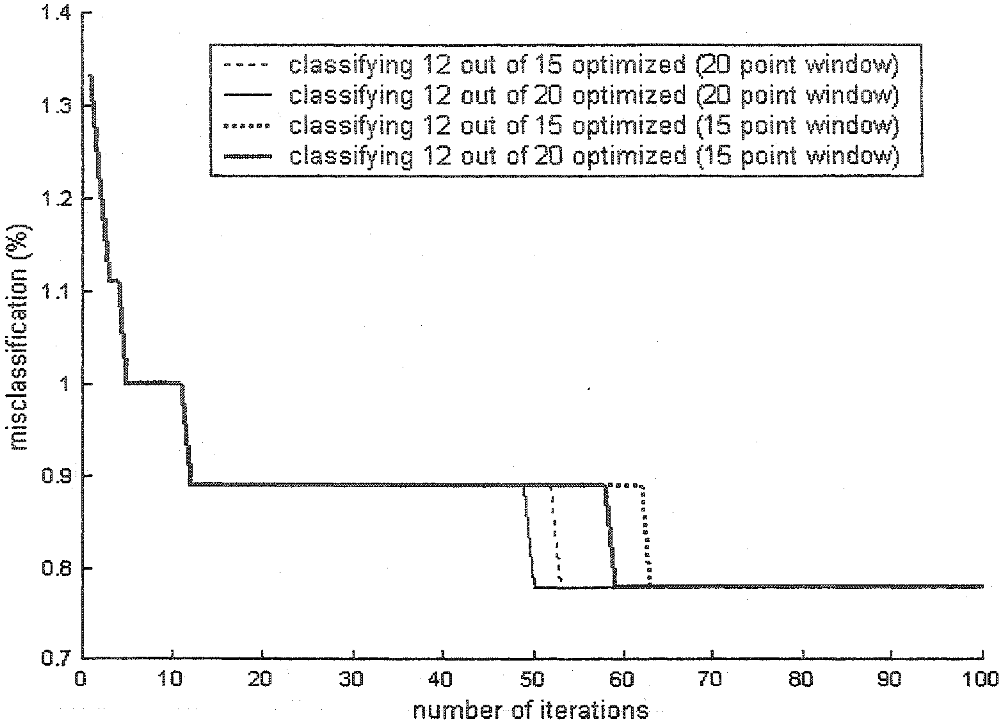


Fig.8.12 The decrease in misclassification rate as a function of time

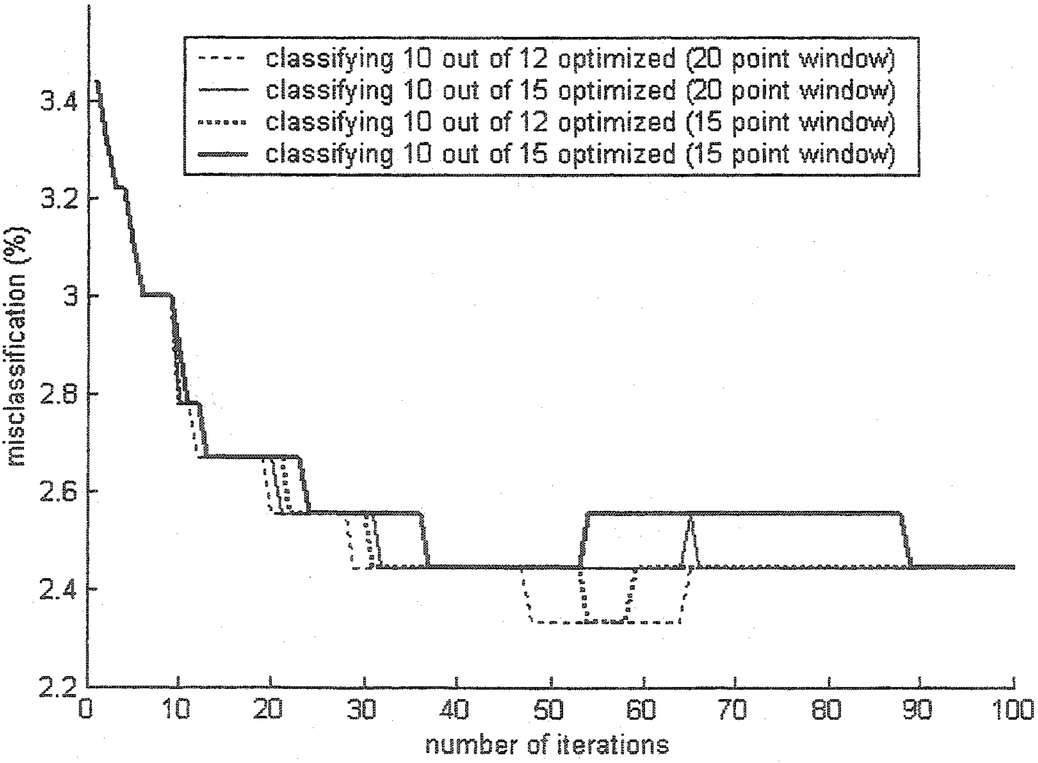


Fig.8.13 The decrease in misclassification rate as a function of time

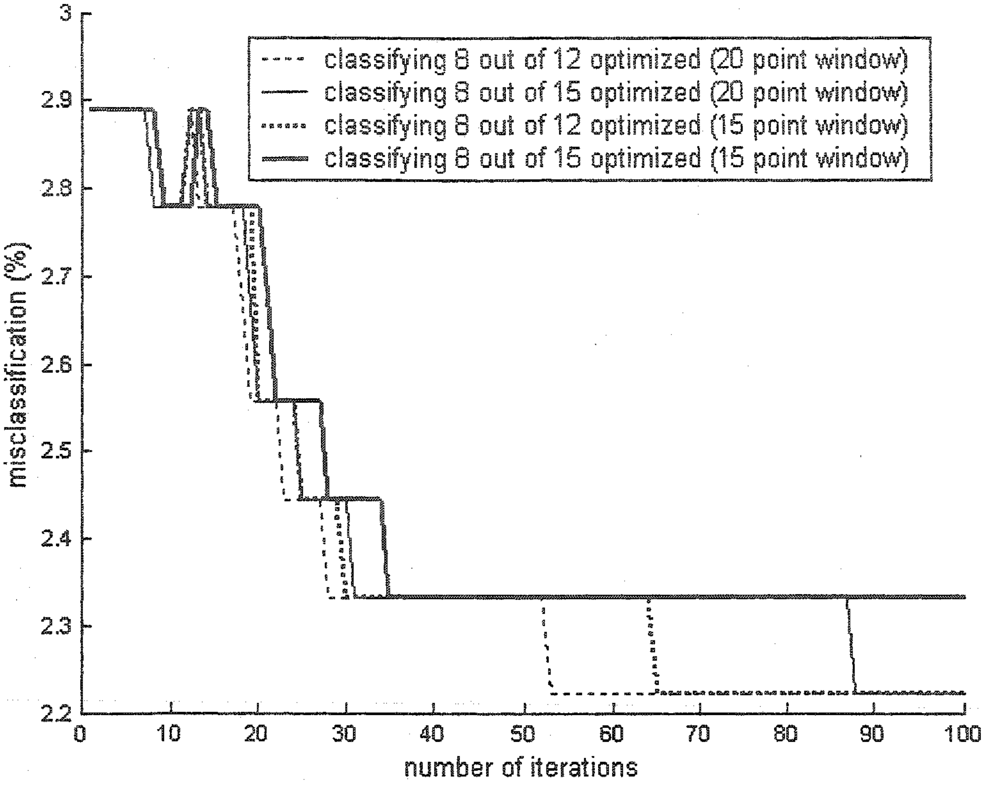
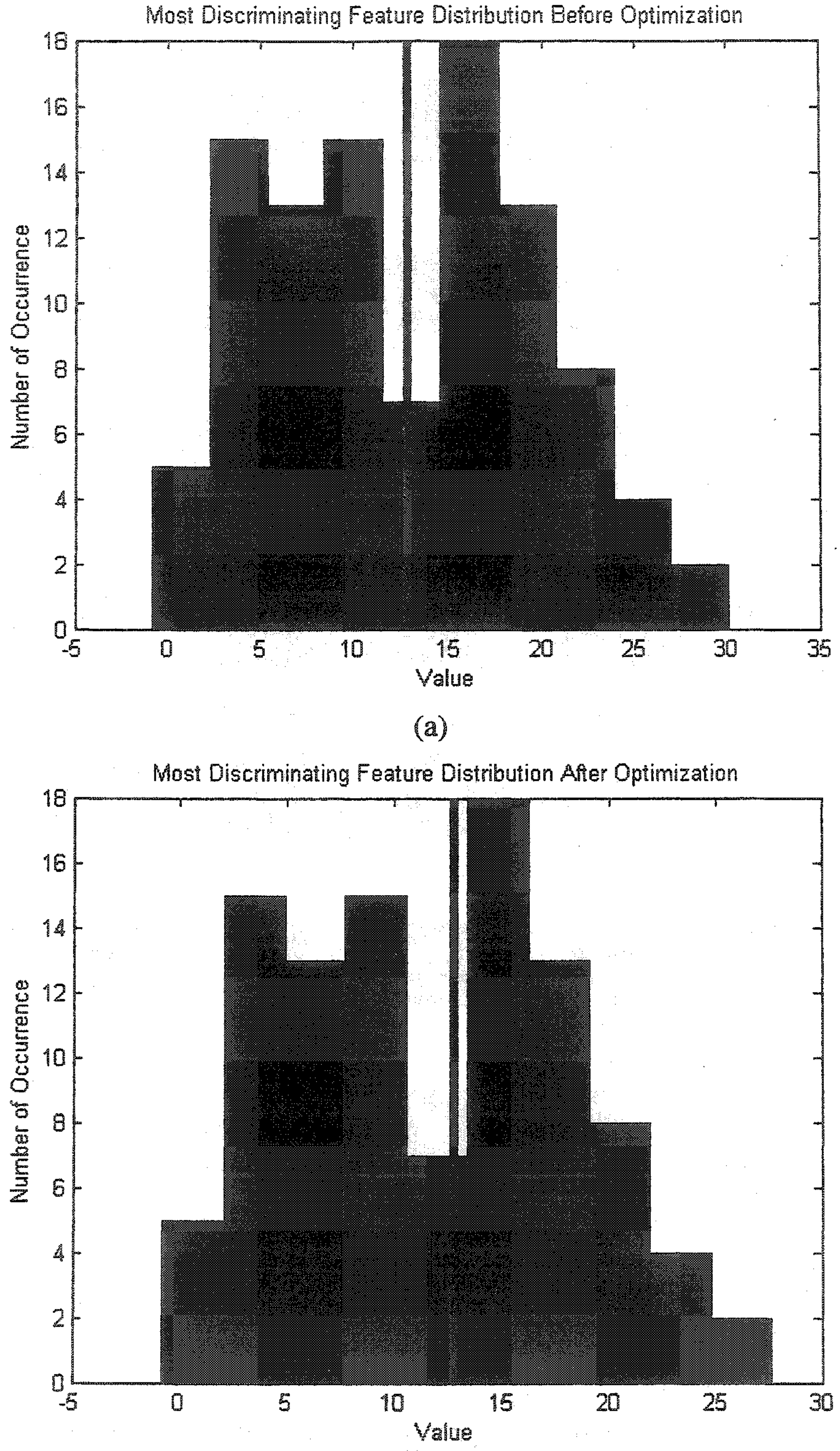


Fig.8.14 The decrease in misclassification rate as a function of time

8.5 Distribution of Features Before and After Optimization Block

Linear discrimination analysis (LDA) [1], [33] is a supervised classification method, which is proven to be the optimal classifier if distributions of features are normal Gaussian distributions. The optimization block as proposed and implemented in this work, helps to improve the accuracy of classification mainly because it linearly transforms the features so that their distributions look more like normal distribution and signals are in the average sense closer to individual clusters as identified by LDA.

The above argument is verified by investigation of feature distributions before and after optimization block. Fig.8.15 depicts the change in the most discriminating LDB coefficient distribution graphically in the form of histograms. Although, the changes are small, the difference between distributions before and after optimization block is easily noticed. The small changes help to fix the misclassifications that happen in LDA due to components on boundaries of clusters.



(a)

(b)

Fig.8.15 Distribution of the most discriminating LDB vector (a) before and (b) after optimization block; the red line shows the average

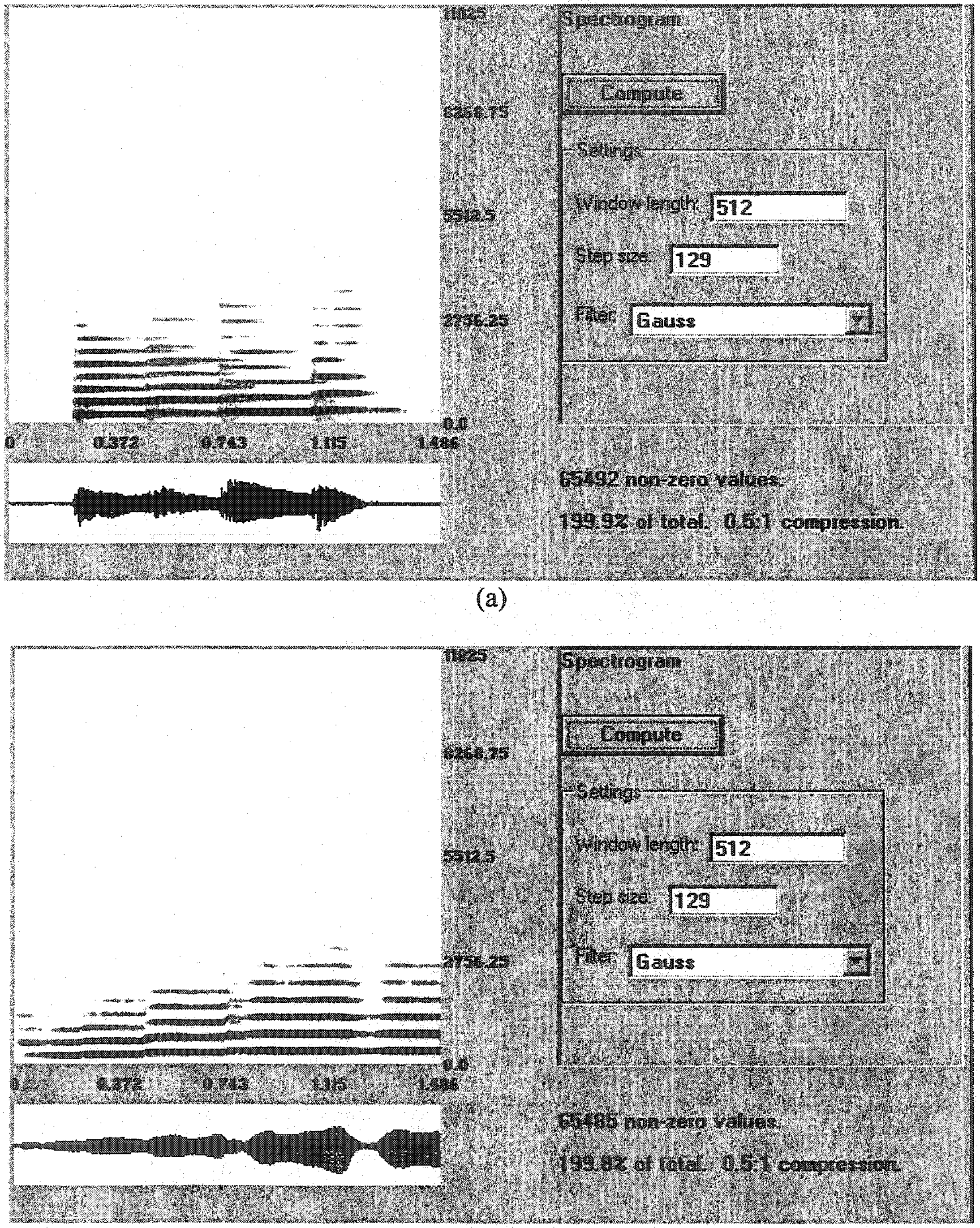
8.6 Audio Signal Classification

In this section, the efficiency of optimized local discriminant basis algorithm (OLDB) is tested on a database of music signal segments. The database constitutes of 5-second music recordings of different types, *i.e.* classical, rock, popular, etc. The sampling frequency is 44.1 kHz. Therefore, each music recording contains enough sound segments to train LDB feature extraction module. The goal is to classify the music segments into different music categories that they belong to. This is a suitable database to test the performance of OLDB on since speech and music are known to be truly non-stationary processes thus warranting a true time-frequency tool [55]. The type of instrument played in the recording has important effect on time-frequency decompositions of the music piece. This is illustrated in Fig. 8.16 that compares the spectrogram of two different instruments, *i.e.* the piano and the flute, playing the notes E_4, F_4, G_4 and A_4. This spectrogram provides a description of the sound signal in the time-frequency plane. As seen in Fig.8.16, the spectra for the individual notes are clearly separated in time. The spectra for individual notes are clearly divided into groups of line segments lying above each note and corresponding to the fundamentals and overtones in each note.

There are clear differences between the "attack" and the "decay" of the spectral line segments for the notes played by the two instruments. For the piano there is a very prominent attack due to the striking of the piano hammer on its strings. This is visible in the gray patches in its spectrogram near the beginning of the fundamental and first overtone line segments for each note. There is also a longer decay for the piano notes due to slow damping down of the piano string vibrations. This is evident in the overlapping of the time intervals underlying each note's line segment. The flute notes arise from a standing wave within the flute created by a gentle breath of the player and rapidly decay when this standing wave collapses at the end of player's breath. The notes generated by the flute show little overlapping of spectral line segments. It is well known that, in addition to the harmonic structure of fundamentals and overtones, the precise features of attack and decay in notes are important factors in human perception of musical quality. This comparison of a piano with a flute illustrates how all of these features of musical notes can be quantitatively captured in time-frequency dictionaries. The type of instruments used in different music categories with their special composition and correlation enables us to differentiate between different types of music in time-frequency plane. In particular music recordings from different categories serve as a standard compression as well as classification criterion database.

The example studied in this section has applications in MPEG-7 standardization by ISO [55], which is more based on information retrieval. Several experiments are performed to extract signal features for classification by using LDB. The following subsections discuss the experiments and their results. Sections 8.6.1, 8.6.2 and 8.6.3 discuss experiments with two classes of signals. Three possible combinations of classical, rock and popular music are fed into LDB feature extractor and LDA classifier. Later on, classification into three categories of classical, rock and popular pieces is considered in

Section 8.6.4 that is more challenging and the 70% classification accuracy obtained in these experiments is outstanding. The effect of having longer duration samples is studied in Section 8.6.5 where sample duration is doubled in two steps.



Window length: 512

Step size: 129

Filter: Gauss

65492 non-zero values.

199.9% of total. 0.5:1 compression.

(a)

Window length: 512

Step size: 129

Filter: Gauss

65485 non-zero values.

199.8% of total. 0.5:1 compression.

(b)

Fig.8.16 (a) The spectrogram of the piano playing the notes E_4, F_4, G_4 and A_4 (b) the spectrogram of the flute playing the same notes

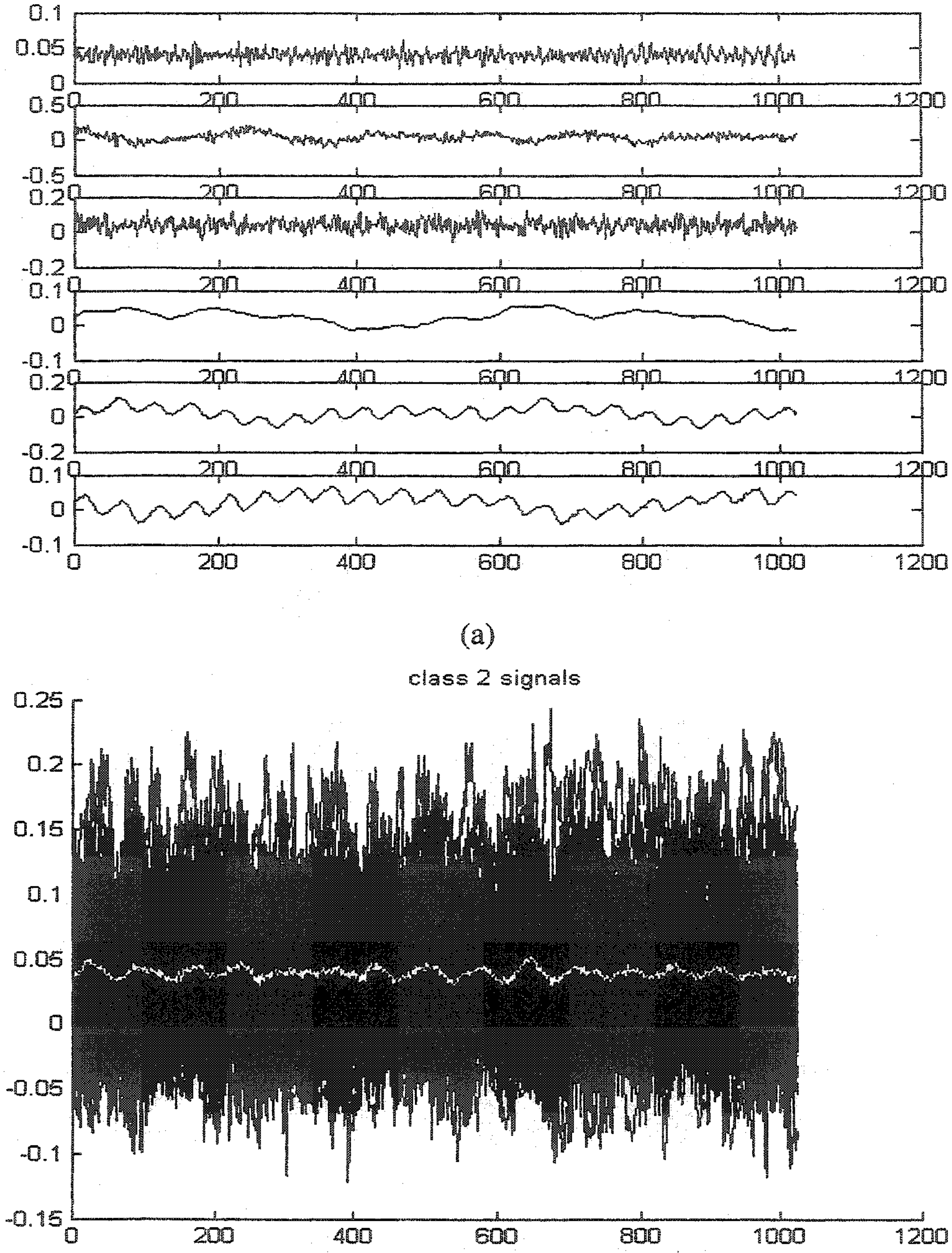
8.6.1 Classifying Classical and Rock Music Segments

The music segments used in this experiment are 23.2 msec segments of classical and rock music recordings of the database described above. Each segment contains 1024 samples. As studied in similar experiments with different signal duration that some of them follow, 25 msec turns out to capture the characteristics of the music segment accurately. 106 music segments from each class are used to train the LDB module. The same number of signals is used to test the performance of this classification scheme. The wavelet packet used in this experiment is a refined Coiflet 24 wavelet packet. The number of classified features is 8. This number is justified when the average class distributions in LDB domain are extracted. Symmetric differential entropy or *J*-divergence is used as discrimination measure.

Fig.8.17 summarizes the results obtained by LDB. Fig.8.17 (a) shows ensemble of training signal. The rock class and classical class segments are depicted together in separate graphs in Fig.8.17 (b) and (c) respectively. It is not easy to tell the difference between classes by looking at the collection of input signals. But in general classical music segments have more correlation within themselves. Fig.8.17 (d) shows a small ensemble of training signals in LDB domain. Training signals in LDB domain are collectively depicted in Fig.8.17 (e) and (f) respectively. LDB has successfully found the best coordinates that capture signal energies. The difference between signal classes is clear in LDB domain. The ensemble averages of the two signal classes are depicted in Fig.8.17 (g) and (h). It is worth to note the difference in ensemble average of the two categories. However, energy normalization in LDB computational engine guarantees that difference between average and signal energies does not cause any problem. This application apparently has simpler statistical complexity than the synthetic problem of Section 8.1. Class averages in LDB domain show that the selection of the eight topmost discriminating LDB functions for classification is a sound decision since class projection energies are concentrated on ten top most discriminating LDB vectors. Fig.8.17 (i) shows the three top most discriminating LDB functions. By looking at the most discriminating feature extracted by LDB it is hard to specify a region of interest in this application (see Fig.8.17 (j)). Thus the whole time axis is used for optimal LDB. The overall misclassification rate is found to be 2.3585% in conjunction with Linear Discriminant Analysis (LDA) classifier.

8.6.2 Classifying Classical and Popular Music Segments

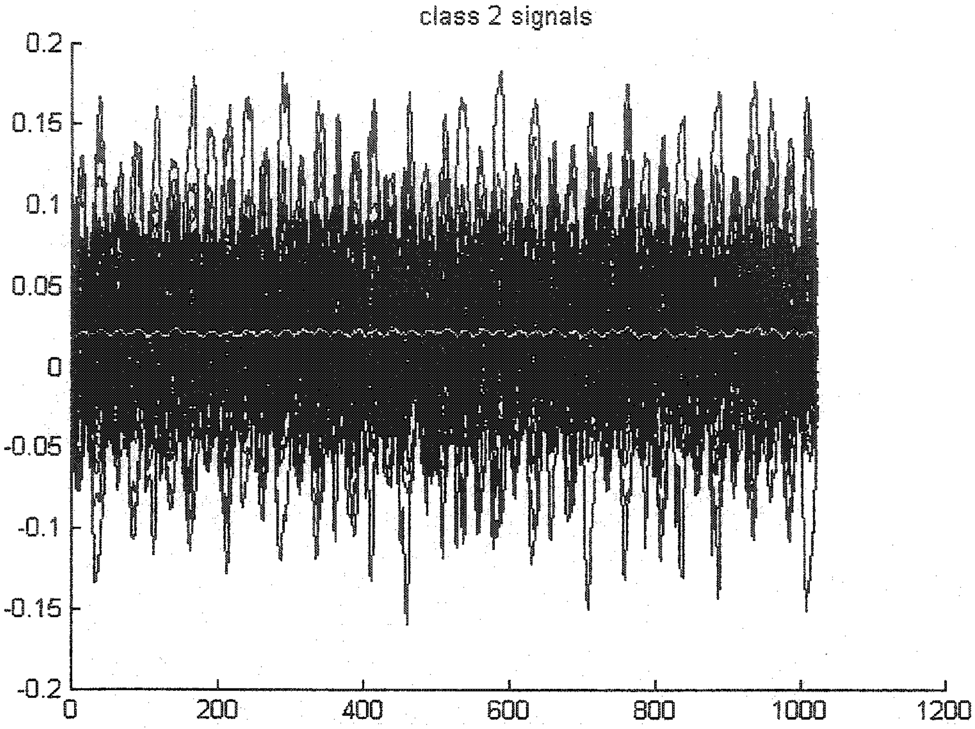
A second experiment is performed to classifying classical and popular music segments. The number and length of the music segment is the same as in the experiment described in Section 8.6.1. LDB algorithm settings are as before. Fig.8.18 summarizes the results of this experiment pictorially. Misclassification rate is found to be 2.3585% with LDA as classifier.



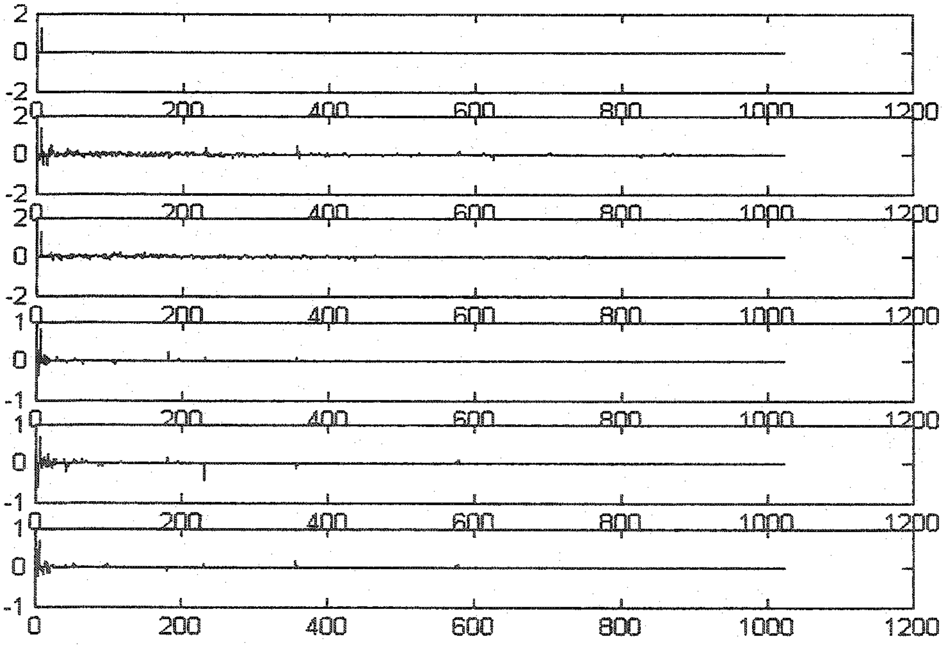
(a)

(b)

Fig.8.17 Rock/Classical Classification: (a) ensemble of three rock and three classical segments from top to bottom (b) rock music signals shown collectively (c) classical music signals shown collectively (d) ensemble of three rock and three classical pieces in LDB domain (e) rock signals shown collectively in LDB domain (f) the same for classical pieces (g) class averages (h) class averages in LDB domain (i) three top most discriminating LDB features (j) most discriminating feature as selected by LDB

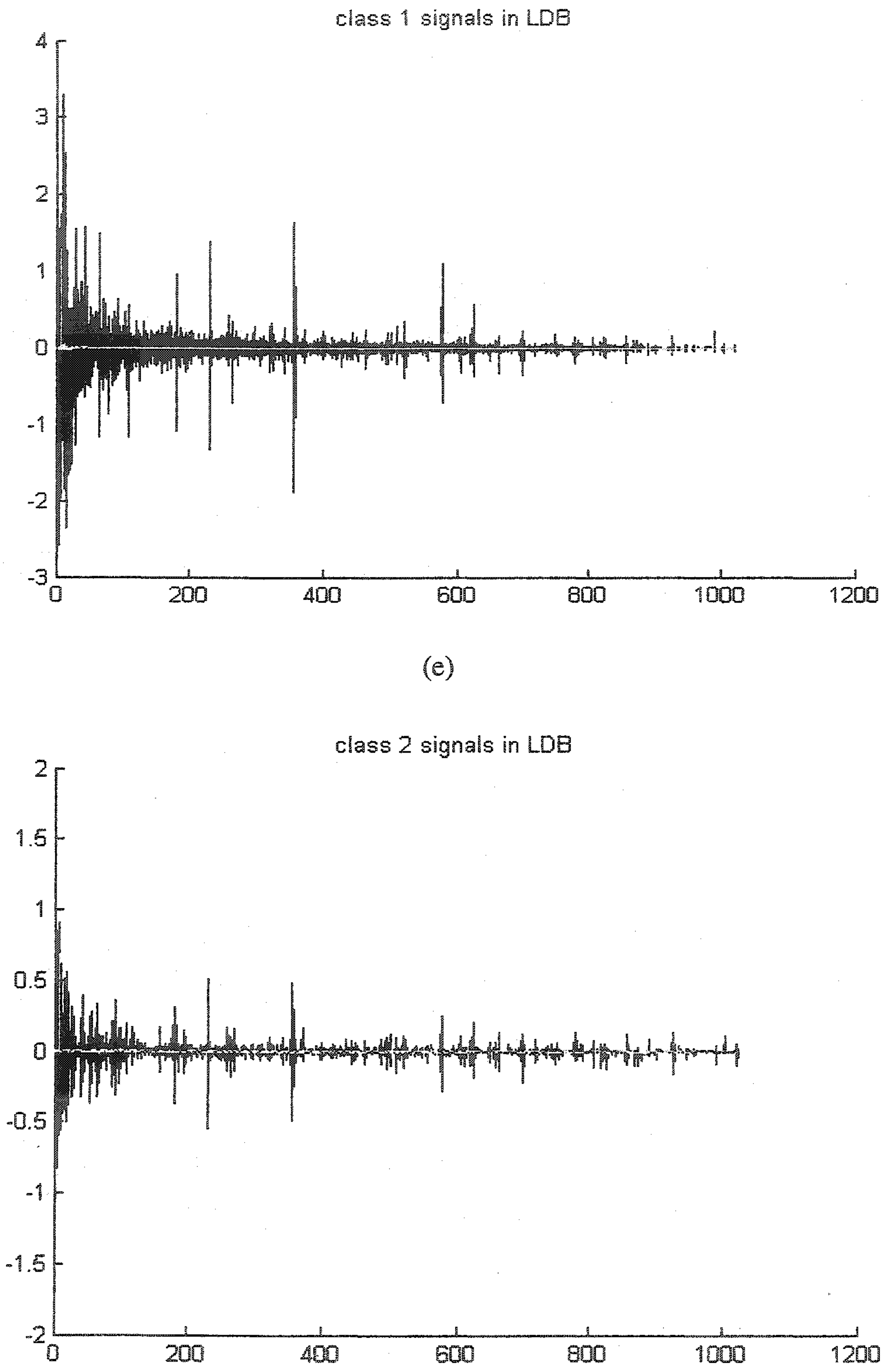


(c)



(d)

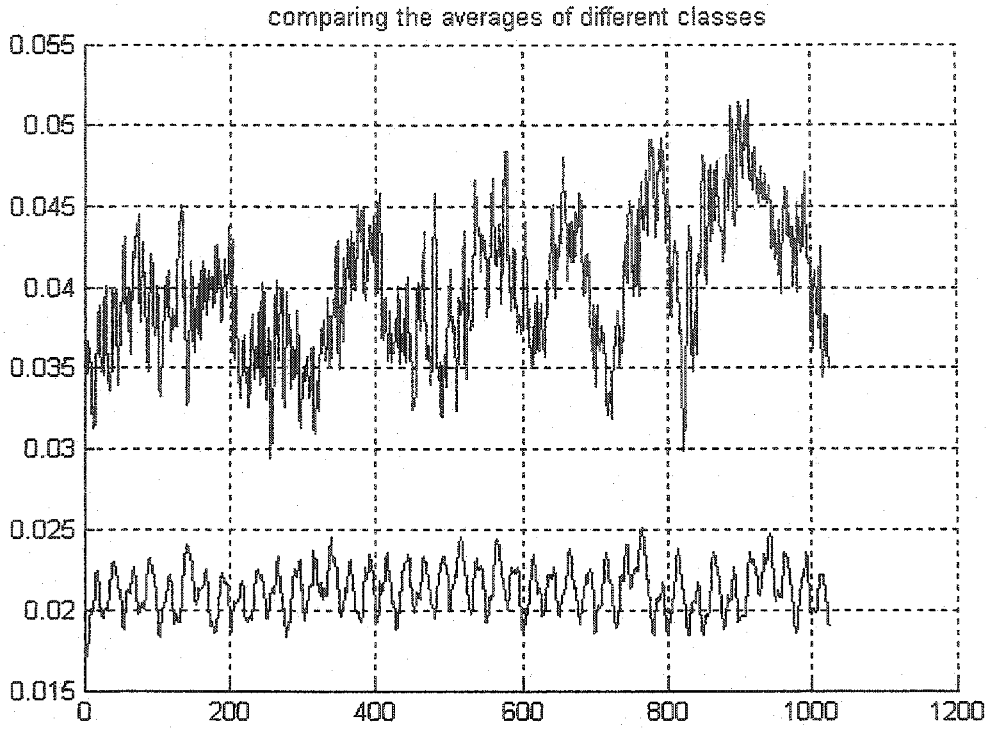
Fig.8.17 (continued)



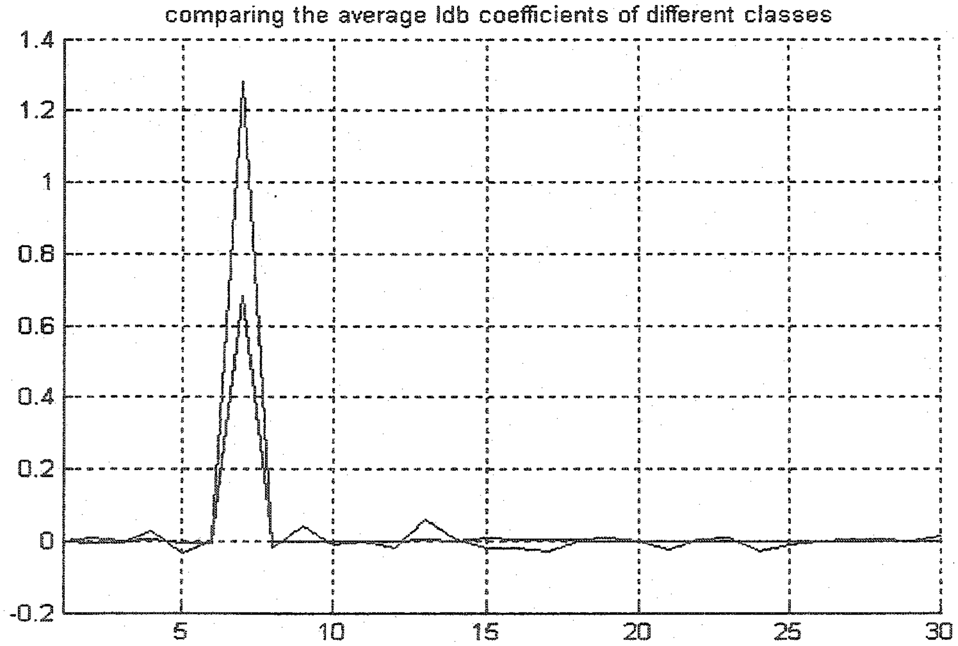
(e)

(f)

Fig.8.17 (continued)

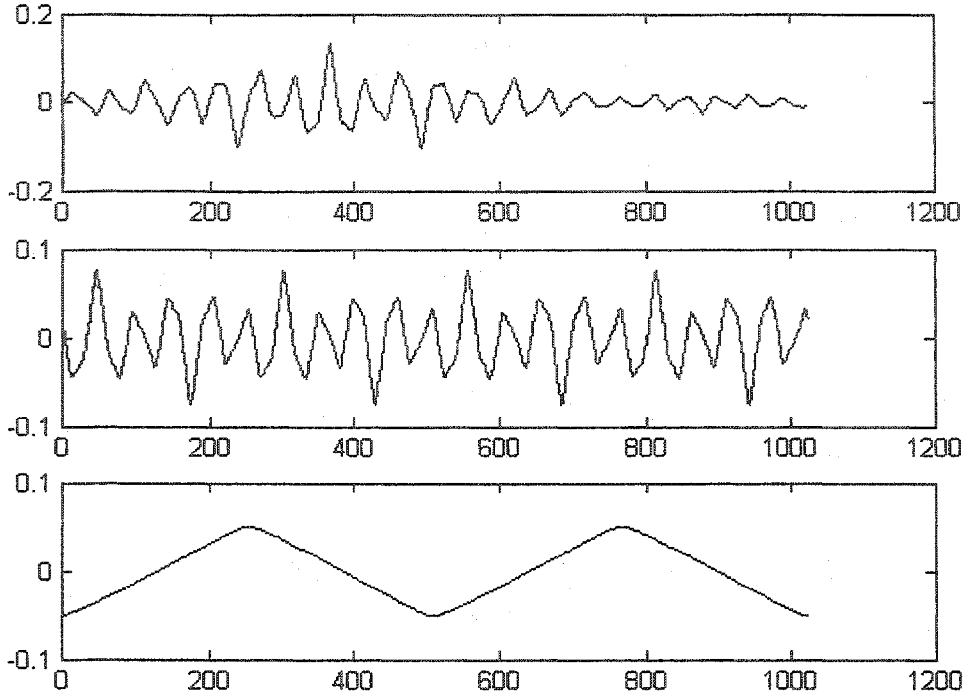


(g)

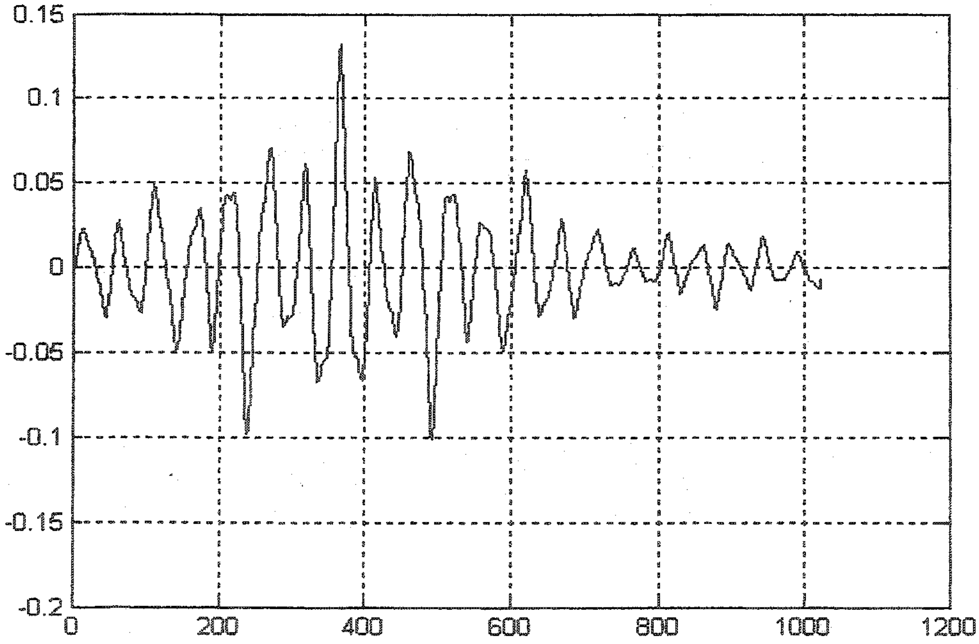


(h)

Fig.8.17 (continued)

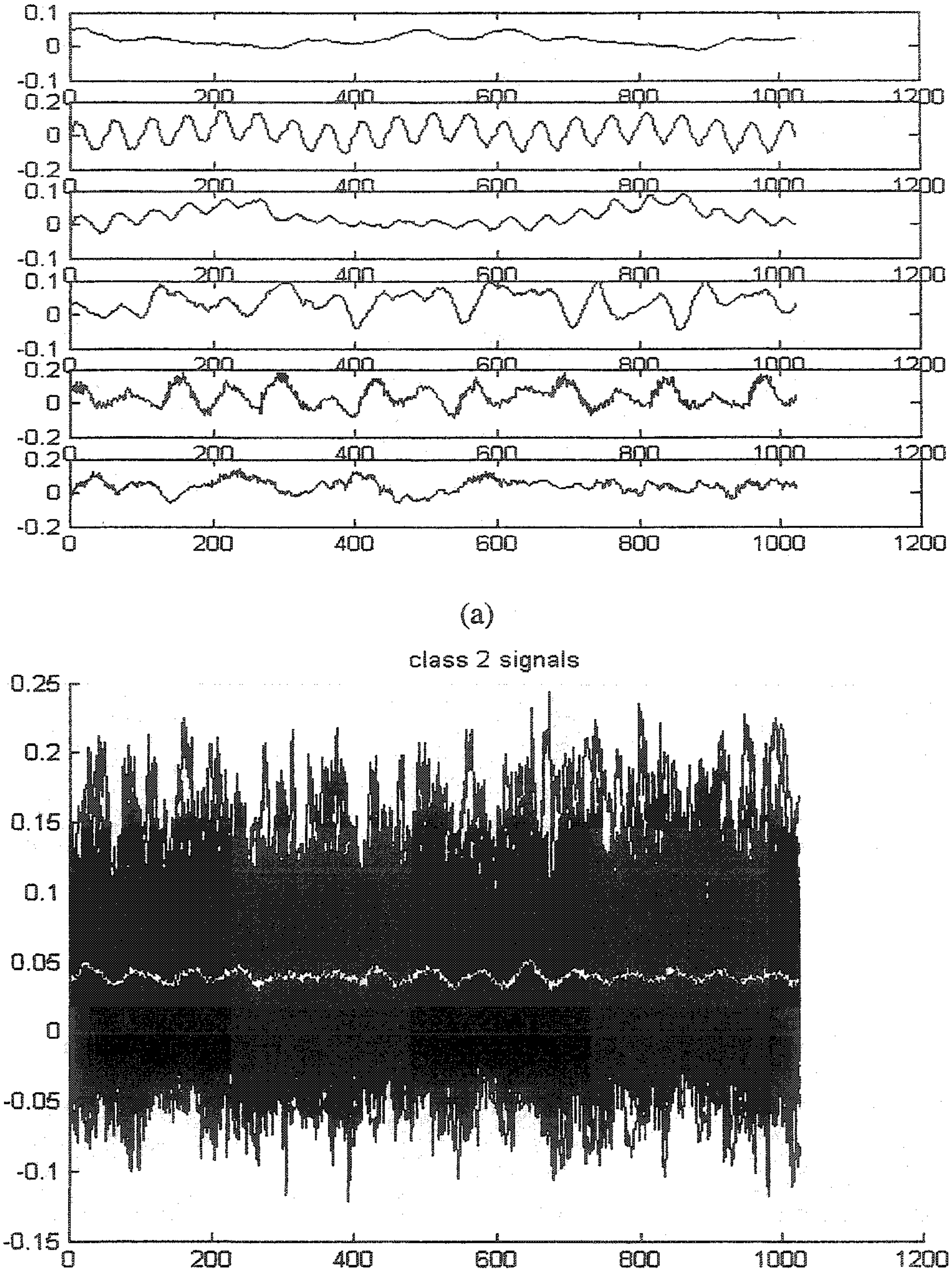


(i)



(j)

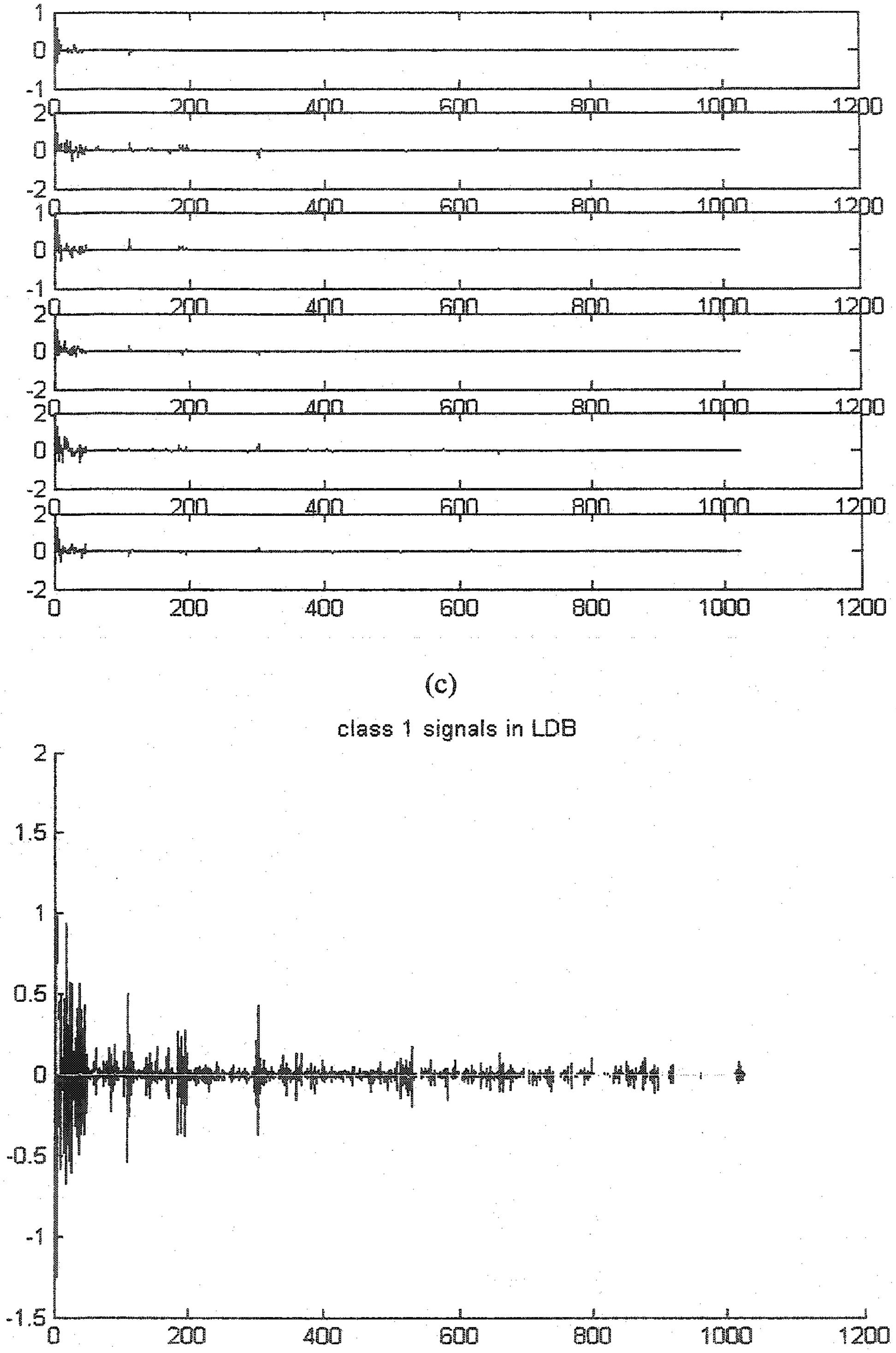
Fig.8.17 (continued)



(a)

(b)

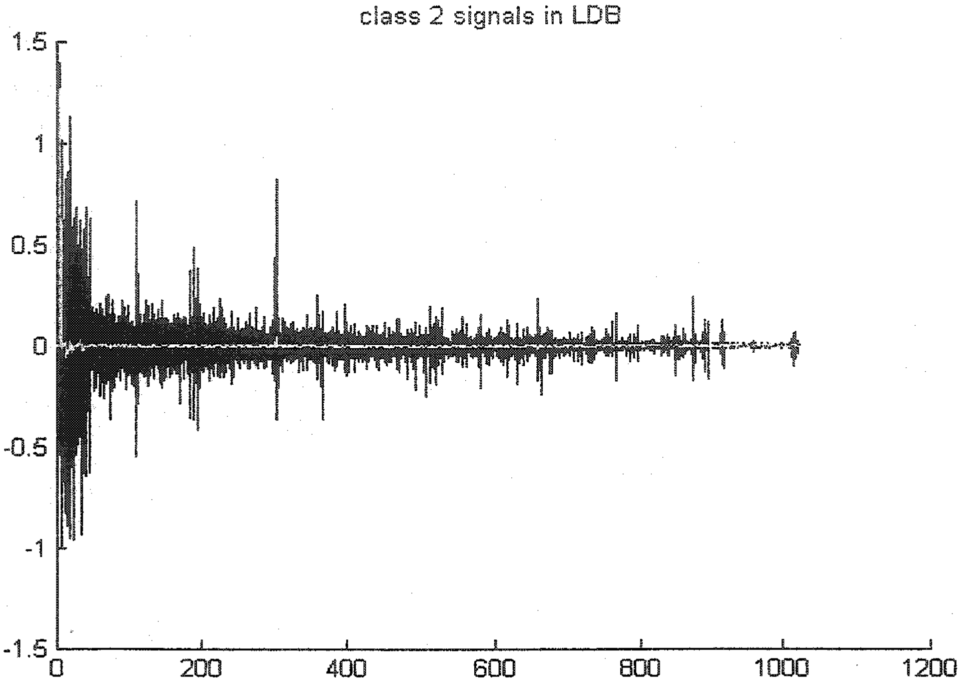
Fig.8.18 Classical/Popular Classification: (a) ensemble of three classical and three popular music segments from top to bottom (b) popular music signals shown collectively (c) ensemble of three rock and three classical pieces in LDB domain (d) pop signals shown collectively in LDB domain (e) the same for classical pieces (f) class averages (g) class averages in LDB domain (h) three top most discriminating LDB features (i) most discriminating feature as selected by LDB



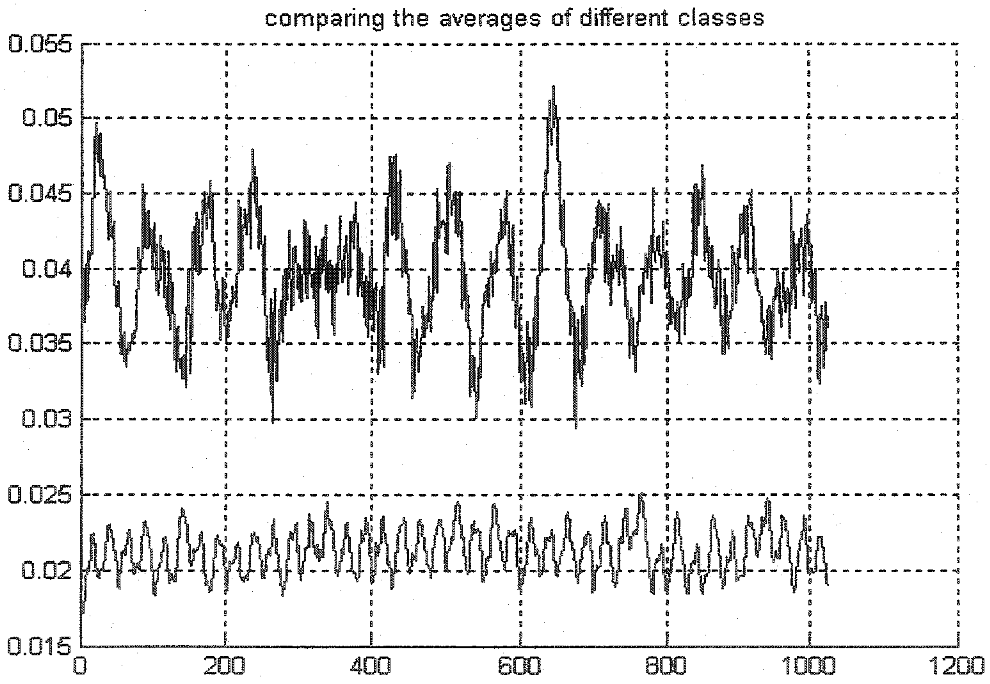
(c)

(d)

Fig.8.18 (continued)

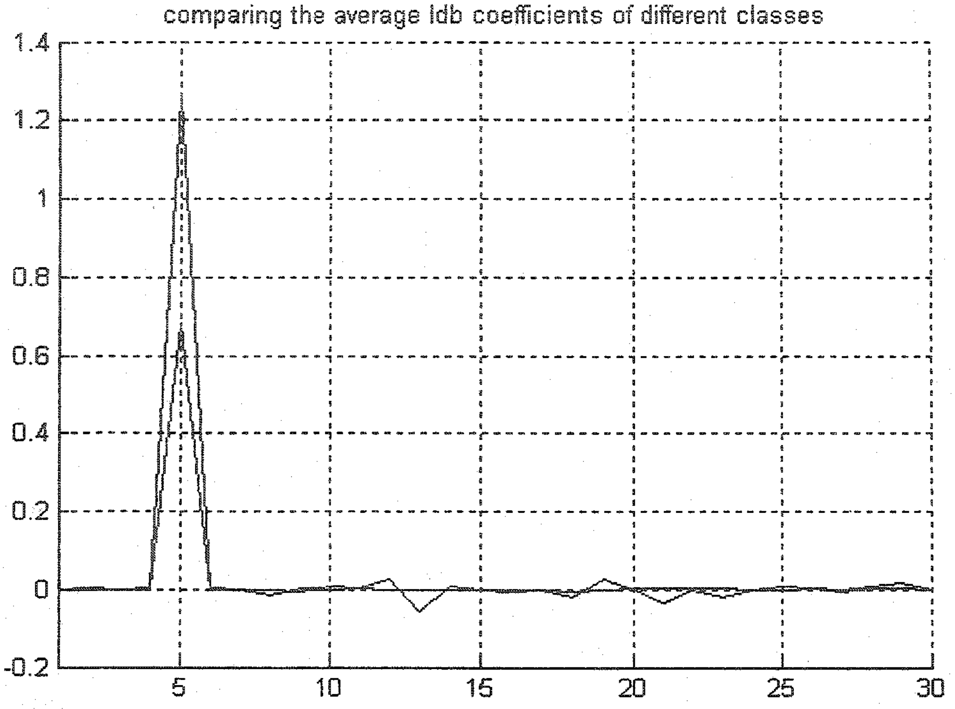


(e)

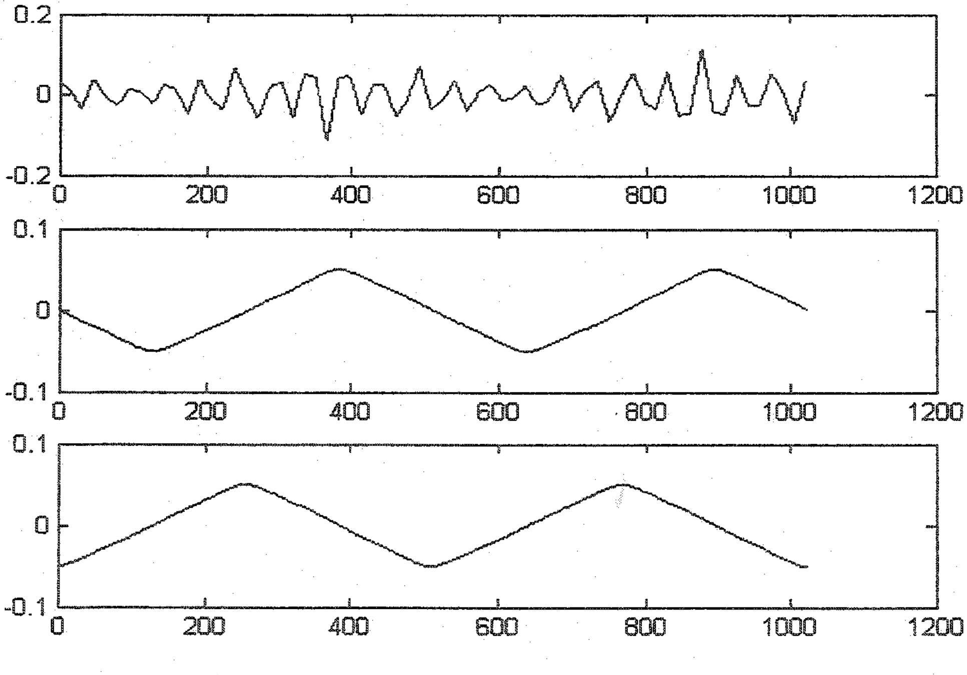


(f)

Fig.8.18 (continued)

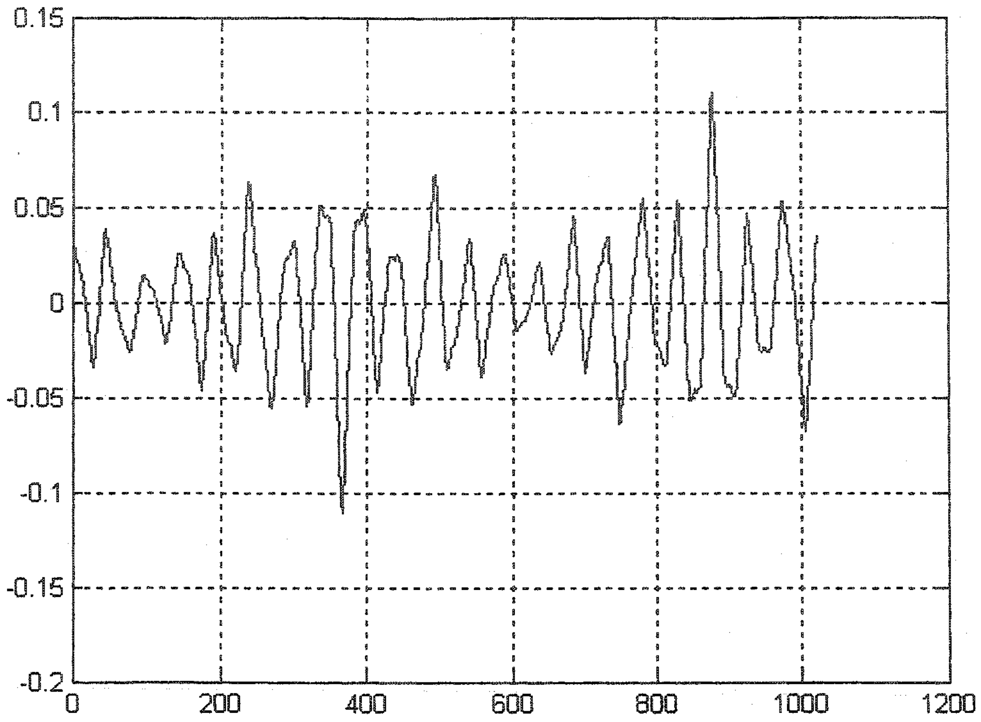


(g)



(h)

Fig.8.18 (continued)

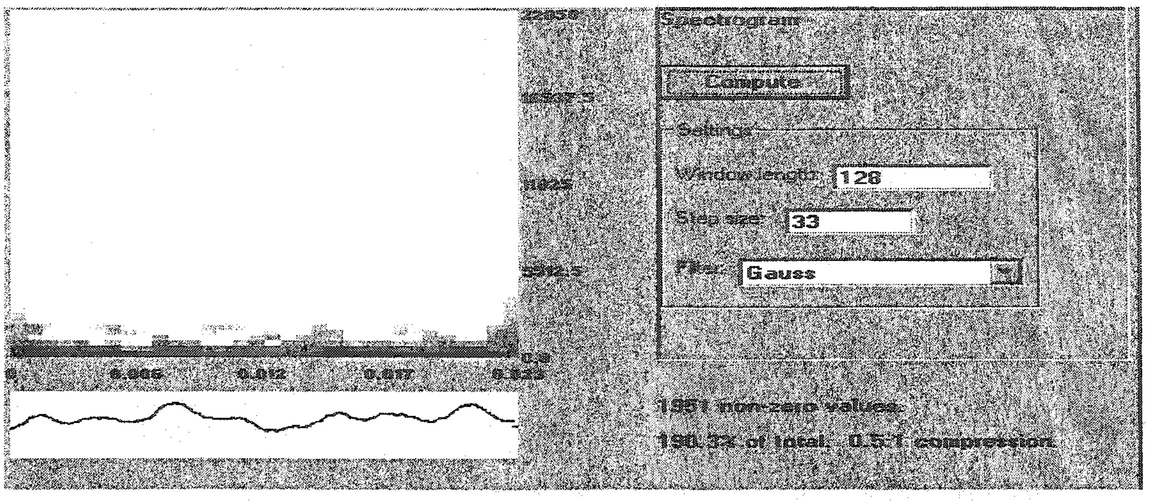


(i)

Fig.8.18 (continued)

8.6.3 Classifying Rock and Popular Music Segments

Herein, the efficiency of LDB algorithm in differentiating between rock and popular music is examined. As it is seen in the spectrograms of Fig.8.19 it is not easy to differentiate between these two classes even in time-frequency plane. This is basically due to the instruments that are used in common in these two categories. The algorithm parameters are set as in Section 8.6.1. Training and test database specifications are also the same as in Section 8.6.1. The results of this experiment are given graphically in Fig.8.20. The overall misclassification rate using LDA as classifier is 44.3396% when classifying the eight top most LDB vectors. Careful examination of Fig.8.20 (d) reveals the fact that signals from two classes have significant projection energy on more than first eight LDB vectors. This means that in order to get better classification one needs to feed more LDB features into the end point classifier. Classification can be improved by classifying more features, however, the number of features to be classified by LDA is upper bounded by the minimum number of training signals in each class. This is the LDA classifier limitation according to its implementation in Matlab®. Misclassification reduces to 35.3764% when the number of classified features is 105.



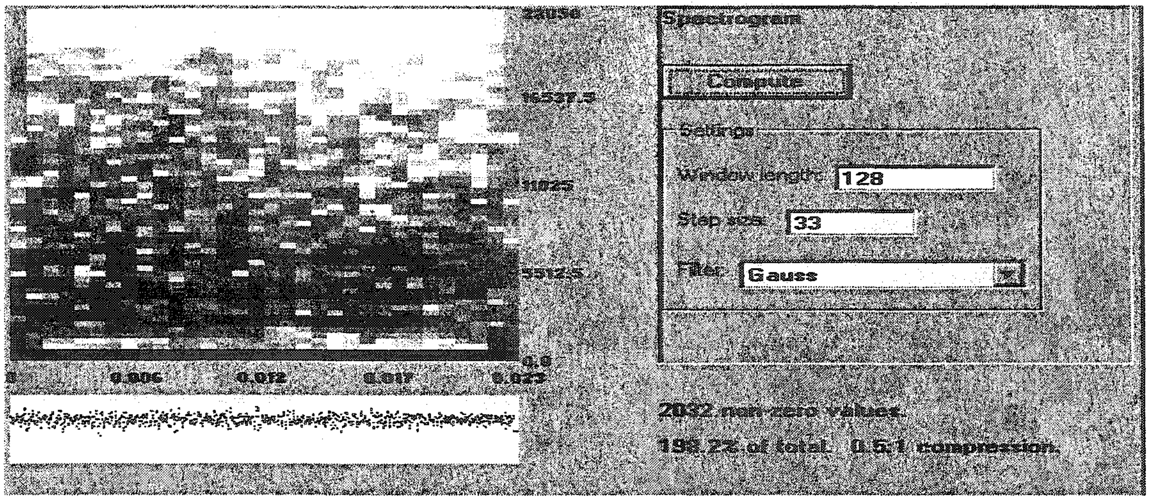
Window length: 128

Step size: 33

Filter: Gauss

1951 non-zero values.

190.3% of total. 0.5:1 compression.



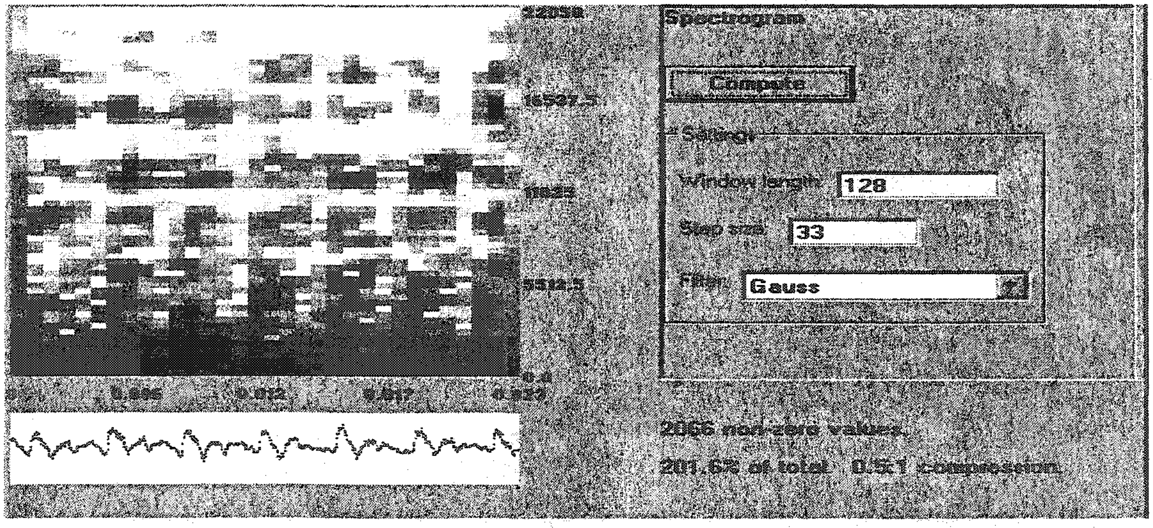
Window length: 128

Step size: 33

Filter: Gauss

2032 non-zero values.

198.2% of total. 0.5:1 compression.



Window length: 128

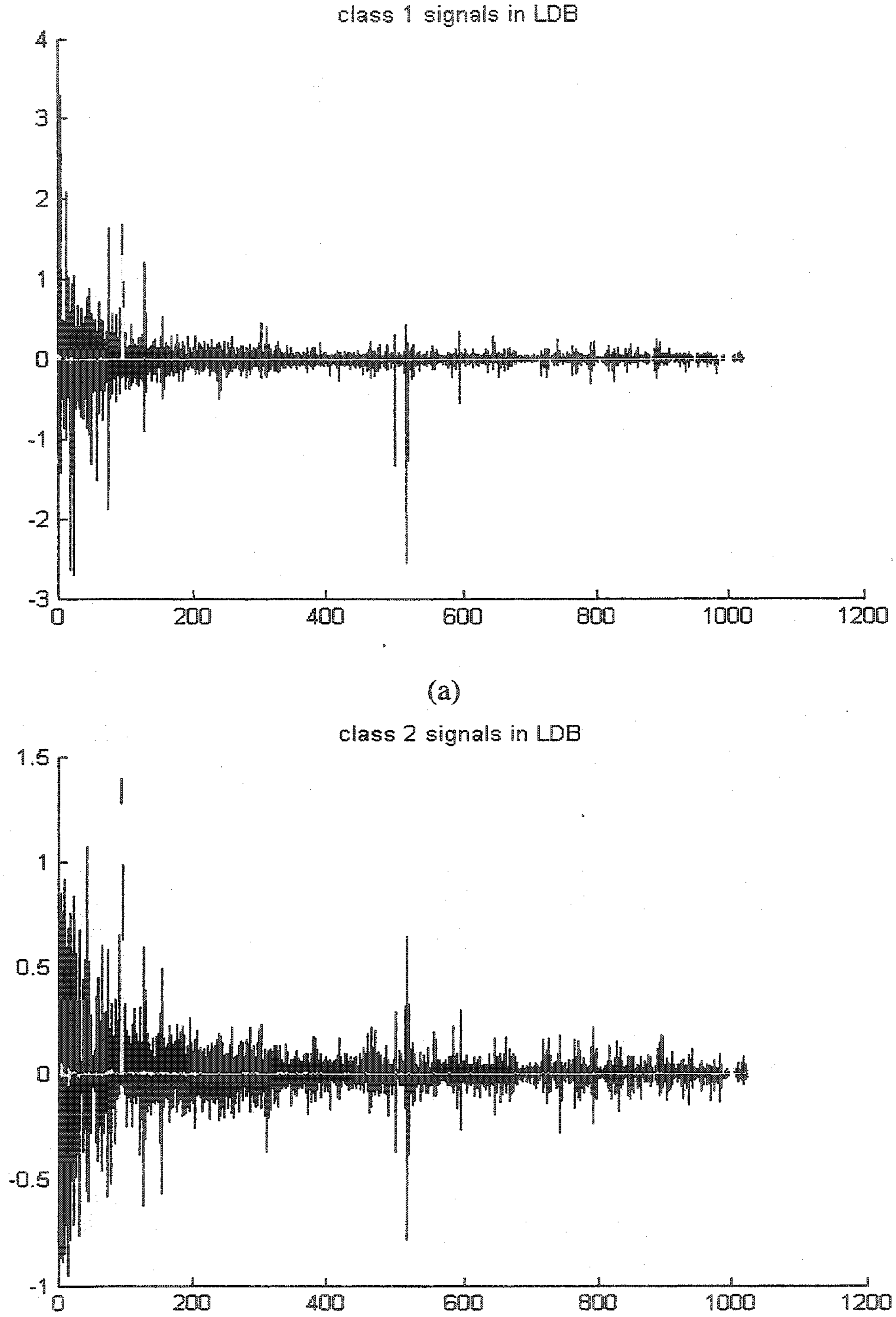
Step size: 33

Filter: Gauss

2066 non-zero values.

201.6% of total. 0.5:1 compression.

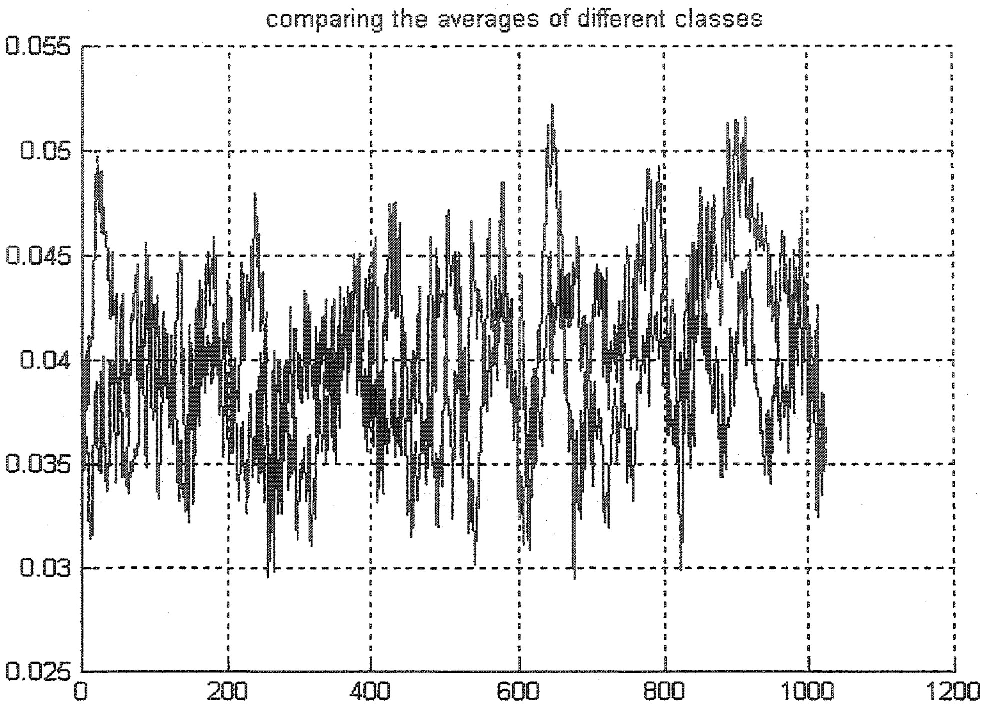
Fig.8.19 From top to bottom: spectrogram of classical, rock and popular 23.2 msec music segments; there is a clear difference between classical and the other two types



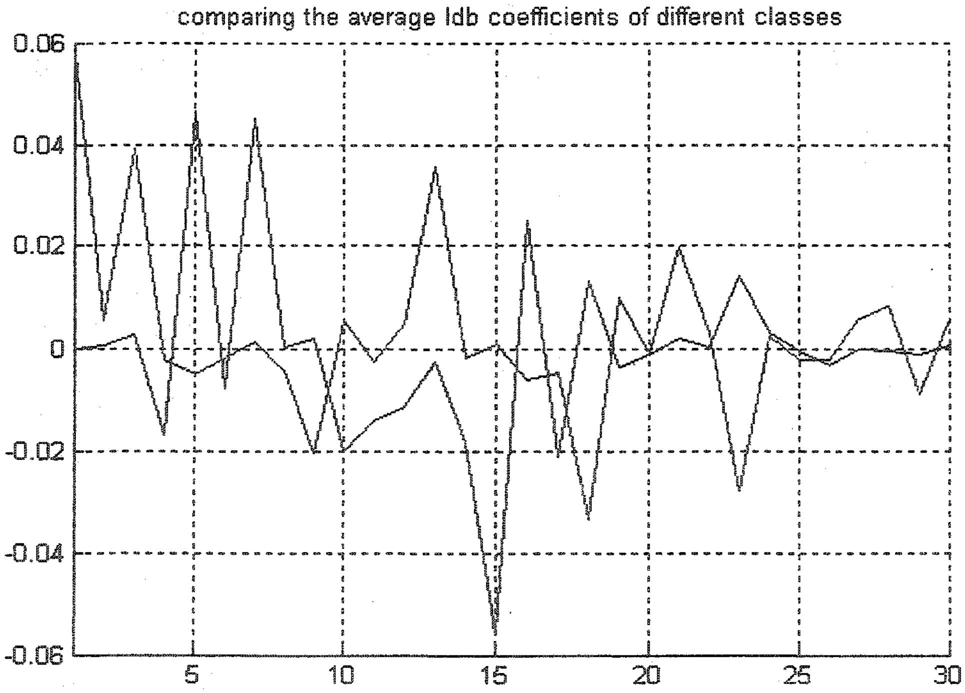
(a)

(b)

Fig.8.20 Rock/Popular Classification: (a) rock signals shown collectively in LDB domain (b) the same for popular pieces (c) class averages (d) class averages in LDB domain



(c)



(d)

Fig.8.20 (continued)

8.6.4 Classifying Classical, Rock and Popular Music Segments

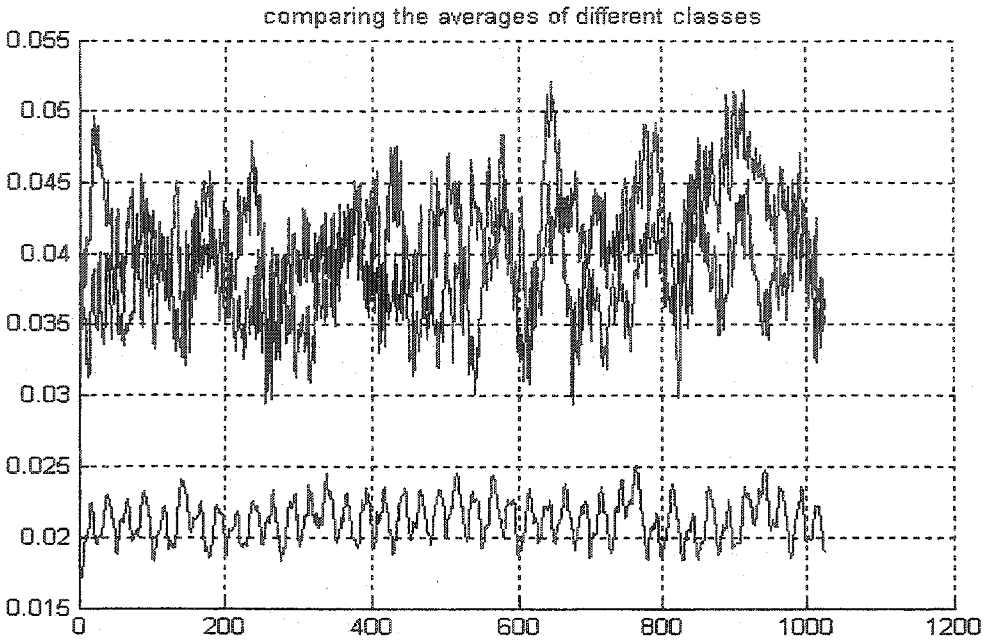
The performance of LDB algorithm in discriminating between three classes of sound signals is examined in this section. The segments used in this experiment are 1024 samples or 23.2 msec long. There are a total of 318 training signals selected equally from three signal classes. A similar database is used to test the performance of LDB. *J*-divergence is used as discrimination measure and LDA is the end point classifier. Coiflet 24 generates the wavelet packet used in this study. The outputs of the LDB as feature extractor are graphically represented in Fig.8.21. The overall misclassification rate between three classes is 29.8742% when only eight top most LDB vectors are fed into LDA classifier for classification.

8.6.5 The Effect of Music Segments Length

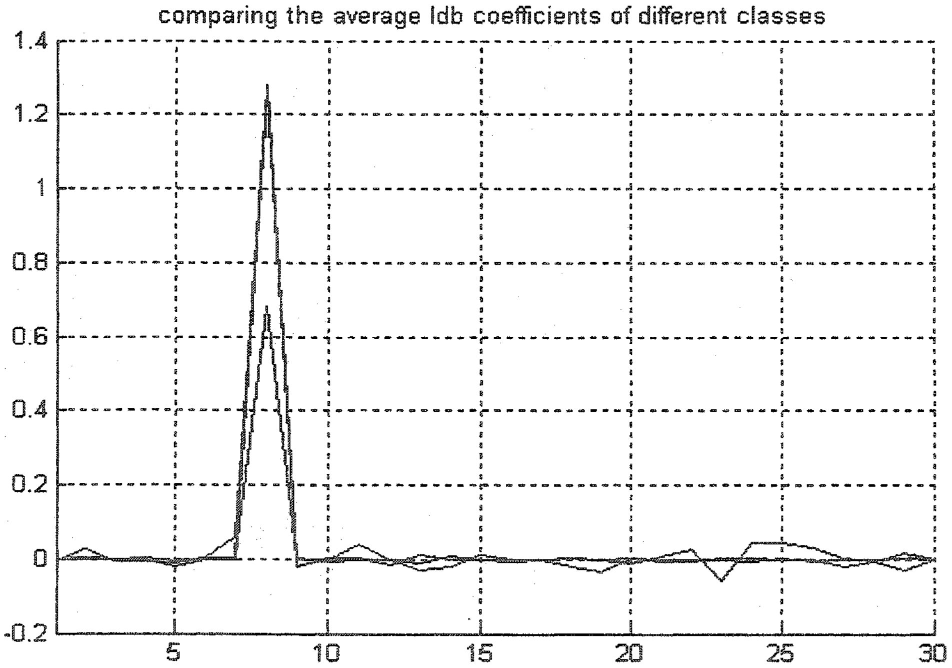
A separate experiment is performed in order to study the effect of length of the recordings used. A similar set up as described in Section 8.6.4 is used to capture the most discriminating features between classical, rock and popular music. However, the duration of each music piece is chosen to be 2048 samples that constitute a 46.4 msec recording. A set of 159 training signal equally distributed between three classes is used to train LDB algorithm. The test database is constructed similar to the training database. Fig.8.22 shows the results of this experiment. The most discriminating feature selected by LDB algorithm as depicted in Fig.8.22 (d) is symmetric around the 23.2 msec point that explains why this length is considered to be capturing audio signal features. The misclassification rate is 28.3019% in conjunction with LDA classifier that has little difference compared to misclassification rate obtained in Section 8.6.4.

In a similar investigation, 75 training signals of 92.8 duration from the three different classes under study are used to train LDB feature extraction module. A similar but independently generated database is used to test the LDB features efficiency in capturing signal characteristics. Fig.8.23 shows the output of the LDB algorithm. Note the pattern in most discriminating feature that repeats itself every 23.2 msec. Misclassification rate is 32 percent in this case when classifying the first 20 LDB vectors. Fig.8.23 (d) is a plot of misclassification rate versus the number of classified LDB features. It turns out that there is no special region of interest in this case so the whole signal duration is used as optimization window.

The length of the processed segment should be kept as small as possible since computational complexity of the algorithm is dependent upon the dimensionality of the input signal space. Thus one should avoid longer signal durations as far as possible to reduce computation time.

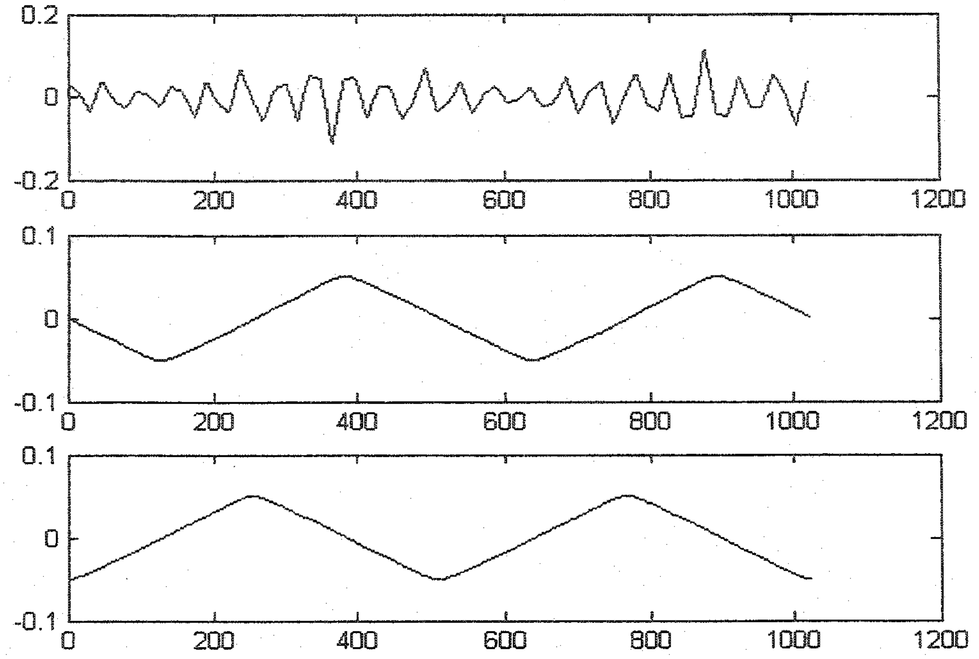


(a)

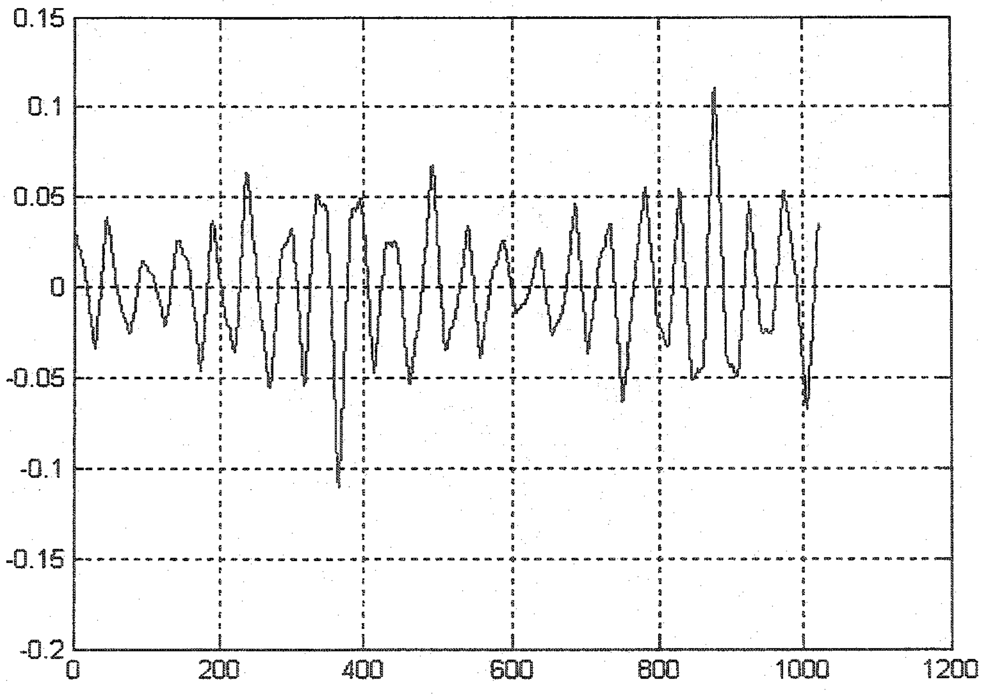


(b)

Fig.8.21 Classical/Rock/Popular Classification over 23.2 msec duration: (a) class averages (b) class averages in LDB domain (c) three most discriminating LDB functions (d) most discriminating LDB feature

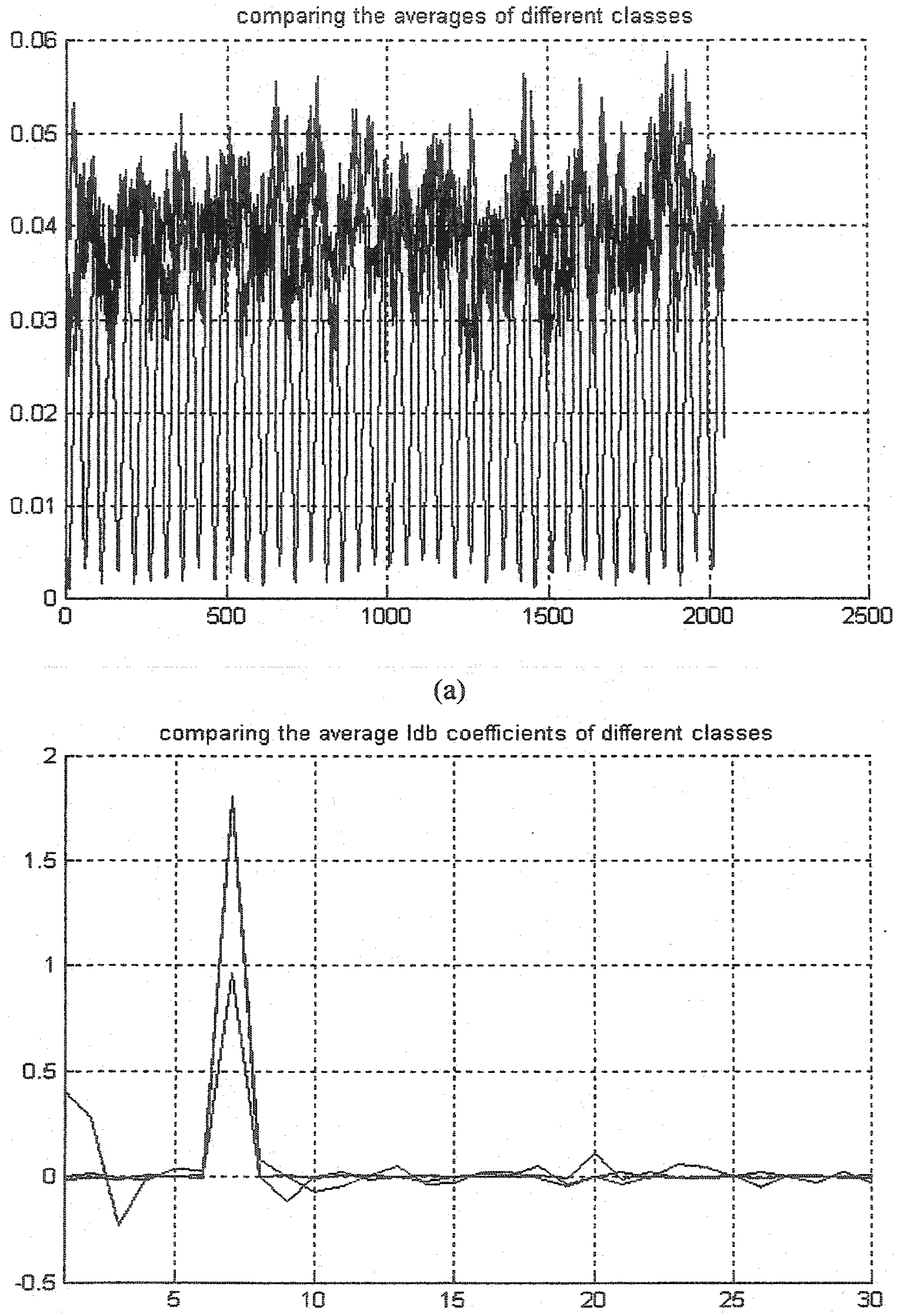


(c)



(d)

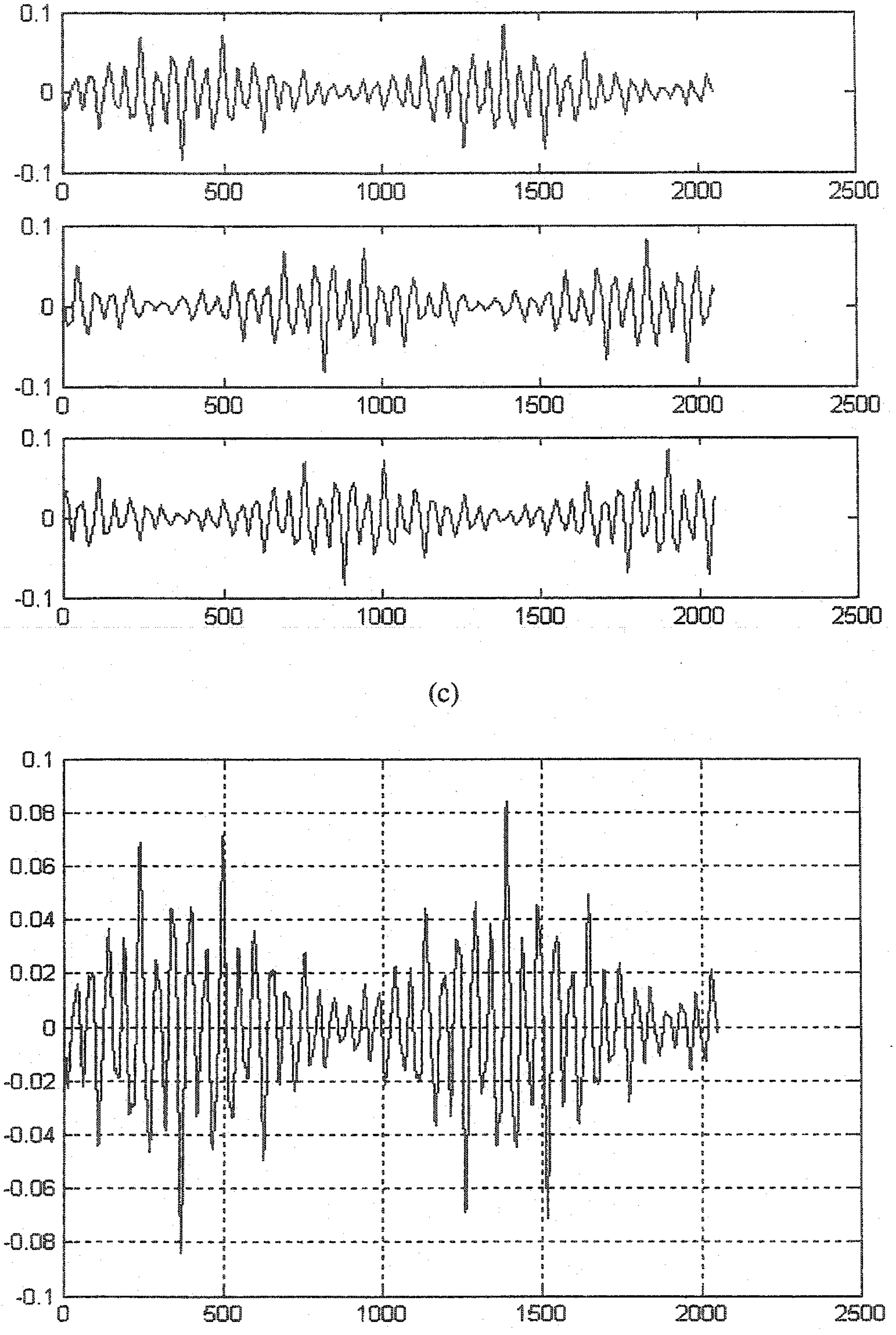
Fig.8.21 (continued)



(a)

(b)

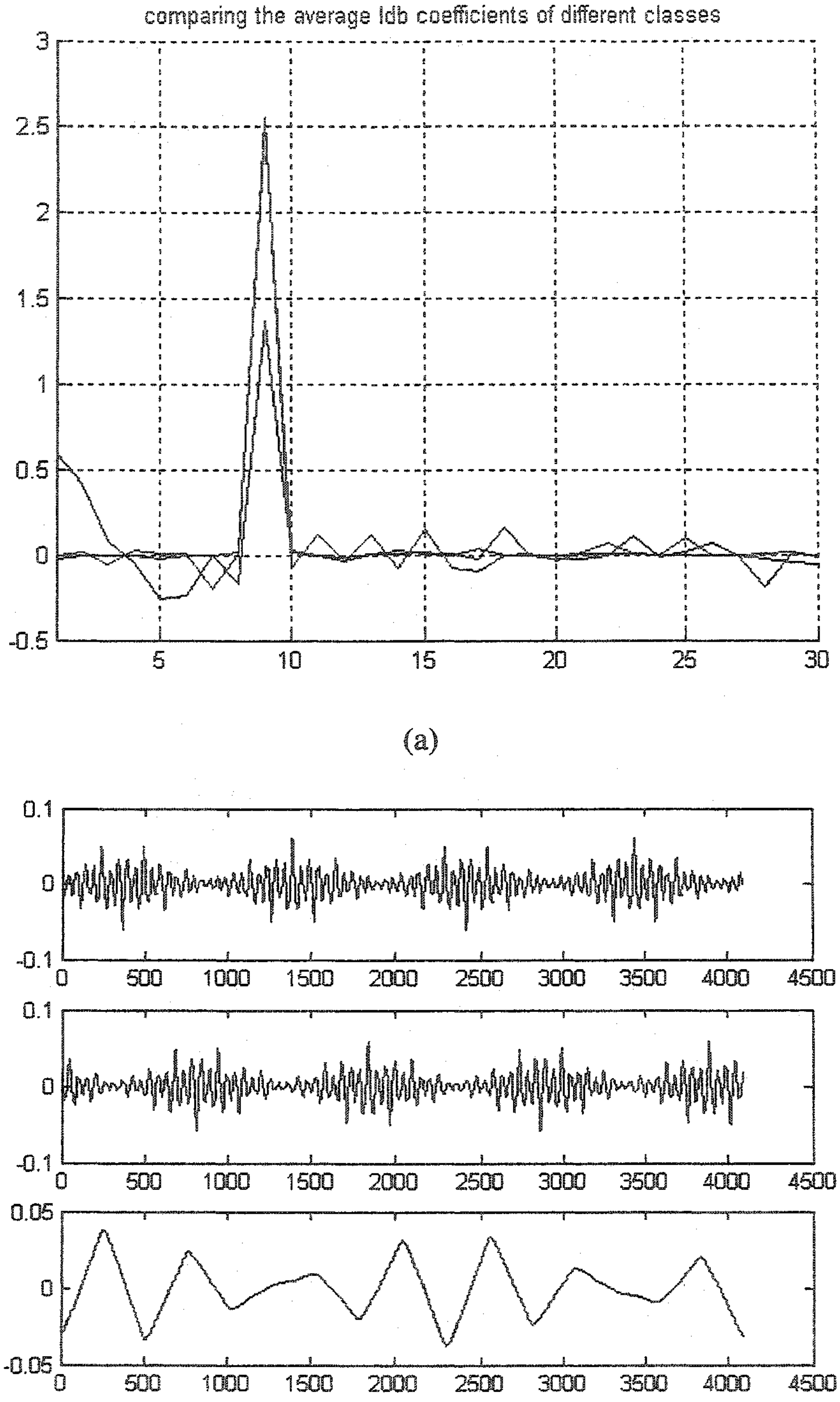
Fig.8.22 Classical/Rock/Popular Classification over 46.4 msec duration: (a) class averages in a 46.4 msec segment (b) class averages in LDB domain (c) three most discriminating LDB functions (d) most discriminating LDB feature



(c)

(d)

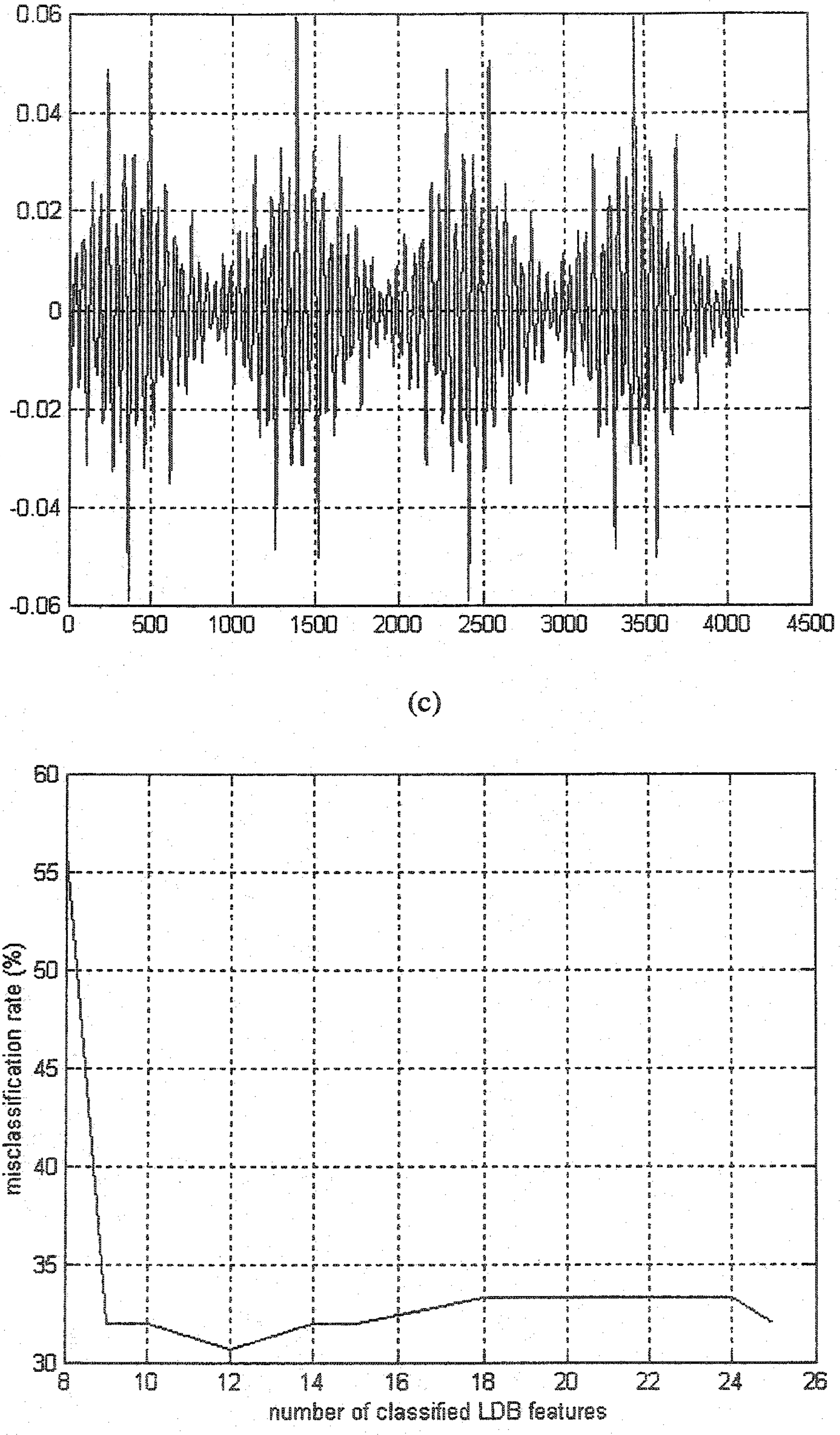
Fig.8.22 (continued)



(a)

(b)

Fig.8.23 Classical/Rock/Popular Classification over 92.8 msec duration: (a) Class averages in LDB domain (b) three most discriminating LDB functions (c) most discriminating LDB feature (d) misclassification rate as a function of classified LDB features



(c)

(d)

Fig.8.23 (continued)

Classes	Number of Training Signals	Duration (msec)	LDB Number Classified	Misclassification Rate (%)
Classical & Rock	212	23.2	8	2.3585
Classical & Pop	212	23.2	8	2.3585
Rock & Popular	212	23.2	8	44.3396
Rock & Popular	212	23.2	105	35.3774
Classical & Rock & Popular	318	23.2	8	29.8742
Classical & Rock & Popular	159	46.4	8	28.3019
Classical & Rock & Popular	75	92.8	20	32

Table 8.2 Summary of OLDB performance in classifying music signal segments

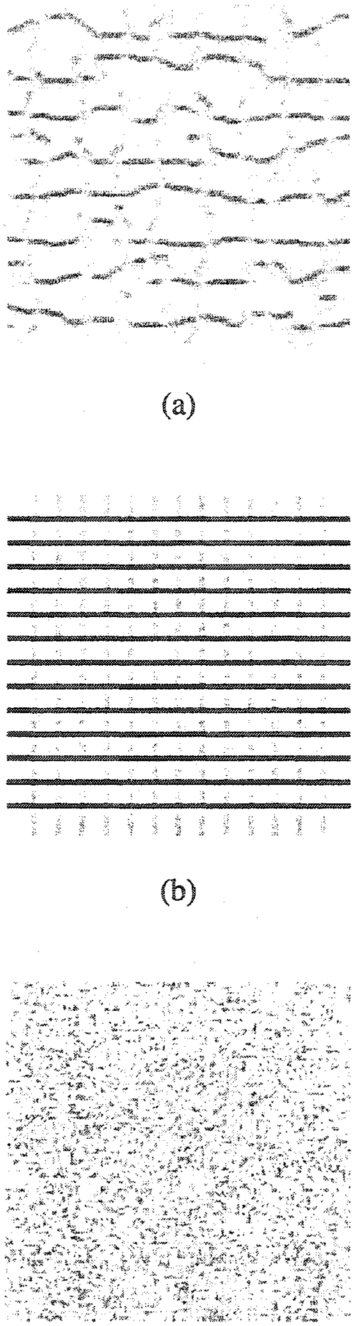
8.6.6 Summary

Table 8.2 summarizes the results obtained in this section for extracting discriminative features between audio signals related to music segments from different categories. Classification into three classes is a challenging problem and OLDB performance in this problem with around 70% accuracy is satisfactory.

8.7 Texture Classification

Motivated by applications of original local discriminant basis algorithm for target detection in defense industry we designed a basic texture classification problem to test the applicability of optimized LDB ideas in image processing context.

In this section, the performance of optimized LDB in image processing applications is studied in a three-class texture classification problem. The textures to be classified are 128×128 grayscale 8-bit images depicted in Fig.8.24. The textures are called "mosaic," "patch" and "grain" from top to bottom. LDB feature extraction module is trained with 64×64 sub-images from these textures. Each 64×64 sub-image is selected by randomly choosing its lower left corner in the original image. Since the goal is to analyze the texture, not the location of individual edges or transients, which form texture elements, lack of translation invariance is problematic in this application. The spin cycle method of Section 5.2 is employed to overcome the problem of translation invariance.



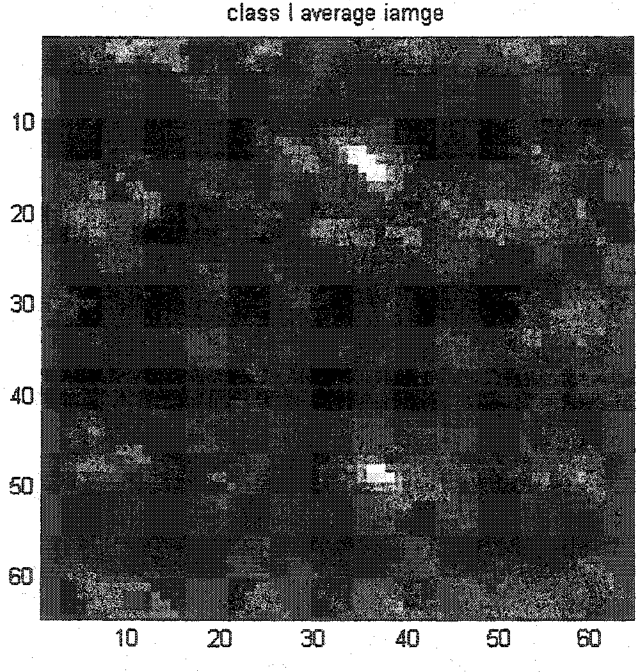
(a)

(b)

(c)

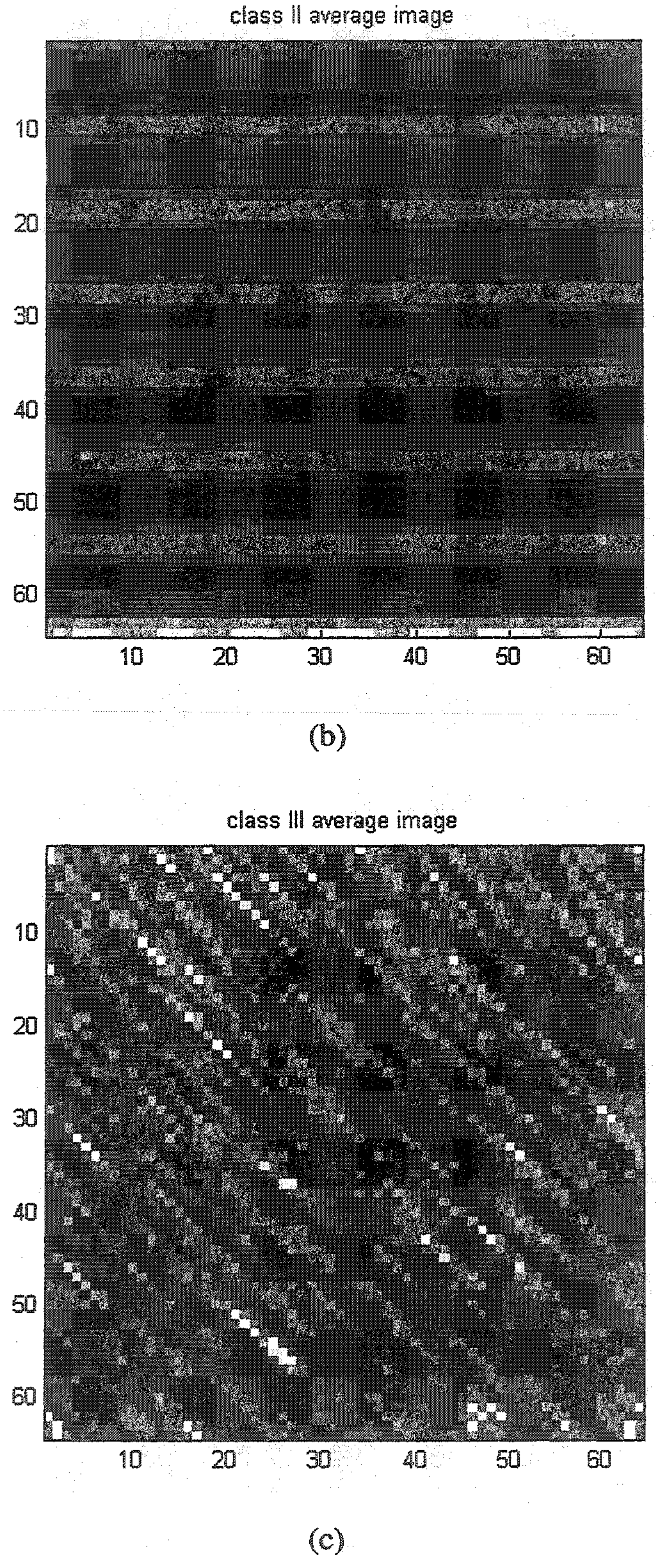
Fig.8.24 Texture classification: (a) “mosaic” texture (b) “patch” texture (c) “grain” texture

A total number of 300 sub-images are constructed for each class by randomly specifying fifty 64×64 sub-images of the original 128×128 texture and translating each sub-image along its diagonal by five pixels. In translation operation the boundary conditions are periodic. A similar but independent set of images is constructed to be used as test images to examine the efficiency of selected LDB. Daubechies 20 quadrature mirror filter generates the wavelet packet structure that is a quadratic tree. *J*-divergence is the discrimination measure. Linear Discriminant Analysis (LDA) is used as classifier in this study. Fig.8.25 to Fig.8.29 summarize the output of 2D local discriminant basis feature extraction module. Class averages are depicted in Fig.8.25 (a), (b) and (c). On the other hand, Fig.8.26 (a), (b) and (c) represent class averages in the selected LDB domain. Top most discriminating LDB features are shown in Fig.8.27 (a) to Fig.8.27 (f). Filtered class images by keeping the first 30 most discriminating LDB functions are depicted in Fig.8.28 (a), (b) and (c). As seen in most discriminating LDB feature shown in Fig.8.27 (a) the upper left 10×10 corner of the image space plays a crucial role in classifying the images. This block seems to contain enough data for discriminating between classes. This is a basic 10×10 block that in this particular problem and database differentiates between classes better than other parts of the image. The optimal weighting idea of Section 8.2 applies in here. Fig.8.29 shows the results of optimal LDB performance after 20 iterations. One is limited to use at most 300 LDB features for classification due to LDA classifier implementation in Matlab®. Classification between three classes of images is challenging and the 85% accuracy obtained by OLDB algorithm is satisfactory.



(a)

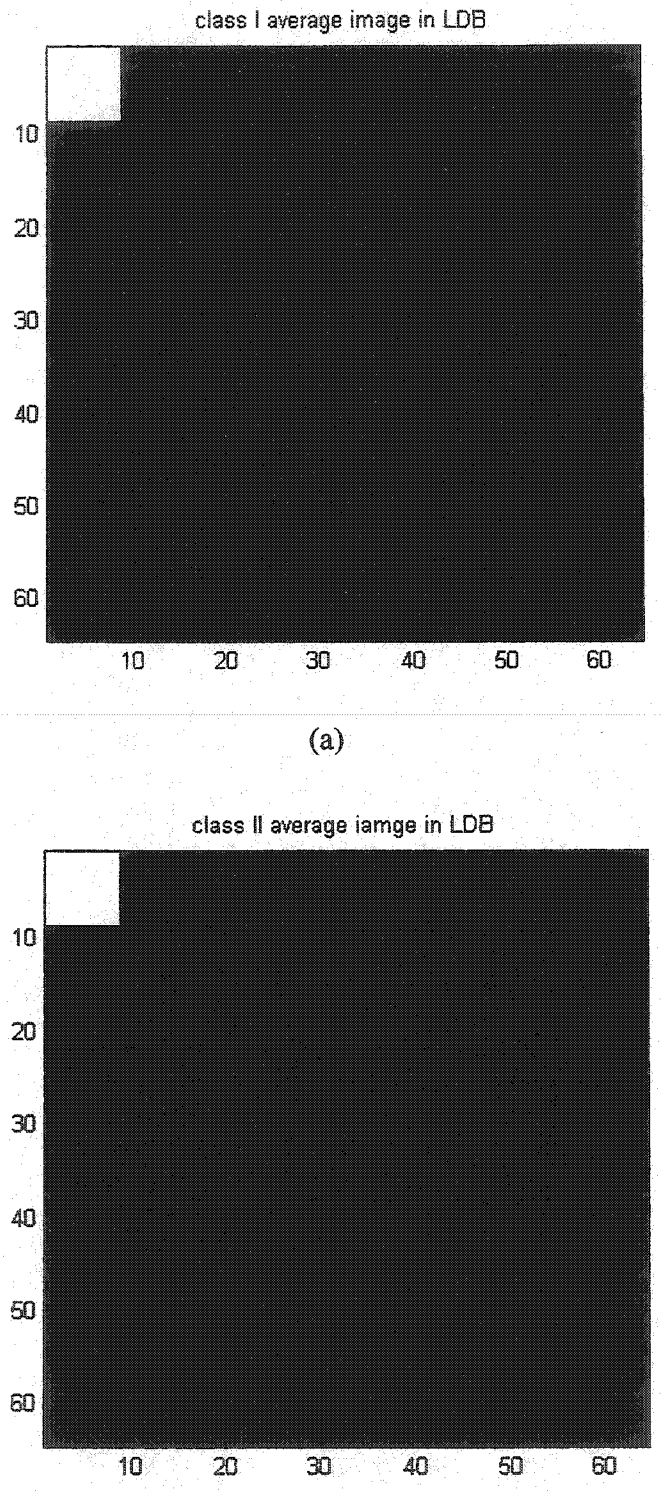
Fig.8.25 "mosaic," "patch" and "grain" texture averages depicted in (a), (b) and (c) respectively



(b)

(c)

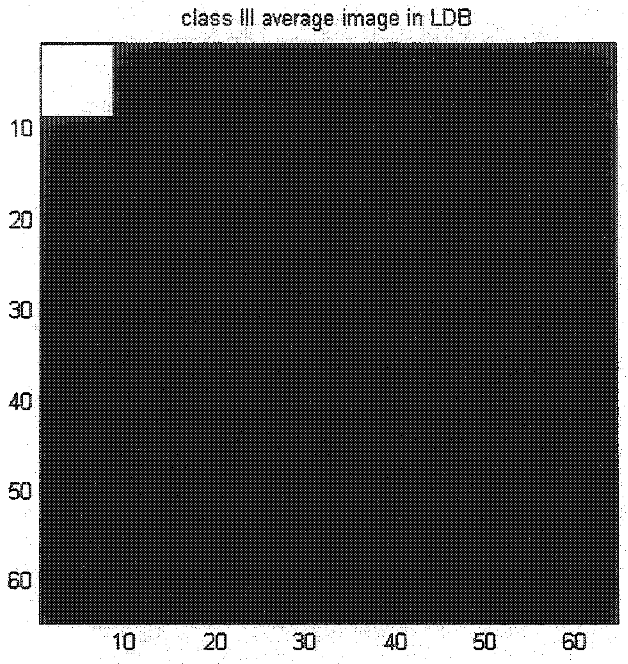
Fig.8.25 (continued)



(a)

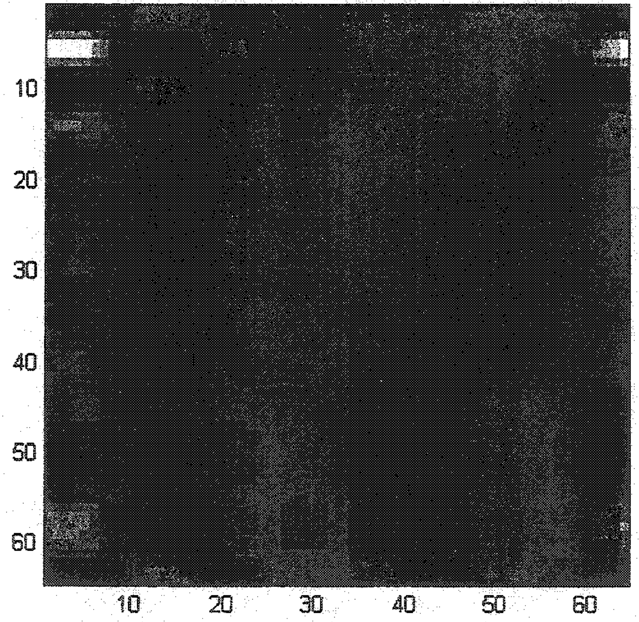
(b)

Fig.8.26 "mosaic," "patch" and "grain" texture averages depicted in (a), (b) and (c) respectively



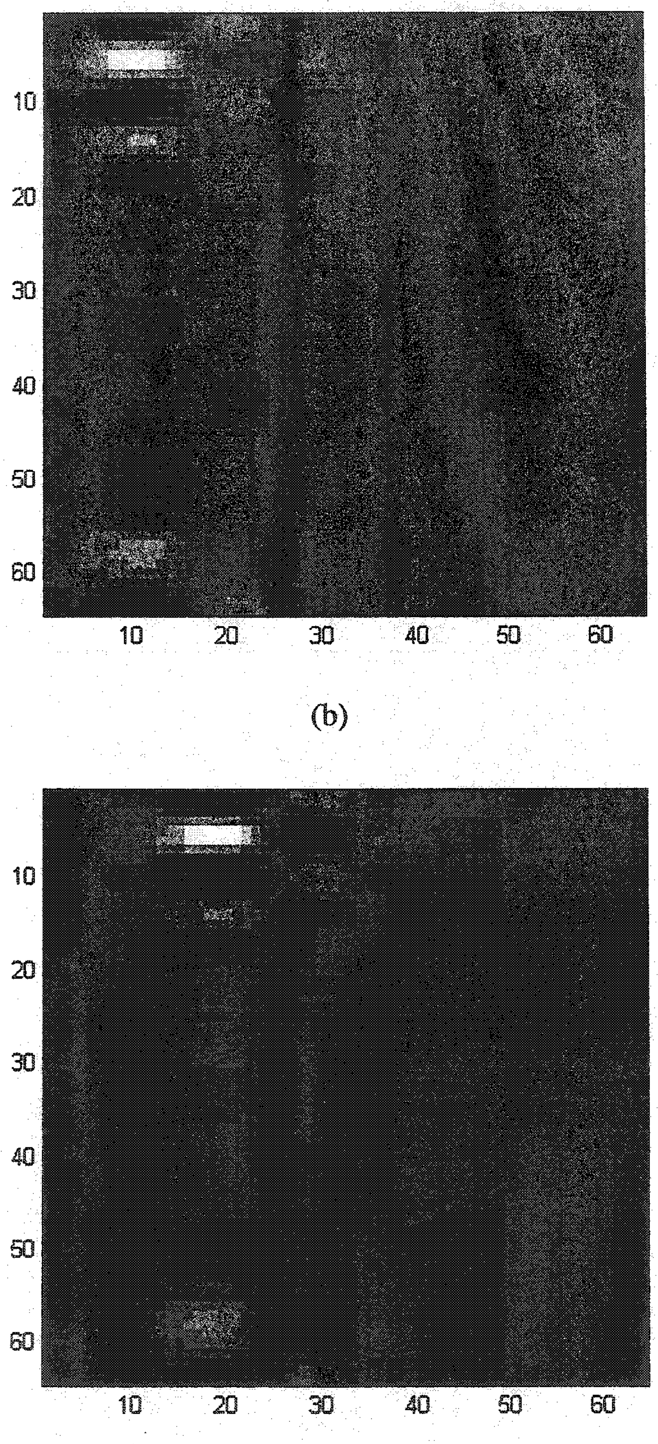
(c)

Fig.8.26 (continued)



(a)

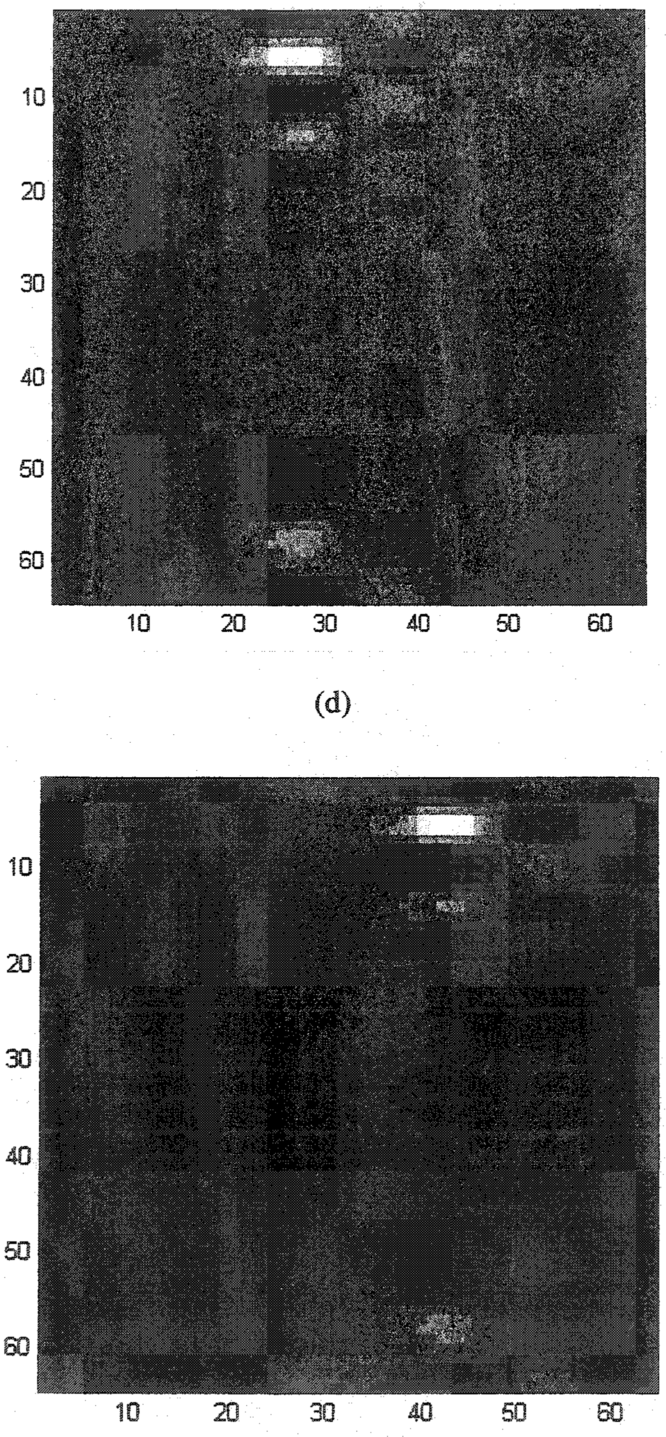
Fig.8.27 Six top most discriminating LDB features ordered from (a) to (f) respectively



(b)

(c)

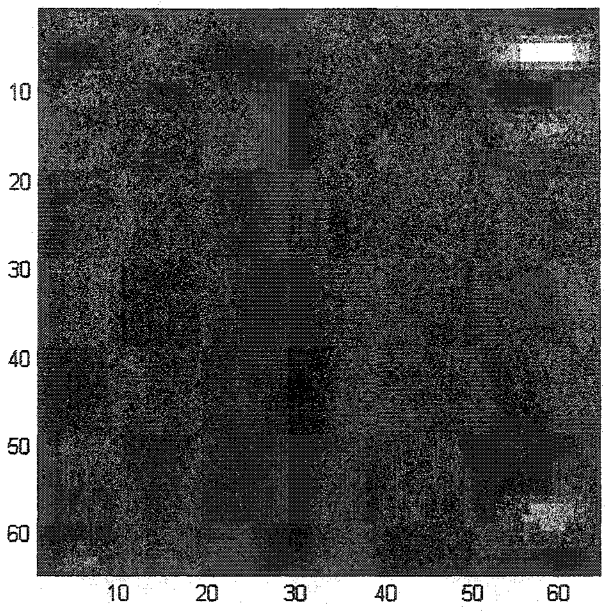
Fig.8.27 (continued)



(d)

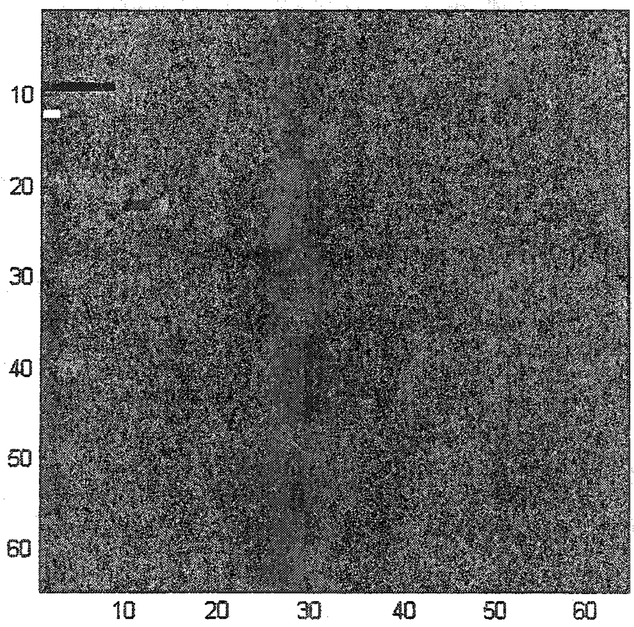
(e)

Fig.8.27 (continued)



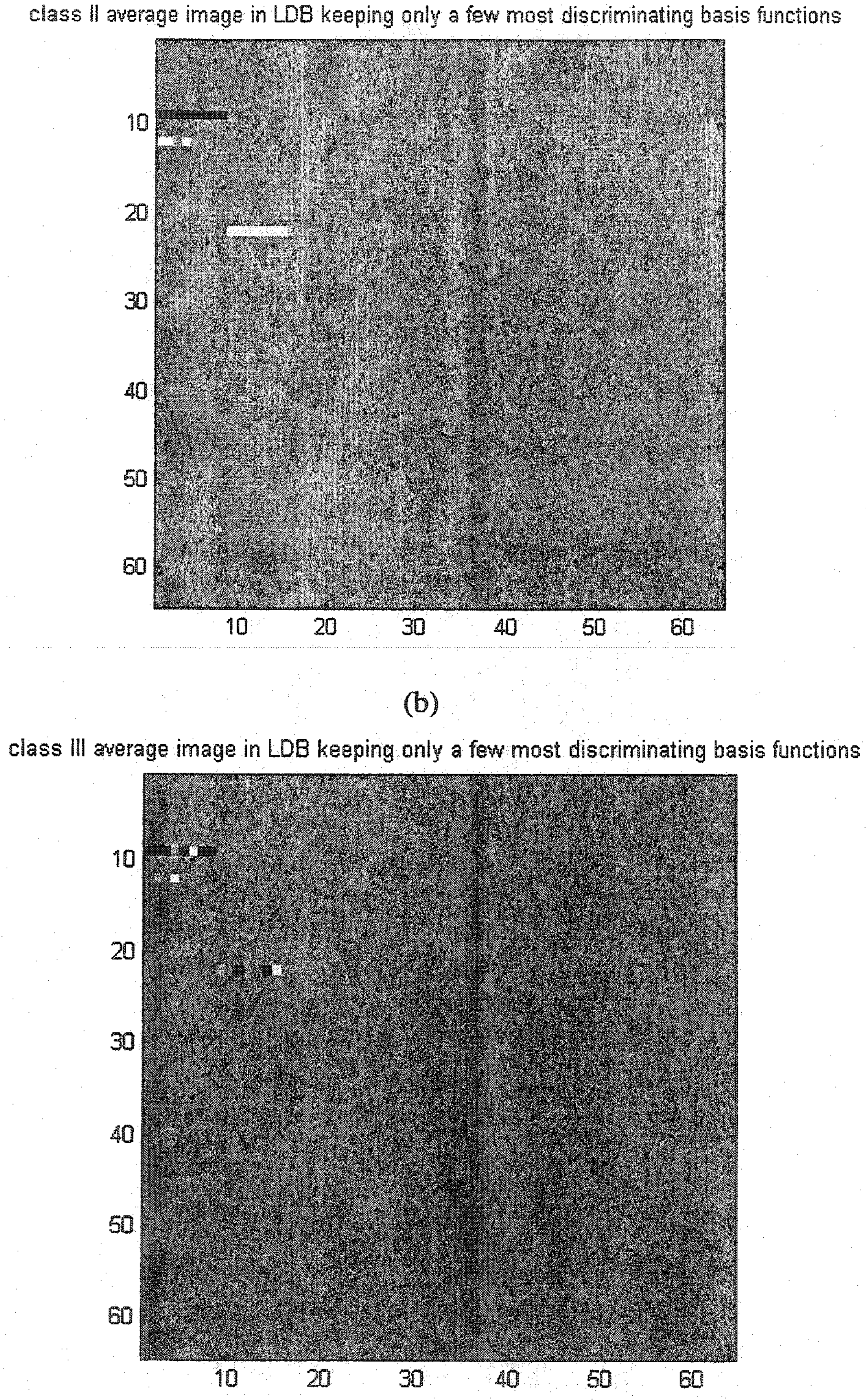
(f)

Fig.8.27 (continued)


class I average image in LDB keeping only a few most discriminating basis functions

(a)

Fig.8.28 “mosaic,” “patch” and “grain” texture averages in LDB domain depicted in (a), (b) and (c) respectively



(b)

(c)

Fig.8.28 (continued)

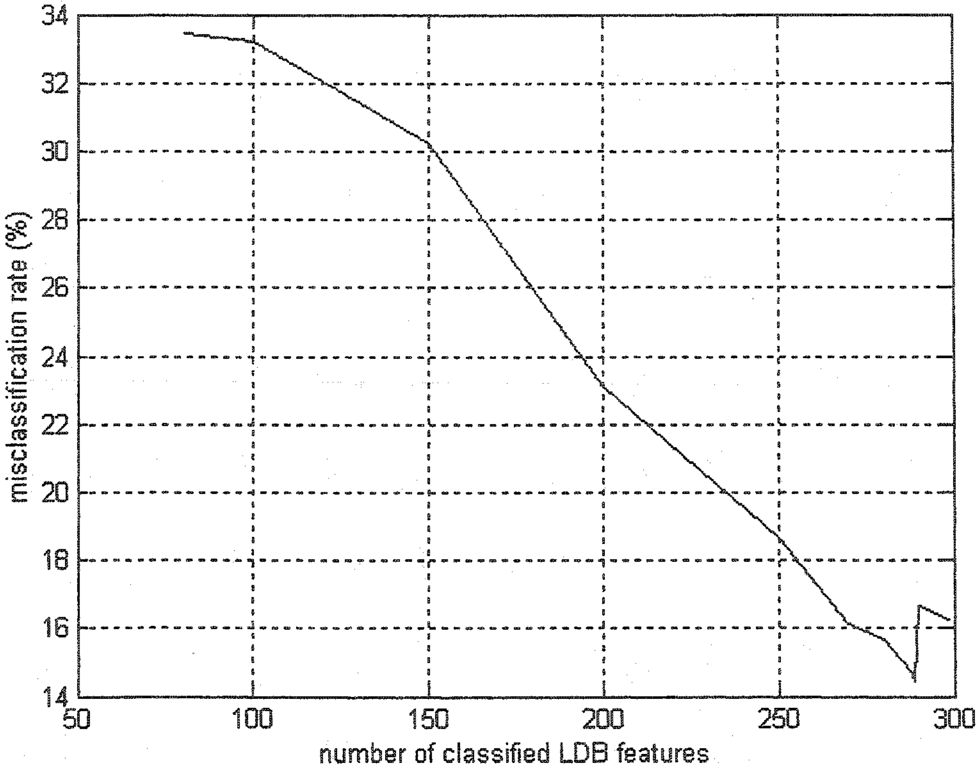


Fig.8.29 Misclassification rate as a function of OLDB features classified

Chapter Nine

Noise Analysis and Other Properties of Optimized Local Discriminant Bases

There are several potential applications for optimized local discriminant basis (OLDB) method in biometrics and biomedical signal processing where its ancestor, Wickerhauser's best basis algorithm [4], has proved to be an efficient technique [5]. Some of the principal properties that need to be studied in practice are the behavior of the algorithm at different noise levels and the effects of changing the number of training signals or the number of optimized and selected LDB features. These are essential steps prior to making primary decision about the applicability of LDB in a certain application area. In this chapter a noise analysis is performed on a traditional 1D criterion classification problem adopted from [34] with different number of selected features for classification and optimization and for both original [7] and optimized versions of the algorithm (Chapter eight). For a detailed description and properties of this example known as "triangular waveform classification" refer to Section 4.7. The effects of changing the number of training signals and type of time-frequency dictionary are studied simultaneously. The results show the effectiveness of the optimization block at different SNR levels. With the aid of the shape classification example of Section 4.8 in image processing context the noise analysis and properties of image LDB are studied in two dimensions. Noise level is increased in five steps that help studying the performance of OLDB algorithm.

A serious problem in practical applications of machine object recognition (Section 4.1) is the presence of colored noise introduced by sensors and data acquisition machinery. While the properties and behavior of systems are best modeled and studied in presence of white Gaussian noise, the analysis and studying the behavior of systems in presence of colored noise poses a challenging problem. Specially colored noise tends to impair the performance of existing feature extraction schemes dramatically. In this chapter, stability of OLDB in presence of colored noise is studied in detail. Two separate experiments are performed to test the performance and stability of OLDB in presence of low frequency colored noise (textured background) in Section 9.4 and high frequency colored noise in Section 9.5.

While a number of existing feature extraction and classification schemes prove to be efficient in discriminating between two classes of input signals, they fail to be as useful in classification into more than two classes, which is a more realistic and challenging approach. OLDB as studied in synthetic examples of Section 4.7 and Section 4.8 and practical applications of Section 8.6 and Section 8.7 is an efficient tool in discriminating between more than two classes. Another related concern in practical applications is the difference in the number of training signals that are available from different signal classes. Signal/image processing engineer is usually faced with a database containing different number of available signals in different classes. This is basically due to the different probability of occurrence for each signal class in real world applications. The distribution of benign cancer in human population and its occurrence is not the same as malignant cases. Some conditions especially in biometrics and biomedical applications happen rarely and the existing databases try to catch as many of all different cases as possible. Section 9.6 discusses the stability of OLDB algorithm when the available number of training signals in different classes varies in a large range including extreme cases.

9.1 OLDB Algorithm Noise Analysis in One Dimension

For the purpose of illustration, we adopt example 5.2 of [2]. In this classification problem three classes of signals are generated by the process given in (78). For a complete description of this problem refer to Section 4.7.

$$\begin{aligned} c(i) &= (6+\varsigma).\chi_{[a,b]}(i) + N\varepsilon(i) \\ b(i) &= (6+\varsigma).\chi_{[a,b]}(i).\frac{i-a}{b-a} + N\varepsilon(i) \\ f(i) &= (6+\varsigma).\chi_{[a,b]}(i).\frac{b-i}{b-a} + N\varepsilon(i) \end{aligned} \tag{78}$$

" a " is an integer-valued uniform random variable on the interval [16,32]. " $b-a$ " also obeys an integer-valued uniform distribution on [32,96]. " ς " and " ε " are the standard normal variates, and $\chi_{[a,b]}(i)$ is the characteristic function on $[a,b]$. $c(i)$ is called the 'cylinder' class whereas $b(i)$ and $f(i)$ are known as 'bell' and 'funnel' classes. $N \geq 0$ determines the *SNR* of the generated signal space. For a discussion on importance of this example as a criterion synthetic problem for classification algorithms refer to [34]. The statistical averages of the each class in the above process over a set of 360 training signals with $SNR \approx 25$ are depicted in Fig.4.7. The statistical average of each class will be windowed around the regions of interest to be used as teacher signal in the proposed optimization block. A slightly different procedure is performed to obtain teacher images that locate microcalcifications in mammograms [38]. The experiments where performed with *N* varying linearly that corresponded to $SNR \approx 37{,}25{,}18{,}13{,}9$. Fig.4.8 shows the average LDB coefficients for different classes of signals and at $SNR \approx 25{,}9$.

It is easily seen in Fig.4.8 that as SNR decreases, the number of significant LDB vectors or selected features also decreases. For signals with low SNR, the features are heavily buried in white Gaussian noise and more difficult to identify. For this reason, the number of LDB vectors used in the optimization block and the number of LDB features fed into LDA classifier are decreased linearly and in five steps from 18 to 10 and from 12 to 8 respectively. Another consideration was toward that of window size that was used for generating teacher signals. As the level of noise is increased, it gets more and more difficult to identify the characteristics of each single class as seen in Fig.4.7. The window was initially set at [40, 60] interval and linearly expanded to [30, 80] in five steps as the noise level increased. This can be considered as guarding regions of interest with a bigger safety factor as discussed in chapter five. The number of signals used in optimization block (*K*) was set to be one tenth of the number of training signals in every experiment. Variable fourth order rational function of Fig.9.1 was used to change the learning rate accordingly (Section 8.3). An equal number of test signals were generated in order to evaluate the efficiency of the overall process. Linear Discriminant Analysis (LDA) classifier was used to classify original and optimized LDB features.

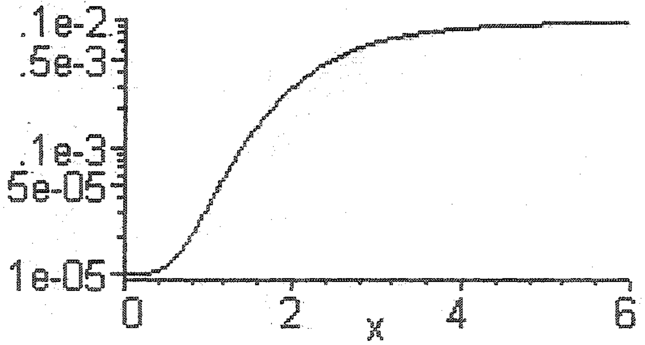


Fig.9.1 The learning rate change pattern as given by $\dfrac{ax^4+b}{cx^4+d}$

9.2 Simulation Results and Discussion in One Dimension

Fig.9.2 shows the optimization process over 360 training signals at different SNR levels, by exploiting Coiflet 4 wavelet packet. In all experiments symmetric differential entropy was used as discrepancy measure (Section 4.5). Note that in each experiment the number of training and test signals are equal. Coiflet 12 wavelet packet was used to generate the data shown in Fig.9.3 with the same number of training signals. Data shown in Fig.9.4 was generated by a set of 120 training signals with Coiflet 4 wavelet packet. Note that since the same number of test signals is used to calculate misclassification rates, the curves in Fig.9.4 are more optimistic compared to graph in Fig.9.2 where a collection of 360 signals is used to train and test the performance. The misclassification rates show a remarkable drop at SNR = 9 in Fig.9.3 and at SNR = 18, 13 in Fig.9.2 and Fig.9.4. This is due to the correlation of noise structure with the statistical feature structure resulting from (92) which is in part dependant on the dictionary used for analysis. In other words, at a certain noise level, noise helps to promote the difference between classes. Optimization process proves to be useful in all experiments. Particularly it is most efficient at SNR = 18, 13 and 9. Decreasing the number of training signals directly impairs the accuracy of LDB algorithm as shown in Fig.9.4.

As observed in other similar experiments, LDB algorithm seems to keep satisfactory performance for $SNR = 10$ and higher. However, the noise behavior is completely non-linear since the algorithm itself is a nonlinear one leading to nonlinear transform of signals. The effect of finer resolution wavelet packet dictionary decomposition is reflected in decreased misclassification rate as depicted in Fig.9.3 at all noise levels. Nevertheless, this is highly application oriented and depends on the selection of most suitable wavelet packet dictionary for analyzing a statistical process that is an open research problem. The effect of the number of signals in training set is most evident. Although a jackknife test is performed to select the training and test signals from the database, but the comparisons in Fig.9.2 and Fig.9.4 directly show the degrading effect of insufficient training signals.

9.3 OLDB Algorithm Noise Analysis in Two Dimensions

The images produced in the statistical process given in Section 4.8 were subjected to different levels of noise and the performance of the LDB algorithm was studied. For the same reason as in one-dimensional case, we decreased the number of LDB features fed into the classifier which was an LDA classifier implemented in Matlab®. The number of classified LDB features was decreased from 8 to 4 in five steps. The QMF representing Daubechies 20 wavelet function was used to generate the analyzing wavelet packet. Symmetric differential entropy or J-divergence was fixed as discrepancy measure in all experiments. The set of training and test signals each consisted of 150 images equivalently distributed between distorted circles, distorted rectangles and distorted combined circles and rectangles. Fig.9.5 shows the misclassification rate of the overall scheme as a function of noise. The performance of optimization block at different noise levels in two-dimensional case is still under study. As previously observed in one-dimensional case, the response and noise behavior of the algorithm is nonlinear with respect to noise level. This is the result that one should expect since the transform into LDB domain itself is a nonlinear one. The curve in Fig.9.5 shows a continuous decreasing trend, however, there is no guarantee as seen in Section 9.2 that misclassification drops with increasing SNR level. This is basically due to statistical nature of the process and noise. At certain noise levels, noise helps the discriminating features of the images under study.

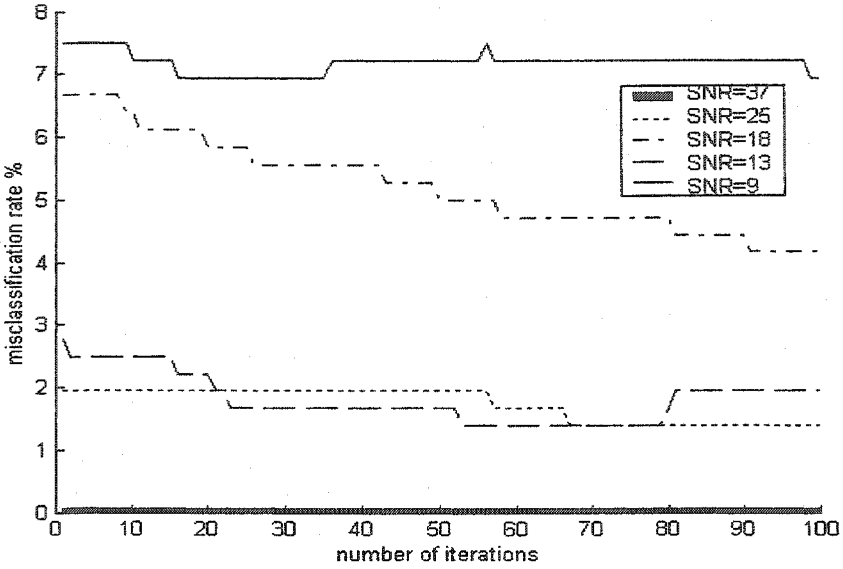


Fig.9.2 Improvement in OLDB performance as a function of time using 360 training signals and Coiflet 4 wavelet packet

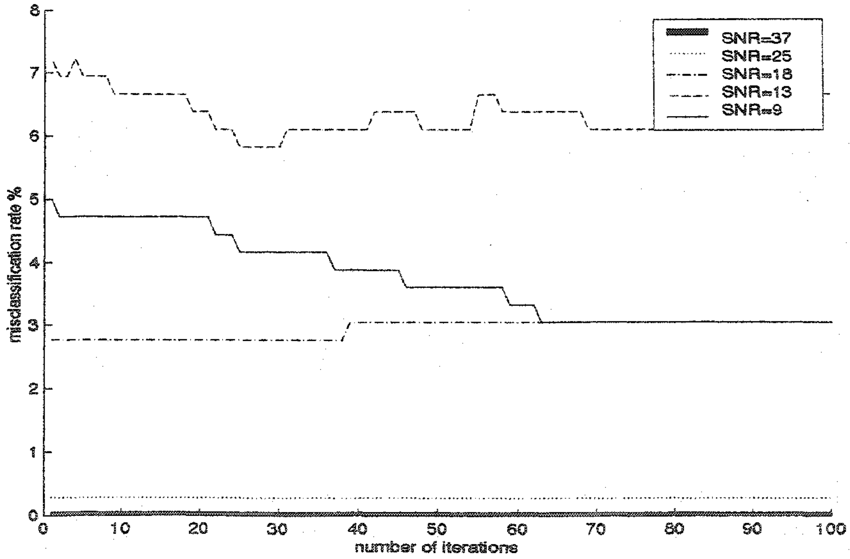


Fig.9.3 Improvement in OLDB performance as a function of time using 360 training signals and Coiflet 12 wavelet packet

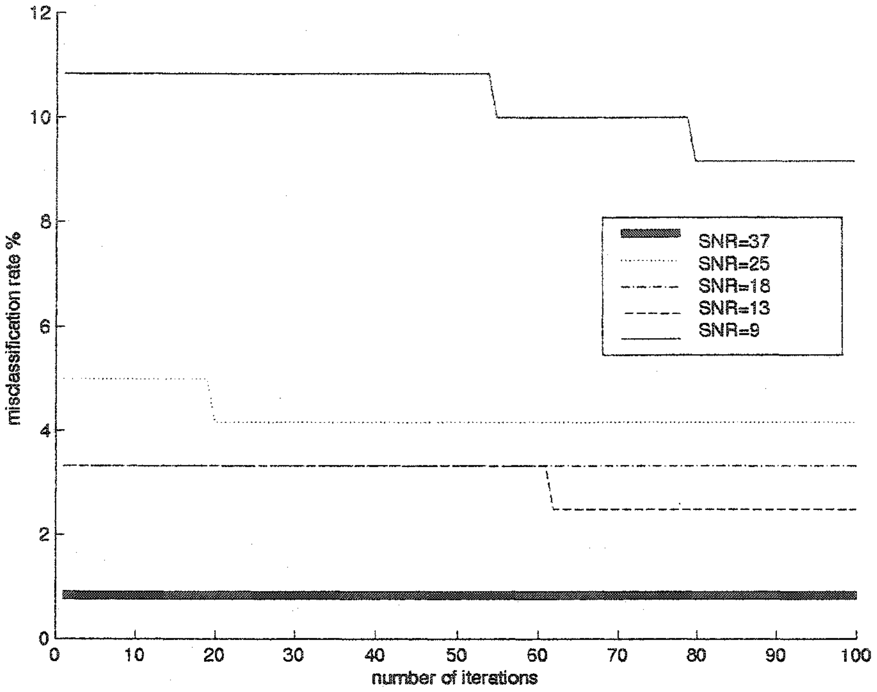


Fig.9.4 Improvement in OLDB performance as a function of time using 120 training signals and Coiflet 4 wavelet packet

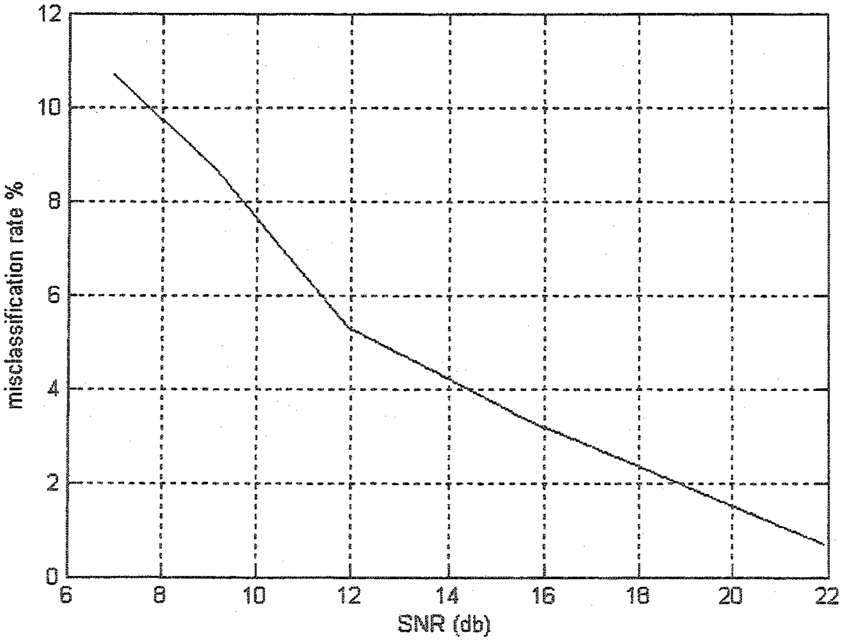


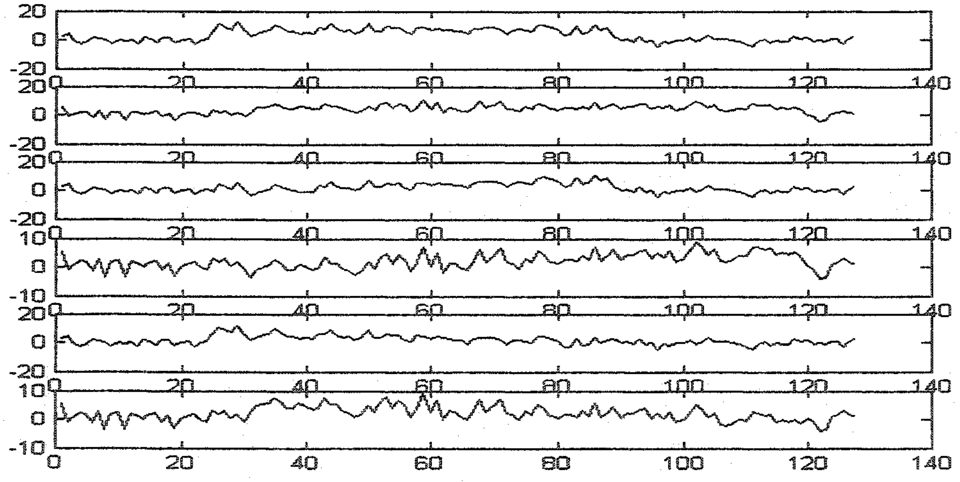
Fig.9.5 Misclassification rate as a function of SNR for two-dimensional OLDB after 20 iterations

9.4 Optimized LDB (OLDB) Performance in Presence of Textured Background

In order to study the effect of background colored noise in the performance of the algorithm, a statistical sinusoidal background texture as given in (79) is added to the process described in Section 9.1,

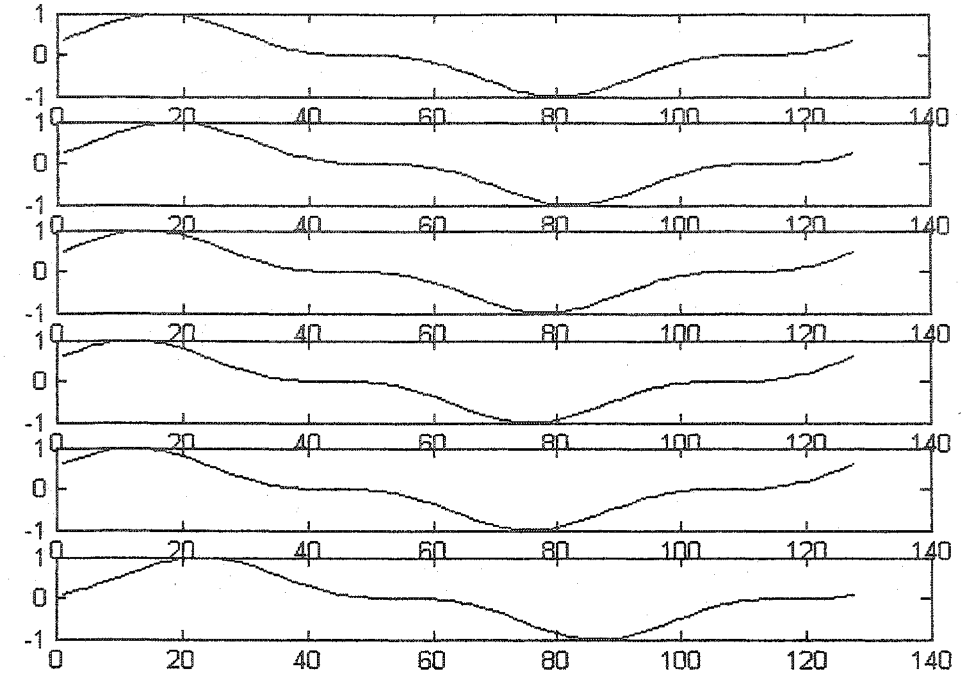
$$s(i) = (5\chi_{[a,b]}(i)+1)\sin^3(2\pi(\frac{5\chi_{[a,b]}(i)+1}{127})i+\beta), \qquad (79)$$

where $\chi_{[a,b]}(i)$ is the same as in (78) and β is a uniform random variable on the closed interval [0,1]. i increases from 1 to 128. The number of training signals is 300 equally distributed between classes. The statistical sinusoidal process of (79) is an extension of background noise studied in [6, Section 7] as a signal/background classification problem. The structured noise given in (79) is more sophisticated due to statistical variations in amplitude and frequency. A top up white noise is added to the overall process to make the signal to noise ratio equal to 14 dB. Fig.9.6 shows the results of OLDB performance in classifying the classes in presence of structured background noise. Note that it is more difficult to differentiate between signals from different classes due to background texture. LDA is used as classifier and symmetric differential entropy is used as discrimination measure. The number of classified LDB features is 12 out of 15 optimized. Optimization helps to improve classification. Teacher signals used in optimization are class averages windowed between [40,60].

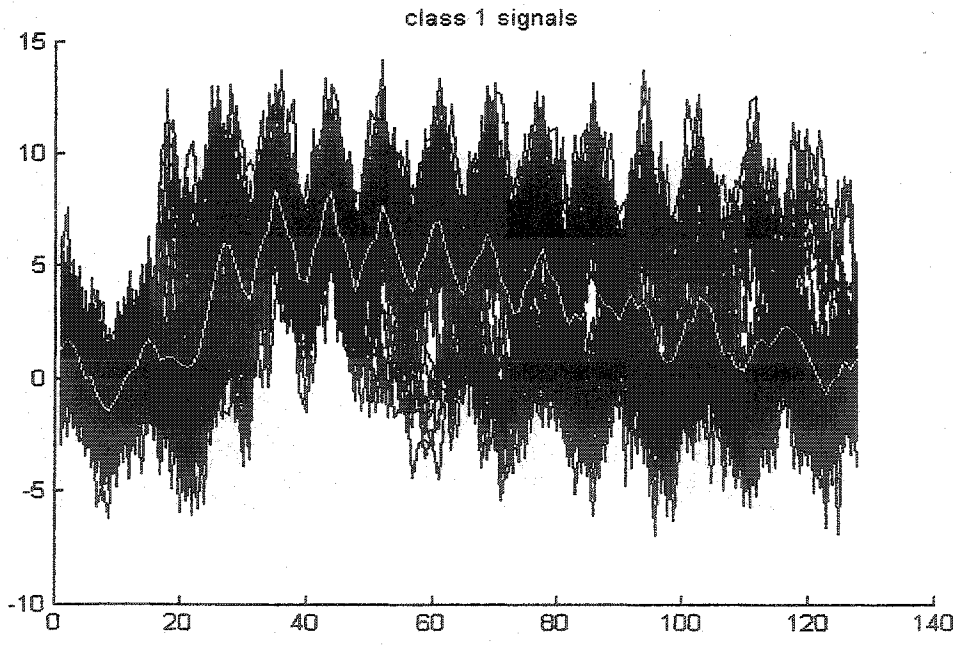


(a)

Fig.9.6 OLDB performance in presence of textured background: (a) ensemble of functions from the process given in (92) with textured background as given in (79): from top to bottom two of each class respectively (b) ensemble of six texture samples (c) class one signals depicted together (d) class two signals (e) class three signals (f) class averages in original domain (g) class averages in LDB domain (h) most discriminating LDB feature (i) decrease in misclassification rate

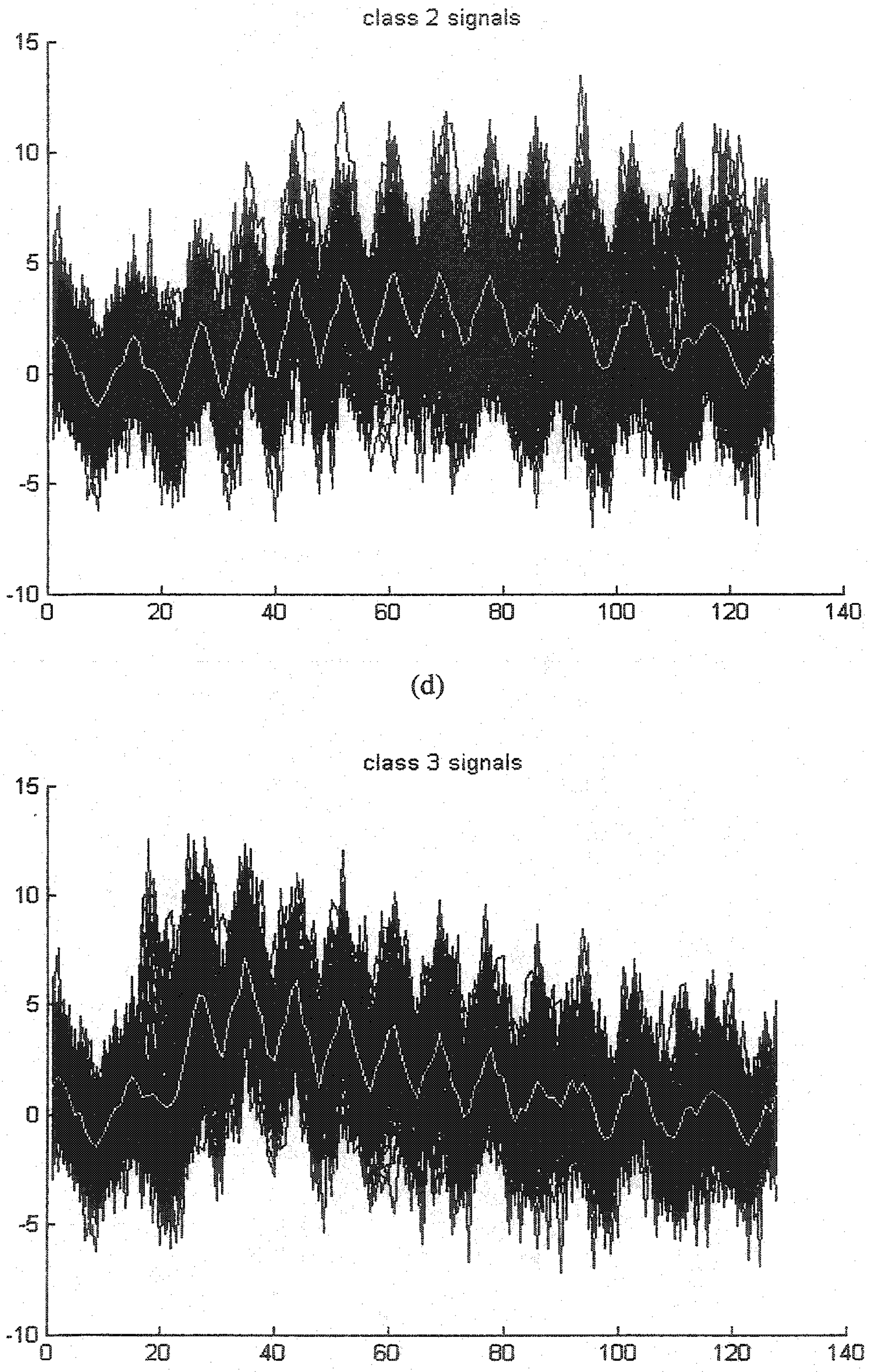


(b)



(c)

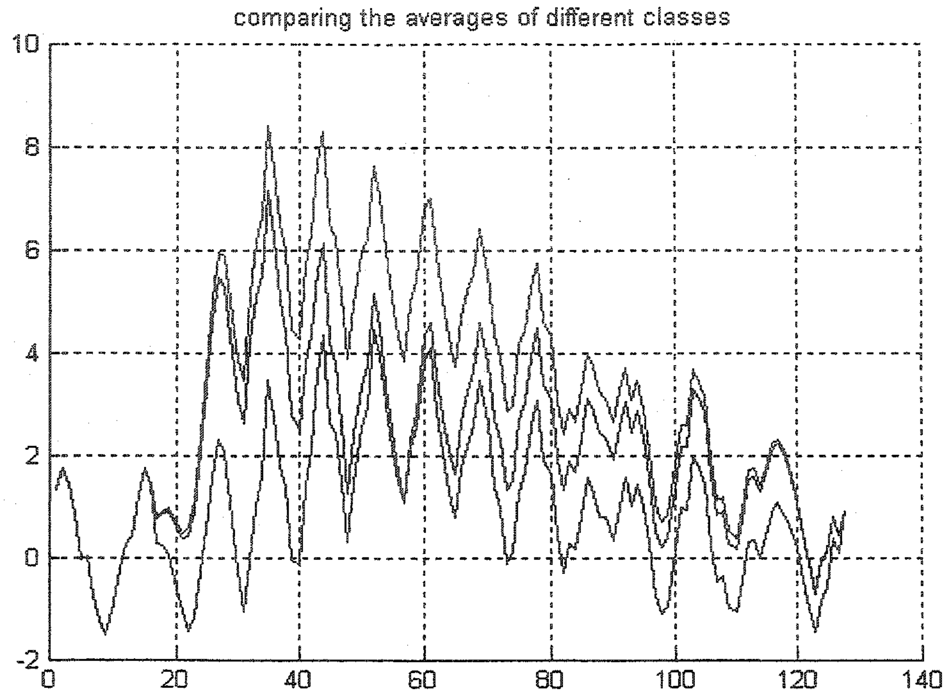
Fig.9.6 (continued)



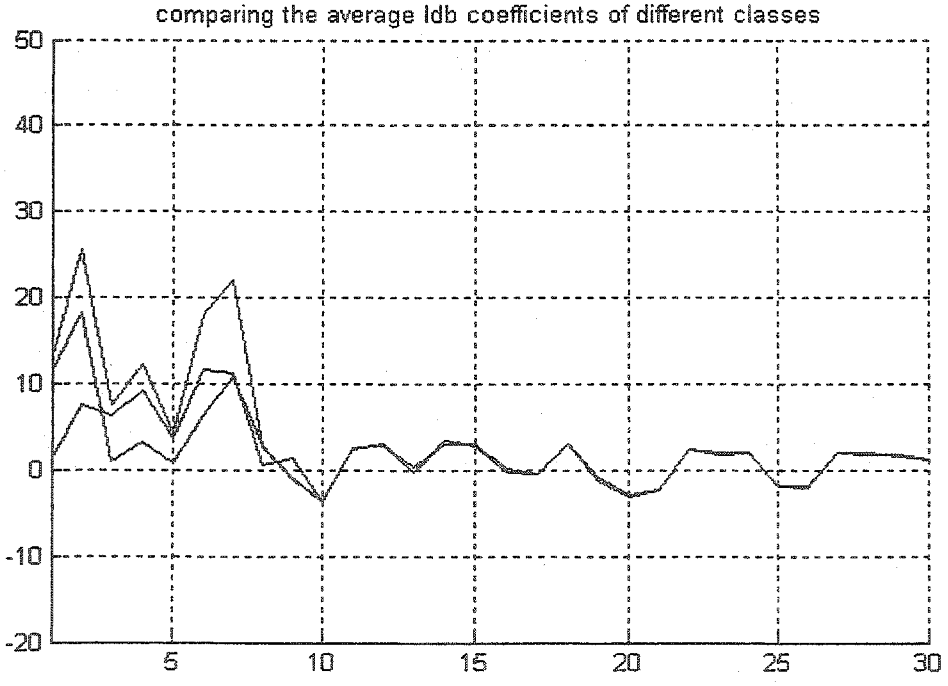
(d)

(e)

Fig.9.6 (continued)

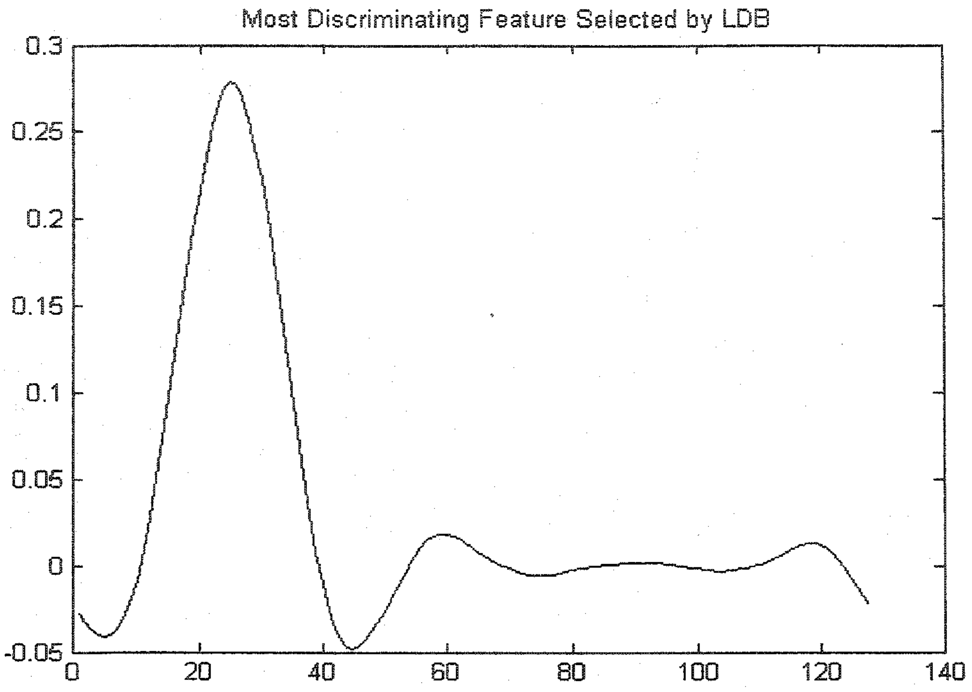


(f)

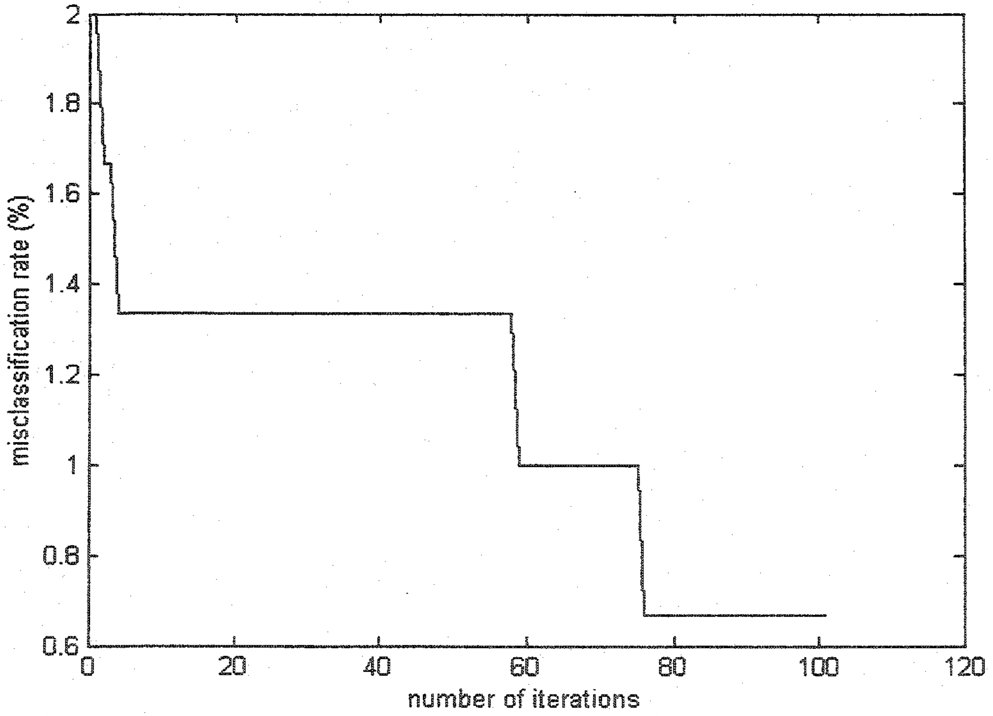


(g)

Fig.9.6 (continued)



(h)



(i)

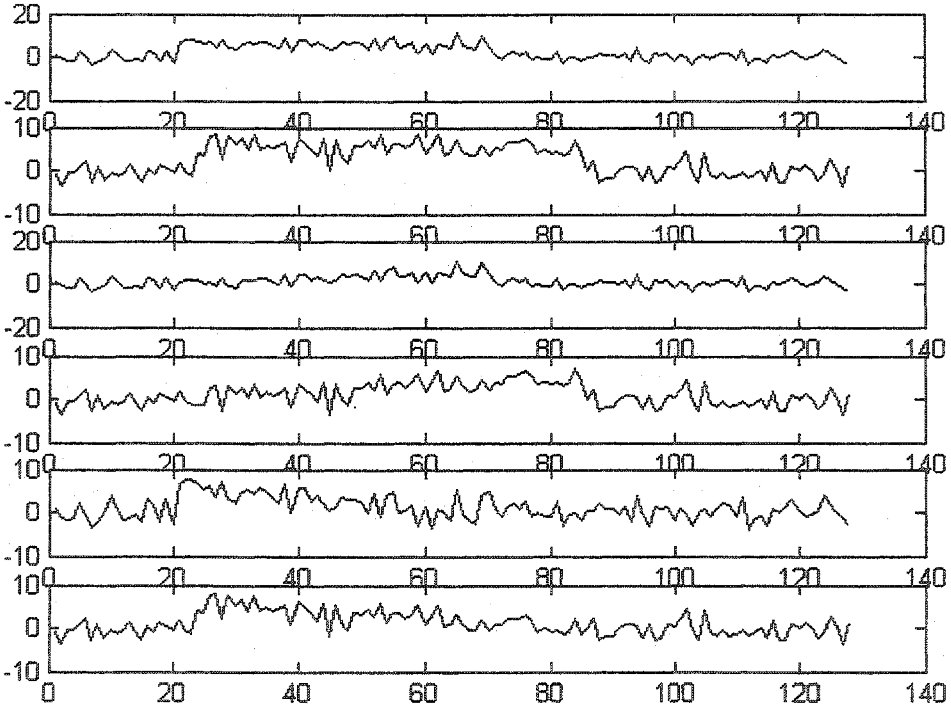
Fig.9.6 (continued)

9.5 Optimized LDB (OLDB) Performance in Presence of Colored Noise

In contrast to what is studied in Section 9.5, the structured noise in this experiment is of higher frequency nature given by (80),

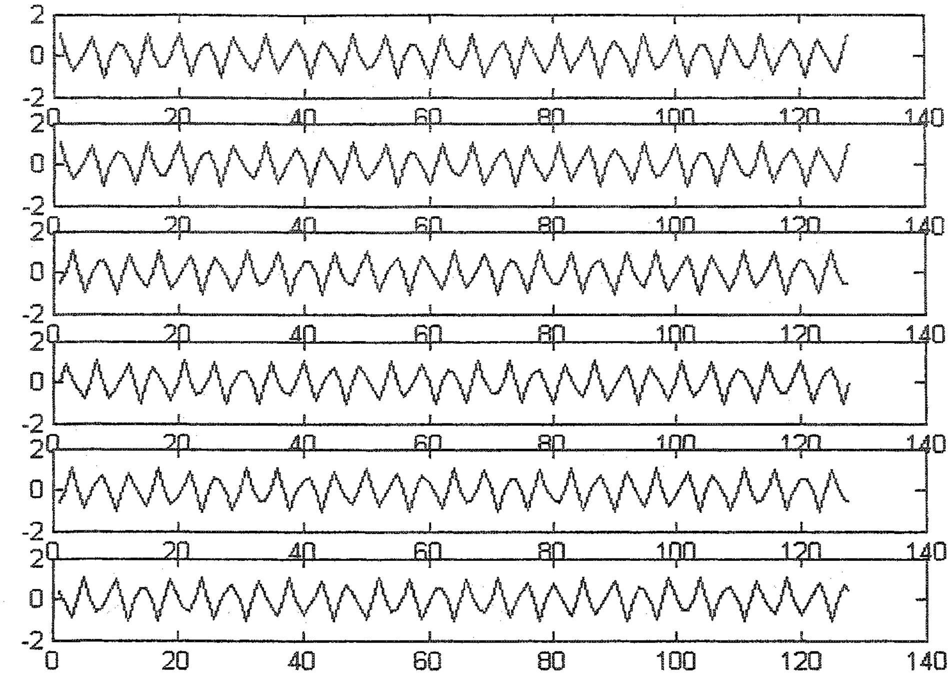
$$s(i) = (.5\chi_{[a.b]}(i) + 2)\sin^3(200\pi(\frac{5\chi_{[a,b]}(i)+1}{127})i + 10\beta), \qquad (80)$$

where $\chi_{[a,b]}(i)$ is the same as in (80) and β is a uniform random variable on the closed interval [0,1]. i increases from 1 to 128. This is a generalization of the structured noise studied in [6, Section 7]. Fig.9.7 shows the results of this experiment. An added white Guassian noise makes the overall SNR equal to 14 dB. Eight LDB features are classified out of optimized 10.

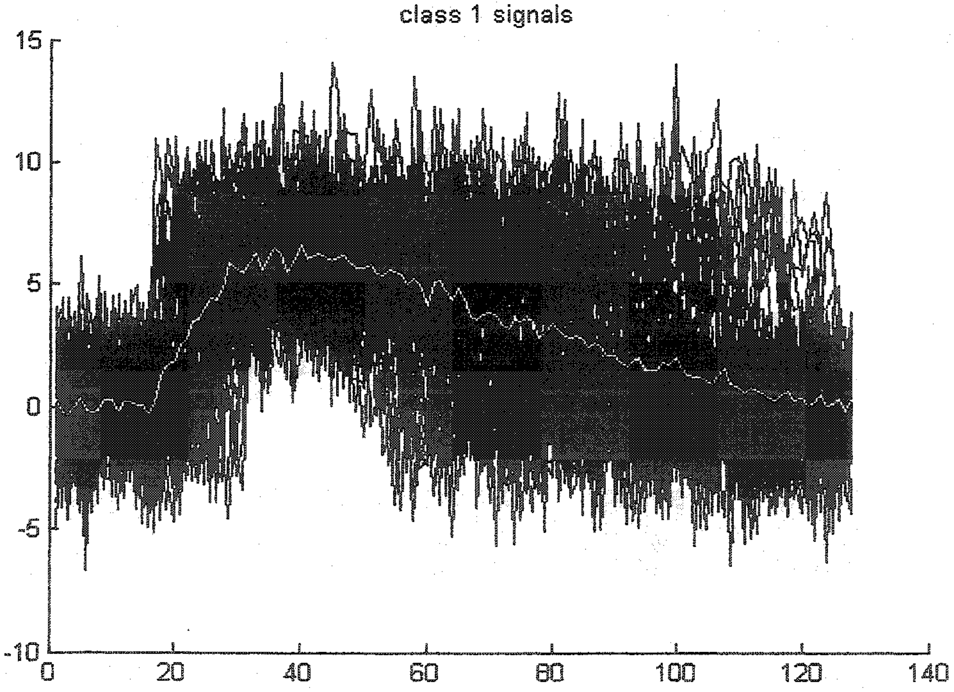


(a)

Fig.9.7 OLDB performance in presence of colored noise: (a) ensemble of functions from the process given in (78) with colored noise as given in (80): from top to bottom two of each class respectively (b) ensemble of six texture samples (c) class one signals depicted together (d) class two signals (e) class three signals (f) class averages in original domain (g) class averages in LDB domain (h) most discriminating LDB feature (i) decrease in misclassification rate

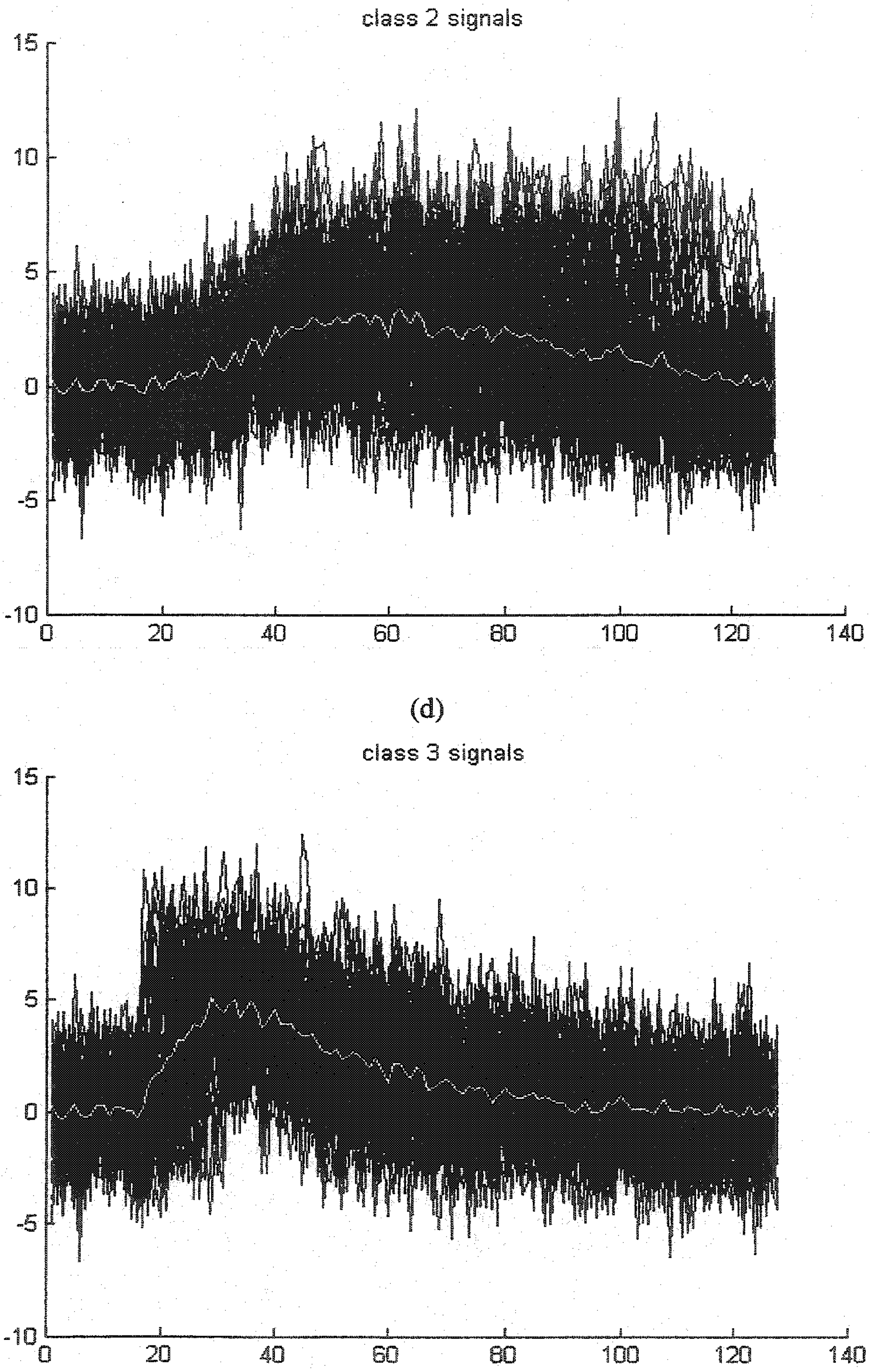


(b)



(c)

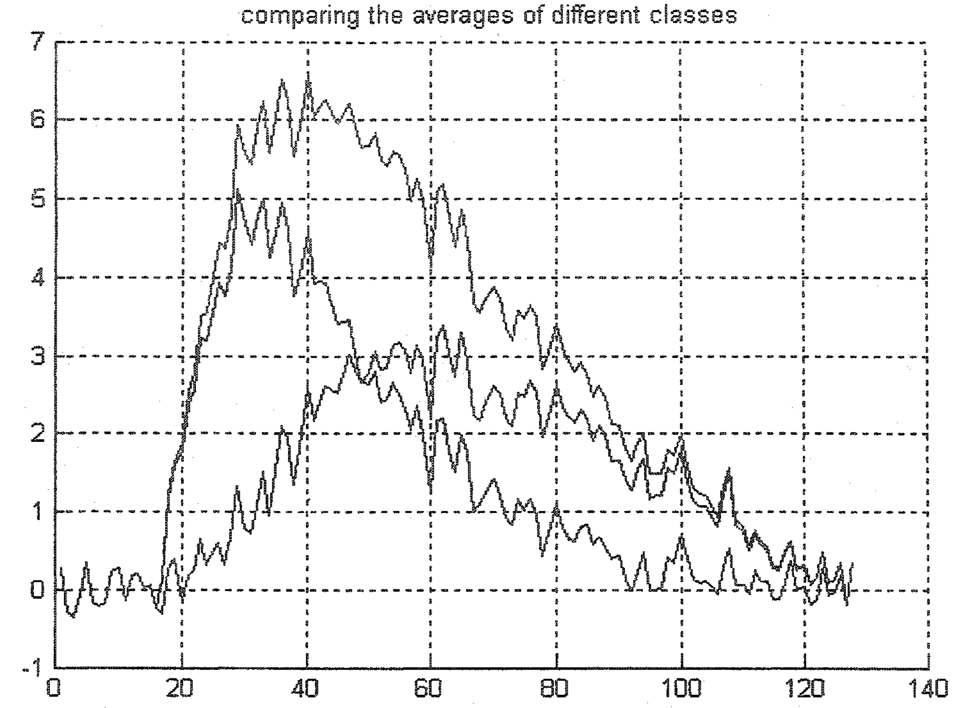
Fig.9.7 (continued)



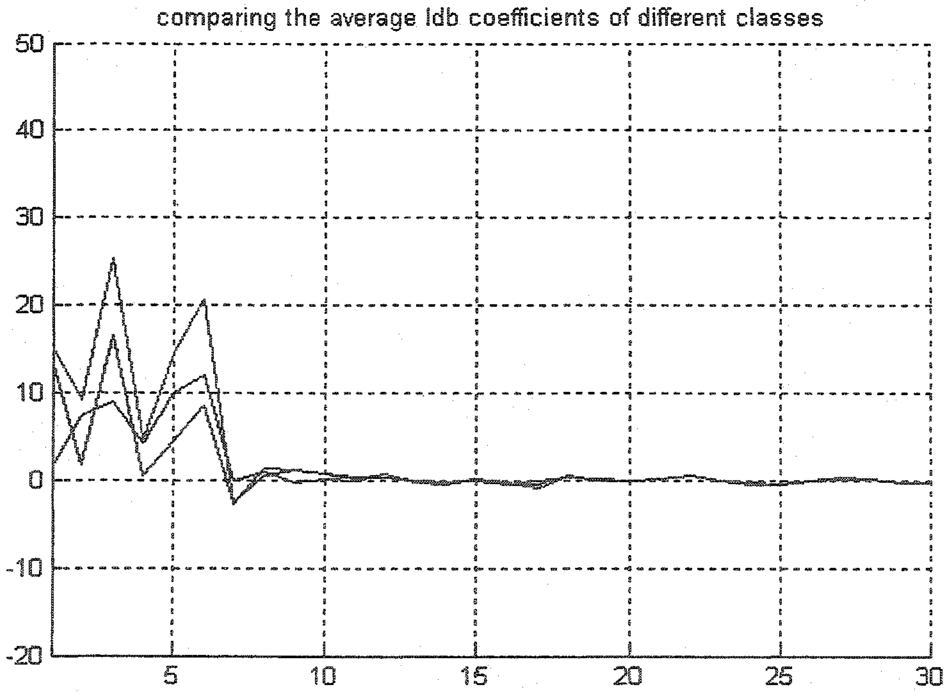
(d)

(e)

Fig.9.7 (continued)

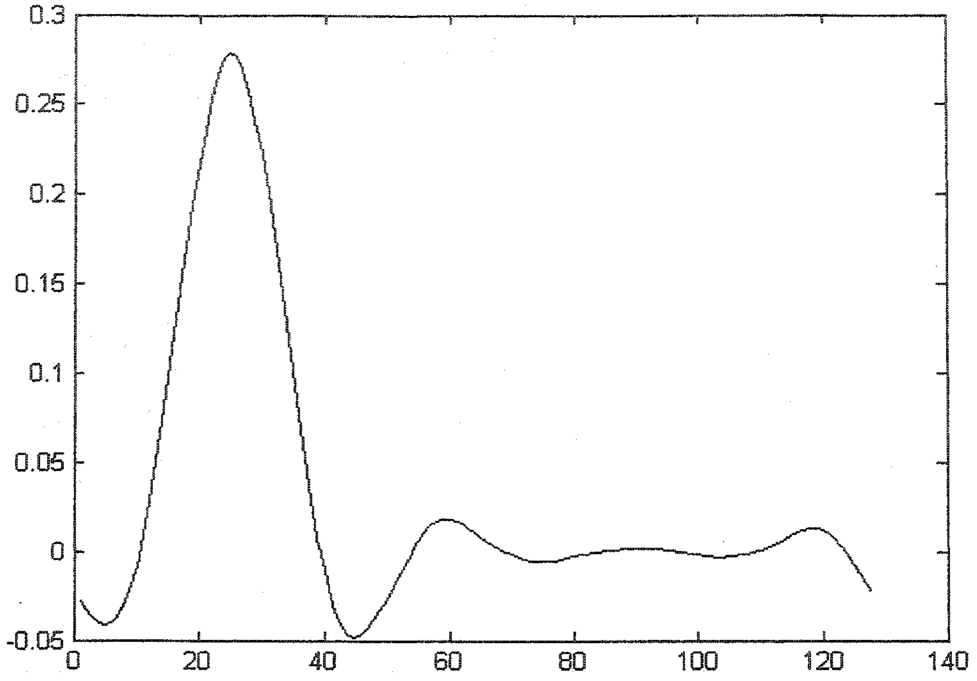


(f)

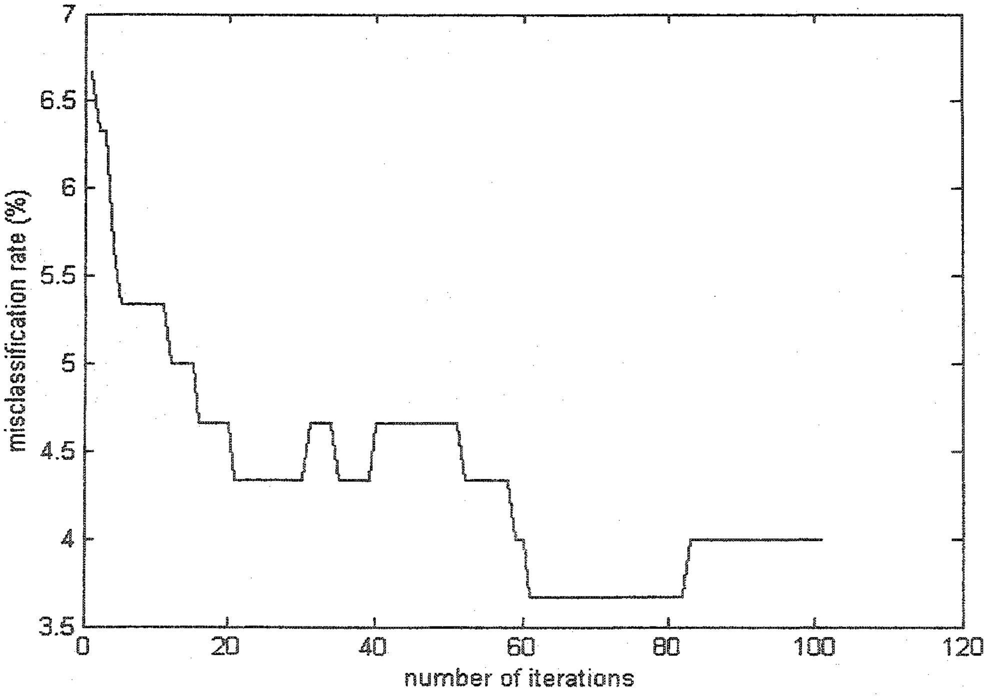


(g)

Fig.9.7 (continued)



(h)



(i)

Fig.9.7 (continued)

9.6 The Effect of Variable Number of Training Signals in Different Classes on OLDB Performance

In the following an experiment is performed to study the effect of having different number of training signals from classes under study. The statistical process under study is the same as in Section 9.4. The number of training signals is changed according to (81),

$$N_1 = (\frac{4}{3} - \alpha) \times 300, \tag{81a}$$

$$N_2 = 100, \tag{81b}$$

$$N_3 = (\alpha - \frac{2}{3}) \times 300, \tag{81c}$$

where N_i is the number of training signals in class i and α is increased from .71 to .98 with .03 step size to study the effect of proportional number of training signals in different classes while keeping the total number of training signals equal to 300. Table 9.1 shows the values of the changing number of training signals versus different values of α used in our study. The same colored noise as described in Section 9.4 is used together with an 18 dB top up white Gaussian noise in all experiments. The number of classified LDB features is 12. Coiflet 4 is used as mother wavelet to generate the wavelet packet structure. Linear Discriminant Classifier (LDA) is the end classifier. The test signals are generated with the same proportionality as training signals but a jackknife test is performed to build a training set and a test set. Fig.9.8 summarizes the result of this experiment.

As it is seen in Fig.9.8, changing the proportion of the training signals from different classes has negligible effect on the performance of the algorithm except at extremes when the number of training signals in a particular class is dramatically low. In order to study the class dependence of the results obtained in this section a second study is performed by changing the number of training signals as in (82),

$$N_1 = (\frac{4}{3} - \alpha) \times 300, \tag{82a}$$

$$N_2 = (\alpha - \frac{2}{3}) \times 300, \tag{82b}$$

$$N_3 = 100. \tag{82c}$$

The results of this experiment are depicted in Fig.9.9. The stability of OLDB performance in extreme cases when the number of training signals in different classes varies considerably is satisfactory.

α	.71	.74	.77	.80	.83	.86	.89	.92	.95	.98
N_1	187	178	169	160	151	142	133	124	115	106
N_2	100	100	100	100	100	100	100	100	100	100
N_3	13	22	31	40	49	58	67	76	85	94

Table 9.1 The change in the number of training signals from each class

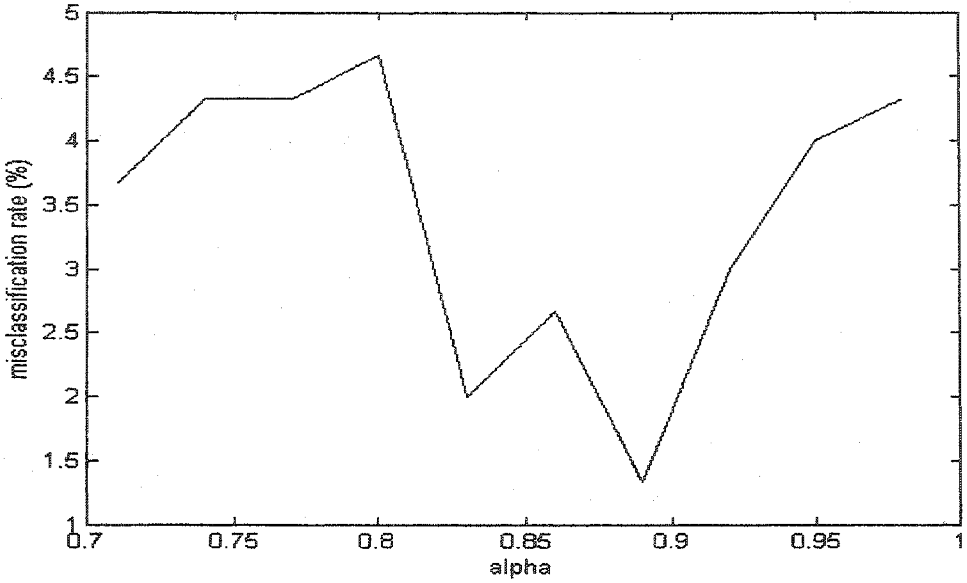


Fig.9.8 The effect of proportional number of training signals given by (81)

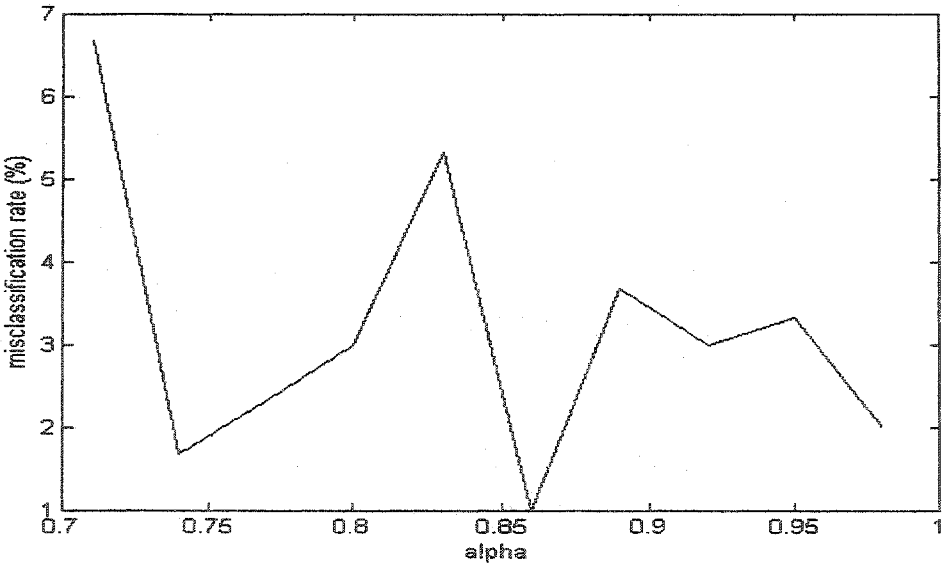


Fig.9.9 The effect of proportional number of training signals given by (82)

Chapter Ten

Conclusion and Future Work

Local discriminant basis (LDB) algorithm is a supervised feature extraction method for non-stationary signal classification. LDB is to target tracking the non-stationarity in the signal at low computational complexity. This is achieved by searching for a most discriminating basis in a redundant set of basis functions. The mathematical foundations of time-frequency transforms as today's best tools for non-stationary signal analysis are studied. The redundant time-frequency collections are the objects that LDB searches for most discriminating basis therein. Theory and applications of LDB are studied in this thesis. Both one-dimensional and two-dimensional versions of the algorithm are implemented in Matlab® computational environment in order to investigate the properties of LDB algorithm. Highly non-stationary synthetic signals produced by processes inside Matlab® are used to study LDB algorithm characteristics.

A novel optimization block is proposed to further improve the accuracy of LDB in conjunction with LDA classifier when the number of selected features is low. Theory of Optimized Local Discriminant Bases (OLDB) is developed. The optimization block proves to be most useful when the signal to noise ratio is between 10 and 30 decibels. This is the typical signal to noise ratio range for most signal acquisition equipment. The improvement in accuracy of the algorithm differs by changing the number of optimized and classified features. However, optimization technique proves to be helpful in all cases.

OLDB is applied to the classification of audio signals to differentiate between pieces of classical, rock and popular music. This is the first application of LDB-based algorithms in the area of audio signal classification. In the most challenging case where three groups of input signals are to be classified, around 70% classification accuracy is achieved. Several experiments are performed to study the effect of music segment durations on OLDB misclassification rate. OLDB is also applied in classifying textured image and its performance in the area of image processing is studied by a three-class texture classification problem.

The noise analysis and stability issues of LDB are studied in detail. Both white Gaussian and colored noise are considered. The noise response of the LDB scheme is nonlinear primarily due to non-linearity of LDB as a transform. The noise behavior of LDB is highly problem dependant. Due to statistical nature of the non-stationary process at hand, correlation between the process and noise as another statistical process leads to nonlinear noise behavior. If the correlation between noise and the signal at a certain signal to noise ratio is significant, noise will improve the classification accuracy. The effect of varying the number of training signals is also studied together with the effect of changing noise power. The number of training signals has a direct influence on the accuracy of the algorithm. OLDB and its optimization block prove to be efficient at all practical SNR levels between 10 and 30 dB. This is the case for all major categories of noise studied in this thesis.

In order to simulate OLDB efficiency in practical applications, the effect of having variable number of training signals from different classes is investigated. OLDB proves to be a stable feature extractor/classifier even in extreme cases when the proportion of training signals from two classes (out of a total three) is close to zero or one. This is an important characteristic of OLDB as a feature extractor/classifier since in most practical applications, the number of available data from different classes vary considerably. The ideal feature extraction scheme should perform equally well in a wide range of available training signal. OLDB is found to be close to ideal in this regard.

In this thesis the steepest decent algorithm is used for optimizing the LDB vectors. A number of other optimization strategies are available that are more efficient than the steepest decent algorithm. The primary difficulty of steepest decent algorithm is to decide about the pattern to change the learning rate. A small learning rate needs more iteration to capture the local minima. A large learning rate causes convergence problems. More effective optimization techniques such as genetic algorithm can better capture the local minima in this case.

The collection of time-frequency dictionaries in this thesis is limited to wavelet packets. A similar study can be done with local trigonometric transforms. A natural continuation of this work is modification of OLDB software to look for most discriminating features inside a chirplet or brushlet dictionary as well. These dictionaries have many potential applications in signal and image processing and can definitely serve as redundant time-frequency dictionaries in LDB.

Another consideration is toward the selection of the dictionary for LDB search. This is still an open problem and needs further investigation to specify which dictionary is the most suitable for the analysis of a particular non-stationary signal. The original LDB algorithm as an ideal method looks for the best basis inside each available dictionary and selects the best basis from all the available dictionaries. However, all the practical implementations of LDB look for the best basis inside a pre-specified dictionary. A methodology to order the available dictionaries for performing the search is highly desirable.

The problem of specifying regions of interest in a signal or image is very controversial. The method by which one specifies regions of interest and how this method can be automated is a major research to be followed in this thesis. The actual performance of different strategies for region of interest selection is to be investigated in a real world biomedical image processing application where the concept of teacher signals originates.

Appendix A

Wavelet Packet Tables

The appendix contains the wavelet packet decomposition, time-frequency energy map and cost tables for one-dimensional and two-dimensional local discriminant basis algorithm implementations.

-1.6682278e-003	-1.3525618e-003	3.1443525e-003	6.2198434e-003	4.4589887e-002	2.1375816e-002	3.9045661e-002	8.8585937e-002
-6.4234862e-003	5.5600406e-003	3.1184773e-003	3.5857296e-002	4.6134843e-002	2.5561526e-002	8.6233773e-002	3.3367034e-002
4.8335608e-004	5.6166422e-004	-2.5644073e-003	2.6184750e-002	4.1839885e-002	6.6527246e-002	-1.7314836e-002	2.5169751e-002
1.1094518e-003	4.9367505e-004	4.7733679e-003	3.2412183e-002	4.1013423e-002	6.3707287e-002	1.8280568e-002	6.8287593e-004
-4.4214769e-003	2.9048904e-003	-5.1736160e-003	3.0263608e-002	1.4930059e-002	-1.2736014e-002	1.4885496e-002	1.9654710e-002
4.5928800e-003	2.6782559e-003	2.6015044e-002	3.3956977e-002	3.6524423e-004	1.3008885e-002	1.2910462e-002	-1.3965602e-003
4.5861260e-003	3.4651862e-003	2.5365619e-002	3.1549911e-002	1.6873985e-002	-1.9366923e-003	-1.8305049e-003	7.5707295e-003
-1.4513635e-004	-1.5318625e-003	1.6874967e-002	3.2347705e-002	4.4811543e-002	1.1627830e-002	8.8761234e-003	4.9820046e-003
1.2622344e-003	-3.0145818e-003	2.4004213e-002	3.0515748e-002	-5.7554124e-003	6.4238068e-004	4.5668023e-003	-3.9459704e-004
6.7351264e-004	3.0068553e-003	1.3971664e-002	2.6898602e-002	-3.8987088e-003	1.3188815e-002	-5.1248468e-003	-6.8530308e-003
-7.2005957e-004	1.5449572e-003	2.7713980e-002	2.6395912e-002	1.6222990e-002	-7.9671818e-003	1.4568672e-002	-1.3673009e-002
2.7990811e-003	2.9089964e-003	2.0877077e-002	2.9974230e-002	9.0305380e-004	-6.6532078e-003	-4.7678833e-003	6.9302041e-003
-2.2688992e-003	-6.7999547e-003	2.1093547e-002	7.9081809e-003	2.2169147e-003	8.6971485e-004	2.1995705e-003	8.8896065e-003
8.4196659e-003	4.5370632e-003	2.3159101e-002	-1.3457605e-003	8.8538245e-004	3.7696810e-003	1.0372232e-002	5.7789440e-003
-5.2602382e-004	2.0832560e-002	2.6731082e-002	4.9081661e-003	-2.7744263e-003	-3.1464435e-003	6.4612601e-003	-2.5247237e-003
4.3938706e-004	2.1192899e-002	1.8805461e-002	2.9639790e-004	1.0904250e-002	1.1732804e-002	2.8907616e-003	6.6128779e-003
4.1140940e-003	1.7913273e-002	2.1977044e-002	-5.6480342e-003	6.8155446e-004	-5.5885821e-003	-5.2619857e-003	6.6619422e-005
2.2862465e-004	1.3525349e-002	2.6914323e-002	1.2578431e-003	1.1801330e-002	-4.8424556e-003	5.3561997e-003	7.5081909e-003
-3.6887726e-004	1.0963787e-002	2.0481371e-002	1.8022566e-003	1.9411647e-003	4.4395396e-003	-4.3388747e-003	7.3517574e-003
-3.2100357e-003	1.5532256e-002	2.3028200e-002	6.5370883e-003	-5.1541007e-004	6.1247369e-003	6.0580803e-003	1.2156619e-003
1.1354236e-003	1.6130831e-002	1.9707356e-002	1.3128506e-002	-8.7156825e-003	1.0054696e-002	-7.4493807e-003	-9.2902605e-003
-5.1531137e-003	1.6087078e-002	2.3550946e-002	-7.7545480e-003	3.8094098e-003	-9.8259644e-004	-5.6890317e-003	1.2447547e-003
2.7548614e-003	7.3860807e-003	1.4757014e-002	-8.6564404e-003	-6.5220942e-003	8.2223768e-003	7.2235374e-003	-1.2288124e-003
6.2614231e-003	1.3572371e-002	2.3372228e-002	-2.3136149e-003	-2.2918442e-003	6.7914812e-004	5.4857342e-003	8.9868121e-003
-2.6678995e-003	2.0961561e-002	1.5280960e-002	2.4019384e-005	1.0455294e-002	5.9483324e-003	7.0449780e-003	-3.1545158e-003
3.3089466e-003	1.8773705e-002	2.2216413e-002	-5.1226157e-003	-1.1703378e-002	-9.1734575e-004	-1.1506137e-002	-1.3117619e-002
4.8361770e-003	1.3838332e-002	1.5122306e-002	5.0152297e-003	-2.2346629e-003	-1.2760335e-002	9.0285706e-004	2.3806540e-003
-6.1463712e-003	1.4139117e-002	2.5712721e-002	1.4895820e-003	2.1206029e-003	1.4203167e-003	4.2696103e-003	3.6574867e-003
-5.5572178e-003	1.5378103e-002	2.1228261e-003	4.6024456e-003	5.5672771e-003	1.0759847e-002	2.7899028e-004	-8.6410284e-003
2.2026857e-003	1.5309183e-002	-4.6050736e-003	6.2599738e-003	1.1260099e-002	5.7492056e-003	-1.2499250e-002	-9.0355803e-003
-1.5421972e-003	1.5037451e-002	9.4619752e-004	3.6341579e-003	5.9776225e-004	4.8988711e-003	8.2504141e-003	-4.6501580e-003
1.3700538e-002	2.0306704e-002	2.6229760e-003	1.3814909e-002	3.7860291e-003	-7.3725445e-003	1.6740976e-003	7.0176895e-003
1.4185023e-002	1.7242783e-002	-1.9882616e-003	3.6783234e-004	-1.9790666e-003	7.6863126e-004	1.3776191e-003	9.3411921e-004
1.3785040e-002	1.7973432e-002	3.8071106e-004	3.3586711e-003	3.3190019e-003	2.1215164e-003	-5.6575039e-005	-1.0141284e-003
1.6015469e-002	1.3816253e-002	4.8706318e-003	-9.4402122e-004	-1.9237146e-003	-5.2288700e-004	1.7593828e-003	-1.7997680e-003
1.3618019e-002	1.0357141e-002	-1.5388308e-003	-2.9598157e-003	-7.6702067e-004	-4.9902229e-004	-7.8587358e-004	6.8837495e-004
1.5632079e-002	1.6273015e-002	-2.0906071e-004	-2.4132535e-003	6.7549564e-004	2.6225350e-004	-2.6023894e-003	-3.3249299e-003
6.4021056e-003	1.9649714e-002	-1.6516470e-003	3.3702207e-003	1.7166279e-003	-1.2612575e-003	-2.0997715e-003	3.5540452e-004
1.0963177e-002	1.7893649e-002	3.0704662e-003	1.6440329e-003	2.3618443e-003	4.6087550e-003	-1.0972188e-003	3.0977803e-003
1.0435102e-002	1.7908629e-002	4.7832589e-003	4.7361758e-004	-7.6107976e-004	-5.1759384e-004	3.2837041e-003	1.5460786e-003
4.8531868e-003	1.5409581e-002	-2.0213415e-003	-2.3049960e-003	2.5773470e-003	-5.6968076e-004	-3.8798051e-003	-5.8005295e-003
1.2031815e-002	1.1531811e-002	7.1947366e-004	-3.7012475e-004	4.7648800e-004	-6.0795241e-003	-4.3233825e-003	-3.1365660e-004
6.9651075e-003	1.7719462e-002	-1.0661834e-003	2.1775099e-003	3.0603674e-003	-1.1392529e-003	-4.0406479e-003	1.6227431e-003
1.6497126e-002	1.6882280e-002	1.6699593e-003	-5.3063978e-003	-1.4361796e-003	-3.8126013e-003	-1.7457426e-003	-4.0915959e-003
7.9345884e-003	1.3016697e-002	3.4180046e-003	1.8520347e-003	3.2220989e-003	2.3366609e-003	-3.1943940e-003	-1.3625173e-003
1.3078645e-002	1.4390383e-002	-5.7946129e-004	3.4419092e-003	1.4573700e-004	-2.4478381e-003	1.2675036e-003	3.1550380e-003
1.1885328e-002	1.7636936e-002	7.8044002e-004	-1.0402424e-004	4.6988260e-004	-1.7758191e-004	1.6989662e-003	-1.6061393e-003
7.4840948e-003	1.4364787e-002	-2.2011524e-004	1.4532816e-003	-6.2627245e-003	1.8818825e-003	-5.7245784e-004	7.9656170e-004
2.6680822e-003	7.9893914e-003	-8.6822766e-005	-2.3915611e-003	2.0832533e-003	-2.4579118e-003	-1.6799856e-003	2.8449143e-003
1.0811233e-002	1.4727116e-002	3.0881815e-003	4.1002332e-003	3.6662381e-003	-2.3951709e-004	5.7033020e-003	5.2207727e-003
7.1418909e-003	1.6403714e-002	1.9824214e-003	4.2101187e-004	4.0203807e-003	4.7207187e-003	-1.3403205e-003	2.4776068e-003
1.3409233e-002	1.4697224e-002	-3.8858432e-003	-1.1599213e-004	-1.7794174e-003	3.6665387e-003	2.1635446e-003	5.8210734e-004
1.2997648e-002	1.1260298e-002	2.2279468e-003	3.1577634e-003	-1.8779925e-003	2.3364486e-003	2.2737366e-003	-8.2658856e-004
1.7566515e-002	9.5000300e-003	9.8927467e-004	-1.5477858e-004	3.1817083e-004	3.3435679e-003	-3.4427093e-003	-4.0421377e-003
1.3319835e-002	1.9288430e-002	-4.1984500e-003	1.3038004e-003	-2.5614304e-003	1.5827927e-004	2.4283052e-003	-1.7660028e-003
8.5574103e-003	1.2160908e-002	1.6362190e-003	8.9246704e-004	4.1774301e-003	-2.5020171e-003	-6.9199895e-005	1.6681393e-003
1.2506304e-002	1.0138618e-002	-1.0089077e-003	6.6177089e-004	2.0673866e-003	1.6406646e-003	1.8520332e-003	1.6661495e-003
7.1477460e-003	1.4269131e-002	1.1273092e-003	-2.2303447e-003	1.2823581e-003	-2.4623873e-004	5.0425812e-004	-9.5302088e-004
1.0964252e-002	1.8814496e-002	2.1354277e-003	2.3911609e-004	-3.6981180e-004	1.8004519e-004	-1.1501070e-003	1.4719367e-003
1.0853529e-002	1.3454280e-002	6.8839285e-004	2.8473295e-003	7.7633792e-004	1.7578280e-003	9.3152580e-004	-1.5456023e-004
1.1039664e-002	8.7512199e-005	-4.4460399e-003	-4.5156281e-003	-2.4301659e-003	-1.2117927e-003	1.3277345e-003	5.7687228e-004
9.8136413e-003	-4.1214420e-003	-1.8577708e-003	6.2398021e-004	1.4181612e-003	2.0937718e-003	-5.1191393e-004	-1.3008279e-003
1.5262484e-002	-2.3176420e-003	-5.0279759e-004	-6.0661414e-004	1.1760806e-003	2.6146644e-003	-4.6541066e-004	2.6379340e-003
3.8122735e-003	-1.5743831e-003	2.8925377e-003	3.6864219e-003	1.6792149e-003	4.6284440e-004	3.2651914e-003	1.9797439e-003
1.2690827e-002	-1.9150808e-003	1.7226582e-003	-2.9987445e-003	2.3626062e-003	5.8210177e-004	-7.7937231e-004	-1.2572073e-003

Table A.1 A 128×8 wavelet packet decomposition table for example in described in Section 4.7

1.3858504e-002	-5.5887266e-003	8.0702999e-003	2.5243393e-003	-3.8467248e-003	8.3573534e-004	2.3887439e-003	-6.0532385e-003
1.3268059e-002	2.7992148e-003	2.9533951e-003	-4.6523836e-004	-3.1702845e-003	-6.2459365e-003	-6.1718280e-003	-2.6750444e-003
1.1194972e-002	3.6698304e-003	-4.5729632e-003	4.2604264e-003	-1.6436710e-003	5.9474840e-003	-2.4077868e-003	-2.6569225e-003
1.3650756e-002	-3.1271413e-003	-3.8271622e-004	-2.6337565e-003	6.3308091e-003	-9.9081446e-004	-1.3496691e-003	7.4820219e-004
1.3233515e-002	5.6993001e-004	-4.6546352e-003	-1.0764820e-003	-4.8296641e-004	-1.9945787e-003	4.8642458e-003	-5.5830996e-003
1.0053576e-002	2.8521000e-003	1.3635567e-003	-4.0178279e-003	3.9717753e-004	-3.7012527e-004	-3.0314494e-003	1.2959828e-003
9.5837517e-003	7.8923828e-003	6.7630953e-003	-1.1824124e-003	-5.1572090e-003	8.1408404e-004	1.6070724e-004	5.7279573e-003
9.8983799e-003	-1.8041950e-003	-8.2548369e-003	-4.2638895e-003	7.1903494e-003	2.5983428e-003	7.9398477e-003	5.5006829e-003
5.3504989e-003	-1.0668431e-003	3.6272409e-003	-2.0940657e-003	-2.4864624e-003	8.7900088e-003	-8.5083753e-004	-2.6728695e-003
1.0137038e-002	-2.1060276e-003	1.8775765e-003	-7.9332905e-004	-2.7667870e-003	-7.4652098e-004	-4.6308459e-003	-3.8761355e-003
1.1496291e-002	-5.9987072e-003	2.3812652e-003	6.3672791e-004	-3.2310099e-003	-4.5990840e-003	3.5220086e-004	1.7535555e-003
1.2253588e-002	5.2372358e-003	2.6073257e-003	-2.9243344e-003	-1.7397457e-003	-1.8033785e-003	2.1277011e-003	1.2554682e-003
1.6606525e-002	1.7162485e-003	-3.2680210e-003	4.1279646e-003	-1.9720932e-003	-5.1055555e-004	-4.4695709e-003	2.9648436e-003
9.6859291e-003	-5.1227822e-003	-1.6248672e-003	8.8524737e-003	6.7051849e-004	2.0084990e-004	-2.7664878e-004	-3.3560841e-003
1.3443058e-002	5.4355211e-003	-3.1687306e-003	-1.0837420e-003	-6.1570351e-003	3.9509903e-003	5.1709408e-003	1.6920724e-003
1.4121105e-002	5.5799748e-003	3.5667541e-003	5.7414019e-003	2.9008233e-003	-1.5893883e-003	-2.7779890e-003	-5.6207422e-003
1.4668129e-002	-2.1014781e-003	-6.7934897e-003	-2.4350984e-003	5.9623973e-003	-4.3975673e-003	-2.2175587e-003	5.2747510e-003
7.2133999e-003	-1.3385408e-003	-3.5200611e-004	6.6935208e-004	-8.1245200e-004	5.4201101e-003	5.2420657e-003	2.1386494e-003
1.1857231e-002	-5.7221505e-003	-2.9391418e-003	3.7514318e-003	3.6202101e-003	3.6900790e-003	9.1459022e-004	-4.6403608e-003
1.1956912e-002	2.4409163e-003	1.3138679e-003	9.5295716e-004	4.1526729e-003	2.8541080e-003	-7.4770514e-003	-5.9337867e-003
7.1529606e-003	4.4013362e-003	-3.0145665e-003	-1.1278606e-002	-3.7161801e-003	2.8478450e-003	1.9954884e-003	-9.2448354e-004
8.1777300e-003	6.7271477e-003	-5.4718586e-003	-1.5468293e-004	-1.1645328e-003	-5.5969056e-003	6.8807120e-004	1.8975632e-003
1.5213471e-002	1.9137726e-003	-1.8113677e-003	2.2570841e-003	4.5279492e-003	1.2652905e-003	3.0428635e-003	-3.5124912e-003
1.0532356e-002	-3.1206555e-004	3.0667777e-003	7.2406028e-004	-9.2272812e-003	-2.7775073e-003	-8.0102762e-003	-7.8157500e-003
1.2543108e-002	5.2138564e-003	-5.2980802e-003	6.7441931e-003	2.2231004e-003	-1.0218400e-002	-2.0569904e-004	5.7315206e-003
1.1378828e-002	1.8343533e-003	4.6120557e-003	2.8941579e-003	6.7045540e-003	4.7056343e-003	7.8998951e-003	5.4406182e-003
8.5887642e-003	3.4076640e-003	-3.8616017e-004	3.5272120e-003	1.5179536e-003	5.9192207e-003	2.1924419e-003	2.9962090e-004
8.8814488e-003	-3.5889725e-003	-2.6754848e-003	-3.0955361e-003	4.8557698e-003	4.8477613e-003	-1.7687139e-003	-2.8009601e-003
1.2750489e-002	-3.2912580e-003	-4.4575505e-003	5.2055703e-003	3.8205236e-003	-1.1827465e-003	7.2269504e-003	-5.1843562e-003
7.3761016e-003	3.2222440e-004	4.9947993e-003	5.7246247e-003	-1.7834131e-004	4.8795469e-004	-1.0483641e-004	5.0360951e-003
1.4052199e-002	-2.0993640e-003	2.7971096e-003	2.7675666e-003	-1.2508897e-003	1.8969815e-003	1.9468171e-003	9.0068908e-004
1.3233748e-002	-8.2536956e-003	-5.3614780e-004	-3.2798448e-003	6.0780091e-003	-1.3408219e-003	-6.7305040e-004	-1.8525261e-003
7.8704768e-003	1.7043222e-003	7.0092718e-003	-3.8336405e-003	-1.2066071e-003	-9.2074416e-004	-1.8819043e-003	2.5858664e-003
1.0015159e-002	1.1586674e-004	-2.9370186e-003	-3.9132134e-003	1.5019431e-003	2.1659627e-003	1.7750630e-003	-7.5548241e-005
6.4587208e-003	1.2834615e-003	-2.8904553e-003	7.9911387e-004	3.5233793e-004	1.7510939e-003	-7.3981970e-003	-6.4083379e-003
2.5460355e-003	-1.4865680e-003	9.5875169e-004	-4.9138536e-004	2.2187625e-003	-5.3158933e-003	-1.6645614e-003	4.0542926e-003
1.4843402e-002	2.2406003e-003	4.4410228e-003	7.1346398e-003	8.6462601e-004	6.4351222e-003	-1.2469203e-003	3.5412899e-003
9.0393311e-003	1.5498316e-003	1.9403459e-004	7.2247112e-004	-6.0106484e-003	6.8543367e-004	3.7612199e-003	1.7778783e-003
1.2302021e-002	-1.3964632e-003	-4.8539305e-003	-1.5235211e-003	3.9630623e-003	-1.8201773e-005	-5.3179412e-004	-2.5517583e-003
1.1942160e-002	-4.5235800e-003	7.7336657e-003	6.6646557e-004	-2.5128600e-003	-3.8900136e-003	-3.0769371e-003	-1.7996879e-003
1.1122283e-002	8.6600002e-004	3.0896412e-003	2.3921974e-003	-1.8230919e-003	-2.8751368e-003	-2.6179918e-003	4.0439797e-003
7.1676886e-003	-5.3663434e-003	4.4324236e-003	-1.6880853e-003	-1.1655861e-003	1.6798355e-003	3.1010592e-003	3.4158023e-004
7.3867372e-003	2.8370870e-003	1.8075350e-003	-8.0203702e-004	-3.6079715e-003	-3.6469666e-003	4.8771772e-004	-6.8966141e-004
9.5954748e-003	-2.3510794e-003	-3.4555845e-003	4.4198132e-003	-1.7512371e-003	-1.0293684e-003	-1.4630462e-003	-1.3793984e-003
6.4660127e-003	-2.8785152e-003	-4.7176217e-003	9.6723790e-004	7.6325455e-004	1.2865386e-003	-4.5733379e-003	3.4175870e-003
6.9673059e-003	9.9115373e-004	-1.6214882e-003	-7.3502172e-004	-2.4925259e-003	-2.7103824e-003	2.5986007e-004	-3.0500894e-003
1.6718257e-002	-9.5447075e-004	3.4128349e-004	-3.9191052e-003	1.4981285e-003	-4.0367044e-003	-4.5071103e-004	3.8314970e-003
1.1254478e-002	-5.1271385e-003	2.1925380e-003	4.5006596e-003	4.3584414e-003	9.0270338e-004	5.8692660e-003	4.4688986e-003
6.3447951e-003	1.8354975e-003	-2.1799413e-003	-1.4979104e-003	5.1096658e-003	1.8481230e-003	4.0229830e-004	5.4511427e-003
1.0880501e-002	2.8718958e-003	-3.3484742e-003	7.6812766e-003	6.2089946e-003	8.9488721e-003	8.1113783e-003	6.0200784e-003
6.6879304e-003	-6.3264884e-003	-5.3582634e-004	1.6150458e-003	-6.9002713e-003	1.8655002e-003	-2.9419770e-003	8.2526202e-004
5.8358785e-003	-3.4510405e-003	-2.3369956e-003	-6.3236319e-003	-6.5938443e-004	-1.5587275e-003	4.1090737e-003	4.9858457e-003
1.0032533e-002	4.6413545e-004	-2.6728382e-003	1.4001778e-003	1.4188898e-003	3.4424067e-004	2.7129681e-003	-8.0171347e-004
1.4716628e-002	-1.9804803e-003	1.3721431e-003	-2.2196051e-003	-1.9735433e-004	5.4189923e-003	1.5791741e-003	3.0350029e-003
1.1535627e-002	2.9505913e-003	-5.2709064e-003	-2.6505043e-003	1.4509176e-003	1.8060267e-003	1.6031314e-003	-3.9945428e-005
1.3571227e-002	-4.0063148e-003	-1.4201888e-003	-1.8886964e-003	-2.4732096e-003	2.5810706e-003	-1.6596227e-003	-2.3071156e-003
6.5356893e-003	-6.0260425e-004	-2.7922176e-004	-4.0012507e-003	-3.1707555e-003	-1.1348202e-003	2.3680892e-003	-4.0601359e-003
9.2631288e-003	4.1280688e-003	-3.2322237e-004	-9.3019797e-004	-4.4459293e-003	-3.3321680e-003	-3.3738100e-003	-7.1115196e-004
-1.0121249e-003	-2.5008682e-003	5.5942850e-004	2.2762435e-003	3.1153644e-003	2.6113903e-003	1.7248827e-003	-3.9753370e-003
-4.6786374e-003	2.9960740e-003	5.0216617e-003	4.1205450e-003	3.9824696e-003	2.9957704e-003	-7.3468583e-003	-6.4146896e-003
-5.0885356e-003	2.5090503e-003	3.2399670e-003	8.3402946e-004	-1.2300947e-004	2.2238068e-003	1.0528473e-003	-6.2909089e-004
3.5913299e-003	5.2907507e-003	-9.4651359e-004	4.4423642e-003	4.0962036e-003	-6.1112475e-003	1.6317841e-004	8.5986001e-004

Table A.1 (continued)

1.5165955e-003	1.3128681e-003	1.6199889e-003	7.6561765e-002	1.7192076e-001	1.0111088e-002	4.3532536e-002	4.5117616e-001
1.5670909e-003	1.2803406e-003	3.3918853e-003	1.0788265e-001	1.2841044e-001	1.9226299e-001	5.8779239e-001	1.8014877e-001
1.6551353e-003	1.2363124e-003	1.1591576e-002	9.8103863e-002	9.0819987e-002	3.5705966e-001	5.6700289e-002	8.3856673e-002
1.3114944e-003	1.7017596e-003	2.5649214e-002	9.5716890e-002	4.8395662e-002	1.7092288e-001	4.2331408e-002	1.5175025e-002
1.3345799e-003	1.7565951e-003	4.1072089e-002	8.8244517e-002	4.8584508e-003	2.0003169e-002	1.6920306e-002	2.6832817e-002
1.5246208e-003	1.2105477e-003	5.2893519e-002	7.7132290e-002	2.8236437e-003	9.1458875e-003	1.6250945e-002	6.3384343e-003
1.4533319e-003	2.9640492e-003	5.2530930e-002	6.8762517e-002	1.2843296e-001	1.1519132e-002	1.1235385e-002	1.2035489e-002
1.3336115e-003	3.9387985e-003	5.2987025e-002	5.2267576e-002	2.0740010e-001	1.2037184e-002	8.2987351e-003	7.4986313e-003
1.5909355e-003	6.8066278e-003	5.0555013e-002	4.9432334e-002	2.0478705e-003	4.9737401e-003	5.4068803e-003	7.1249267e-003
1.3731512e-003	1.0638112e-002	4.7142255e-002	3.6400115e-002	9.1662528e-003	4.0048665e-003	8.5210990e-003	6.8030526e-003
1.7649336e-003	1.3774134e-002	5.0254237e-002	2.8117091e-002	4.5924427e-003	9.5447546e-003	4.7298297e-003	4.0284790e-003
1.6275751e-003	1.7791296e-002	4.7249801e-002	1.2476211e-002	2.5108963e-003	3.8858961e-003	3.7514482e-003	4.4527989e-003
1.5301263e-003	2.2314872e-002	4.6396352e-002	2.8302723e-003	4.1877009e-003	2.9784946e-003	3.1828204e-003	3.1822834e-003
1.4202526e-003	2.4155354e-002	4.0360809e-002	1.6136414e-003	3.8478458e-003	2.9108287e-003	2.9093336e-003	2.9098706e-003
1.1534347e-003	2.7517478e-002	3.9728984e-002	1.6238986e-003	3.8714374e-003	2.5907045e-003	2.4874942e-003	2.7480988e-003
1.3770243e-003	2.7055769e-002	3.8422764e-002	1.9585827e-002	3.4650115e-003	2.8001729e-003	2.7005526e-003	2.4399480e-003
2.2060700e-003	2.7723167e-002	3.4437640e-002	1.5056109e-003	2.3385115e-003	2.3511862e-003	2.8916638e-003	3.0106126e-003
2.5733947e-003	2.6040345e-002	3.2352361e-002	1.3249896e-003	2.2368986e-003	2.1820483e-003	2.3043352e-003	2.1853864e-003
2.7800680e-003	2.7943863e-002	2.6068612e-002	2.5401423e-003	2.0369168e-003	2.2413864e-003	2.3023103e-003	2.3878959e-003
3.4733805e-003	2.6651396e-002	2.5616495e-002	4.5819694e-003	4.5392269e-003	2.7663227e-003	2.0426343e-003	1.9570487e-003
4.6036381e-003	2.5419158e-002	2.6860998e-002	2.7324237e-003	1.7738663e-003	2.1594762e-003	2.0077356e-003	2.8759704e-003
4.4953217e-003	2.6099004e-002	2.2970042e-002	1.6566404e-003	2.2962087e-003	2.0577027e-003	2.7542084e-003	1.8859736e-003
6.0584040e-003	2.3627798e-002	1.8410418e-002	1.8808842e-003	1.8881174e-003	2.1567839e-003	2.3380437e-003	2.4358188e-003
7.8294662e-003	2.5173878e-002	1.7525202e-002	1.8024044e-003	2.1474237e-003	3.3422633e-003	2.6162384e-003	2.5184634e-003
7.0368978e-003	2.5526425e-002	1.4325112e-002	1.8839116e-003	2.4232903e-003	1.9779283e-003	2.1966379e-003	1.6267643e-003
9.2158095e-003	2.5796005e-002	1.2499794e-002	1.9585441e-003	1.7073839e-003	1.9829816e-003	1.7958863e-003	2.3657599e-003
9.9727768e-003	2.3798863e-002	7.2822528e-003	2.0833272e-003	1.7484258e-003	2.1198862e-003	2.0446548e-003	1.7696448e-003
1.0752435e-002	2.4166956e-002	4.4137176e-003	1.9163579e-003	1.6095467e-003	1.4203097e-003	1.4639269e-003	1.7389369e-003
1.1738157e-002	2.4808124e-002	2.3907416e-003	2.0028162e-003	1.7939128e-003	1.8618041e-003	1.8506603e-003	2.0741048e-003
1.3372858e-002	2.1567015e-002	1.5048370e-003	2.0901567e-003	2.2201375e-003	1.6866349e-003	1.9186693e-003	1.6952248e-003
1.2477913e-002	2.2173139e-002	1.4490137e-003	2.3418821e-003	1.5103187e-003	2.3940482e-003	1.9518530e-003	1.8646210e-003
1.3985802e-002	1.8934600e-002	1.1793754e-003	2.0795382e-003	2.1114133e-003	1.6808360e-003	1.9021406e-003	1.9893726e-003
1.3692363e-002	2.1215128e-002	1.4758428e-003	1.4405683e-003	2.4487332e-003	1.7660939e-003	1.6405042e-003	1.6666428e-003
1.5774909e-002	2.0727469e-002	1.4023106e-003	1.4173274e-003	1.8567553e-003	1.2950437e-003	2.0073615e-003	1.9812229e-003
1.3750607e-002	1.9214536e-002	1.3301610e-003	1.2274987e-003	1.5262601e-003	2.2930154e-003	1.5424839e-003	1.5120091e-003
1.3895052e-002	2.0104723e-002	1.5926583e-003	1.5966340e-003	1.5141004e-003	1.4203607e-003	1.5841640e-003	1.6146388e-003
1.4268512e-002	1.7592805e-002	1.5579284e-003	2.2009694e-003	1.4727139e-003	1.7263284e-003	1.8293189e-003	1.8474700e-003
1.5166846e-002	1.7464885e-002	1.8689565e-003	2.4691638e-003	1.9816663e-003	1.5755628e-003	1.5217439e-003	1.5035928e-003
1.3354486e-002	1.6451895e-002	2.3883151e-003	1.5058941e-003	1.3749138e-003	1.8508774e-003	1.5589918e-003	1.7761763e-003
1.3577058e-002	1.6254097e-002	2.0791697e-003	1.5129126e-003	1.1721705e-003	1.4200312e-003	1.6627453e-003	1.4455608e-003
1.5430145e-002	1.3221474e-002	1.9381677e-003	1.5909668e-003	1.7244077e-003	1.3722312e-003	1.6092624e-003	1.7414731e-003
1.4054886e-002	1.4024322e-002	1.1981886e-003	1.8903335e-003	1.7395167e-003	1.9028816e-003	1.8304217e-003	1.6982109e-003
1.3693717e-002	1.3873892e-002	1.5636855e-003	1.5136601e-003	1.4249581e-003	1.7672122e-003	1.7491758e-003	2.0035436e-003
1.4420777e-002	1.3285018e-002	1.5857622e-003	1.6001354e-003	1.7439603e-003	1.7919862e-003	1.6454514e-003	1.3910836e-003
1.3271535e-002	1.4418206e-002	1.3621084e-003	1.5584754e-003	1.8446196e-003	1.5412423e-003	1.4981566e-003	1.7876433e-003
1.3615416e-002	1.4060552e-002	1.7410626e-003	1.4333794e-003	1.4579317e-003	1.7858386e-003	1.9289647e-003	1.6394780e-003
1.3467814e-002	1.1386653e-002	9.4931923e-004	2.0559854e-003	1.7776636e-003	1.5047682e-003	1.3639540e-003	1.4642312e-003
1.3839879e-002	1.2199446e-002	1.8649968e-003	1.6861084e-003	1.6396415e-003	1.6865389e-003	1.7273127e-003	1.6270355e-003
1.1810973e-002	9.8071116e-003	1.5028581e-003	1.4237635e-003	1.7323129e-003	1.2913206e-003	1.1683406e-003	1.1778292e-003
1.3154752e-002	1.0010399e-002	1.4878804e-003	1.9419801e-003	1.5353250e-003	1.4571423e-003	1.4657248e-003	1.4562362e-003
1.3323916e-002	8.7922080e-003	1.4296773e-003	1.4598217e-003	1.9293976e-003	1.5789432e-003	1.5282727e-003	1.5372922e-003
1.3023255e-002	9.8619459e-003	1.5445332e-003	1.3504995e-003	1.3994949e-003	1.3803882e-003	1.5454561e-003	1.5364367e-003
1.3595177e-002	7.1279525e-003	1.5222569e-003	1.6174736e-003	1.2219612e-003	1.4843030e-003	1.5419895e-003	1.4129160e-003
1.4014895e-002	7.6556918e-003	1.6971904e-003	1.0534957e-003	1.3258478e-003	1.6225899e-003	1.6916291e-003	1.8207026e-003
1.3145500e-002	7.7205530e-003	1.7334695e-003	1.5023300e-003	1.7328889e-003	1.8464963e-003	1.5769714e-003	1.6390319e-003
1.3874634e-002	5.5163634e-003	1.4866342e-003	1.3234109e-003	1.5390035e-003	1.7550483e-003	1.8978476e-003	1.8357870e-003
1.2483257e-002	3.9759002e-003	1.7917793e-003	1.7163514e-003	1.4643278e-003	1.5158057e-003	1.6203906e-003	1.8914916e-003
1.2805318e-002	3.4383188e-003	1.8805655e-003	1.8808181e-003	1.4707359e-003	1.3594357e-003	1.5107436e-003	1.2396425e-003
1.3304691e-002	3.0982045e-003	1.5222188e-003	1.5025850e-003	1.7066936e-003	1.4310792e-003	1.2501055e-003	1.3126787e-003
1.2590112e-002	2.3697494e-003	1.8115852e-003	1.5752994e-003	1.7575263e-003	1.7918733e-003	1.7169542e-003	1.6543811e-003
1.2943986e-002	1.8279051e-003	1.4390099e-003	1.7573775e-003	1.4404991e-003	1.6405176e-003	1.3825819e-003	1.6272253e-003
1.3008299e-002	1.6909579e-003	1.7024464e-003	1.4399255e-003	1.4859220e-003	1.2552542e-003	1.6511566e-003	1.4065131e-003
1.1487764e-002	1.4806322e-003	1.2141187e-003	1.2748592e-003	1.3077499e-003	1.4655091e-003	1.5817721e-003	1.5603944e-003
1.1418628e-002	1.5194268e-003	1.4951581e-003	1.6400115e-003	1.4103162e-003	1.5842960e-003	1.3300662e-003	1.3514439e-003
1.1948246e-002	1.4189615e-003	1.4409206e-003	1.2433341e-003	1.8540624e-003	1.6120679e-003	1.4712369e-003	1.6941508e-003

Table A.2 The 128×8 time-frequency map of class I signals as explained in Section 4.7

1.1535798e-002	1.5945760e-003	1.2029605e-003	1.5060539e-003	1.7537086e-003	1.4925629e-003	1.9574271e-003	1.7345132e-003
9.2753660e-003	1.8252657e-003	1.6662498e-003	1.6141522e-003	1.9288593e-003	2.0331163e-003	1.6163820e-003	1.3472294e-003
1.1453156e-002	1.2080393e-003	1.3774982e-003	1.8240229e-003	1.6756426e-003	1.6218854e-003	1.7145865e-003	1.9837391e-003
1.1381459e-002	1.6306702e-003	1.5700913e-003	1.4032345e-003	1.6480729e-003	1.3031680e-003	1.5559203e-003	1.5996897e-003
1.1084443e-002	1.7292737e-003	1.7098473e-003	1.7141010e-003	1.3479605e-003	1.2647399e-003	2.0258753e-003	1.9821059e-003
1.1561992e-002	1.5194694e-003	1.2820682e-003	1.4870224e-003	1.2030418e-003	1.6006795e-003	1.2077320e-003	1.4242673e-003
1.0900987e-002	1.5448988e-003	1.6155080e-003	1.7818467e-003	1.5687977e-003	2.0519257e-003	1.4309855e-003	1.2144502e-003
1.0143115e-002	1.2027259e-003	1.7238007e-003	1.7900474e-003	1.5059335e-003	1.1318488e-003	1.0612881e-003	1.3199119e-003
1.0065196e-002	1.1688457e-003	2.0455518e-003	1.7032546e-003	1.6627795e-003	1.6946396e-003	1.8886584e-003	1.6300346e-003
1.1138595e-002	1.5105422e-003	1.5396901e-003	1.9232222e-003	1.6392537e-003	1.6679097e-003	1.5979692e-003	1.7376055e-003
9.6638864e-003	1.7559416e-003	1.5483550e-003	1.5136174e-003	1.5960963e-003	1.6884642e-003	1.6349467e-003	1.4953104e-003
9.3543236e-003	1.4641228e-003	1.2485075e-003	1.3967962e-003	1.3398774e-003	1.4014039e-003	1.7164694e-003	1.3391241e-003
9.7671626e-003	2.0028336e-003	2.0239800e-003	1.4113258e-003	1.3404870e-003	1.9211483e-003	1.3837854e-003	1.7611307e-003
9.7226695e-003	1.5950242e-003	1.5907771e-003	1.5360303e-003	1.8633146e-003	1.5360806e-003	1.3156082e-003	1.3016868e-003
9.4152560e-003	1.9619566e-003	1.2963012e-003	1.5789335e-003	1.4991075e-003	1.4053541e-003	1.8481240e-003	1.8620455e-003
8.8425769e-003	1.9621351e-003	1.3084365e-003	1.4627998e-003	1.3976561e-003	1.8207002e-003	1.5472908e-003	1.2878157e-003
9.4152499e-003	1.6457491e-003	1.7083098e-003	1.3554066e-003	1.8461530e-003	1.3946512e-003	1.5209455e-003	1.7804206e-003
8.9494876e-003	1.2044899e-003	1.6595574e-003	1.3823264e-003	1.4123504e-003	1.7640560e-003	1.9108250e-003	1.5939804e-003
8.0043355e-003	1.3889059e-003	1.4743823e-003	1.4655630e-003	1.4463904e-003	1.2918656e-003	1.2922117e-003	1.6090564e-003
6.8563465e-003	1.4652595e-003	1.4886058e-003	1.8999782e-003	1.6798802e-003	1.5881341e-003	1.4080943e-003	1.4008700e-003
7.9022417e-003	1.3487082e-003	1.4408522e-003	1.3552061e-003	1.7342717e-003	1.2786260e-003	1.3996988e-003	1.4069231e-003
8.1256443e-003	1.2223609e-003	1.6435320e-003	1.8669537e-003	1.3890219e-003	1.5984901e-003	1.4725446e-003	1.7828479e-003
7.1707101e-003	1.9349940e-003	1.4992344e-003	1.6735937e-003	1.3096314e-003	1.4788319e-003	1.6637444e-003	1.3534412e-003
7.8390582e-003	1.3340097e-003	1.8105694e-003	1.6191067e-003	1.9079822e-003	1.2159753e-003	1.4893243e-003	1.4414643e-003
7.8658308e-003	1.3378776e-003	1.6186590e-003	1.3085376e-003	1.5214646e-003	1.5228529e-003	1.4716438e-003	1.5195038e-003
7.1449436e-003	1.2696189e-003	1.3503426e-003	1.4249170e-003	1.4887421e-003	1.6688050e-003	1.3557753e-003	1.4152873e-003
7.8626702e-003	1.5119401e-003	1.8342304e-003	1.2728303e-003	1.5250636e-003	1.5381686e-003	1.6290585e-003	1.5695464e-003
8.0373465e-003	1.5362574e-003	1.3792094e-003	1.4240038e-003	1.2668152e-003	1.3122106e-003	1.3483539e-003	1.6420258e-003
7.9810160e-003	1.4642080e-003	1.5649371e-003	1.3341099e-003	1.2070924e-003	1.4194572e-003	1.6797851e-003	1.3861132e-003
7.5864669e-003	2.2051498e-003	1.3325803e-003	1.4569470e-003	1.5191688e-003	1.5899276e-003	1.2113104e-003	1.3690309e-003
7.3018065e-003	1.4070504e-003	1.4766401e-003	1.7429109e-003	1.3935066e-003	1.5624383e-003	1.6445843e-003	1.4868638e-003
6.9012818e-003	1.4618958e-003	1.7773038e-003	1.0897763e-003	1.3922216e-003	1.4690094e-003	1.5060249e-003	1.2527682e-003
5.9682190e-003	1.9873979e-003	1.4266925e-003	1.6260586e-003	1.1015196e-003	1.3573555e-003	1.1643715e-003	1.4176282e-003
7.6575437e-003	1.3545823e-003	1.2565164e-003	1.4092640e-003	1.7031508e-003	1.4180703e-003	1.3771999e-003	1.5815472e-003
5.9127150e-003	1.4347606e-003	1.3195097e-003	1.3046226e-003	1.3192090e-003	1.5697754e-003	1.7666142e-003	1.5622669e-003
5.8157370e-003	1.7565151e-003	1.6385506e-003	2.0061882e-003	1.5345444e-003	1.3847546e-003	1.4980629e-003	1.1410949e-003
5.4986658e-003	1.6314980e-003	2.0723262e-003	1.2642386e-003	1.4218209e-003	1.3991495e-003	1.1810360e-003	1.5380040e-003
5.9137558e-003	1.1994018e-003	1.6658737e-003	1.0504483e-003	1.0295735e-003	9.9138349e-004	1.3258097e-003	1.2706963e-003
5.4076611e-003	1.0642752e-003	1.3319550e-003	1.3077678e-003	1.5691767e-003	1.4817183e-003	1.2520974e-003	1.3072107e-003
5.0988880e-003	1.8034444e-003	1.2687207e-003	1.8564730e-003	1.7222300e-003	1.4402115e-003	1.3036345e-003	1.5938416e-003
5.4921966e-003	1.6961646e-003	1.1563350e-003	1.1623748e-003	1.5840656e-003	1.7050187e-003	1.5944233e-003	1.3042162e-003
6.1469385e-003	1.5589458e-003	1.3456780e-003	1.2950618e-003	1.7571196e-003	1.7256403e-003	1.7437202e-003	1.7365193e-003
4.3873594e-003	1.6348661e-003	1.2221232e-003	1.3682195e-003	1.4735811e-003	1.6180391e-003	1.8471315e-003	1.8543325e-003
4.3746710e-003	1.3788464e-003	1.7619103e-003	1.4468088e-003	1.5225269e-003	1.2992854e-003	1.3596634e-003	1.5024626e-003
4.0687279e-003	1.6666212e-003	2.0519852e-003	1.6816424e-003	1.2997377e-003	1.1828815e-003	1.3617027e-003	1.2189035e-003
4.5878486e-003	1.5615495e-003	1.4120426e-003	1.4231048e-003	1.2624831e-003	1.2983929e-003	1.3051180e-003	1.3827215e-003
4.5033086e-003	1.5386230e-003	1.6230540e-003	1.6442625e-003	1.2433516e-003	1.5956262e-003	1.3497019e-003	1.2720984e-003
4.7770440e-003	1.7781502e-003	1.8476447e-003	1.6711006e-003	1.7957170e-003	1.7777651e-003	1.9644161e-003	1.5816596e-003
4.2020766e-003	1.3323748e-003	1.0946949e-003	1.9132803e-003	1.1869534e-003	1.6221988e-003	1.3889637e-003	1.7717202e-003
3.3598534e-003	1.9402230e-003	1.5334699e-003	1.2644349e-003	1.7282996e-003	1.4909860e-003	1.4358233e-003	1.3075303e-003
2.9184660e-003	1.5995496e-003	1.7096037e-003	1.3641715e-003	1.2963870e-003	1.3737894e-003	1.4755362e-003	1.6038292e-003
2.6358113e-003	1.3142691e-003	1.4434811e-003	1.7398476e-003	1.8561343e-003	1.8389552e-003	1.5526245e-003	1.3868717e-003
2.7840358e-003	1.6934036e-003	1.3391702e-003	1.9203240e-003	1.8077331e-003	1.3597694e-003	1.6233238e-003	1.7890766e-003
2.6971099e-003	1.1409350e-003	1.8732996e-003	1.9789723e-003	1.5755510e-003	1.5918830e-003	1.6348281e-003	1.5367377e-003
2.3279462e-003	1.5483663e-003	1.5590567e-003	1.7114368e-003	1.5159427e-003	1.7073713e-003	1.6872025e-003	1.7852929e-003
2.3048626e-003	1.3694964e-003	1.3603727e-003	1.2708776e-003	1.7188978e-003	2.0803959e-003	1.9327811e-003	1.3417081e-003
1.8977261e-003	1.2214860e-003	1.4267576e-003	1.6006004e-003	1.8048044e-003	1.3620507e-003	1.5404223e-003	2.1314953e-003
1.9065490e-003	1.5263041e-003	1.2223825e-003	1.4297266e-003	1.4302534e-003	1.7354177e-003	1.3707261e-003	1.7168390e-003
1.4845319e-003	1.4570145e-003	1.5867198e-003	1.3451971e-003	1.4743209e-003	1.5937590e-003	1.9276938e-003	1.5815809e-003
1.4411847e-003	1.7346743e-003	1.6843781e-003	1.7824602e-003	1.4173595e-003	1.1060492e-003	1.5035969e-003	1.6226960e-003
1.7215900e-003	1.6206864e-003	1.3280432e-003	1.3600613e-003	1.6924214e-003	1.4213447e-003	1.6881582e-003	1.5690592e-003
1.6790973e-003	1.2406632e-003	1.3270884e-003	1.7507708e-003	1.6524405e-003	1.6994367e-003	1.0477290e-003	1.2080193e-003
1.4637701e-003	1.0880842e-003	1.5680048e-003	1.1951705e-003	1.3452163e-003	1.5372604e-003	1.5246070e-003	1.3643166e-003

Table A.2 (continued)

2.7635075e-003	2.3922799e-003	2.9412609e-003	4.8756644e-001	7.3266615e-002	3.7278358e-002	4.5440214e-002	2.3699427e-001
2.8555192e-003	2.3330089e-003	5.4294064e-003	3.7895983e-001	2.4749904e-001	8.2793721e-001	2.9925714e-001	5.5658033e-002
3.0159518e-003	2.2527817e-003	5.8858465e-002	1.1013849e-001	4.1037122e-001	1.4041900e-001	1.1727038e-001	1.8980237e-002
2.3897768e-003	3.1006952e-003	1.4999300e-001	3.4165379e-002	3.5983359e-001	7.9492513e-001	1.4084506e-001	1.0544480e-002
2.4318426e-003	3.2048498e-003	2.3389375e-001	3.2835681e-002	1.0053000e-002	6.7892885e-002	1.1636128e-002	1.3575068e-003
2.7781312e-003	2.2687289e-003	2.4798814e-001	7.5063768e-002	4.9419140e-003	7.1900625e-003	3.6477859e-002	1.2962296e-002
2.6482301e-003	3.8897898e-003	1.5990943e-001	1.2473079e-001	8.0496559e-001	4.2116268e-002	2.0110101e-002	1.2328527e-004
2.4300781e-003	8.0902178e-003	1.0966960e-001	1.5525477e-001	2.8425512e-001	5.8651096e-002	1.6939144e-002	1.3167055e-003
2.8989684e-003	2.2925188e-002	4.7838839e-002	1.9761523e-001	2.2385214e-003	9.6635642e-003	1.1684921e-002	1.7527722e-003
2.5021264e-003	4.8310445e-002	2.5277657e-002	1.9875500e-001	4.0237755e-002	7.0569259e-003	2.8790931e-002	5.9785194e-003
3.2160239e-003	6.8047292e-002	1.4308325e-002	1.6558888e-001	1.2604413e-002	4.2086735e-002	1.6149829e-003	4.1251419e-003
2.9657322e-003	8.4902526e-002	1.1375118e-002	6.3494928e-002	2.5658840e-003	2.1881693e-003	1.9650433e-003	1.8857101e-003
2.7881632e-003	9.1354114e-002	1.5094190e-002	4.2219818e-003	9.0985091e-003	4.6403111e-003	3.7649922e-003	1.9929018e-003
2.5879537e-003	1.2090127e-001	2.2136584e-002	2.9215527e-003	6.8614029e-003	4.6764576e-003	2.5303895e-003	2.0933316e-003
2.1017638e-003	1.0958966e-001	3.7901187e-002	2.5001620e-003	8.3608127e-003	2.3905715e-003	4.3868589e-003	2.4491567e-003
2.5091838e-003	8.8286834e-002	4.2752346e-002	1.1844940e-001	5.8044069e-003	4.9697004e-003	2.3470741e-003	2.5165295e-003
2.8627160e-003	6.7447595e-002	5.8375877e-002	2.4318336e-003	3.2783254e-003	2.6699782e-003	2.9987888e-003	2.0522882e-003
3.2474032e-003	6.2821368e-002	6.5731108e-002	2.1502334e-003	3.2326583e-003	2.9398784e-003	2.9641678e-003	3.6692797e-003
3.3384132e-003	5.1087671e-002	7.0736137e-002	3.4717514e-003	2.9018042e-003	3.4245940e-003	2.0142314e-003	2.1374871e-003
5.5069829e-003	2.9063985e-002	8.1522268e-002	1.1237062e-002	1.3344730e-002	2.5547193e-003	2.0204948e-003	2.3535456e-003
1.1134433e-002	2.2665083e-002	8.8334945e-002	4.5037684e-003	2.8492205e-003	3.1651356e-003	3.6020651e-003	1.9181191e-003
1.1733224e-002	1.3687740e-002	8.2590518e-002	3.1150678e-003	3.5395441e-003	2.8510751e-003	5.5304299e-003	1.4504957e-003
1.6383236e-002	9.6037481e-003	8.2277847e-002	3.0665531e-003	2.5720476e-003	2.6967655e-003	2.8836810e-003	2.9445745e-003
2.1562297e-002	7.7289544e-003	7.5362717e-002	2.9370579e-003	3.2887721e-003	7.4921142e-003	2.5873610e-003	2.5382104e-003
2.7235168e-002	5.4371503e-003	6.2946420e-002	2.5947808e-003	3.3514854e-003	2.6881233e-003	2.9362160e-003	2.6401163e-003
2.8870769e-002	5.0136489e-003	5.4527706e-002	2.7227029e-003	3.1317838e-003	2.3311091e-003	2.4785857e-003	2.6702457e-003
3.3514485e-002	4.7897579e-003	2.6742345e-002	3.0681580e-003	2.3746601e-003	3.0270930e-003	2.0084346e-003	1.4010295e-003
3.8729566e-002	4.0599406e-003	9.5777863e-003	2.7776231e-003	2.3721497e-003	2.0292570e-003	2.0648610e-003	2.6265336e-003
3.5030742e-002	5.0528664e-003	3.2098936e-003	3.2706929e-003	2.8452719e-003	2.9203337e-003	3.1906799e-003	2.8701173e-003
4.2284673e-002	8.8696177e-003	2.6006107e-003	2.9268758e-003	3.1057207e-003	2.8713390e-003	2.5629166e-003	2.8870899e-003
4.5384781e-002	9.4290532e-003	2.6411573e-003	3.2984712e-003	2.6845358e-003	3.2251258e-003	3.0034799e-003	2.5747340e-003
5.1047434e-002	1.1300440e-002	2.1399932e-003	2.9253818e-003	2.7605547e-003	2.4908872e-003	2.6598346e-003	2.2465747e-003
4.0335418e-002	1.6812115e-002	2.6390547e-003	2.4904019e-003	3.2214021e-003	2.6957351e-003	2.4546942e-003	2.4699783e-003
4.3259257e-002	2.0165572e-002	2.5613919e-003	2.4723403e-003	2.6305128e-003	2.5499658e-003	2.8984581e-003	2.5220179e-003
3.6436198e-002	1.7331481e-002	2.4249975e-003	2.2039560e-003	2.6129507e-003	3.0639473e-003	2.5338732e-003	2.7306269e-003
2.9667401e-002	2.2978062e-002	2.9099187e-003	2.9267089e-003	2.1453097e-003	2.3716107e-003	2.5131716e-003	2.3297558e-003
2.4532331e-002	2.1372339e-002	2.4883987e-003	2.9409849e-003	2.1333836e-003	2.4818881e-003	2.9989508e-003	2.5478234e-003
3.4930107e-002	3.1495318e-002	2.5242885e-003	3.3095395e-003	3.0536598e-003	2.4900756e-003	1.9464956e-003	2.1913432e-003
2.1608450e-002	2.5145860e-002	3.2626935e-003	2.2554466e-003	2.5631506e-003	2.5727488e-003	2.1530671e-003	2.4353171e-003
2.2881802e-002	2.6529853e-002	2.8036314e-003	2.9313553e-003	2.0447340e-003	2.0243560e-003	2.2900919e-003	1.9107080e-003
2.0271305e-002	2.4930522e-002	2.6138518e-003	2.3117611e-003	2.9187426e-003	2.2620547e-003	2.6597828e-003	2.2548434e-003
1.8010809e-002	2.9519972e-002	2.1056339e-003	3.0691362e-003	3.1059215e-003	2.9404205e-003	2.5971094e-003	3.0033010e-003
1.2892116e-002	3.4444951e-002	2.8383076e-003	2.1783809e-003	2.4657201e-003	3.0228510e-003	2.8839238e-003	3.0211286e-003
6.8022229e-003	2.5942074e-002	2.9420994e-003	2.3258086e-003	2.8377647e-003	2.6519239e-003	2.6374628e-003	2.4975605e-003
1.0930748e-002	4.0356539e-002	2.5620353e-003	2.3930727e-003	2.5075455e-003	2.0793326e-003	2.1784496e-003	2.8204949e-003
5.5819238e-003	3.5537454e-002	2.4579388e-003	2.5637204e-003	2.4325602e-003	2.7781107e-003	3.2934771e-003	2.6908110e-003
6.9836429e-003	2.9901850e-002	1.6378092e-003	3.0683994e-003	2.7692474e-003	2.7807888e-003	1.8388930e-003	2.3011617e-003
2.5852270e-003	3.7960090e-002	3.1366957e-003	3.2866665e-003	2.5486700e-003	3.0350750e-003	3.2221590e-003	2.6805467e-003
2.6252159e-003	2.9256364e-002	2.3402213e-003	2.3879750e-003	2.8748299e-003	2.2133827e-003	2.1562305e-003	1.9768627e-003
4.7375328e-003	2.7552208e-002	2.3246006e-003	2.5970007e-003	2.7570002e-003	2.5049349e-003	2.5948742e-003	2.7945539e-003
2.3569856e-003	2.8020814e-002	2.0321595e-003	1.9801802e-003	2.6299615e-003	2.7282247e-003	2.6683167e-003	2.2273999e-003
1.6787467e-003	2.7778651e-002	2.6113705e-003	2.4459296e-003	2.3048405e-003	1.9043089e-003	1.8888176e-003	2.1124012e-003
1.5951925e-003	2.1666237e-002	2.4109981e-003	2.8090515e-003	2.0508882e-003	2.5978767e-003	2.6015853e-003	2.1589867e-003
1.2833528e-003	2.3686367e-002	2.7561551e-003	1.7777570e-003	2.1535042e-003	2.9055334e-003	2.7996777e-003	3.2453304e-003
1.6643478e-003	2.2229406e-002	3.2189708e-003	2.5094830e-003	3.2272644e-003	2.8624908e-003	2.7517527e-003	2.6518261e-003
8.9338792e-004	1.2636238e-002	2.8845044e-003	2.1024680e-003	2.5510408e-003	2.7035292e-003	2.9520992e-003	3.0571038e-003
1.2962689e-003	9.7076861e-003	3.0025694e-003	2.9826011e-003	2.6092047e-003	2.6178573e-003	2.4796873e-003	2.7887627e-003
6.4209557e-004	7.8708798e-003	2.9442770e-003	3.6214938e-003	2.4614023e-003	1.9209259e-003	2.5508157e-003	2.2606752e-003
1.3427311e-003	4.3175175e-003	2.3231863e-003	2.7892571e-003	3.0365432e-003	2.6294398e-003	1.9594848e-003	2.4295205e-003
1.1339554e-003	3.2655958e-003	3.3016928e-003	2.6547696e-003	3.3407300e-003	3.1990653e-003	3.2811613e-003	2.7697750e-003
1.7876721e-003	2.6549143e-003	2.6701910e-003	2.9289665e-003	2.4205435e-003	2.8142418e-003	1.9536116e-003	2.7095364e-003
2.3034009e-003	2.7537936e-003	2.9039795e-003	2.3865109e-003	2.6717970e-003	1.9089277e-003	3.0060041e-003	2.2317837e-003
2.8875962e-003	2.7393119e-003	2.0800014e-003	2.4445986e-003	2.1237386e-003	2.3434979e-003	3.0447598e-003	2.6605225e-003

Table A.3 The 128×8 cost table for the classification problem of Section 4.7

2.4583475e-003	2.7642261e-003	2.5993634e-003	2.9645846e-003	1.8885574e-003	2.8841099e-003	1.9618572e-003	2.3398204e-003
3.7617993e-003	2.5845981e-003	2.4897936e-003	2.2245358e-003	3.1435441e-003	2.7505085e-003	2.4963755e-003	2.9623204e-003
2.8693092e-003	2.7964934e-003	2.0804500e-003	2.1820631e-003	2.8337717e-003	2.4481380e-003	3.1168598e-003	2.6358107e-003
3.7910935e-003	3.3110325e-003	2.9287446e-003	2.2315492e-003	3.2073775e-003	3.2641716e-003	2.6194295e-003	2.0042835e-003
4.9230690e-003	2.2022992e-003	2.5094848e-003	2.9403532e-003	3.0510006e-003	2.6087265e-003	2.8109002e-003	3.4299605e-003
6.5482740e-003	2.9713196e-003	2.8638655e-003	2.5848360e-003	2.8063388e-003	2.2531179e-003	2.2981886e-003	2.7663473e-003
6.7120587e-003	3.1510452e-003	3.1070639e-003	2.8290816e-003	2.2304386e-003	2.0949542e-003	3.6224126e-003	3.1734023e-003
7.1434398e-003	2.7687443e-003	2.2731861e-003	2.5516112e-003	2.1755085e-003	2.6732864e-003	2.2632399e-003	2.5318946e-003
5.8905011e-003	2.8150813e-003	2.4035951e-003	3.0173707e-003	2.4133158e-003	3.6843483e-003	2.5224024e-003	2.2537026e-003
6.7838761e-003	2.1892409e-003	2.3919315e-003	2.8914706e-003	2.5214459e-003	1.9489099e-003	1.8899457e-003	1.9204806e-003
8.7453601e-003	1.9460750e-003	2.7645670e-003	3.2409390e-003	2.8608453e-003	2.8766770e-003	2.8218741e-003	2.7819805e-003
1.0287359e-002	2.8388112e-003	2.5889538e-003	3.2970403e-003	2.3624824e-003	2.7204308e-003	2.8110718e-003	2.8756824e-003
7.8454766e-003	2.5642975e-003	2.8497100e-003	2.7307059e-003	2.4011892e-003	2.6430085e-003	2.6318863e-003	2.5481315e-003
8.3327087e-003	2.3048687e-003	2.2684668e-003	2.3339844e-003	2.3745476e-003	2.3545775e-003	2.9261334e-003	1.8882131e-003
9.4126048e-003	2.8013970e-003	3.6870168e-003	2.3547659e-003	2.2929496e-003	3.3695854e-003	2.1282416e-003	3.1735502e-003
1.0732018e-002	2.1817274e-003	2.7356890e-003	2.6988406e-003	3.4231495e-003	2.6716878e-003	2.2837783e-003	2.3577701e-003
1.0702643e-002	2.8618951e-003	2.0960021e-003	2.7582122e-003	2.6333142e-003	2.1525009e-003	3.2119238e-003	3.1144028e-003
7.8601475e-003	2.7644901e-003	2.3051756e-003	2.4392075e-003	2.2258458e-003	2.9125181e-003	2.4715262e-003	2.1925617e-003
9.5271514e-003	2.7279712e-003	3.0546240e-003	2.2034227e-003	3.0863500e-003	2.3174173e-003	2.6003710e-003	2.8632492e-003
9.8881289e-003	2.2535810e-003	2.9302884e-003	2.4072528e-003	2.3275786e-003	2.9563974e-003	2.9709429e-003	2.5624021e-003
8.8077266e-003	2.5288957e-003	2.4469838e-003	2.6478518e-003	2.7298551e-003	2.0708954e-003	2.1963000e-003	2.6005557e-003
8.1257202e-003	2.6702039e-003	2.6246361e-003	3.6169627e-003	2.8118209e-003	2.7725714e-003	2.6740414e-003	2.7908484e-003
1.0751465e-002	2.4575870e-003	2.4412774e-003	2.3088558e-003	2.8692947e-003	2.2485633e-003	2.5052838e-003	2.3913973e-003
1.0680806e-002	2.2273596e-003	3.1339119e-003	2.5406461e-003	2.3957748e-003	2.7180882e-003	2.5134312e-003	2.7571933e-003
9.4445963e-003	3.5259044e-003	2.5873320e-003	2.9798791e-003	2.3216534e-003	2.7787311e-003	2.8182111e-003	2.5839276e-003
1.2069217e-002	2.4421433e-003	3.2692378e-003	2.9879157e-003	3.3991063e-003	2.5960329e-003	2.9493685e-003	2.5203922e-003
1.3524888e-002	2.2997927e-003	2.7588759e-003	2.2703183e-003	2.1832235e-003	2.5672877e-003	2.3350165e-003	2.7908965e-003
8.3688245e-003	2.1014618e-003	2.5465518e-003	2.5471739e-003	2.6682626e-003	2.5087686e-003	2.4159985e-003	2.4356744e-003
1.2897151e-002	2.6066530e-003	3.1062902e-003	2.3043314e-003	2.7772051e-003	2.7333930e-003	2.6302421e-003	2.6056508e-003
1.5833629e-002	2.6608089e-003	2.5056696e-003	2.6790382e-003	2.6034008e-003	2.5539428e-003	2.2507711e-003	2.9246272e-003
1.3840523e-002	2.6665527e-003	2.6352981e-003	2.2831104e-003	2.3918636e-003	2.4694369e-003	2.8534989e-003	2.1849811e-003
1.3345780e-002	3.9015446e-003	2.1741963e-003	2.7694506e-003	2.6322436e-003	2.9110340e-003	2.4749470e-003	2.7151514e-003
1.2623532e-002	2.5495297e-003	2.3807830e-003	3.0160638e-003	2.4201141e-003	2.6946429e-003	3.0589647e-003	2.8154364e-003
1.1473270e-002	2.7085790e-003	3.2391303e-003	1.7154242e-003	2.1414771e-003	2.5674306e-003	2.8420247e-003	2.2332012e-003
9.4731398e-003	3.2388829e-003	2.6089642e-003	3.0215262e-003	1.9828063e-003	2.4441643e-003	2.0763398e-003	2.6853036e-003
1.5271776e-002	2.3611875e-003	2.2639812e-003	2.5745302e-003	3.2982694e-003	2.3418921e-003	2.4105511e-003	2.9830367e-003
1.2599284e-002	2.4460765e-003	2.4672814e-003	2.6654834e-003	2.4646127e-003	3.0135556e-003	3.0351858e-003	2.4566691e-003
1.1391141e-002	3.2483081e-003	2.5974120e-003	2.8130153e-003	2.5888619e-003	2.4162200e-003	2.7241466e-003	2.2020428e-003
9.5920566e-003	2.8487697e-003	2.9627111e-003	2.1121055e-003	2.5653063e-003	2.3872743e-003	2.1788743e-003	2.7029404e-003
1.0392994e-002	2.1936572e-003	2.4656567e-003	1.8794554e-003	1.7562367e-003	1.7152372e-003	2.1555932e-003	2.2513234e-003
9.7960137e-003	1.8416665e-003	2.3225378e-003	2.3391134e-003	2.8336817e-003	2.6829890e-003	2.1460812e-003	2.0441946e-003
8.6795506e-003	3.0232423e-003	2.3181039e-003	3.3833763e-003	2.8210620e-003	2.4917942e-003	2.2899754e-003	2.8213018e-003
1.2747950e-002	3.2003862e-003	2.1016169e-003	2.0472448e-003	2.7145048e-003	2.9020246e-003	2.8255504e-003	2.2876565e-003
1.1351538e-002	2.8435973e-003	2.3855274e-003	2.3710845e-003	3.0510764e-003	2.9676976e-003	3.0840922e-003	2.7794546e-003
8.8521645e-003	2.9678682e-003	2.2824962e-003	2.5255248e-003	2.5056232e-003	2.7614712e-003	2.9355472e-003	3.2578773e-003
8.0162777e-003	2.2824953e-003	3.1286183e-003	2.4400050e-003	2.3934859e-003	1.9588774e-003	2.1785131e-003	2.3020823e-003
7.7525450e-003	3.1037296e-003	3.6724704e-003	2.7631855e-003	2.2081736e-003	2.0487072e-003	2.2810620e-003	2.1587844e-003
9.1997896e-003	2.7004833e-003	2.3813219e-003	2.4880851e-003	2.1650520e-003	2.1686902e-003	2.0466783e-003	2.3465435e-003
1.1325803e-002	2.5617846e-003	2.8290360e-003	2.8444254e-003	2.2495274e-003	2.7745289e-003	2.4071826e-003	2.1014102e-003
8.5856378e-003	3.0258629e-003	3.4615231e-003	2.5879931e-003	3.0602591e-003	3.1114427e-003	3.4611997e-003	2.6154313e-003
8.1570650e-003	2.5367240e-003	1.9708038e-003	3.1184449e-003	2.1811377e-003	2.6247880e-003	2.2745463e-003	3.1176202e-003
5.6079947e-003	3.2031822e-003	2.5761067e-003	2.2419745e-003	2.7494601e-003	2.5619960e-003	2.3861715e-003	2.2345851e-003
3.8770811e-003	2.8241550e-003	3.1217443e-003	2.5335079e-003	2.3070502e-003	2.3806442e-003	2.5294216e-003	2.6792376e-003
4.0184737e-003	2.2713305e-003	2.5704158e-003	3.0465625e-003	3.2452828e-003	3.2834233e-003	2.7220460e-003	2.5328524e-003
4.1751334e-003	2.9565306e-003	2.2913772e-003	3.2574501e-003	3.0807339e-003	2.3320143e-003	2.8043017e-003	2.9994141e-003
4.8051101e-003	1.9857631e-003	3.1333034e-003	3.5617489e-003	2.7185297e-003	2.6598914e-003	2.9095237e-003	2.6860874e-003
2.9302362e-003	2.6606959e-003	2.5731136e-003	2.9639251e-003	2.5231381e-003	2.8985231e-003	2.7282649e-003	2.9500170e-003
3.0811641e-003	2.4830153e-003	2.4542491e-003	2.3127772e-003	3.0353315e-003	3.7068312e-003	3.2832676e-003	2.4830195e-003
2.5417563e-003	1.9947707e-003	2.4178565e-003	2.6763033e-003	3.2745879e-003	2.3390914e-003	2.6379070e-003	3.4368763e-003
2.5127896e-003	2.4271645e-003	2.3231331e-003	2.5417160e-003	2.5394383e-003	2.9707656e-003	2.4423964e-003	3.0121988e-003
2.1360645e-003	2.3673425e-003	2.5823797e-003	2.3173072e-003	2.6611781e-003	2.8355763e-003	3.5134518e-003	2.9261272e-003
2.2584030e-003	2.9297077e-003	2.7253849e-003	2.9257193e-003	2.3501270e-003	1.7643669e-003	2.4327179e-003	2.7272362e-003
2.6916609e-003	2.6033595e-003	2.0935356e-003	2.3930784e-003	3.0195378e-003	2.2833340e-003	2.9504370e-003	2.6513997e-003
2.8560830e-003	2.0824627e-003	2.4778770e-003	3.2267587e-003	2.5174901e-003	2.9964587e-003	1.7894527e-003	2.0190995e-003
2.6672503e-003	1.9190601e-003	2.8711282e-003	1.9640433e-003	2.4120974e-003	2.8768821e-003	2.7284439e-003	2.5002710e-003

Table A.3 (continued)

1.6912032e-004	7.0328164e-005	3.1719838e-003	1.0859881e-003
-2.0456646e-003	2.4140426e-003	-5.9950294e-005	3.2765393e-003
7.0617145e-005	-3.4799717e-004	1.9184545e-003	5.8781499e-003
9.4984225e-004	2.9572114e-004	5.6361269e-004	-1.4021841e-003
-2.0665855e-003	-7.8095666e-004	9.2050809e-004	2.1914352e-003
-1.3576826e-003	1.7326900e-004	8.9636784e-004	8.1755336e-004
-9.9529667e-005	6.0849256e-004	4.5324470e-003	7.2243719e-004
-2.7549870e-004	7.8899217e-004	-3.0663766e-003	3.5355701e-003
5.7261393e-004	3.7100427e-003	-2.8701706e-003	-1.5204369e-003
-2.0622722e-003	4.4338451e-004	6.7349784e-005	-6.4919216e-004
-3.1788299e-005	1.4981601e-003	3.3576949e-003	-3.1277253e-003
-2.6153497e-003	1.2859910e-004	2.5538053e-003	-3.0360027e-003
-1.2865878e-004	-1.6019216e-003	-1.7309867e-003	4.2487257e-003
1.0566512e-003	2.5556287e-003	6.5428227e-003	5.2920638e-003
-2.6517532e-003	2.2697940e-003	1.2528857e-003	-1.0496315e-003
1.1729271e-004	1.5995283e-003	-3.1619216e-004	2.7111969e-003
-1.0323905e-003	-5.9257497e-004	-2.1552496e-003	6.5384184e-005
5.6046818e-003	-2.4352193e-003	8.2955569e-005	2.5197637e-004
-1.4270181e-003	-2.2451775e-003	2.7753180e-003	6.0175410e-004
2.3665953e-003	-9.6681787e-004	-1.3027072e-003	1.7875255e-003
-5.8695451e-004	4.5173583e-003	-4.2485920e-003	7.5177194e-004
6.1162740e-004	3.2226323e-003	1.3501041e-003	-1.9383081e-003
-1.0250531e-003	-1.3472201e-003	3.7706723e-003	-2.9158638e-004
-6.5437382e-004	1.4021715e-003	-3.3934850e-005	3.4815851e-004
3.6209973e-003	3.4520702e-003	2.5022465e-003	-1.8550259e-003
-8.7615589e-004	-4.7280560e-004	-1.9538871e-003	-4.4323189e-004
1.5534758e-003	-3.6956533e-003	1.6537234e-003	1.2137803e-003
2.2509432e-003	3.1133515e-003	-3.3029471e-003	-8.6277184e-004
-1.6649259e-003	-2.4507485e-003	3.2258988e-003	-2.5437427e-003
-2.1404016e-003	1.8534619e-003	2.2715520e-003	3.3487581e-004
-2.6200791e-003	1.2843475e-003	3.9918820e-003	-4.4440554e-003
7.6229051e-004	-1.0396572e-003	4.3779783e-003	-9.1161934e-004
2.5806439e-004	8.0812052e-005	-2.2898575e-004	-1.5386916e-003
5.7373309e-005	1.2980881e-003	7.7767269e-004	1.0195972e-003
-2.9332939e-004	-1.1589964e-003	-2.0451282e-003	1.0127959e-003
-2.5547725e-003	-1.9004332e-004	6.0477575e-004	4.9953692e-004
3.9887201e-003	-1.8695483e-004	-1.6542405e-003	-1.0216612e-003
-1.5312310e-003	-1.2984622e-003	1.4955498e-004	-4.3071950e-003
1.7594751e-003	2.4349969e-003	-3.1418633e-003	2.7962592e-003
-1.8473480e-003	5.5549831e-004	-1.1240728e-003	7.6604696e-004
3.3167099e-003	8.4833410e-005	-1.2921591e-003	-2.7278979e-003
2.0163106e-003	2.2179421e-003	-4.6760152e-003	2.5688095e-003
-1.6680776e-003	1.5307357e-003	-8.5647559e-004	-2.0004377e-003
8.2398604e-004	-1.2577875e-003	-1.3853188e-003	-2.4201699e-003
-9.3602526e-004	3.2476264e-003	-1.1139522e-004	-1.6330312e-003
-6.3260158e-004	9.1337706e-005	2.9559695e-003	-1.3431484e-003
2.6177997e-005	9.7784474e-004	5.9873650e-004	2.1699254e-003
9.6537177e-004	1.5084380e-003	-3.4670385e-003	1.3749351e-003
-5.8761748e-004	-2.9766939e-003	-2.6133467e-003	2.5562199e-003
1.7789408e-003	-8.5781130e-004	-6.5641023e-003	-3.3456623e-003
-8.4463518e-004	-2.9936751e-003	-1.4596257e-003	-1.0738698e-003
1.3489206e-004	5.1485426e-004	3.4534008e-003	-3.1156392e-006
-1.5844223e-003	1.5492028e-004	2.0232919e-003	-3.6697339e-004
-3.6108464e-003	-1.4136028e-003	-8.7411245e-004	-5.4206902e-005
2.7544115e-003	6.7778316e-004	-8.9286371e-004	1.2535462e-003
1.9908344e-003	-4.9202474e-003	-9.0736403e-004	1.6225907e-003
1.9494777e-003	1.7038370e-003	-1.1825075e-003	4.5396379e-003
-8.0634421e-005	-4.1019337e-003	-2.1192349e-003	-5.9951547e-004
1.2994925e-003	-1.9363101e-003	2.6828604e-003	-3.5976722e-003
-1.1677846e-003	-3.5527013e-003	4.4368075e-004	2.1874383e-003

Table A.4 The first 64 rows of the 4096×4 wavelet packet decomposition table for a 64×64 pixel image as described in Section 4.8

1.5053417e-004	1.8137877e-004	2.1067494e-004	2.9350071e-004
1.8171573e-004	2.2913854e-004	2.5030264e-004	3.9577521e-004
1.3832844e-004	2.4115953e-004	2.4434576e-004	1.1420044e-003
1.9994330e-004	1.2956448e-004	2.9415958e-004	2.6808356e-004
1.6122399e-004	2.4023048e-004	2.2083449e-004	3.5398663e-004
2.1160072e-004	2.3451895e-004	2.3423686e-004	5.2754695e-004
2.3287151e-004	2.2125468e-004	2.5720816e-004	3.7852911e-004
1.5101773e-004	3.5080073e-004	2.8269306e-004	3.1789137e-004
2.4847897e-004	2.9268982e-004	3.2549018e-004	1.9699598e-004
1.9666671e-004	1.8179452e-004	2.9570727e-004	2.2709930e-004
1.9785051e-004	2.1331017e-004	3.1417650e-004	2.3999788e-004
1.6249931e-004	2.2024984e-004	2.6649074e-004	2.6212969e-004
1.6485893e-004	2.0943712e-004	2.5157092e-004	2.5621150e-004
2.5216374e-004	2.8029987e-004	4.5025289e-004	2.6213659e-004
2.4542511e-004	1.7993891e-004	2.6131293e-004	2.4403486e-004
2.0033149e-004	3.2423147e-004	2.4711663e-004	2.7445513e-004
2.0224681e-004	1.9363140e-004	2.9448250e-004	1.9778194e-004
2.2672058e-004	2.2767288e-004	2.0198725e-004	2.2342736e-004
2.6557142e-004	2.2009004e-004	2.1556908e-004	1.8544368e-004
1.4498092e-004	2.1951469e-004	1.6668506e-004	1.9480301e-004
2.1565064e-004	1.7092649e-004	2.8497312e-004	1.7456692e-004
2.8520535e-004	2.2379979e-004	1.5737538e-004	1.8696171e-004
1.7333607e-004	2.6748722e-004	2.0024445e-004	1.4548522e-004
2.7238433e-004	1.7946487e-004	2.2139022e-004	1.5460858e-004
2.1654208e-004	2.4643364e-004	1.5755635e-004	2.6585585e-004
2.3978567e-004	2.7588832e-004	1.7610579e-004	2.0593810e-004
2.2785433e-004	2.7145163e-004	2.1970055e-004	1.7933779e-004
2.2246940e-004	2.6407763e-004	2.0457492e-004	2.7839828e-004
2.0394655e-004	2.7782661e-004	1.8376618e-004	3.2666984e-004
1.5027131e-004	2.7345072e-004	3.9336091e-004	3.0146367e-004
2.5317677e-004	2.1016304e-004	2.4054671e-004	2.7188218e-004
1.8888507e-004	2.5767552e-004	3.3863727e-004	2.6413887e-004
2.4851339e-004	1.8774075e-004	2.7164285e-004	2.1664423e-004
3.0170483e-004	1.7593524e-004	2.4188035e-004	2.6497100e-004
2.7982463e-004	2.0635353e-004	1.9359529e-004	2.9540601e-004
1.7374824e-004	2.7589576e-004	1.3796737e-004	2.3498110e-004
1.6541179e-004	1.4822934e-004	1.7242000e-004	2.3301998e-004
2.1482404e-004	1.6710990e-004	1.9229215e-004	3.4799286e-004
2.2503186e-004	1.5514124e-004	2.2500451e-004	2.2222571e-004
1.6914188e-004	2.4487083e-004	1.4171561e-004	1.6415804e-004
2.1553866e-004	2.2231357e-004	2.0904124e-004	1.9977796e-004
2.7824785e-004	1.9300600e-004	2.1012356e-004	1.8391803e-004
2.4019733e-004	2.6314476e-004	1.7344190e-004	1.5738404e-004
2.5418524e-004	2.2451609e-004	2.3846056e-004	2.2380683e-004
2.6876594e-004	3.3539432e-004	2.2159726e-004	2.3128876e-004
2.0382201e-004	2.8143218e-004	1.9574245e-004	2.3146168e-004
1.6769950e-004	1.7845587e-004	2.7604849e-004	1.8228760e-004
2.8144156e-004	2.4937742e-004	2.6774468e-004	2.8233003e-004
1.9711947e-004	3.1744744e-004	2.3947728e-004	1.4805538e-004
2.7937269e-004	3.0010514e-004	2.9897360e-004	2.7755448e-004
1.8416640e-004	1.7940979e-004	1.5929693e-004	1.9947193e-004
1.5885246e-004	1.4819415e-004	2.2148854e-004	2.3446632e-004
2.1616075e-004	1.4651419e-004	1.8758424e-004	2.6304720e-004
2.4450188e-004	1.5516091e-004	1.9557284e-004	1.4964503e-004
1.3761543e-004	1.9698132e-004	1.8563492e-004	2.0108302e-004
1.9447243e-004	1.9697326e-004	2.4577141e-004	2.8116160e-004
1.2845596e-004	2.0533438e-004	2.2147556e-004	2.3265663e-004
1.9402245e-004	2.3746374e-004	1.9024556e-004	2.5611598e-004
2.0436429e-004	2.3911099e-004	2.7236834e-004	2.7169762e-004
1.7712219e-004	1.3994618e-004	2.2424672e-004	1.7798169e-004

Table A.5 The first 64 rows of the 4096×4 energy map table for class I signals generated by the statistical process described in Section 4.8

2.2896252e-005	3.1896327e-005	9.9920974e-005	1.4378064e-002
1.1561855e-004	2.6295277e-005	6.9021854e-005	1.4418862e-002
3.5395633e-005	2.6001959e-005	7.0742147e-006	3.3644941e-003
2.0765261e-004	1.1273998e-004	5.9048574e-005	1.4509244e-003
3.1119121e-005	4.6621201e-006	2.4982772e-005	5.5795478e-005
3.1786942e-005	1.7432847e-006	6.0714806e-005	1.7266264e-005
8.9124829e-006	5.0063915e-005	5.4460591e-005	1.9257374e-006
1.8848903e-004	1.3240359e-004	4.4807460e-005	2.7778219e-004
1.1373219e-005	6.6962624e-005	1.2404129e-004	4.2018180e-005
4.6175560e-005	5.1551470e-005	1.6456546e-005	5.4309356e-006
8.5757492e-006	1.4542734e-005	1.3184354e-004	8.5476180e-006
1.7667504e-004	1.7121814e-005	1.7200533e-004	6.7866851e-005
1.8171239e-004	3.4261425e-005	1.5838410e-006	4.5139202e-005
5.0712564e-005	1.1004192e-005	1.3978806e-004	1.2352005e-006
4.8185531e-006	3.5528945e-005	3.0623013e-005	6.7702853e-005
4.2408498e-005	9.2009194e-005	2.1757025e-005	5.3385359e-005
8.7636629e-005	1.1211253e-004	8.5763170e-005	1.8261965e-005
2.6103440e-005	7.8562336e-005	1.5959075e-005	4.8717515e-006
7.9186013e-006	1.9276107e-005	4.3066389e-006	9.7835711e-005
8.7194845e-005	1.4014061e-005	5.6885835e-005	3.3581756e-005
4.0698280e-007	1.2365649e-004	1.2377422e-004	1.0427434e-005
1.1681977e-005	3.9662885e-006	7.7005849e-005	1.0536503e-004
3.2868664e-005	7.5449585e-005	1.4003414e-005	2.5634126e-004
7.1293609e-005	3.1737183e-005	1.3009753e-006	1.9303885e-005
2.9049808e-005	3.4012353e-005	8.6160861e-005	1.1935587e-004
6.0637109e-007	4.4376784e-006	8.4820408e-006	2.7401171e-005
3.4630219e-005	9.8247971e-005	2.2098872e-005	5.6177605e-005
2.2886945e-005	1.6731250e-005	3.8135085e-005	2.9504620e-005
4.7430399e-005	6.5313896e-006	1.0308174e-004	1.9603012e-004
1.3597843e-004	3.0815633e-005	2.3755199e-004	6.9922063e-005
1.0759515e-004	7.8296060e-005	3.1493880e-005	7.8855363e-005
8.8319432e-005	1.6655239e-005	1.9617992e-004	1.9904195e-005
2.7027252e-005	6.3060091e-005	3.1860602e-005	3.5001324e-005
7.0204507e-005	2.6691311e-006	2.1022422e-005	5.8794686e-005
2.4685133e-005	5.7568351e-006	5.5901975e-005	3.2883231e-005
9.6829222e-005	1.0757262e-004	5.5059853e-005	1.0210845e-005
3.3249133e-005	1.0494934e-004	3.1623921e-005	9.0258082e-005
3.6942397e-006	3.3608139e-005	1.7943582e-005	5.3782429e-005
6.2138727e-005	5.0799800e-005	1.2821643e-005	1.2025931e-006
5.1571618e-005	5.9081231e-005	6.4041905e-005	1.8671587e-005
2.8370870e-005	3.4715622e-005	3.0073994e-005	8.2572896e-006
1.5255547e-004	5.2861896e-005	3.5002552e-005	2.1154984e-005
2.1775449e-005	1.2253844e-005	1.9295944e-005	4.3460026e-005
7.5201639e-006	4.9711726e-005	8.6666129e-006	1.1212290e-005
5.0038416e-005	9.1644610e-005	8.3333539e-005	4.2804076e-005
9.8813814e-006	7.2323754e-005	1.9839458e-005	2.9838550e-005
8.7261331e-005	2.7903675e-005	8.3554796e-005	7.5457749e-005
6.7245499e-005	9.9064235e-005	4.4119177e-005	3.3391908e-005
9.8931092e-006	4.7661450e-005	3.1670376e-005	1.8943416e-004
5.4545348e-005	5.3792471e-005	1.1965271e-004	8.9476813e-005
2.4273831e-005	8.6275316e-005	1.5501877e-004	1.3264273e-004
3.5042739e-005	1.4656790e-004	1.0002323e-005	9.6505717e-005
4.5581508e-005	2.0018186e-004	2.4743684e-005	5.5150443e-005
9.0478835e-006	8.6645328e-006	5.3207667e-005	5.0720807e-005
1.0946209e-004	8.1717504e-006	2.3587362e-005	6.2942546e-006
9.4296616e-005	1.7585680e-005	1.2950805e-005	3.4043079e-005
7.6946214e-005	1.2078916e-005	3.3740409e-005	4.6850678e-005
2.5226490e-005	3.6069582e-005	1.3069840e-004	2.6880833e-005
1.7127062e-005	2.4316380e-005	1.4449901e-004	5.1807572e-005
1.1598265e-004	1.3676060e-004	2.1682643e-005	9.9616366e-005

Table A.6 The first 64 rows of the 4096×4 cost table for classification of signals generated by statistical process described in Section 4.8

Bibliography

[1] R. O. Duda, P. E. Hart, D. G. Stork, *Pattern Classification*, New York: Wiley, second edition, 2001.

[2] B. J. Frey, *Bayesian Networks for Pattern Classification, Data Compression and Channel Coding*, Ph.D. Thesis, University of Toronto, 1997.

[3] R. R. Coifman and M.V. Wickerhauser, "Entropy-based Algorithms for Best Basis Selection," *IEEE Trans. Inform. Theory,* Vol.38, no.2, pp. 713-719, 1992.

[4] M.V. Wickerhauser, *Adapted Wavelet Analysis from Theory to Software*, A K Peters, Ltd. Wellesley, MA, 1994.

[5] R. R. Coifman and M.V. Wickerhauser, "Adapted Waveform "De-noising" for Medical Signals and Images," *IEEE Engineering in Medicine and Biology Magazine*, Vol.14, no.5, pp. 578 –586, 1995.

[6] N. Saito and R. R. Coifman, "Local Discriminant Bases and Their Applications," *J. Mathematical Imaging and Vision,* Vol.5, no.4, pp. 337-358, 1995, *Invited paper*.

[7] N. Saito, *Local Feature Extraction and Its Application Using a Library of Bases*, Ph.D. Thesis, Dept. of Math, Yale University, New Haven, CT 06520 USA, Dec. 1994.

[8] L. Debnath, *Wavelet Transforms and Their Applications*, Boston, Birkhauser, 2002.

[9] S., G., Mallat, *A Wavelet Tour of Signal Processing*, San Diego, Academic Press, 1998.

[10] L. Cohen, "Time-frequency Distributions-A Review," *Proceedings of the IEEE*, Vol.77, no.7, July 1989, *Invited Paper*.

[11] S. G. Mallat, Z. Zhang, "Matching Pursuits with Time-frequency Dictionaries," *IEEE Trans. Signal Processing*, Vol. 41, pp. 3397-3415, 1993.

[12] J. H. Friedman and J. W. Tukey, "A Projection Pursuit Algorithm for Exploratory Data Analysis," *IEEE Trans. Comput.*, Vol. 23, pp. 881-890, 1974.

[13] A. Bultan, "A Four Parameter Atomic Decomposition of Chirplets," *IEEE Transactions on Signal Processing,* Vol.43, no.2, March 1999.

[14] F. G. Meyer and R. R. Coifman, "Brushlets: A Tool for Directional Image Analysis and Image Compression," *Appl. Comput. Harmon. Anal.*, Vol.4, no.2, pp.147-187, 1997.

[15] T. M. Apostol, *Introduction to Mathematical Analysis*, Addison-Wesley, Reading, Massachusetts, second edition, January 1975.

[16] T. M. Apostol, *Calculus*, Volume II, John Wiley & Sons, New York, second edition, 1969.

[17] G. Zhang, *Lecture of Functional Analysis*, Volume I, Beijing University Press, 1986.

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

[19] P. M. Woodward, *Probability and Information Theory with Applications to Radar*, Pergamon Press, London, 1953.

[20] D. E. Newland, *An Introduction to Random Vibrations, Spectral and Wavelet Analysis*, third edition, Longman Group Limited, London 1993.

[21] C. S. Burrus, R. A. Gopinath, H. Gua, *Introduction to Wavelets and Wavelet Transforms-A Primer*, Prentice-Hall, NJ, 1998.

[22] E. Wesfreid, M. V. Wickerhauser, "Adapted Local Trigonometric Transform and Speech Processing, *IEEE Transactions on Signal Processing*, Vol.41, no.2, pp. 3596-3600, December 1993.

[23] M. V. Wickerhauser, "Smooth Localized Orthonormal Bases," *Comptes Rendus de l'Academie des Sciences de Paris*, Vol. 316, pp. 423-427, 1993.

[24] B. Boashash, B. Lovell, and H. J. Whitehouse, ``High-resolution Time-frequency Signal Analysis by Parametric Modeling of the Wigner-Ville Distribution," *in Proc. IASTED Int. Symp. Signal Process. Appl.*, pp. 297-301, 1987.

[25] B. Boaushash and H. J. Whitehouse, ``Seismic Applications of the Wigner--Ville Distribution," *in Proc. IEEE Int. Conf. Circuits Syst.*, pp. 34-37, 1986.

[26] S. Mann and S. Haykin, ``Chirplet Tranform: Physical Considerations," *IEEE Trans. Signal Processing*, Vol. 43, pp. 2745--2761, Nov. 1995.

[27] S. Mann and S. Haykin, `` `Chirplets' and `Warblets': Novel Time-frequency Methods," *Electron. Lett.*, Vol. 28, pp. 114-116, Jan. 1992.

[28] D. Mihovilovic and R. N. Bracewell, ``Adaptive Chirplet Representation of Signals of Time-frequency Plane," *Electron. Lett.*, Vol. 27, pp. 1159-1161, June 1991.

[29] R. G. Baraniuk and D. L. Jones, ``Shear Madness: New Orthonormal Bases and Frames Using Chipr Functions," *IEEE Trans. Signal Processing*, Vol. 41, pp. 3543-3549, Dec. 1993.

[30] L. B. Almeida, ``The Fractional Fourier Transform and Time-frequency Representations," *IEEE Trans. Signal Processing*, Vol. 42, pp. 3084-3091, Nov. 1994.

[31] R.R. Coifman, Y. Meyer, *Remarques sur l'analyse de fourier `a fen"etre*. C.R. Acad. Sci. Paris I, pp. 259-261, 1991.

[32] D. W. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization*, John Wiley & Sons, New York, 1992.

[33] R. A. Fisher, "The Use of Multiple Measurements in Taxonomic Problems," *Ann. Eugenics*, Vol. 7, pp 179-188, 1936.

[34] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*, Chapman and Hall, Inc., New York, 1993.

[35] K. Fukunaga, *Introduction to Statistical Pattern Recognition,* second edition, Academic Press, San Diego, 1990.

[36] B. D. Ripley, "Statistical Aspects of Neural Networks," *Networks and Chaos: Statistical and Probabilistic Aspects*, O. E. Barndorff-Nielson, J. L. Jensen, D. R. Cox, and W. S. Kendall, eds., pp 40-123, Chapman and Hall, Inc., New York, 1993.

[37] N. Saito and R. R. Coifman, "Improved Local Discriminant Bases Using Empirical Probability Density Estimation," Proc. *on Statistical Computing*, American Statistic. Assoc., 1996.

[38] H. Yashida, "Matching Pursuit with Optimally Weighted Wavelet Packets for Extraction of Microcalcifications in Mammograms," *Applied Signal Processing*, no.5, pp 127-141, 1998.

[39] G. Kronquist and H. Storm, "Target Detection with Local Discriminant Bases and Wavelets," *Proc. of SPIE*, v.3710, pp 675-683, 1999.

[40] N. Saito, R. R. Coifman, "Extraction of Geological Information from Acoustic Well-Logging Waveforms Using Time-Frequency Wavelets," *Geophysics*, v.62, n.6, 1997.

[41] L. S. Rogers, C. Johnston, "Land Use Classification of SAR Images Using a Type II Local Discriminant Basis for Preprocessing," *Proc. of IEEE Conference on Acoustics, Speech and Signal Processing*, v.5, pp 2729-2732, 1998.

[42] M. L. Cassabaum, H. A. Schmitt, H. W. Chen, J. G. Riddle, "Application of Local Discriminant Bases Discrimination Algorithm for Theater Missile Defense," *Proc. of SPIE*, Vol.4119, pp 886-893, 2000.

[43] M. L. Cassabaum, H. A. Schmitt, H. W. Chen, J. G. Riddle, "Fuzzy Classification Algorithm Applied to Signal Discrimination for Navy Theater Wide Missile Defense," *Proc. of SPIE*, Vol.4120, pp 134-145, 2000.

[44] J. M. Caballero Bejar, S. Krishnan, "Application of Time-Frequency Distributions in Ultrasound Imaging of Apoptosis," *Internal Report*, Electrical and Computer Engineering Department, Ryerson University.

[45] G. Czarnota, M. Kolios, M. Sherar, F. Ottensmeyer, J. Hunt, "High-Frequency Ultrasound Imaging of Apoptosis in Vitro, in Situ and in Vivo," *Faseb Journal*, Vol. 13(7), 1999.

[46] G. Czarnota, M. Kolios, M. Portnoy, F. Ottensmeyer, J. Hunt, M. Sherar, "Ultrasound Imaging of Apoptosis: High-Resolution Non-invasive Monitoring of Programmed Cell Death in Vitro, in Situ and in Vivo," *British Journal of Cancer*, Vol. 81(3), pp. 520-527, July 1999.

[47] WaveLab V.8.02, http://www.stat-stanfors.edu/~wavelab

[48] S. Watanabe, "Karhunen-Loeve Expansion and Factor Analysis: Theoretical Remarks and Applications," in *Trans. 4th Prague Conf. Inform. Theory*, Publishing House of Czechoslovak Academy of Sciences, pp 635-660, 1965.

[49] R. R. Coifman, N. Saito, "The Local Karhunen-Loeve Bases," *Time-Frequency and Time-Scale Analysis, Proceedings of the IEEE-SP International Symposium on*, pp 129 –132, 1996.

[50] P. Qua, Z. Lei, Z. Hongcai, D. Guanzhong, "Adaptive wavelet based spatially de-noising," Signal *Processing Proceedings, 1998 Fourth International Conference on*, pp 486-489,1998.

[51] P. Frossard, P. Vandegheynst, "Redundancy in Non-orthogonal Transforms," *Proc. of IEEE International Symposium on Inform. Theory*, n.01CH3725, p 196, 2001.

[52] C. M. Spooner, "Application of Local Discriminant Bases to HRR-based ATR," *Signals, Systems and Computers, Conference Record of the Thirty-Fifth Asilomar Conference on*, v.2, pp 1067 -1073, 2001.

[53] J. B. Kruskal, "Toward a Practical Method which Helps Uncover the Structure of a Set of Multivariate Observations by Finding the Linear Transformation which Optimizes a New 'Index of Condensation'", *Statistical Computation* (R. C. Milton and J. A. Nelder, eds.), Academic Press, New York, pp 427-440, 1969.

[54] FAWAV – A Fourier/Wavelet Analyzer (freeware), available form : http://www.uwec.edu/academic/curric/walkerjs/pages/downloadFAWAV.html

[55] J. Foote, "Content-based Retrieval of Music and Audio," in *Multimedia Storage and Archiving Systems II, Proc. of SPIE*, pp.138-147, 1997.

[56] N. Saito, R. R. Coifman, F. B. Geshwind, F. Warner, "Discriminant Feature Extraction Using Empirical Probability Density Estimation and A Local Basis Library," *Pattern Recognition, Journal of Pattern Recognition Society*, v.35, pp. 2841-2852, 2002.