Wavelet Denoising Techniques with Applications to High-Resolution Radar

by

Dewey S. Tucker

B.E.E., Georgia Institute of Technology (1995)

Submitted to the Department of Electrical Engineering
and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 1997

Author . . . .
Department of Electrical Engineering
and Computer Science
May 23, 1997

Certified by . . . .
Dr. Hamid Krim
Research Scientist, Laboratory for Information and Decision Systems
Thesis Supervisor

Accepted by . . . .
Arthur C. Smith
Chairman, Departmental Committee on Graduate Students

Wavelet Denoising Techniques with Applications to High-Resolution Radar

by

Dewey S. Tucker

Submitted to the Department of Electrical Engineering
and Computer Science
on May 23, 1997, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering

Abstract

The classical estimation problem, that of estimating an unknown signal in additive noise, has recently been revisited by researchers. Wavelets and wavelet packets can be attributed to the resurgence of interest in this area. One of the most significant properties of wavelets, which we exploit, is their ability to represent signals in a given smoothness class with very few large magnitude coefficients. In this research, we find an "optimal" representation of a given signal in a wavelet packet tree, such that removing noisy coefficients at a given threshold improves the signal quality and minimizes the error in reconstructing the signal.

To apply these denoising techniques, we seek a methodology that will extract significant features from high-resolution radar (HRR) returns for the purpose of Automatic Target Recognition (ATR). HRR profiles are one-dimensional radar returns that provide a "fingerprint" of a target. Using returns from a particular target, we create a database that contains signals representative of the target at different aspect angles, and the database is extended to include many possible targets. The ATR problem then reduces to a database search procedure in order to determine a target's identity and spatial orientation.

Thesis Supervisor: Dr. Hamid Krim
Title: Research Scientist, Laboratory for Information and Decision Systems

Research Support

This research was supported in part by the Air Force Office of Scientific Research (AFOSR BUGC12391NGD) and (AFOSR F49620-98-1-0349).

This material is based upon work also supported under a National Science Foundation Graduate Research Fellowship. Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the author and do not necessarily reflect the views of the National Science Foundation.

Acknowledgments

First and foremost, I express my sincere gratitude to Dr. Hamid Krim, my research supervisor, for his guidance and expertise. I thank him for all of the helpful and constructive comments that he provided concerning my thesis and research. He was always willing to discuss any problems that I had, even if he was busy at the time. I thank him for allowing me the privilege to work with him. I would also like to thank Alan Willsky for his advice in the small group meetings that I attended each week. His guidance was extremely helpful in developing the ATR algorithm proposed in this thesis. I thank him for all of his wise suggestions and the meetings he organized at AlphaTech.

I thank all of my professors at Georgia Tech and MIT for sharing their knowledge and allowing me to further my education. I would especially like to thank those who were kind enough to write recommendations, Professors Alfred Andrew, Mark Clements, John Dorsey, Mary Ann Ingram, Laurence Jacobs, Jim McClellan, and Russell Mersereau from Georgia Tech and Professor Munther Dahleh from MIT. Without their support, I would not have been able to attend MIT or have the NSF scholarship. I would also like to thank the National Science Foundation for providing me with three years of funding. This scholarship has allowed me the opportunity to focus all of my energies on classes and research.

I thank all of the members of the Stochastic Systems Group for helping me to make it through this thesis. Thanks go to Mike S., Cedric L., John R., Andrew K., Mike D., Terrence H., Austin F., Ben H., Andy T., Ilya P., Seema J., and Gildas S. Thanks for being great friends. I especially thank my good friend Andrew Kim for all of his technical advice. He was always willing to help me with my computer

problems, even when he had other work to do. I also thank him for giving me advice concerning my research. I thank Seema Jaggi and Mike Daniel for all of their moral support. They always listened to my problems and offered me advice about graduate life in general. I wish them both the best of luck in the future. I finally thank Andy Tsai, my new office-mate, for being a good friend, who supported me during some of those late nights. He was always willing to offer advice concerning my thesis, and I thank him for that.

I also thank Walter Sun, my current roommate and good friend from Georgia Tech. He has always been willing to answer my technical questions and offer advice. I thank him for his willingness to help other people and volunteer at a moment's notice. I also thank my friend Nick Laneman for all of his support and advice. I thank him for all of his help in 6.241, 6.432, 18.100, and in practicing for the OQE. I finally thank Beth Gerstein for being one of my greatest supporters and friends. Her enthusiasm for life in general has truly been a blessing to me, and I wish her luck in her future career.

I thank my extended family at First Baptist Church, Fitzgerald, for all of their kindness over the years. I especially thank Gene, Pat, Gerald, Marlene, Blane, Barbara, Susan, and Marty for their friendship and guidance. These acknowledgments would be incomplete if I did not thank God for all He has given me. I owe everything to Him, and I thank Him for providing me with the talents and opportunities that He has. I finally thank my family and the friends that I did not have the space to mention. I especially thank my parents for always supporting me in everything that I have done. They have provided me with everything I have ever needed. They have made many sacrifices to help me in my endeavors, and I thank them so much.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, wavelets have been the focus of intense research interest. Wavelets have, in fact, permeated every application area and scientific discipline in one form or another. On account of their unique properties, they almost challenge the Fourier transform as the analysis tool of choice in many applications. The power of wavelets lies in their ability to analyze transient signals, with inherently non-stationary statistics. A wavelet basis is well-adapted to these types of signals, providing information at different scales and translations. The analysis is simple since all of the wavelet basis functions are obtained from scales and shifts of a single "mother" wavelet.

Wavelets, however, are really only the beginning, since they represent a few of the many possible basis representations. Several different *overcomplete* representations of signals, that offer even more freedom, have been proposed. In this thesis, we have chosen to focus on wavelet packets, which provide an overcomplete representation with the wavelet basis as a special case. Since wavelet packets provide many choices of bases, a "best" basis must be selected, given a specific goal or application.

Compression is one popular goal, and will be the driving force in a significant portion of subsequent analyses.

Wavelets have certainly gained popularity in a multitude of new application areas, but they have also allowed investigators to revisit some of the classical engineering problems. One classic problem that underlies all fields of study involves estimating an unknown signal in noise. The traditional solution to this problem has been to minimize the estimation error given a model or statistics for the underlying signal and additive noise. These approaches lead to such techniques as the least-squares solution via the normal equations, the causal and non-causal Wiener filters, and the Kalman filter.

Wavelet-based denoising, however, leads to somewhat different approaches. One popular technique is to obtain the most compressed representation of a signal and then discard coefficients with magnitude less than a predetermined threshold. Compression is, in fact, one of the most important properties of wavelets. This property is important in communications applications because it allows a significant portion of a signal to be sent with only a few coefficients. These coefficients can then be represented by longer codewords as a safeguard against errors. Since the smaller coefficients only provide details, they are not as important and can be coded with a fewer number of bits.

The goal of compression in communications systems is, however, very similar to the goal of compression in the denoising problem. We place more confidence in the larger wavelet coefficients of a noisy signal than the smaller coefficients, since the large coefficients are more likely to contain some signal information. By removing the smaller coefficients, we essentially remove noise and thereby improve the quality of a signal. A more compressed representation of a signal therefore leads to less degradation in the underlying signal when thresholding is used, and as a result, an appropriate choice for the thresholding strategy will be an important topic in the wavelet denoising problem.

While compression may be an important goal that leads to effective denoising, it does not lead to an optimal solution in any measurable sense. An alternative solution is to find the "best" representation of a noisy signal which minimizes an appropriate error criterion. The advantage of this method over compression is that it accounts for the thresholding rule and the statistics of the additive noise. Formalizing this approach will be a significant contribution provided in this thesis.

As a second topic, we consider the problem of Automatic Target Recognition (ATR) using High-Resolution Radar (HRR) profiles. While ATR and denoising may seem somewhat disconnected, the two topics are actually not so distant. One specific reason for the connection is that we propose an algorithm for ATR which uses techniques very similar to those associated with denoising. Also, the HRR profiles chosen here can be modeled as an underlying signal plus noise. These profiles are a reasonable choice for the ATR problem because they capture important information about an object. The algorithm that we propose will use this information to determine a target's identity along with its spatial orientation.

1.1 Denoising Techniques

In this thesis, we treat the classical denoising problem of estimating an unknown signal in additive noise. The noisy observations $\{x(m)\}$ are given by,

$$x(m) \quad = \quad s(m) + v(m), \quad 0 \leq m \leq N-1, \tag{1.1}$$

where $\{v(m)\}$ is zero-mean white Gaussian noise with variance σ^2. Performing effective noise removal, in this case, consists of three important steps.

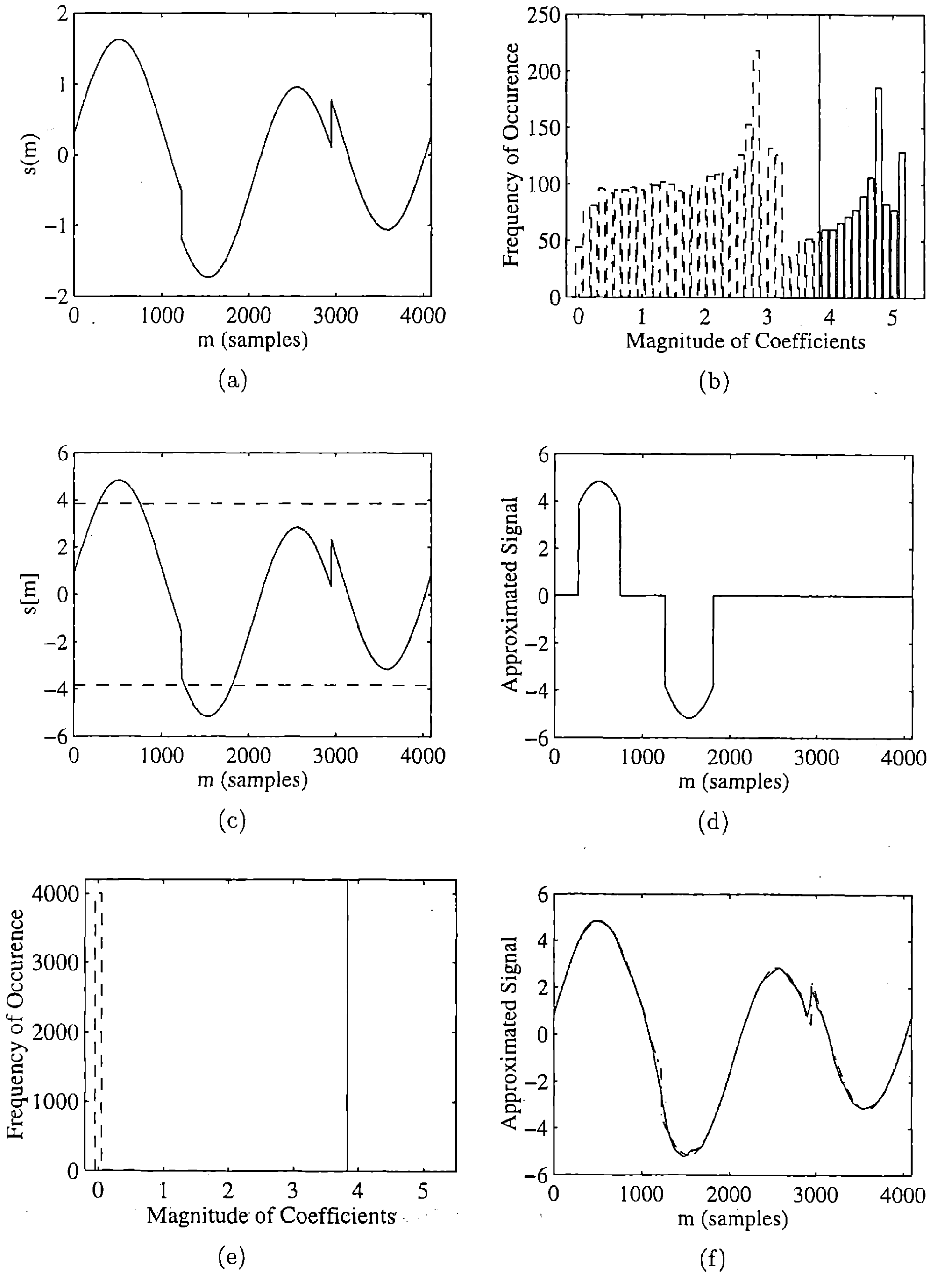
- **Best Signal Representation** – From a given class of possible representations, a "best" signal representation must be found based on some specified goal. Wavelet packets are used here to provide this family of representations, and a best basis search algorithm is used to efficiently identify the "best" representation. For the goal of compression, the optimal representation is the one that requires the fewest number of significant coefficients to represent a signal.
- **Cost Functions** – Cost functions are necessary ingredients in the denoising scheme because they are directly related to the meaning of "best". The "best" representations are the ones which minimize an appropriate cost function. For compression, we search for cost functions that penalize representations with a fairly uniform distribution of coefficients and that prefer the representations with more skewed distributions.
- **Thresholding** – Thresholding is necessary to remove unwanted noise from a signal. Since the statistics of white noise are invariant under orthogonal transformations such as the wavelet packet decomposition, the noisy wavelet coefficients will suffer the same degradation as the noisy signal. If the most compressed representation of a signal is found, thresholding smaller coefficients

will primarily remove noise without significantly reducing the quality of the underlying signal.

The above points have been presented assuming that the primary goal is compression. To illustrate these concepts, we present a simple example. A test signal composed of sinusoids with two discontinuities is shown in Figure 1.1(a). The histogram in Figure 1.1(b) shows the magnitude of the signal coefficients which are distributed somewhat uniformly between 0 and 5. To obtain a simpler representation of this signal, only the most significant coefficients must be used, and this corresponds to thresholding the coefficients at a given level T. Figures 1.1(c) and (d) show that thresholding is disastrous since the smaller coefficients contain a significant amount of information about the signal in this representation.

This problem is less troublesome if wavelet packets are used to find a better representation for the signal, as shown in Figure 1.1(e). The histogram shows that a significant portion of the coefficients are approximately zero, and consequently, most of the signal information is contained in a very small number of larger coefficients. Figure 1.1(f) shows an approximation to the original signal obtained by thresholding the coefficients below T and applying the inverse transformation. Since most of the signal information is contained in the larger coefficients, removing the smaller coefficients results in a good approximation to the original signal. This example shows how wavelet packets and an appropriately chosen cost function are useful in finding a compressed representation of a signal.

A more important issue concerns the utility of wavelet packets in the denoising problem, and to address this issue, we return to the problem statement given in Equation (1.1). The goal is to find the best estimate of an underlying signal $\{s(m)\}$ in the presence of additive noise $\{v(m)\}$, given N noisy observations $\{x(m)\}$. Knowing



(a) (b)

(c) (d)

(e) (f)

Figure 1.1: Illustration of the power of wavelet packets in finding a compressed representation of a signal. (a) HeaviSine signal. (b) Histogram of the signal. (c) Shows the thresholding region. (d) Shows the thresholded signal. (e) Histogram of the transformed signal obtained by a wavelet packet decomposition. (f) Signal approximated by the coefficients with magnitude above the threshold.

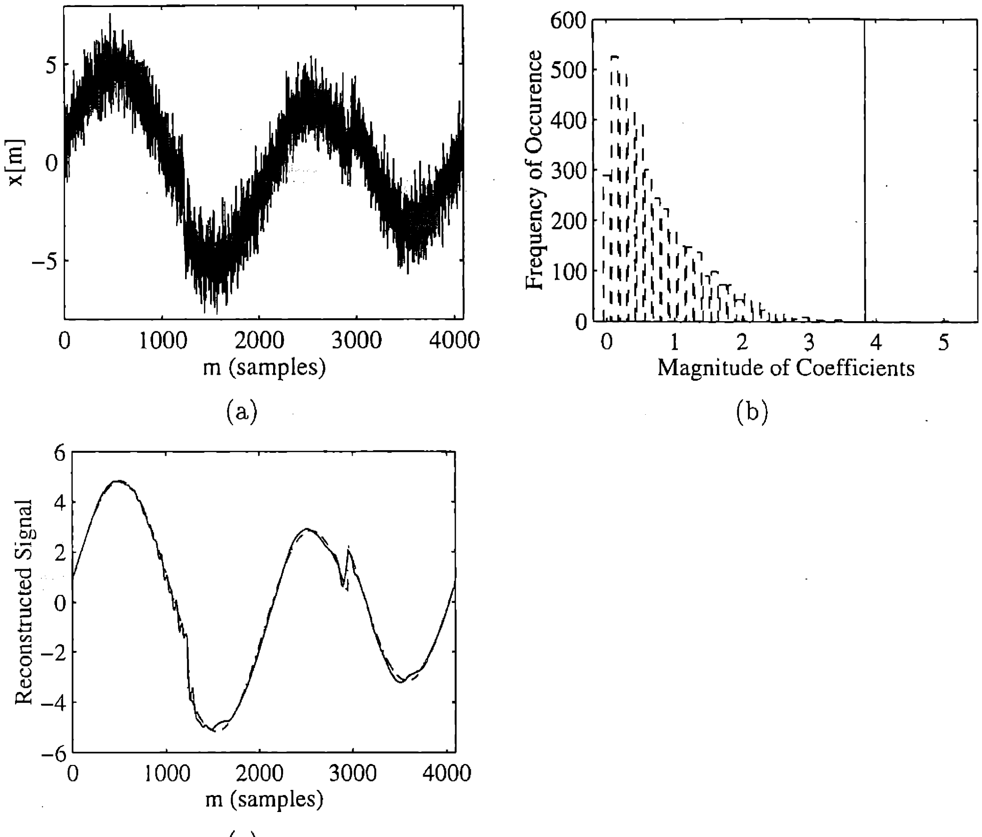
the statistics of the underlying signal $\{s(m)\}$ and the noise $\{v(m)\}$, we could use the linear least-squares estimator to determine the best estimate of $\{s(m)\}$. We do not, however, assume any particular distribution for the signal, and as a result, estimating $\{s(m)\}$ using Equation (1.1) is a very difficult problem. A partial solution can be obtained by applying an orthogonal transformation to the noisy observations. The denoising problem then becomes easier in this new representation. Applying a transformation to the observed signal, we obtain a different denoising problem,

$$\mathbf{W}\mathbf{x} = \mathbf{W}\mathbf{s} + \mathbf{W}\mathbf{v} \tag{1.2}$$

$$\boldsymbol{w}_x = \boldsymbol{w}_s + \boldsymbol{w}_v. \tag{1.3}$$

The coefficients in this new basis are therefore partitioned into coefficients belonging to the underlying signal and coefficients belonging to the noise. Since we have applied an orthogonal transformation (*i.e.* $\mathbf{W}^T\mathbf{W} = \mathbf{W}\mathbf{W}^T = \mathbf{I}$), the noise statistics remain invariant in the new basis. The denoising problem of Equation (1.3) is therefore identical to the denoising problem in Equation (1.1), but by choosing a "good" transformation, the noise removal process is more effective.

Consider another example which shows the utility of this transformation. A noisy version of the signal previously shown in Figure 1.1(a) is given in Figure 1.2(a). The variance of the additive noise is appropriately chosen to provide a signal-to-noise ratio (SNR) of 10 dB. Wavelet packets are then used to find the most compressed representation of the noisy signal. The histogram of the coefficients, shown in Figure 1.2(b), resembles a Gaussian distribution as expected. If we remove the coefficients below a predetermined threshold T, the noise is almost completely removed. At the same time, we retain the most important coefficients needed to reconstruct the underlying signal, as shown in Figure 1.2(c). This example focused on the specific goal of compression, but there are other goals. In this thesis, we present an alternative,



(a)

(b)

(c)

Figure 1.2: Illustration of how signal compression can lead to effective denoising. (a) Noisy HeaviSine signal with SNR level of 10 dB. (b) Histogram of the noisy signal coefficients. (c) Signal reconstruction obtained by removing coefficients at a threshold level T.

namely to minimize the error in reconstructing the underlying signal.

1.2 Applications to High-Resolution Radar

Anyone who has received a speeding ticket before, is familiar with the power of radar technology. The problem of automatic target recognition, however, requires more information about a target than its speed and range. In combat situations, ATR systems must also be able to distinguish targets as either friend or foe, and be able to determine the orientation of a target to know if it is in a position to attack. ATR systems must, therefore, be able to identify targets and their spatial orientation in addition to performing detection and tracking operations.

In this thesis, we present an algorithm for ATR. We specifically believe that simultaneously robust and efficient ATR algorithms must involve pattern matching, in the form of a database search. We therefore propose an "overcomplete" database of "patterns" or signals representative of the targets of interest. The difficult task is to find the appropriate signals and discrimination statistics that allow targets and orientations to be distinguished from one another. In addition, noise added to the received radar returns introduces a factor of uncertainty and must be handled accordingly.

The overcomplete representation of HRR returns is obtained by successive averaging over larger and larger angular sectors of the orientation space. This averaging is an intuitively appealing method for combining the information in several signals into one representative signal. By forming the database from overlapping sectors, the representation of a given target is overcomplete and therefore more robust. Searching the database involves optimizing the match between a received radar return and the signals in the database. The cost function used for this optimization is therefore important for the purpose of discrimination. We present three such statistics that are based on the Maximum Likelihood estimator.

1.3 Contributions and Organization

This section provides a road map for the thesis, along with some of the important results and contributions provided in each chapter.

Chapter 2: Wavelets and Wavelet Packets

This chapter provides an overview of wavelet theory from the point-of-view of multiresolution analysis and filter banks. Orthogonal filter banks are of specific interest because they are directly related to orthonormal wavelet bases. Two methods of solving the dilation equation for the scaling function are presented, along with several choices of filter coefficients which lead to "good" solutions. In addition, a method for iteratively computing the wavelet coefficients is presented, which ties the wavelet theory back to filter banks. The final section of Chapter 2 introduces the notion of wavelet packets and the overcomplete representation that they provide.

Chapter 3: Majorization and Best Basis Search

This chapter provides a definition of *compression*, using the framework offered by majorization theory. The first section introduces some of the main definitions and theorems of the general majorization theory. Several well-known convex and concave functionals are then presented for the purpose of preserving an underlying majorization. The chapter concludes with the discussion of an efficient algorithm to find a "best" signal representation. This algorithm, termed the Best Basis Search, optimizes additive cost functions over all possible combinations of wavelet packet coefficients. The primary contribution of this chapter is to introduce a new framework for understanding compression. Using this, we can interpret the information cost functions as well as present new possibilities.

Chapter 4: Denoising Techniques Using Wavelet Packets

This chapter discusses the techniques involved in denoising with the specific goals of compression and minimal reconstruction error. Two thresholding techniques are introduced and interpreted in the context of the denoising problem. Compression-based denoising is then discussed, along with several results to illustrate its performance. A second denoising strategy which involves minimizing the reconstruction error of the underlying signal is also introduced. Both biased and unbiased estimators are derived, and their performance is verified through simulation.

Chapter 5: Applications to High-Resolution Radar

This chapter applies some of the denoising techniques to the problem of Automatic Target Recognition. The viability of using High-Resolution Radar profiles for the ATR problem is verified by presenting an efficient and robust algorithm. We begin by describing both real and simulated HRR data. We then define the ATR problem with specific emphasis on ground-based targets. The significant contribution of this chapter is the algorithm that we propose. We present the methodology for constructing as well as searching a database of HRR returns. Several results are presented to verify the performance of the algorithm.

Chapter 6: Conclusions

This chapter summarizes the significant results of this thesis. In addition, it discusses several possibilities for improving and extending some of the current denoising techniques. The proposed ATR algorithm also has future possibilities, and some suggestions are offered for extending its capabilities.

Chapter 2

Wavelets and Wavelet Packets

Wavelets and wavelet packets have become popular tools in many scientific disciplines because of their nice properties. Wavelets, like the Fourier transform, represent a signal by weighted basis functions from an orthonormal set but allow more freedom than the Fourier transform in choosing the basis functions (*i.e.* the functions are not restricted to sinusoids). While sinusoids are useful for representing the frequency content of a signal, they are not particularly useful in analyzing transient signals, which require basis functions that are well-localized in time. The wavelet basis functions, though, are typically well-localized in both time and frequency. One of the most redeeming properties of wavelets, and the one that we most exploit here, is that through their vanishing moment properties, wavelets are well-suited to represent signals using only a few basis functions. The idea that a signal is more efficiently represented using a wavelet basis leads, as we shall see, to effective denoising.

The wavelet basis functions are obtained from a single mother wavelet, $\psi(t)$, by dilations and translations, or

$$\psi_{jk}(t) = 2^{j/2}\psi(2^j t - k). \tag{2.1}$$

This results in a doubly indexed basis for all $j \in \mathbb{Z}$ and $k \in \mathbb{Z}$. From this basis, all functions $f(t)$ in $L^2(\mathbb{R})$ can be represented by a weighted sum of the basis functions,

$$f(t) = \sum_j \sum_k w_{jk}\psi_{jk}(t). \tag{2.2}$$

Since we only consider orthonormal basis functions here, the coefficients $\{w_{jk}\}$ can be found by projecting $f(t)$ onto the appropriate basis function to obtain

$$w_{jk} = \int_{-\infty}^{\infty} f(t)\psi_{jk}^*(t)dt, \tag{2.3}$$

where "$*$" represents conjugation. The mother wavelet is obtained from the scaling function $\phi(t)$ by the *wavelet equation*,

$$\psi(t) = 2\sum_k h_1(k)\phi(2t - k), \tag{2.4}$$

while the scaling function is a solution to the *dilation equation*,

$$\phi(t) = 2\sum_k h_0(k)\phi(2t - k). \tag{2.5}$$

In this chapter, we will show how the wavelet and dilation equations arise from multiresolution analysis and filter bank theory. We will see that designing good wavelets is strongly dependent upon these two equations. In the last part of this chapter, we will extend the wavelet decomposition to wavelet packets, which provide additional freedom in representing a signal efficiently, freedom that will be useful in the denoising problem.

2.1 Multiresolution Analysis

The theory of wavelets is a tightly woven web of different ideas, dating back to the early 1900's. It combines results from mathematics, physics, and signal processing, and it is therefore a challenging task to describe the development of the wavelet theory in a succinct, yet intuitive format without inevitably overlooking some important details. To this end, we search for a starting point in this web that will unravel all of the useful results and simultaneously spare the reader from the long history of the wavelet theory. The path that we take was pioneered by Mallat [1] and Meyer [2] and is commonly termed *multiresolution analysis*. Multiresolution analysis is a mathematical framework, described by a set of axioms, from which wavelet and subband decompositions of signals can be understood. In this section, we will describe these axioms and show how they lead to important ideas in wavelet theory.

The essence of the multiresolution theory is that a signal can be represented at different scales, where important features of a signal become more apparent. One can imagine analyzing a signal under a microscope. By increasing the magnification of the microscope, one can see more and more details of the signal. In choosing a wavelet basis, we are essentially looking for a good microscope to examine a particular signal. In the past, the most popular analysis tool has been the Fourier transform, which examines a signal based on its frequency components. The major drawback to Fourier analysis is that a signal is synthesized from exponential functions which are not compactly supported in time. In essence, the Fourier microscope can resolve frequency components infinitely close together if the integration time is infinitely large. In designing a good wavelet, we thus seek a time-frequency *atom* that is limited by some measure in both time and frequency. The Heisenberg Uncertainty Principle provides a lower bound on the area of the time-frequency support, where the region of support in the time and frequency domain is characterized by a variance measure,

$$\begin{aligned}\sigma_t^2 &= \int_{-\infty}^{\infty} t^2 |f(t)|^2 dt \\ \sigma_f^2 &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \omega^2 |F(\omega)|^2 d\omega.\end{aligned}$$

For signals of unit energy, the area of a time-frequency atom is then given by $\sigma_t \sigma_f$, which has a lower bound of 1/2. Extending this idea, we can illustrate the region of support in the time-frequency plane by a two-dimensional plot as shown in Figure 2.1. Figure 2.1(a) shows a typical tiling for the short-time Fourier transform, while Figure 2.1(b) shows a typical tiling for the wavelet transform. The difference between the wavelet transform and the short-time Fourier transform is in the dyadic tiling of the time-frequency plane. This type of tiling is useful because short windows capture high-frequency transient effects, while longer windows capture low-frequency trends in a signal. The time-frequency atoms resulting from the dyadic tiling are ideally suited for the problems of interest, and as shown below, are a direct result of multiresolution analysis.

Multiresolution analysis can be axiomatically defined by the following four conditions.

Definition 1 – **Multiresolution Analysis** *[3, 4] The subspaces V_j must satisfy the following four conditions:*

1) Nested spaces: $V_j \subset V_{j+1}$
Upward Completeness: $\overline{\bigcup V_j} = \mathrm{L}^2(\mathbb{R})$
Downward Completeness: $\bigcap V_j = \{0\}$

2) Scale Invariance: $f(t) \in V_j \Longleftrightarrow f(2t) \in V_{j+1}$

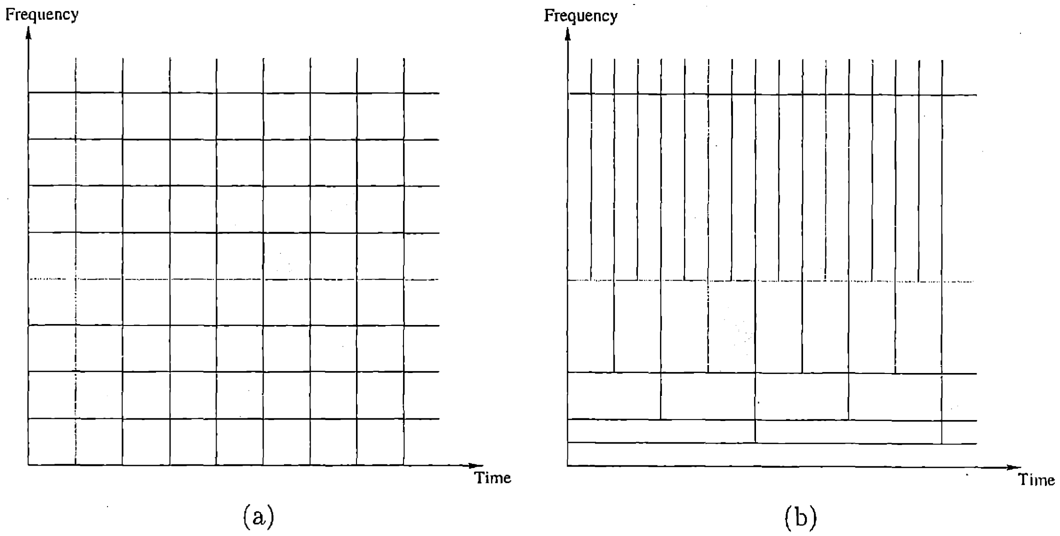


Figure 2.1: Tiling of the time-frequency plane. (a) Atomic decomposition of the short-time Fourier transform. (b) Atomic decomposition of the wavelet transform.

3) Shift Invariance: $f(t) \in V_0 \iff f(t-k) \in V_0$ *for all* $k \in \mathbb{Z}$

4) Existence of a Basis: V_0 *has an orthonormal basis* $\{\phi(t-k) | k \in \mathbb{Z}\}$

The first condition not only stipulates the increasing nature of the subspaces $\{V_j\}$ but also their completeness in $L^2(\mathbb{R})$. This is tantamount to saying that

$$\lim_{j \to -\infty} \|f_j(t)\| = 0 \qquad (2.6)$$

$$\lim_{j \to \infty} \|f(t) - f_j(t)\| = 0. \qquad (2.7)$$

where $f_j(t)$ is the projection of $f(t)$ onto V_j. Note that the increasing nature of the subspaces necessarily implies the nested structure of the subspaces. The second condition introduces the notion of scale, leading to the dyadic tiling of the time-frequency plane, while the third condition requires the subspaces to be closed under functional translation. The final condition requires that a basis exist for the subspace V_0, and using the second and third conditions, this is equivalent to the existence of an orthonormal basis $\{2^{j/2}\phi(2^j t - k)\}$ for V_j.

The above multiresolution framework leads to a new set of subspaces W_j which represent the details obtained in moving from one scale to the next. We specifically define W_j to be the orthogonal complement of V_j in the larger subspace V_{j+1}. As a result, V_{j+1} is the direct subspace sum of V_j and W_j, or

$$V_{j+1} = V_j \bigoplus W_j \quad \text{and} \quad V_j \bigcap W_j = \{0\}. \tag{2.8}$$

Letting $\Delta f_j(t)$ equal the projection of $f(t)$ onto W_j, gives $f_{j+1}(t) = f_j(t) + \Delta f_j(t)$, which indicates that $\Delta f_j(t)$ provides additional details to $f_j(t)$ in order to generate the refined function $f_{j+1}(t)$. By iterating this procedure, two important facts about the subspaces can be derived,

$$\begin{aligned} V_{j+1} &= V_0 \bigoplus W_0 \bigoplus W_1 \bigoplus \cdots \bigoplus W_j && (2.9) \\ \mathrm{L}^2(\mathbb{R}) &= V_0 \bigoplus \left(\bigoplus_{j=0}^{\infty} W_j \right) = \bigoplus_{j=-\infty}^{\infty} W_j. && (2.10) \end{aligned}$$

As a result, the subspaces W_j are also complete in $\mathrm{L}^2(\mathbb{R})$, and by noting that W_l is a subset of V_j, for $l < j$, and that V_j is orthogonal to W_j, one concludes that W_j is orthogonal to W_l for $l \neq j$. Given these facts and the conditions of Definition 1, one can define an orthonormal basis for $\mathrm{L}^2(\mathbb{R})$,

$$\psi_{jk}(t) = 2^{j/2}\psi(2^j t - k), \quad j \in \mathbb{Z} \text{ and } k \in \mathbb{Z}, \tag{2.11}$$

where $\{\psi_{jk}(t) | k \in \mathbb{Z}\}$ is an orthonormal basis for W_j. We refer the interested reader to [4] for a more detailed discussion of this result.

Since the space V_0 is contained in V_1, the function $\phi(t)$ is in V_1, and can be written as a linear combination of the basis functions for V_1. This leads to the dilation equation,

$$\phi(t) \;=\; \sqrt{2}\sum_k c(k)\phi(2t-k) = 2\sum_k h_0(k)\phi(2t-k), \tag{2.12}$$

where the coefficients $\{c(k)\}$ follow the convention $\sum_k c(k) = \sqrt{2}$ and $\{h_0(k)\}$ satisfy $\sum_k h_0(k) = 1$. To determine the coefficients $\{c(k)\}$ in the dilation equation, one can use the orthogonality property of translates to obtain,

$$\begin{aligned}\sqrt{2}\int_{-\infty}^{\infty}\phi(t)\phi(2t-n) \;&=\; 2\int_{-\infty}^{\infty}\left(\sum_k c(k)\phi(2t-k)\right)\phi(2t-n)dt \\ &=\; 2c(n)\int_{-\infty}^{\infty}\phi^2(2t-n)dt = c(n).\end{aligned} \tag{2.13}$$

Since W_0 is contained in V_1, it may be written in terms of the basis functions for V_1. This leads to the wavelet equation,

$$\psi(t) \;=\; \sqrt{2}\sum_k d(k)\phi(2t-k) = 2\sum_k h_1(k)\phi(2t-k). \tag{2.14}$$

In addition, the orthogonality of the basis $\{\phi(t-k)\}$ imposes constraints on the coefficients $\{c(k)\}$,

$$\begin{aligned}\int_{-\infty}^{\infty}\phi(t)\phi(t-m)dt \;&=\; 2\int_{-\infty}^{\infty}\left(\sum_k c(k)\phi(2t-k)\right)\left(\sum_l c(l)\phi(2t-2m-l)\right)dt \\ &=\; 2\int_{-\infty}^{\infty}\sum_k\sum_l c(k)c(l)\phi(2t-k)\phi(2t-2m-l)dt.\end{aligned}$$

Due to orthogonality, the only non-zero terms occur when $k = 2m + l$, which gives

$$\begin{aligned}\int_{-\infty}^{\infty} \phi(t)\phi(t-m)dt &= \delta(m) = 2\int_{-\infty}^{\infty} \sum_k c(k)c(k-2m)\phi^2(2t-k)dt \\ &= \sum_k c(k)c(k-2m).\end{aligned}$$

This shows that the coefficients $\{c(k)\}$ have unit energy and double-shift orthogonality, or

$$\sum_k c(k)c(k-2m) = \delta(m) \quad \text{and} \quad \sum_k |c(k)|^2 = 1. \tag{2.15}$$

The functions $\phi(t)$ and $\psi(t-m)$ must also be orthogonal for any value of m, and $\psi(t)$ must be orthogonal to $\psi(t-m)$ for all $m \neq 0$. This leads to the following two constraints on the coefficients,

$$\sum_k c(k)d(k-2m) = 0 \tag{2.16}$$

$$\sum_k d(k)d(k-2m) = \delta(m). \tag{2.17}$$

Similar conditions will appear again in the next section as requirements for orthogonal filter banks.

2.2 Orthogonal Filter Banks

The popularity of the wavelet theory in applied fields is perhaps due, for the most part, to its efficient implementation via filter banks. Filter banks have gained prominence in signal processing for their frequency discrimination, important for subband coding. The typical structure of a two-channel filter bank, introduced in the 1980's [5], is shown in Figure 2.2. The filters $\mathbf{H}_0$ and $\mathbf{F}_0$ are typically lowpass filters, while $\mathbf{H}_1$ and $\mathbf{F}_1$ are highpass filters. The downsampling operation $(\downarrow \mathbf{2})$ removes the odd components of a signal,

$$v_i(n) = y_i(2n), \tag{2.18}$$

and as a result, both signals $\{v_0(n)\}$ and $\{v_1(n)\}$ are half the length of the original signal. Downsampling can equivalently be represented by the following matrix operation,

$$\mathbf{v_i} = (\downarrow \mathbf{2})\mathbf{y_i} \tag{2.19}$$

where $(\downarrow \mathbf{2})$ is the identity matrix with odd rows removed and given by,

$$(\downarrow \mathbf{2}) = \begin{bmatrix} & \ddots & \ddots & & & & & & \\ \cdots & 0 & 1 & 0 & 0 & \cdots & & & \\ & & \cdots & 0 & 1 & 0 & 0 & \cdots & \\ & & & & \cdots & 0 & 1 & 0 & \cdots \\ & & & & & & \ddots & \ddots & \end{bmatrix}. \tag{2.20}$$

For ease of analysis, it is often useful to carry out the operations in the z-domain. It is simple to show [3] that downsampling in time "expands" the z-transform and adds an aliasing term, or

$$v_i(n) = y_i(2n) \quad \longleftrightarrow \quad V_i(z) = \frac{1}{2}\left[Y_i(z^{1/2}) + Y_i(-z^{1/2})\right]. \tag{2.21}$$

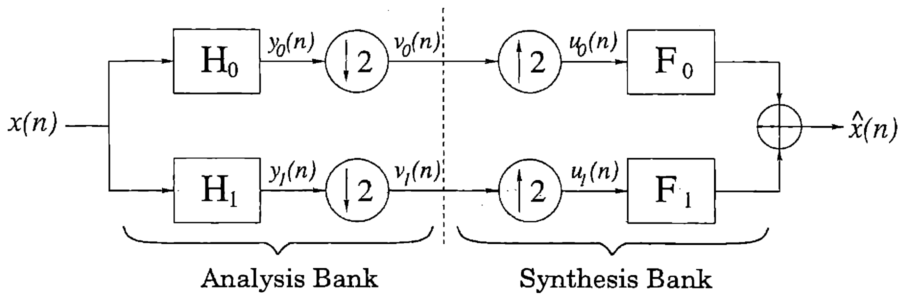


Figure 2.2: Simple two-channel filter bank.

The combined filtering and downsampling operations can also be represented in terms of matrices. First writing the equations for convolution and downsampling gives

$$\begin{aligned} y_i(n) &= \sum_k h_i(k)x(n-k) && (2.22) \\ v_i(n) &= y_i(2n) = \sum_k h_i(k)x(2n-k), && (2.23) \end{aligned}$$

which shows that $\{v_i(n)\}$ can be obtained from $\{x(n)\}$ directly by the matrix operation,

$$\mathbf{v}_\mathrm{i} = (\downarrow 2)\mathbf{H}_i\mathbf{x} \qquad (2.24)$$

where

$$(\downarrow 2)\mathbf{H}_i = \begin{bmatrix} & \ddots & \ddots & & & & & & \\ \cdots & h_i(3) & h_i(2) & h_i(1) & h_i(0) & & & & \\ & & \cdots & h_i(3) & h_i(2) & h_i(1) & h_i(0) & & \\ & & & & \cdots & h_i(3) & h_i(2) & h_i(1) & h_i(0) \\ & & & & & & \ddots & \ddots & \end{bmatrix}. \qquad (2.25)$$

In the synthesis bank, the signals are upsampled and then filtered. The upsampling operation $(\uparrow \mathbf{2})$ adds zeros between successive signal samples,

$$u_i(n) = \begin{cases} v_i(n/2) & n \text{ even} \\ 0 & n \text{ odd} \end{cases}, \tag{2.26}$$

and the equivalent matrix operation is given by

$$\mathbf{u_i} = (\uparrow \mathbf{2})\mathbf{v_i} \tag{2.27}$$

where

$$(\uparrow \mathbf{2}) = \left[\begin{array}{cccccccc} & \ddots & \ddots & & & & & \\ \cdots & 0 & 1 & 0 & 0 & \cdots & & \\ \cdots & 0 & 0 & 0 & 0 & \cdots & & \\ & \cdots & 0 & 1 & 0 & 0 & \cdots & \\ & \cdots & 0 & 0 & 0 & 0 & \cdots & \\ & & \cdots & 0 & 1 & 0 & 0\cdots & \\ & & & & & & \ddots & \ddots \end{array} \right]. \tag{2.28}$$

The matrix $(\uparrow \mathbf{2})$ is the identity matrix with rows of zeros added in between. Comparing Equations (2.20) and (2.28) shows that upsampling and downsampling are transpose operations, (*i.e.* $(\uparrow \mathbf{2})^T = (\downarrow \mathbf{2})$). In the z-domain, upsampling "shrinks" the z-transform, or

$$u_i(n) = \begin{cases} v_i(n/2) & n \text{ even} \\ 0 & n \text{ odd} \end{cases} \longleftrightarrow U_i(z) = V_i(z^2). \tag{2.29}$$

As a final observation, downsampling followed by upsampling replaces the odd components of a signal with zeros. The equivalent z-domain operation is given by,

$$u_i(n) = \begin{cases} y_i(n) & n \text{ even} \\ 0 & n \text{ odd} \end{cases} \longleftrightarrow U_i(z) = \frac{1}{2}\left[Y_i(z) + Y_i(-z)\right]. \tag{2.30}$$

To design a perfect reconstruction filter bank, the synthesis bank must perform the inverse operation of the analysis bank. We will, however, require all the filters to be causal, to make the recovered signal a delayed version of the input signal. Following the signal path through the lowpass and highpass branches of the filter bank shown in Figure 2.2 and using Equation (2.30), yields the following,

$$\text{Lowpass Branch}: \quad \frac{1}{2}F_0(z)\left[H_0(z)X(z) + H_0(-z)X(-z)\right] \tag{2.31}$$

$$\text{Highpass Branch}: \quad \frac{1}{2}F_1(z)\left[H_1(z)X(z) + H_1(-z)X(-z)\right]. \tag{2.32}$$

Adding the two outputs and combining terms leads to an expression for the reconstructed signal,

$$\begin{aligned} \hat{X}(z) &= \underbrace{\frac{1}{2}\left[F_0(z)H_0(z) + F_1(z)H_1(z)\right]}_{\text{Distortion term}} X(z) + \\ &\quad \underbrace{\frac{1}{2}\left[F_0(z)H_0(-z) + F_1(z)H_1(-z)\right] X(-z)}_{\text{Aliasing term}}. \end{aligned} \tag{2.33}$$

In designing a perfect reconstruction filter bank, the aliasing term is ideally zero and the distortion term is unity (or a delay to make the filters causal). The design equations for perfect reconstruction filter banks are then given by,

$$F_0(z)H_0(z) + F_1(z)H_1(z) = 2z^{-l} \tag{2.34}$$

$$F_0(z)H_0(-z) + F_1(z)H_1(-z) = 0. \tag{2.35}$$

To cancel the aliasing in Equation 2.33 the simple choice of $F_0(z) = H_1(-z)$ and $F_1(z) = -H_0(-z)$ is made, and substituting this choice into the distortion equation, gives

$$F_0(z)H_0(z) - F_0(-z)H_0(-z) \;=\; 2z^{-l}, \tag{2.36}$$

Letting $P(z) = z^l F_0(z)H_0(z)^1$, provides the final design equation,

$$P(z) + P(-z) \;=\; 2, \tag{2.37}$$

which requires $P(z)$ to be a halfband filter. Filters of this type are only composed of odd powers of z and a constant term equal to 1. A perfect reconstruction filter bank thus requires a halfband $P(z)$ which subsequently leads to the other filters via spectral factorization of $P(z)$.

The above discussion presents a general framework for designing perfect reconstruction filter banks. We now focus our discussion on designing orthogonal filter banks. An orthogonal two-channel filter bank is shown in Figure 2.3, where the normalized coefficients $\{c(n)\}$ and $\{d(n)\}$ have been used for the lowpass and highpass FIR filters respectively. In addition, the synthesis branch has filters $\{c(-n)\}$ and $\{d(-n)\}$ represented by the matrices $\mathbf{C}^T$ and $\mathbf{D}^T$ respectively. Non-causal filters are used for the moment to make the analysis simpler. We will show that $\mathbf{C}^T$ and $\mathbf{D}^T$ are the required matrix operations to construct an orthogonal filter bank, and we will give the conditions on $\{c(n)\}$ and $\{d(n)\}$ that lead to perfect reconstruction.

If the entire filter bank operation is represented by a matrix $\mathbf{M}$, then $\mathbf{M}$ must equal the identity matrix for perfect reconstruction. For an orthogonal filter bank,

[1] l must be odd because the left side of Equation (2.36) is an odd function.

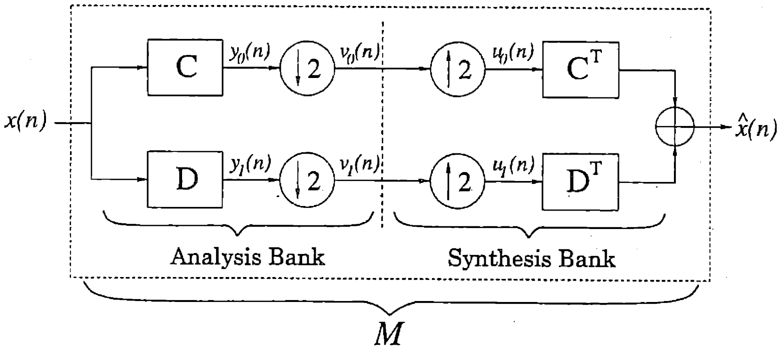


Figure 2.3: Simple two-channel orthogonal filter bank.

$\mathbf{M} = \mathbf{H}\mathbf{H}^T = \mathbf{I}$ is required, where $\mathbf{H}$ is the entire analysis bank and $\mathbf{H}^T$ is the entire synthesis bank. From the previous discussion about matrix operations, $\mathbf{H}$ can be written as

$$\mathbf{H} = \begin{bmatrix} (\downarrow 2)\mathbf{C} \\ \cdots\cdots \\ (\downarrow 2)\mathbf{D} \end{bmatrix} = \begin{bmatrix} & \ddots & \ddots & & & & \\ \cdots & c(3) & c(2) & c(1) & c(0) & & \\ & & \cdots & c(3) & c(2) & c(1) & c(0) \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & d(3) & d(2) & d(1) & d(0) & & \\ & & \cdots & d(3) & d(2) & d(1) & d(0) \\ & & & & & \ddots & \ddots \end{bmatrix}. \quad (2.38)$$

Taking the transpose of $\mathbf{H}$, gives

$$\mathbf{H}^{\mathbf{T}} = \begin{bmatrix} (\downarrow 2)\mathbf{C} \\ \cdots\cdots \\ (\downarrow 2)\mathbf{D} \end{bmatrix}^T = \begin{bmatrix} \mathbf{C}^{\mathbf{T}}(\uparrow 2) & \vdots & \mathbf{D}^{\mathbf{T}}(\uparrow 2) \end{bmatrix}, \quad (2.39)$$

where we have used the fact that upsampling and downsampling are transpose operations. This shows that $\mathbf{C}^T$ and $\mathbf{D}^T$ are the filters that should be used in the synthesis bank as shown in Figure 2.3.

We now determine the conditions on the filters $\{c(n)\}$ and $\{d(n)\}$. Since $\mathbf{M} = \mathbf{H}\mathbf{H}^T = \mathbf{I}$, it must also be true that $\mathbf{H}^T\mathbf{H} = \mathbf{I}$ because $\mathbf{H}$ is an orthogonal matrix. Writing these matrix equations in terms of the filters, results in the following three conditions which must be satisfied[2],

$$\begin{aligned} \sum_k c(k)c(k-2m) &= \delta(m). && (2.40) \\ \sum_k c(k)d(k-2m) &= 0 && (2.41) \\ \sum_k d(k)d(k-2m) &= \delta(m). && (2.42) \end{aligned}$$

The same equations were derived in the previous section while defining multiresolution analysis, and as a result, designing an orthogonal filter bank is strongly connected to finding an orthonormal basis $\{\phi(t-k)\}$. There are many coefficients that satisfy Equations (2.40)–(2.42), but the number of design possibilities can be limited by making a simple causal choice for the filter $\{d(n)\}$,

$$d(n) \;=\; (-1)^n c(N-n), \quad 0 \le n \le N, \tag{2.43}$$

where $N+1$ is the length of the filter. If the coefficients $\{c(n)\}$ satisfy Equation (2.40), then this choice of $\{d(n)\}$ will automatically satisfy Equations (2.41) and (2.42). As a consequence of this choice, selecting good values for $\{c(n)\}$ is the primary concern in designing orthogonal filter banks. Once these coefficients have been selected, it is then possible to solve the dilation equation, in order to determine the scaling function.

[2] See [3] for more details.

2.3 The Scaling Function

Given a set of coefficients $\{h_0(n)\}$ (or equivalently $\{c(n)\}$) that correspond to an orthogonal filter bank, the scaling function can be found by solving the dilation equation,

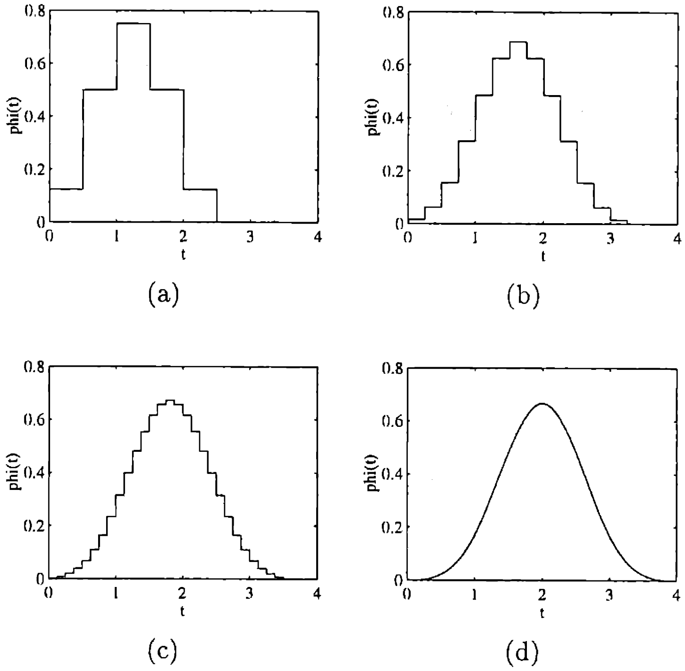
$$\phi(t) = 2\sum_k h_0(k)\phi(2t-k). \tag{2.44}$$

Out of the numerous methods available for solving Equation (2.44) we discuss two. The first method, called the *cascade algorithm*, is an iterative approach to solving the dilation equation. The algorithm is initialized using a simple function such as the box function on the interval $[0,1]$. This initial estimate of $\phi(t)$ is then updated via the dilation equation to find a new estimate. Specifically, let

$$\phi^{(0)}(t) = \begin{cases} 1 & 0 \le t \le 1 \\ 0 & \text{otherwise} \end{cases} \tag{2.45}$$

$$\phi^{(i+1)}(t) = 2\sum_k h_0(k)\phi^{(i)}(2t-k), \tag{2.46}$$

and if the cascade algorithm converges, the solution to the dilation equation is given by $\phi(t) = \lim_{i\to\infty}\phi^{(i)}(t)$. For an arbitrary set of coefficients $\{h_0(n)\}$, however, the cascade algorithm is not guaranteed to converge. The necessary and sufficient conditions for existence and convergence are given in [3]. To illustrate the convergence of the cascade algorithm, we consider an example from the family of splines, using the coefficients $\{h_0(0), \cdots, h_0(4)\} = \{\frac{1}{16}(1,4,6,4,1)\}$. The results for different iteration numbers are shown in Figure 2.4, where the box function on $[0,1]$ is used as an initial guess for $\phi^{(0)}(t)$. For this example, the cascade algorithm rapidly converges to the true solution of the dilation equation. It can be shown that the coefficients used in this example generate a cubic spline scaling function. Examining Figure 2.4, note that the cubic



(a) (b) (c) (d)

Figure 2.4: Different iterations of the cascade algorithm for the filter coefficients $\frac{1}{16}(1,4,6,4,1)$. (a) $\phi^{(1)}(t)$ (b) $\phi^{(2)}(t)$ (c) $\phi^{(3)}(t)$ (d) $\phi^{(10)}(t)$

spline is supported on $[0,4]$ for the set of coefficients $\{h_0(0),\cdots,h_0(4)\}$. It can be shown, in general, that if the cascade algorithm converges, the scaling function will be supported on $[0,N]$ for a set of coefficients $\{h_0(0),\cdots,h_0(N)\}$.

An alternative approach to solving the dilation equation is based in the Fourier domain. Taking the Fourier transform of both sides of Equation (2.44) results in

$$\hat{\phi}(\omega) \;=\; H_0\left(\frac{\omega}{2}\right)\hat{\phi}\left(\frac{\omega}{2}\right), \tag{2.47}$$

where $\hat{\phi}(\omega)$ is the Fourier transform of $\phi(t)$ and $H_0(\omega)$ is the discrete-time Fourier transform of $\{h_0(n)\}$. We can iteratively determine $\hat{\phi}\left(\frac{\omega}{2^j}\right)$ in terms of $\hat{\phi}\left(\frac{\omega}{2^{j+1}}\right)$ and

$H_0\left(\frac{\omega}{2^{j+1}}\right)$ for all values of j, to obtain in the limit

$$\hat{\phi}(\omega) = \prod_{j=1}^{\infty} H_0\left(\frac{\omega}{2^j}\right). \tag{2.48}$$

Note that $\lim_{j\to\infty} \hat{\phi}\left(\frac{\omega}{2^j}\right) = \hat{\phi}(0) = 1$, since the normalization $\int_{-\infty}^{\infty} \phi(t) = 1$ is used. For Equation (2.48) to converge, the conditions $H_0(\omega)|_{\omega=0} = 1$ and $H_0(\omega)|_{\omega=\pi} = 0$ are clearly required, and other conditions are discussed in [3] and [4]. As a simple example, consider the coefficients $\{h_0(n)\} = \{(\frac{1}{2}, \frac{1}{2})\}$, with a corresponding Fourier transform $H_0(\omega) = \frac{1}{2}(1+e^{-i\omega})$. Using the infinite product formula in Equation (2.48) gives

$$\hat{\phi}(\omega) = \lim_{n\to\infty} \prod_{j=1}^{n} H_0\left(\frac{\omega}{2^j}\right) = \lim_{n\to\infty} \frac{1}{2^n}\left(\frac{1-e^{-i\omega}}{1-e^{-i\frac{\omega}{2^n}}}\right) = \frac{1-e^{-i\omega}}{i\omega}, \tag{2.49}$$

which corresponds to the box function on $[0,1]$, or

$$\phi_{\mathbf{Haar}}(t) = \begin{cases} 1 & 0 \le t \le 1 \\ 0 & \text{otherwise} \end{cases}. \tag{2.50}$$

The box function is therefore the solution to the dilation equation with filter coefficients $\{h_0(n)\} = \{(\frac{1}{2}, \frac{1}{2})\}$, and we denote this solution by $\phi_{\mathbf{Haar}}(t)$ because it is the scaling function for the famous Haar wavelet.

The above results can, in fact, be extended to a more general class of scaling functions, which result from a set of filter coefficients

$$\{h_0^{(l)}(n)\} = \left\{ \underbrace{\left(\frac{1}{2}, \frac{1}{2}\right) * \left(\frac{1}{2}, \frac{1}{2}\right) * \cdots * \left(\frac{1}{2}, \frac{1}{2}\right)}_{l \text{ times}} \right\}, \tag{2.51}$$

obtained by l convolutions of the coefficients $\{(\frac{1}{2},\frac{1}{2})\}$. For example, the first two sets of coefficients are $\{h_0^{(0)}(n)\} = \{(\frac{1}{2},\frac{1}{2})\}$ and $\{h_0^{(1)}(n)\} = \{(\frac{1}{4},\frac{1}{2},\frac{1}{4})\}$. Using Equation (2.48) and the results obtained in Equation (2.49), the frequency and time domain solutions become respectively,

$$\hat{\phi}_l(\omega) = \left(\frac{1-e^{-i\omega}}{i\omega}\right)^l \tag{2.52}$$

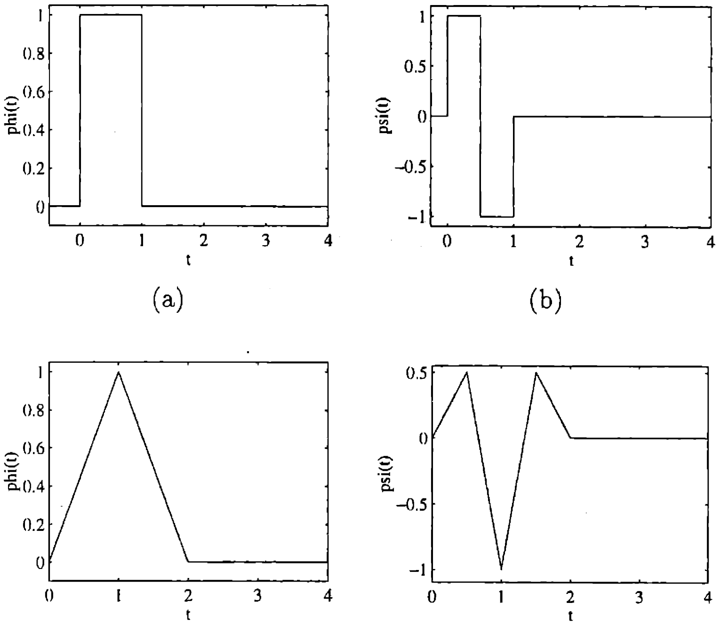
$$\phi_l(t) = \underbrace{\phi_{\text{Haar}}(t) * \phi_{\text{Haar}}(t) * \cdots * \phi_{\text{Haar}}(t)}_{l \text{ times}}. \tag{2.53}$$

These solutions correspond to the family of splines obtained from repeated convolutions of box functions. For example, the coefficients $\{h_0^{(0)}(n)\}$ lead to the box, and $\{h_0^{(1)}(n)\}$ result in a linear spline, sometimes referred to as the hat function. The example previously shown in Figure 2.4 is a cubic spline scaling function, obtained from three convolutions of the box function, with corresponding coefficients $\{h_0^{(3)}(0), \cdots, h_0^{(3)}(4)\} = \{\frac{1}{16}(1,4,6,4,1)\}$ obtained from three convolutions of the coefficients $\{(\frac{1}{2},\frac{1}{2})\}$. While this section has focused on the family of splines and the methodologies to construct them, the next section will present other choices of filter coefficients and the corresponding wavelets.

2.4 Wavelets

Once the scaling function has been determined, the wavelet immediately follows from the wavelet equation,

$$\psi(t) = 2\sum_k h_1(k)\phi(2t-k), \tag{2.54}$$



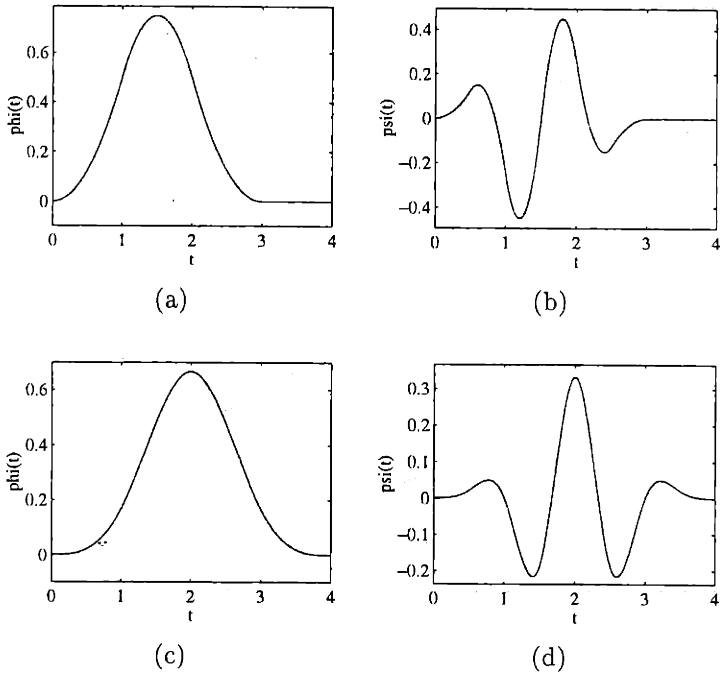
(a) (b)

(c) (d)

Figure 2.5: (a) Haar scaling function. (b) Haar wavelet. (c) Linear spline. (d) Wavelet for the linear spline.

where the coefficients are chosen to be $\{h_1(n)\} = \{(-1)^n h_0(N-n)\}$ in order to ensure the orthogonality constraint. Figures 2.5 and 2.6 show some of the scaling functions and wavelets from the family of splines discussed in the previous section. From the multiresolution analysis, we know that $\phi(t)$ and $\psi(t)$ must be orthogonal for all integer shifts, and $\psi(t)$ must also be orthogonal to all dilations.

Another important family of wavelets was pioneered by Daubechies [6]. Her design procedure began from the filter coefficients with the goal of placing as many zeros at $\omega = \pi$ as possible, while still satisfying the constraint that $P(z)$ from Equation (2.37) must be halfband. This design procedure leads to many possible choices for filters and hence wavelets. One popular choice is the four-tap Daubechies' filter, given by the coefficients $\{h_0(n)\} = \{\frac{1}{8}(1+\sqrt{3}, 3+\sqrt{3}, 3-\sqrt{3}, 1-\sqrt{3})\}$. This



(a) (b)

(c) (d)

Figure 2.6: (a) Quadratic spline. (b) Wavelet for the quadratic spline. (c) Cubic spline. (d) Wavelet for the cubic spline.

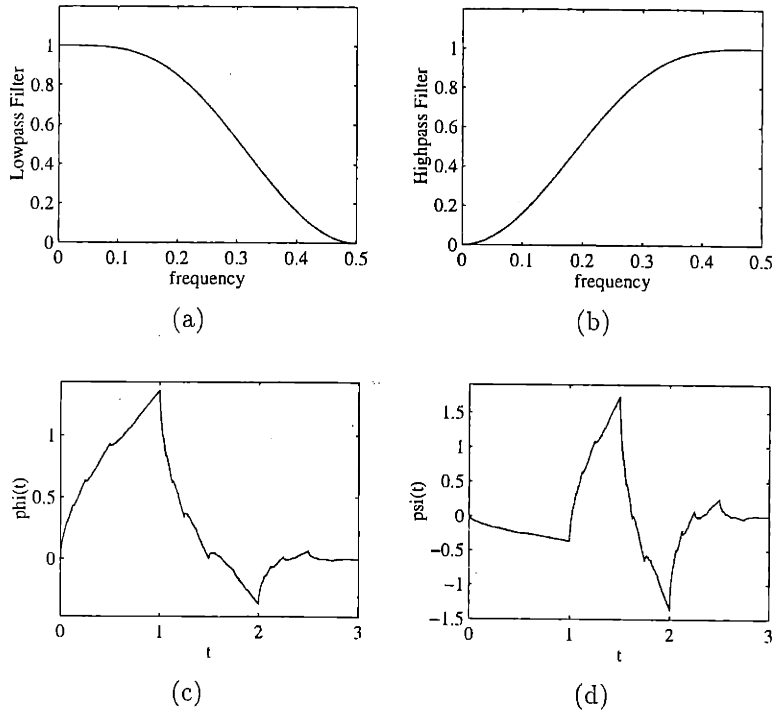


Figure 2.7: Results for Daubechies' four-tap filter. (a) Lowpass filter (b) Highpass filter (c) Scaling function (d) Wavelet

choice of coefficients places two zeros at $\omega = \pi$ and leads to a wavelet with two vanishing moments, that is,

$$\int_{-\infty}^{\infty} \psi(t)dt = \int_{-\infty}^{\infty} t\psi(t)dt = 0. \tag{2.55}$$

The Fourier transforms of the lowpass and highpass coefficients of the four-tap filter are shown in Figures 2.7(a) and 2.7(b) respectively. Note that this filter is maximally flat at the endpoints but fails to have the sharp transition region typically emphasized in filter design. The corresponding scaling function and wavelet are given in Figures 2.7(c) and 2.7(d), which show that this choice of filter coefficients leads to wavelets with a more complicated structure than the spline family.

Having constructed a wavelet basis, our goal is to use this new representation to describe arbitrary functions. We are essentially looking for a basis that will allow us to represent a function with a minimal number of significant coefficients. Prior to this, though, we require an efficient implementation of the transform. The solution to this problem was given in 1989 by Mallat [1] who proposed an algorithm to iteratively determine the wavelet coefficients, given the projection of a function $f(t)$ onto an initial subspace. For simplicity, assume that $f(t) \in V_1$ and that the projection of $f(t)$ onto the basis functions of V_1 yields the coefficients

$$f^{(1)}(n) \quad = \quad \langle f(t), \sqrt{2}\phi(2t-n)\rangle, \quad n \in \mathbb{Z}. \tag{2.56}$$

The function $f(t)$ can therefore be synthesized from these coefficients,

$$f(t) \quad = \quad \sqrt{2}\sum_{l=-\infty}^{\infty} f^{(1)}(l)\phi(2t-l). \tag{2.57}$$

From the multiresolution analysis, we know that V_1 can be separated into two orthogonal subspaces V_0 and W_0. We now wish to determine the projection of $f(t)$ onto these subspaces, using the coefficients $\{f^{(1)}(n)\}$. Projecting $f(t)$ onto the basis functions for V_0 and using Equation (2.57) along with the dilation equation gives the following,

$$\begin{aligned}
f^{(0)}(n) &= \langle f(t), \phi(t-n)\rangle = \langle f(t), \sqrt{2}\sum_k c(k)\phi(2t-2n-k)\rangle \\
&= \sqrt{2}\sum_k c(k)\langle f(t), \phi(2t-2n-k)\rangle \\
&= 2\sum_k\sum_l c(k)f^{(1)}(l)\langle \phi(2t-l), \phi(2t-2n-k)\rangle.
\end{aligned}$$

Since $\phi(t)$ is orthogonal to its integer shifts, the inner products are non-zero only when $l = 2n + k$, and the equation above simplifies to

$$f^{(0)}(n) = \sum_{l} c(l - 2n) f^{(1)}(l). \tag{2.58}$$

Substituting $\{\tilde{c}(n)\} = \{c(-n)\}$, provides the final result

$$f^{(0)}(n) = \sum_{l} \tilde{c}(2n - l) f^{(1)}(l). \tag{2.59}$$

Comparing this result to Equation (2.23), we see that $\{f^{(0)}(n)\}$ can be obtained from $\{f^{(1)}(n)\}$ by filtering with $\{\tilde{c}(n)\}$ followed by downsampling. As a result, Mallat's algorithm essentially ties the wavelet theory back to filter banks. We can similarly determine the coefficients $\{g^{(0)}(n)\}$ by projecting $f(t)$ onto the subspace W_0,

$$g^{(0)}(n) = \sum_{l} \tilde{d}(2n - l) f^{(1)}(l), \tag{2.60}$$

which corresponds to filtering $\{f^{(1)}(n)\}$ with the coefficients $\{\tilde{d}(n)\} = \{d(-n)\}$ and then downsampling.

More generally, assume that $f(t) \in V_j$ with corresponding coefficients $\{f^{(j)}(n)\}$. The goal is to determine the coefficients in V_{j-1} and W_{j-1}. The above arguments suggest the more general result

$$f^{(j-1)}(n) = \sum_{l} \tilde{c}(2n - l) f^{(j)}(l) \tag{2.61}$$

$$g^{(j-1)}(n) = \sum_{l} \tilde{d}(2n - l) f^{(j)}(l). \tag{2.62}$$

This procedure can be iterated to obtain coarser and coarser approximations of $f(t)$. This results in a filter bank that iterates the lowpass branches, as shown in Figure 2.8. The wavelet coefficients are obtained from the output of the highpass filters, and the

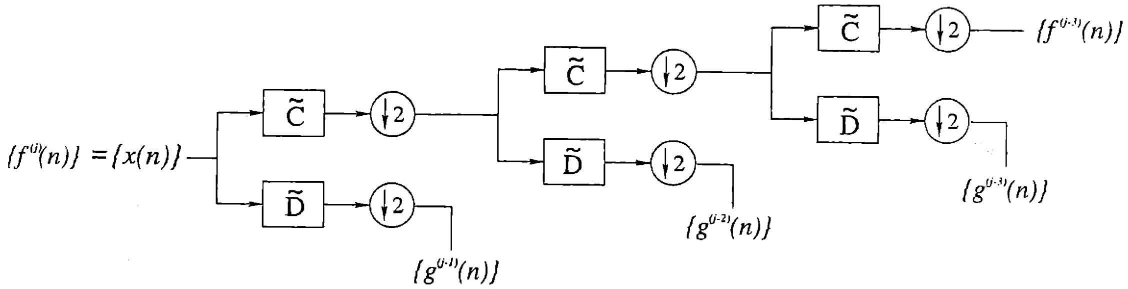


Figure 2.8: Analysis bank used to iteratively compute the wavelet coefficients.

output of the final lowpass filter contains the remaining information necessary to perfectly represent $f(t)$ in this new basis.

In subsequent analyses, we are specifically interested in the discrete-time wavelet transform, used to find the wavelet basis for a discrete-time signal $\{x(n)\}$. The discrete-time decomposition lends itself directly to the preceding analysis. We simply assume that there is some function $f(t) \in V_J$, such that when projected onto the basis functions of V_J, we obtain the discrete-time signal $\{x(n)\}$. A signal of length $N = 2^J$ can therefore be decomposed into J sets of coefficients via the filter bank operation.

From the wavelet decomposition of a signal, we would also like to iteratively determine the projection coefficients onto the larger subspaces in order to synthesize a signal. The projection of $f(t)$ onto V_j can be obtained from the coefficients $\{f^{(j-1)}(n)\}$ and $\{g^{(j-1)}(n)\}$, corresponding to the subspaces V_{j-1} and W_{j-1} respectively, by the following equation,

$$f^{(j)}(n) \quad = \quad \sum_l c(2l-n)f^{(j-1)}(l) + \sum_l d(2l-n)g^{(j-1)}(l). \qquad (2.63)$$

This corresponds to upsampling the coefficients $\{f^{(j-1)}(n)\}$ and $\{g^{(j-1)}(n)\}$ followed

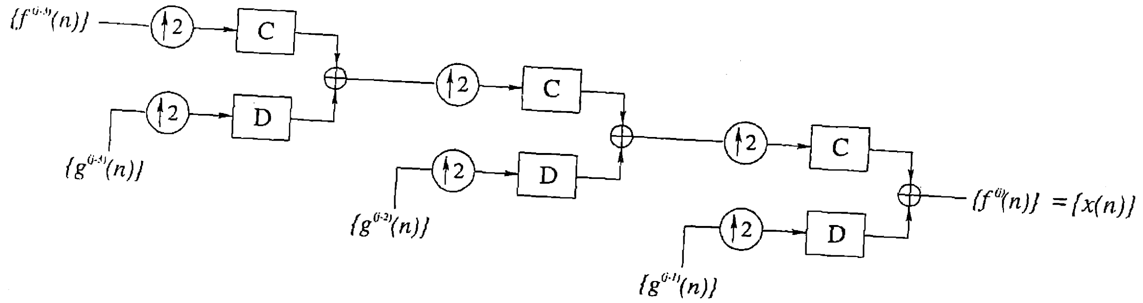


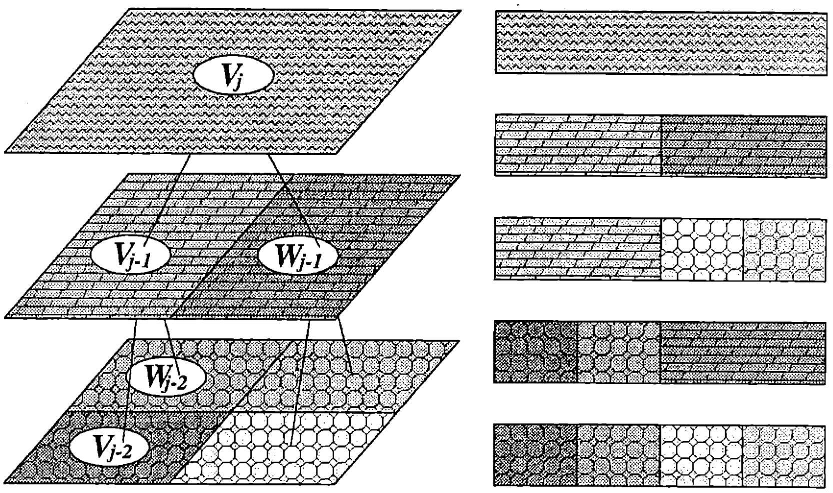
Figure 2.9: Synthesis bank used to reconstruct a signal from its wavelet coefficients.

by filtering with $\{c(n)\}$ and $\{d(n)\}$ respectively. The results are then added. These operations are equivalent to the synthesis branch of a filter bank, as shown in Figure 2.9.

Mallat's algorithm therefore allows a signal to be decomposed into a set of wavelet coefficients represented by a new set of basis functions. This method is, however, limited to only one such basis for a particular signal, and depending upon the signal, this representation may not be well-adapted. We thus seek an alternative to the simple wavelet basis that will offer more freedom in finding a "good" signal representation. One possible improvement may be obtained by searching for a basis in an *overcomplete* representation of a signal provided by wavelet packets.

2.5 Wavelet Packets

To obtain the wavelet decomposition of a signal, we split the subspaces V_j into two orthogonal subspaces V_{j-1} and W_{j-1}, and iterate this procedure by splitting V_{j-1} into two orthogonal subspaces. This, in a sense, is restrictive since W_j may be further decomposed and provide additional freedom in the representation. Wavelet packets



(a) (b)

Figure 2.10: (a) Binary tree of subspaces resulting from a wavelet packet decomposition. (b) Graphical illustration of five possible orthogonal bases for V_j.

lift this limitation, by further decomposing any subspace W_j into two orthogonal subspaces, which can, in turn, be further decomposed. This subspace splitting results in a hierarchy of parent and children nodes structured in a binary tree of subspaces as illustrated in Figure 2.10(a). To form a basis for the initial subspace V_j, we must choose a sufficient number of basis functions from the children subspaces to span V_j. This procedure is illustrated graphically in Figure 2.10(b), where a basis is obtained if the children subspaces "cover" V_j.

The idea of splitting both sets of subspaces leads to an extension of the multiresolution framework discussed earlier, and at the same time, provides an adaptive representation of a given signal. To show that this is in fact possible, we use the *splitting trick*, given in Lemma 1 and proven in [7].

Lemma 1 - The Splitting Trick *[7]*

Suppose that the functions $\{\varphi(t-k)|k \in \mathbb{Z}\}$ form an orthonormal basis for a subspace $\mathcal{S}$. Let φ_0, φ_1 be,

$$\hat{\varphi}_e(\omega) = H_e\left(\frac{\omega}{2}\right)\hat{\varphi}_e\left(\frac{\omega}{2}\right); \quad e = 0, 1.$$

Then, $\{\varphi_{0;k}(t) = \frac{1}{\sqrt{2}}\varphi_0\left(\frac{t}{2}-k\right)|k \in \mathbb{Z}\}$ and $\{\varphi_{1;k}(t) = \frac{1}{\sqrt{2}}\varphi_1\left(\frac{t}{2}-k\right)|k \in \mathbb{Z}\}$ constitute an orthonormal basis for $\mathcal{S}$.

Starting from an initial subspace V_j, we can recursively split subspaces at each level. Carrying this procedure through L levels of resolution and using the notation defined in [8], we obtain the basis functions,

$$\psi^L_{e_1,\cdots,e_L;j,k}(t) = 2^{(j-L)/2}\psi^L_{e_1,\cdots,e_L}(2^{j-L}t-k), \tag{2.64}$$

where

$$\hat{\psi}^L_{e_1,\cdots,e_L}(\omega) = \prod_{i=1}^{L} H_{e_i}\left(\frac{\omega}{2^i}\right)\hat{\phi}\left(\frac{\omega}{2^L}\right). \tag{2.65}$$

The scaling and wavelet functions are directly related to this new set of basis functions by $\phi = \psi^L_{0,\cdots,0}$ and $\psi = \psi^L_{1,0,\cdots,0}$. From an initial subspace V_j, we obtain 2^L sets of basis functions after L splittings, where the sequences $\{e_L, \cdots, e_1\}$ represent the order of filtering operations performed in the successive splittings. These sequences represent a binary indexing scheme where $(m)_2 = e_L e_{L-1}\cdots e_1$ is the binary representation of m. As a result, m indexes the set of basis functions obtained on the L^{th} splitting and takes values $0 \leq m < 2^L$. One possible basis for V_j is therefore,

$$\{\psi^L_{m;j,k}(t) = 2^{(j-L)/2}\psi^L_m(2^{j-L}t-k)|k \in \mathbb{Z}, 0 \leq m < 2^L\}. \tag{2.66}$$

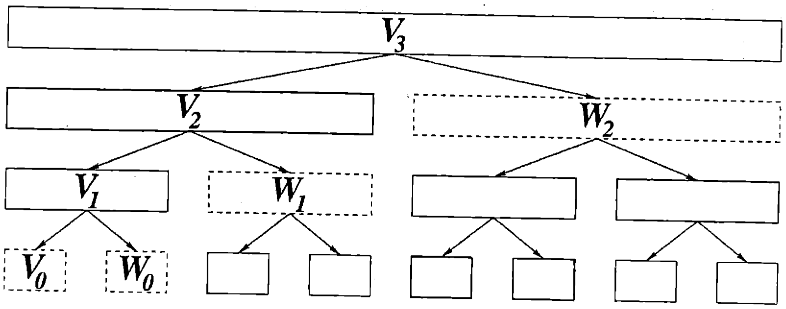


Figure 2.11: Illustration of the splitting process for $L = 3$ and $j = 3$.

Figure 2.11 illustrates the splitting process for $L = 3$ and $j = 3$.

The basis given in Equation (2.66) is obtained by subdividing each subspace at every stage, and results in a basis for the bottom row of subspaces in the tree shown in Figure 2.11. Splitting every subspace, however, is not necessary to find a basis for V_j, since each subspace does not have to be divided in order to represent all functions in V_j. Figure 2.11 shows two other possible bases. The dashed rectangles correspond to the wavelet decomposition, $V_3 = V_0 \bigoplus W_0 \bigoplus W_1 \bigoplus W_2$, and the bold rectangles correspond to a different subspace splitting with basis functions,

$$\{\psi_0^1(4t-k), \psi_{1,1}^2(2t-k), \psi_{0,0,1}^3(t-k), \psi_{1,0,1}^3(t-k) | k \in \mathbb{Z}\}.$$

From Figure 2.11, it is visually evident that a basis is formed if the proper "covering" subspaces of the original space V_3 are selected. We now formalize this idea by letting the subspace V_j be equivalent to the unit interval $[0, 1]$. We then partition the unit interval into subintervals $I_{l,m}$. Let $I_{l,m} = \left[2^{-l}m, 2^{-l}(m+1)\right)$ be a subinterval on $[0, 1]$ corresponding to a set of basis functions $\{2^{(j-l)/2}\psi_m^l(2^{j-l}t - k) | k \in \mathbb{Z}\}$. A basis is then formed if we pick a set of subintervals that cover the entire interval $[0, 1]$. Each set of such intervals then forms one possible partition $p \in \mathcal{P}$, corresponding to the basis $\mathcal{B}^p = \{2^{(j-l)/2}\psi_m^l(2^{j-l}t - k) | k \in \mathbb{Z}, \{(l, m) | I_{l,m} \in p\}\}$.

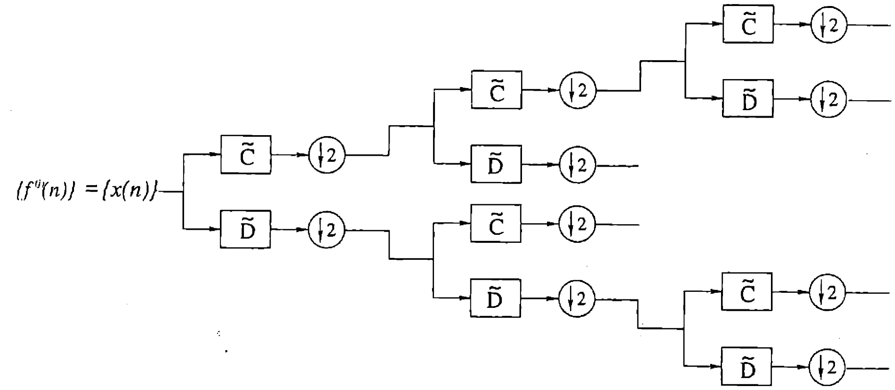


Figure 2.12: Analysis bank of one possible wavelet packet decomposition.

With the wavelet decomposition, a signal $f(t) \in V_j$ can be represented by its coefficients in the spaces $V_0, W_0, \cdots, W_{j-1}$. With wavelet packets, there are many more possible representations. To compute the coefficients in the different subspaces, Mallat's approach may again be used to develop an efficient algorithm. By using the splitting trick, all subspaces are divided using the same set of coefficients $\{h_0(n)\}$ and $\{h_1(n)\}$, and consequently, Mallat's algorithm applies to all of these subspaces. It can be shown that the coefficients may be computed recursively by filtering and then downsampling, where the highpass branches of the filter bank are also iterated. Figure 2.12 shows one possible wavelet packet decomposition that splits the initial subspace three times. As before, the coefficients are recursively computed by filtering with $\{\tilde{c}(n)\} = \{c(-n)\}$ and $\{\tilde{d}(n)\} = \{d(-n)\}$ and then downsampling. The only difference between the wavelet decomposition and the wavelet packet decomposition is therefore in the structure of the filter bank. To reconstruct a signal from its wavelet packet coefficients, we use a synthesis bank that is a mirror image of the analysis bank, and we replace the filter coefficients $\{\tilde{c}(n)\}$ and $\{\tilde{d}(n)\}$ with $\{c(n)\}$ and $\{d(n)\}$ respectively.

In this section, we have introduced a method to obtain many different signal representations. All of these representations are equivalent in the sense that any function $f(t) \in V_j$ can be perfectly represented by a set of coefficients. Some combinations of coefficients, however, may result in representations that are more adapted to various signal features. In the denoising problem, one main goal is to find a basis that allows a signal to be represented by the smallest number of significant coefficients. With this goal in mind, we seek an efficient algorithm to find such a representation. This goal and the resulting solution are further discussed in the next chapter.

Chapter 3

Majorization and Best Basis Search

In this chapter, we present an approach to find the "best" representation of a signal. Wavelet packets will be essential in this process because they provide many possible representations of a signal, and from these representations, we will choose the "best". The meaning of "best" is, however, an important point to address. We must first define the goals to be fulfilled, and then, based on these goals, the meaning of "best" will become more apparent. For example, the goal in communications is to send large amounts of information over a noisy channel in a short amount of time and with minimal degradation. To solve this problem, information theorists try to find a minimal representation of the transmitted information. Compression, therefore, is one possible solution to the communications problem, and as discussed in the introduction, compression is also one possible solution to the denoising problem. By compression, we mean that a signal can be represented faithfully by a small number of large coefficients. In this chapter, we will focus on finding such a minimal representation.

The theory of majorization is directly related to finding a compressed representation of a signal, and we will introduce this theory in Sections 3.1 and 3.2 as a framework for obtaining and understanding further results. In Section 3.3, we will discuss various cost functions that can be used to measure compression, and we will show how these costs fall out of the majorization framework. We conclude this chapter in Section 3.4 by discussing an efficient algorithm to find the "best" basis in a dictionary of bases afforded by wavelet packets. This basis will be optimal in the sense that it minimizes one of the functionals discussed in Section 3.3 over all possible combinations of coefficients in a wavelet packet tree.

3.1 Majorization Theory

Measuring inequality has been an important topic in the history of mathematics and economics. Some of the earlier results date back to Lorenz (1905) and Dalton (1920) who tried to measure the distribution of wealth. Hardy, Littlewood, and Pólya (1929) [9] and later Marshall and Olkin (1979) [10] established important results that provide the framework for our discussion. To measure inequality, we search for a "metric" that will distinguish vector $\mathbf{x}$ from vector $\mathbf{y}$ if the components of one of the vectors are somewhat more equal than the components of the other. Majorization is such a "metric" that measures the distribution of the components in a vector. If $\mathbf{x}$ and $\mathbf{y}$ satisfy the conditions of Definition 2 below, we say that "$\mathbf{x}$ is majorized by $\mathbf{y}$" or equivalently "$\mathbf{y}$ majorizes $\mathbf{x}$", denoted by $\mathbf{x} \prec \mathbf{y}$.

Definition 2 - **Majorization** *[10, 11]*

Let $\mathbf{x}$ *and* $\mathbf{y}$ *be two column vectors with* $\mathbf{x} = [x_1, \cdots, x_n]^T$ *and* $\mathbf{y} = [y_1, \cdots, y_n]^T$. *Let the components of the vectors be ordered according to the following convention:*

$$x_{[1]} \geq x_{[2]} \geq \cdots \geq x_{[n]}, \qquad y_{[1]} \geq y_{[2]} \geq \cdots \geq y_{[n]}$$
$$x_{(1)} \leq x_{(2)} \leq \cdots \leq x_{(n)}, \qquad y_{(1)} \leq y_{(2)} \leq \cdots \leq y_{(n)}.$$

Then, $\mathbf{y}$ *is said to majorize* $\mathbf{x}$ *or equivalently* $\mathbf{x} \prec \mathbf{y}$ *if*

(i) $\sum_{i=1}^{m} x_{[i]} \leq \sum_{i=1}^{m} y_{[i]}$ *holds for* $m = 1, 2, \cdots, (n-1)$, *and*

(ii) $\sum_{i=1}^{n} x_{[i]} = \sum_{i=1}^{n} y_{[i]}$.

An equivalent definition of $\mathbf{x} \prec \mathbf{y}$ *is,*

(i') $\sum_{i=1}^{m} x_{(i)} \geq \sum_{i=1}^{m} y_{(i)}$ *holds for* $m = 1, 2, \cdots, (n-1)$, *and*

(ii') $\sum_{i=1}^{n} x_{(i)} = \sum_{i=1}^{n} y_{(i)}$.

The equivalent definition given by (i') and (ii') will be useful when discussing the ideas proposed by Lorenz. This definition follows from (i), if (ii) is true,

$$\sum_{i=1}^{m} x_{[i]} \leq \sum_{i=1}^{m} y_{[i]} \quad \text{and} \quad \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i$$
$$\Longrightarrow \sum_{i=1}^{n-k} x_{(i)} = \sum_{i=k+1}^{n} x_{[i]} \quad \geq \quad \sum_{i=k+1}^{n} y_{[i]} = \sum_{i=1}^{n-k} y_{(i)},$$

which shows that (i') is true.

Majorization as defined above, attempts to distinguish two vectors $\mathbf{x}$ and $\mathbf{y}$ by comparing partial sums of the ordered components. This essentially measures the variability of the components, given that both vectors must have the same sum, and as a result, $\mathbf{x} \prec \mathbf{y}$ distinguishes $\mathbf{y}$ as a more diverse n-tuple than $\mathbf{x}$. Also note that the following two results always hold,

(a) $\bar{\mathbf{y}} \prec \mathbf{y}$, where $\bar{\mathbf{y}} = [\bar{y}, \cdots, \bar{y}]^T$ and $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$, and

(b) $\mathbf{y} \prec \left[\sum_{i=1}^{n} y_i, 0, \cdots, 0\right]^T$ is true for all $\mathbf{y}$ with non-negative coefficients.

Both (a) and (b) provide intuitively appealing results based on Definition 2. The first result shows that a vector $\bar{\mathbf{y}}$ with the most even distribution, is always majorized by a vector whose average is $\bar{y}$. The second result shows that a vector with positive coefficients and a fixed sum s is always majorized by a vector with a single positive coefficient equal to s. In addition, both of these examples show that majorization is related in some way to the notion of compression, and consequently, this framework will prove useful in the denoising problem.

Additional insight about Definition 2 may be gained by considering the ideas proposed by Lorenz. He was interested in a measure that would indicate how evenly wealth was distributed among a population. He introduced a graph, now called the Lorenz curve, that visually depicts the distribution of income in a population, as shown in Figure 3.1. For simplicity, assume that a population is composed of n individuals whose respective incomes x_i are ranked in increasing order, $x_{(1)}, \cdots, x_{(n)}$. The cumulative wealth of the poorest m people is $S_x^m = \sum_{i=1}^{m} x_{(i)}$, and the Lorenz curve is formed by plotting the normalized points $\left(\frac{m}{n}, \frac{S_x^m}{S_x^n}\right)$ for $m = 0, \cdots, n$. Figure 3.1 shows three possible wealth curves. Curve A corresponds to an even distribution of wealth because each individual in the population contributes an equal amount to the cumulative wealth S_x^m. Curve B corresponds to an unequal distribution of wealth because the poorer individuals contribute a small amount of income, while the richer individuals add a large amount of income, and consequently, curve C corresponds to a population with an even greater income disparity than A or B. The Lorenz curve therefore depicts income disparity by its shape, with greater disparity corresponding to curves that are further away from the equal distribution A. This idea exactly corresponds to Definition 2. Using the terminology above, (i') and (ii') imply that

$$\begin{aligned} \frac{S_x^m}{S_x^n} &\geq \frac{S_y^m}{S_y^n}, && m = 1, \cdots, (n-1) \quad \text{and} \\ \frac{S_x^n}{S_x^n} &= \frac{S_y^n}{S_y^n} = 1 && \text{if and only if } \mathbf{x} \prec \mathbf{y}. \end{aligned}$$

For the example given in Figure 3.1, if the vectors **a**, **b**, and **c** correspond to the curves A, B, and C respectively, then $\mathbf{a} \prec \mathbf{b} \prec \mathbf{c}$.

Both the definition of majorization and the ideas proposed by Lorenz are intuitively appealing as measures of disparity or equivalently as measures of compression. These measures, however, are somewhat strict in the sense that they require both vectors to have an equal sum. Removing this restriction leads to the weaker form of majorization given in Definition 3 below.

Definition 3 – **Weak Majorization** *[10, 11]*

(a) $\mathbf{y}$ *is said to weakly submajorize* $\mathbf{x}$, *or* $\mathbf{x} \prec_w \mathbf{y}$, *if* $\sum_{i=1}^{m} x_{[i]} \leq \sum_{i=1}^{m} y_{[i]}$ *holds for* $m = 1, 2, \cdots, n$.

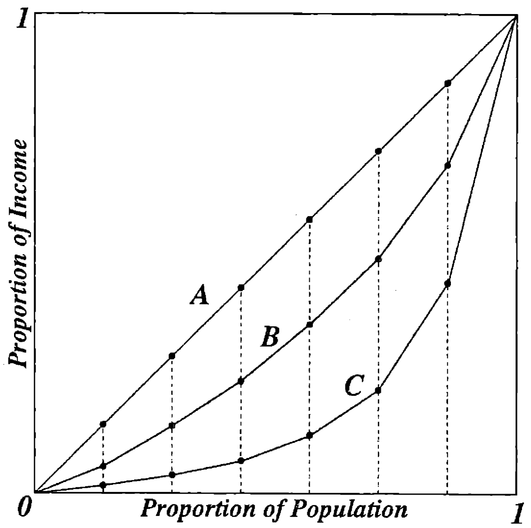


Figure 3.1: Plot of Lorenz curves showing different distributions of wealth.

(b) $\mathbf{y}$ *is said to weakly supermajorize* $\mathbf{x}$, *or* $\mathbf{x} \prec^w \mathbf{y}$, *if* $\sum_{i=1}^{m} x_{(i)} \geq \sum_{i=1}^{m} y_{(i)}$ *holds for* $m = 1, 2, \cdots, n$.

These weaker forms are useful when the constraint of equal sums cannot be enforced.

3.2 Convex Functionals and Majorization

To find the best representation of a signal, we will particularly be interested in functions that provide a measure of compression, rather than the conditions imposed by the previous definitions. In general, however, the conditions for majorization are closely related to convex functions, and the following theorem due to [9] relates the two ideas.

Theorem 1 – **Convex Functionals and Majorization** *[11]*

Let I be an interval in $\mathbb{R}$, and let $\mathbf{x}$ and $\mathbf{y}$ be two n-dimensional vectors such that $x_i, y_i \in I$ for all $i = 1, \cdots, n$. Then,

$$\sum_{i=1}^{n} \gamma(x_i) \leq \sum_{i=1}^{n} \gamma(y_i)$$

holds for every continuous convex function $\gamma : I \longrightarrow \mathbb{R}$ if and only if $\mathbf{x} \prec \mathbf{y}$ holds.

As a result, convex functions as a whole provide a measure of compression. We also include the following theorem[1] for completeness and to relate the ideas presented in Definition 3 to convex functions.

Theorem 2 – **Convex Functions and Weak Majorization** *[11]*

Let I be an interval in $\mathbb{R}$, and let $\mathbf{x}$ and $\mathbf{y}$ be two n-dimensional vectors such that $x_i, y_i \in I$ for all $i = 1, \cdots, n$. Then,

(a) $\sum_{i=1}^{n} \gamma(x_i) \leq \sum_{i=1}^{n} \gamma(y_i)$ *holds for every continuous increasing convex function $\gamma : I \longrightarrow \mathbb{R}$ if and only if $\mathbf{x} \prec_w \mathbf{y}$ holds, and*

(b) $\sum_{i=1}^{n} \gamma(x_i) \leq \sum_{i=1}^{n} \gamma(y_i)$ *holds for every continuous decreasing convex function $\gamma : I \longrightarrow \mathbb{R}$ if and only if $\mathbf{x} \prec^w \mathbf{y}$ holds.*

In this analysis, we are particularly interested in the notion of Schur-convexity, due to Schur (1923). Since $\mathbf{x} \prec \mathbf{y}$ provides an ordering on two vectors $\mathbf{x}$ and $\mathbf{y}$, we are

[1] Theorem 2 is due to Weyl (1949) and Tomić (1949).

interested in functions Υ that will always preserve this ordering (i.e. $\Upsilon(\mathbf{x}) \leq \Upsilon(\mathbf{y})$ or $\Upsilon(\mathbf{x}) \geq \Upsilon(\mathbf{y})$). The definition below links the ideas of majorization to the convex and concave functions Υ.

Definition 4 - Schur-Convex Functions *[10, 11]*

(i) A real-valued function Υ defined on a set $\mathcal{D} \subset \mathbb{R}^n$ is said to be Schur-convex on $\mathcal{D}$ if

$$\mathbf{x} \prec \mathbf{y} \text{ on } \mathcal{D} \implies \Upsilon(\mathbf{x}) \leq \Upsilon(\mathbf{y}).$$

If equality holds only when $\mathbf{x}$ is a permutation of $\mathbf{y}$, then Υ is said to be strictly Schur-convex on $\mathcal{D}$.

(ii) A real-valued function Υ defined on a set $\mathcal{D} \subset \mathbb{R}^n$ is said to be Schur-concave on $\mathcal{D}$ if

$$\mathbf{x} \prec \mathbf{y} \text{ on } \mathcal{D} \implies \Upsilon(\mathbf{x}) \geq \Upsilon(\mathbf{y}).$$

If equality holds only when $\mathbf{x}$ is a permutation of $\mathbf{y}$, then Υ is said to be strictly Schur-concave on $\mathcal{D}$.

Definition 4 also implies that Υ is Schur-concave on $\mathcal{D}$ if and only if $(-\Upsilon)$ is Schur-convex on $\mathcal{D}$. This definition therefore reveals that Schur-convex functions are order preserving, and the identification of such functions will lead to appropriate ordering when an underlying majorization is present. The power of this argument can be illustrated by an example. Schur showed that the diagonal elements $\{h_{ii}\}$ for a positive semidefinite Hermitian matrix H are majorized by its eigenvalues,

$$(h_{11}, \cdots, h_{nn}) \prec (\lambda_1, \cdots, \lambda_n).$$

Since there exists an underlying majorization of the form above, any Schur-concave or Schur-convex functions on $\mathbb{R}^n_+$ will satisfy the inequalities in Definition 4. One famous inequality, called Hadamard's determinant inequality, is given by

$$\prod_{i=1}^{n} h_{ii} \;\geq\; \prod_{i=1}^{n} \lambda_i(H) = \det(H),$$

and is a direct result of Definition 4 because the product function is Schur-concave on $\mathbb{R}^n_+$.

An additional property of majorization which will prove useful in the sequel concerns compositions of Schur-concave and Schur-convex functions. We restrict our discussion to a result that relates Schur-convex functions in $\mathbb{R}^n$ to one-dimensional convex functions, and we refer the interested reader to [10] for a more complete treatment. The following theorem essentially restates Theorem 1, but is necessary to concretely establish the relationship between majorization theory and the additive cost functions discussed in the next section.

Theorem 3 – Additive Compositions *[11]*

Let I be an interval in $\mathbb{R}$. Let $\mathbf{x}$, $\mathbf{y}$ be two n-tuples such that $x_i, y_i \in I$ for all $i = 1, \cdots, n$ and define $\Upsilon(\mathbf{x}) : I^n \rightarrow \mathbb{R}$ to be a real-valued function such that $\Upsilon(\mathbf{x}) = \sum_{i=1}^{n} \gamma(x_i)$ for some continuous function $\gamma : I \rightarrow \mathbb{R}$. Then Υ is a Schur-convex function on I^n if and only if γ is a convex function on I.

In the next section, we will discuss several additive convex (concave) functions that will lead to Schur-convex (concave) functions. These functions will then provide an ordering on two sets of coefficients when an underlying majorization is present, but since Definition 4 is not an "if and only if" statement, satisfying inequalities of the form $\Upsilon(\mathbf{x}) \leq \Upsilon(\mathbf{y})$ will not necessarily imply $\mathbf{x} \prec \mathbf{y}$. However, functions that provide a "good" measure of compression will serve as an approximation to $\mathbf{x} \prec \mathbf{y}$.

3.3 Cost Functions

In this section, we will use the previous results concerning majorization to analyze some of the cost functions commonly used to determine the best basis in a wavelet packet tree. In addition, we will introduce some novel cost functions. In light of the above majorization theory, these functions must either be Schur-convex or Schur-concave to preserve the order of the majorization. The functions, however, must also be good indicators of compression in a wavelet packet basis, and this determination is generally made by running computer simulation to assess the quality of compression on the average. Among the "good" functions, we will find that information cost functions work well. We will also show that cost functions derived from the Lorenz curve may be used. We will finally establish that additive cost functions will ensure that the order of majorization is preserved.

3.3.1 Information Cost Functions

The following information cost functions have been proposed in [12].

(1) *Number of coefficients above a threshold T*

This function provides an additive measure of information by giving the number of coefficients with magnitude greater than an arbitrary threshold T. This is a measure of compression because it indicates the number of coefficients required to represent a signal. This cost function is given by,

$$\Upsilon_T(\mathbf{x}) = \sum_{i=1}^{n} \mathcal{I}_T(x_i), \quad \text{where} \tag{3.1}$$

$$\mathcal{I}_T(x_i) = \begin{cases} 1 & |x_i| \geq T \\ 0 & \text{otherwise} \end{cases} . \tag{3.2}$$

The discontinuity of this function makes it neither Schur-convex nor Schur-concave. This function, therefore, does not preserve the order of majorization, and in general, will not provide an accurate measure of compression. As an example, recall that $\mathbf{x} = \bar{\mathbf{y}} \prec \mathbf{y}$ is always true. Choosing the threshold T such that $\bar{y} < T < \max_i\{y_i\}$, necessarily implies $\Upsilon_T(\mathbf{x}) < \Upsilon_T(\mathbf{y})$, which should be true for a Schur-convex function. By choosing $\min_i\{y_i\} < T < \bar{y}$, $\Upsilon_T(\mathbf{x}) > \Upsilon_T(\mathbf{y})$ will always be true, indicating that $\Upsilon_T(\cdot)$ is a Schur-concave function. The choice of T is therefore critical , which shows that this information cost function does not always preserve the order of the majorization. Figure 3.2(a) shows this function for the case $\Upsilon_T : \mathbb{R}^2 \rightarrow \mathbb{R}$.

(2) *Concentration in the ℓ^p norm*

Another information measure is the classical ℓ^p norm,

$$\Upsilon_p(\mathbf{x}) = \|\mathbf{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p} . \tag{3.3}$$

It can be shown [10] that $\Upsilon_p(\cdot)$ is strictly Schur-convex for $p > 1$ and strictly Schur-concave for $p < 1$. From the majorization theory, $\mathbf{x} \prec \mathbf{y}$ then implies

$\Upsilon_p(\mathbf{x}) \leq \Upsilon_p(\mathbf{y})$ for $p > 1$ and $\Upsilon_p(\mathbf{x}) \geq \Upsilon_p(\mathbf{y})$ for $p < 1$. As a result, for $p > 1$ the best signal representation in a wavelet packet basis will be found by maximizing the ℓ^p cost function, whereas $p < 1$ will require that ℓ^p be minimized. Figure 3.2(b) shows that the ℓ^p cost function in $\mathbb{R}^2$ is concave for $p = 0.2$, while Figure 3.2(c) shows that ℓ^p is convex for $p = 2.5$. The ℓ^p norm for $p = 1$ shown in Figure 3.2(d) exhibits both Schur-convexity and Schur-concavity. In practice, this function should be minimized, as will be illustrated in Section 3.3.3.

(3) *Entropy cost function*

The entropy cost function is an extremely important measure of information. In information theory, it measures the amount of uncertainty in a random variable. For a fixed alphabet size, the uniform distribution has the largest uncertainty and hence the largest possible entropy when compared to all other distributions with the same size alphabet. This idea is very reminiscent of the fact that $\bar{\mathbf{y}} \prec \mathbf{y}$ is satisfied for all $\mathbf{y}$, as discussed in Section 3.1. The entropy measure therefore assigns a compressed signal a lower cost than it would to a more uniformly distributed signal. The entropy is defined for probability mass functions as

$$\mathcal{H}_E(\mathbf{x}) = -\sum_{i=1}^{n} x_i \log\{x_i\} \quad \text{for} \quad 0 \leq x_i \leq 1 \quad \text{and} \quad \sum_i x_i = 1. \qquad (3.4)$$

As a function in $\mathbb{R}^n$, $\mathcal{H}_E(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}$ is strictly Schur-concave and is therefore a good measure of spread. In terms of majorization, $\mathbf{x} \prec \mathbf{y}$ will then imply $\mathcal{H}_E(\mathbf{x}) \geq \mathcal{H}_E(\mathbf{y})$. The goal, therefore, in finding the best signal representation will be to minimize this cost function over all possible wavelet packet bases. The concavity of the entropy cost is depicted in Figure 3.2(e), where the magnitudes of x_1 and x_2 are used to compute the entropy cost and a constant sum of 1 is not enforced.

(4) *Logarithm of energy*

This final information cost function is defined to be,

$$\mathcal{H}_L(\mathbf{x}) = \sum_{i=1}^{n} \log\{|x_i|^2\}, \tag{3.5}$$

with the convention $\log\{0\} = 0$. According to [12], this is a measure of entropy for a Gauss-Markov process with n uncorrelated Gaussian random variables, each with a variance equal to the square of the components of $\mathbf{x}$. This function is strictly Schur-concave as shown in Figure 3.2(f), and therefore should be minimized over all possible wavelet packet coefficients.

3.3.2 Lorenz Cost Functions

The following cost functions are based on the Lorenz curve and its meaning in terms of income inequality. The first two functions are given in [10], and the third is another possible function that we propose.

(1) *Gini Coefficient*

This function was originally proposed by Gini (1912) and is based on the Lorenz curve shown in Figure 3.3. The cost function of Gini is twice the shaded area in Figure 3.3 which is twice the area between the Lorenz curve and the 45° line. This area is given by,

$$\Upsilon_G(\mathbf{x}) = 1 + \frac{1}{n} - \frac{2}{n^2\bar{x}} \sum_i i x_{[i]}, \tag{3.6}$$

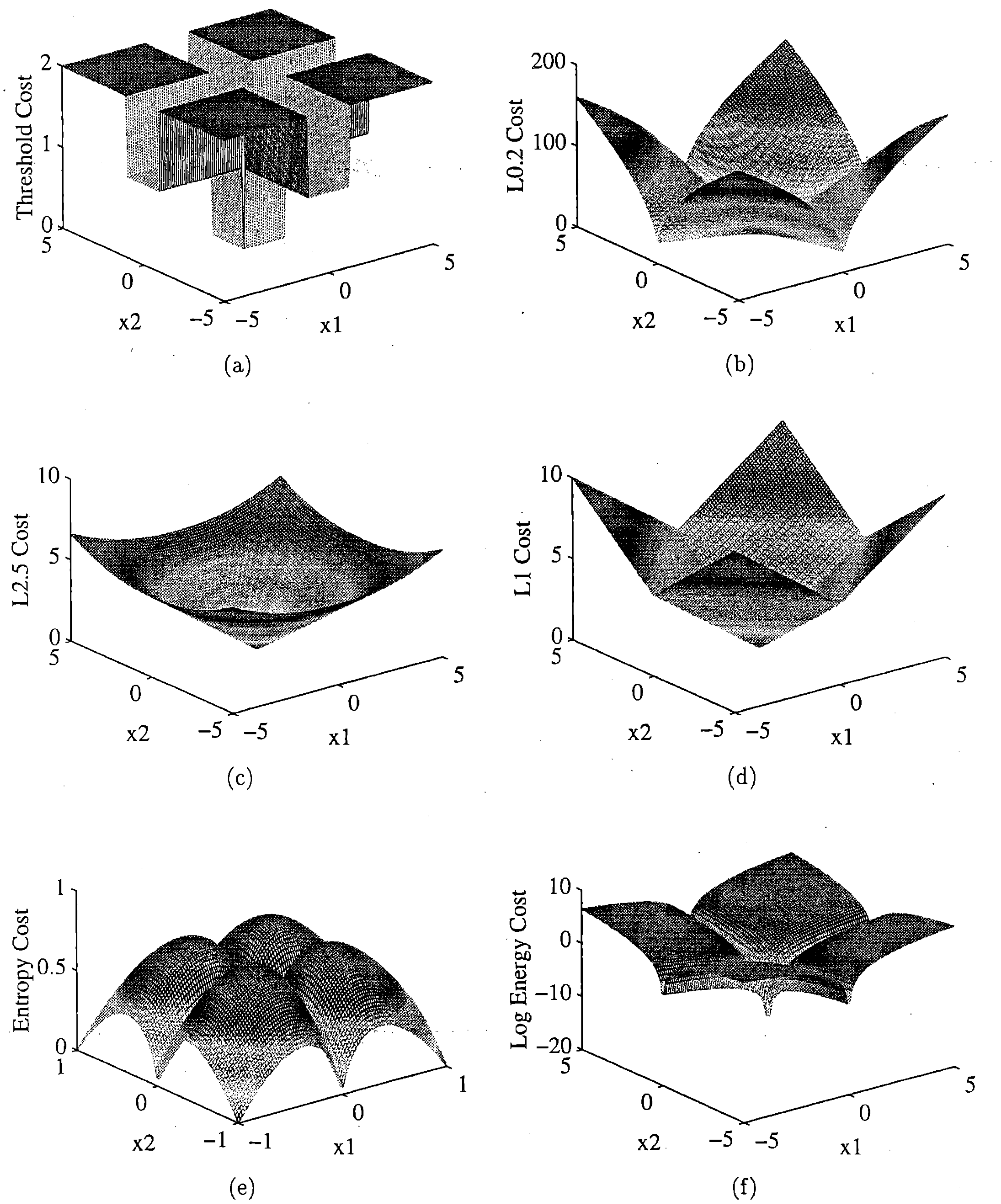


Figure 3.2: Information cost functions. (a) Number of coefficients above a threshold T. (b) Concentration in the l^p norm for $p = 0.2$. (c) Concentration in the l^p norm for $p = 2.5$. (d) Concentration in the l^p norm for $p = 1.0$. (e) Entropy cost function. (f) Logarithm of the energy.

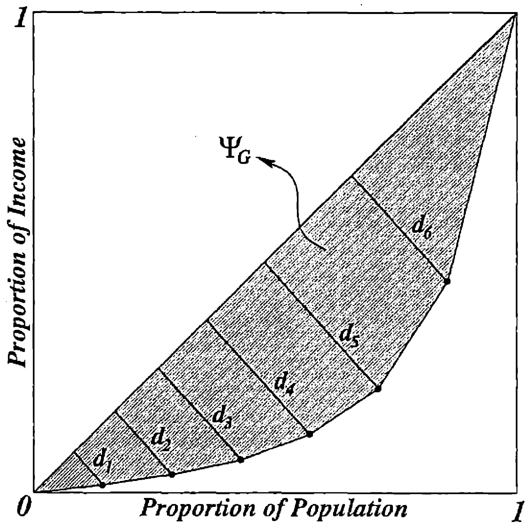


Figure 3.3: Illustrates different measures for the amount of bow in the Lorenz curve.

and is strictly Schur-convex on $\mathbb{R}^n_+$. To achieve good compression, this cost function should therefore be maximized. This agrees with the notion of majorization because it corresponds to maximizing the area in Figure 3.3. Figure 3.4(a) shows the somewhat elaborate graph of the Gini function.

(2) *Schutz Coefficient*

The Schutz coefficient is a measure proposed by Schutz (1951), again used to measure income inequality, and is given by

$$\Upsilon(\mathbf{x}) = \frac{1}{\bar{x}}\sum_{i=1}^{n}(x_i - \bar{x})^+, \quad \text{where } u^+ = max\{u, 0\}. \tag{3.7}$$

This cost function is a normalized measure that indicates the total excess income above the mean. It is Schur convex on $\mathbb{R}^n_+$ but not strictly Schur convex and is shown in Figure 3.4(b). In terms of the Lorenz curve, this coefficient can be related to the slopes of the line segments.

(3) *Lorenz Bow Distance*

This measure, which we propose, is very similar to the Gini coefficient and is a function of the distances $\{d_j\}$ shown in Figure 3.3. The lengths $\{d_j\}$ are the perpendicular distances from the points $(\frac{j}{n}, \tilde{x}_j)$ to the 45° line, where

$$\tilde{x}_j = \frac{\sum_{i=1}^{j} x_{(i)}}{\sum_{i=1}^{n} x_{(i)}}.$$

For simplicity, we assume that all of the coefficients are positive, but in computing this cost, the absolute value of the coefficients should be used. The distances are then determined to be,

$$d_j = \frac{1}{\sqrt{2}} \left| \frac{j}{n} - \tilde{x}_j \right|. \tag{3.8}$$

This measure can be manipulated slightly to provide more intuition,

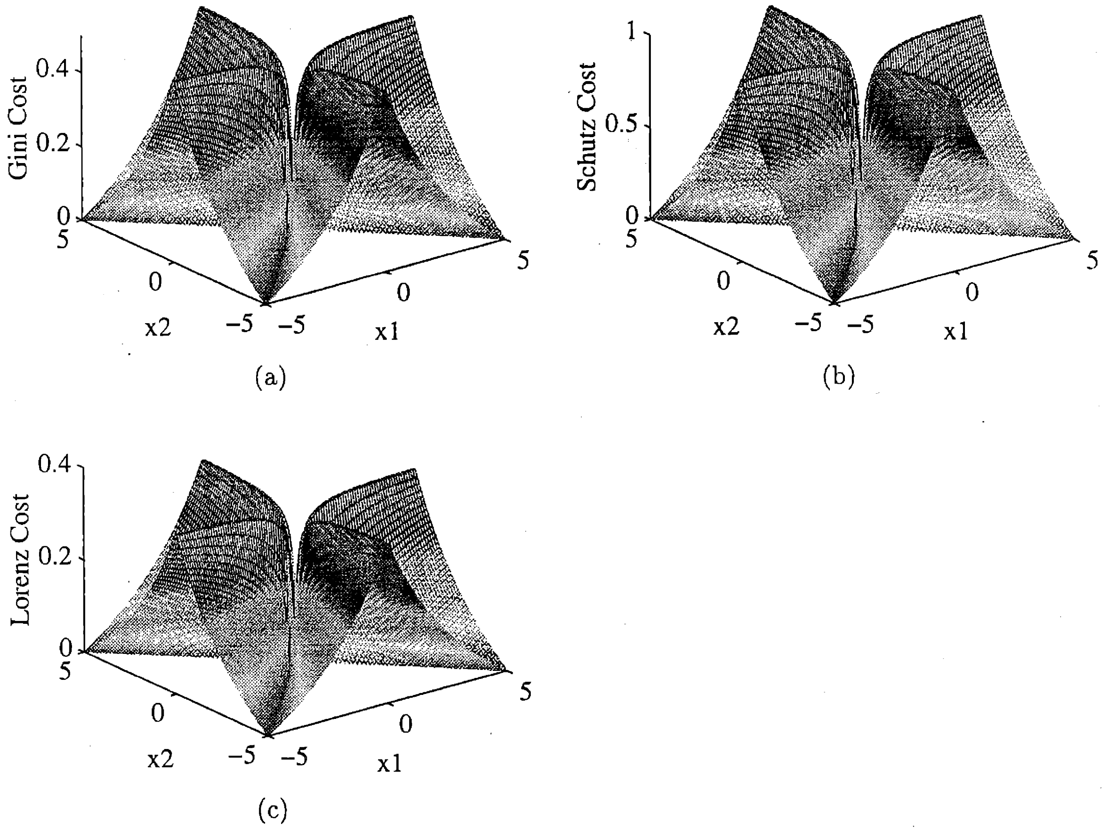
$$\begin{aligned} d_j &= \frac{1}{\sqrt{2}} \left| \frac{j}{n} - \tilde{x}_j \right| = \frac{1}{\sqrt{2}} \left| \frac{j}{n} - \frac{\sum_{i=1}^{j} x_{(i)}}{\sum_{i=1}^{n} x_{(i)}} \right| \\ &= \frac{1}{\sqrt{2}} \left| \frac{\frac{j}{n} \sum_{i=1}^{n} x_{(i)} - \sum_{i=1}^{n} x_{(i)}}{\sum_{i=1}^{n} x_{(i)}} \right| = \frac{1}{\sqrt{2}} \frac{\left| \sum_{i=1}^{j} \left(\bar{x} - x_{(i)}\right) \right|}{\sum_{i=1}^{n} x_{(i)}}. \end{aligned} \tag{3.9}$$

This shows that d_j is a normalized measure of the total distance that the sorted coefficients $\{x_{(i)}\}$ up to j deviate from the mean. From these distances, two possible measures can be computed,

$$\Upsilon_{L,max} = \max_j \{d_j\} \tag{3.10}$$

$$\Upsilon_{L,dist} = \sum_{j=1}^{n} d_j, \tag{3.11}$$

and both of these measures should be maximized to preserve the underlying majorization. Figure 3.4(c) shows the cost function $\Upsilon_{L,dist}(\cdot)$. Admittedly, all



(a)

(b)

(c)

Figure 3.4: Cost functions based on the Lorenz curve. (a) Gini Coefficient (b) Schutz Coefficient (c) Lorenz Bow Distance

of the graphs in Figure 3.4 have the same shape for the proposed cost functions, but in higher dimensions, they will not be equivalent.

3.3.3 Exponential Cost Function

In this section, we show that minimizing the ℓ^1 norm is reasonable to achieve a compressed representation of a signal among a set of representations afforded by wavelet packets. We assume that the magnitudes of the wavelet packet coefficients have an exponential distribution of the form,

$$p_Y(y;\theta) \;=\; \frac{1}{\theta}e^{-\frac{y}{\theta}}, \;\; y > 0. \tag{3.12}$$

This is a good assumption in practice because the wavelet coefficients generally have a large number of small coefficients and a small number of large coefficients. It is known that the Maximum-Likelihood estimator for this type of distribution is the mean of the observations, or

$$\hat{\theta} \;=\; \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i. \tag{3.13}$$

If the coefficients truly come from an exponential distribution, then $\hat{\theta}$ should be minimized over the wavelet packet coefficients so that the drop-off is as rapid as possible. Since $1/n$ is simply a scale factor, the results will be the same if $n\hat{\theta}$ is minimized. The cost function for all $x_i \in \mathbb{R}$ is then given by,

$$\Upsilon_E(\mathbf{x}) \;=\; \sum_{i=1}^{n} |x_i|, \tag{3.14}$$

which is exactly the ℓ^1 norm. As a result, we see that this type of measure is very useful for maximizing the decay in the exponential distribution and consequently, in obtaining a minimal representation of a signal. Note that this function was previously shown in Figure 3.2(d) and may be viewed as another ℓ^p based search approach.

3.3.4 Additive Cost Functions

In Section 3.4, we will describe an efficient method for finding the best representation of a signal by minimizing or maximizing an appropriate cost function. This search will involve comparisons of the form

$$\begin{aligned} \Upsilon(\mathbf{p}) &\leq \Upsilon(\mathbf{c}_1) + \Upsilon(\mathbf{c}_2) \quad \text{or} & (3.15) \\ \Upsilon(\mathbf{p}) &\geq \Upsilon(\mathbf{c}_1) + \Upsilon(\mathbf{c}_2) & (3.16) \end{aligned}$$

where $\mathbf{p} \in \mathbb{R}^n$ and $\mathbf{c}_1, \mathbf{c}_2 \in \mathbb{R}^{n/2}$. These comparisons are different from $\Upsilon(\mathbf{x}) \leq \Upsilon(\mathbf{y})$ or $\Upsilon(\mathbf{x}) \geq \Upsilon(\mathbf{y})$ where $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. The functions $\Upsilon(\cdot)$ must therefore be chosen so that $\Upsilon(\mathbf{c}_1) + \Upsilon(\mathbf{c}_2)$ is Schur-convex (concave) if $\Upsilon(\cdot)$ is Schur-convex (concave). In doing so, the comparisons in Equations (3.15) and (3.16) will preserve the order of the underlying majorization. In general, however, it is not trivial to determine the convexity or concavity of $\Upsilon(\mathbf{c}_1) + \Upsilon(\mathbf{c}_2)$ based on knowledge of $\Upsilon(\cdot)$, but one valid choice is available based on the results of Theorem 3. If $\Upsilon(\cdot)$ is composed of additive functions $\gamma(\cdot)$ then $\Upsilon(\mathbf{c}_1) + \Upsilon(\mathbf{c}_2)$ will also be composed of additive functions $\gamma(\cdot)$, and the Schur-convexity (concavity) will be guaranteed in the additive composition. The requirement can then be formally stated as

$$\Upsilon(\mathbf{c}_1 \oplus \mathbf{c}_2) = \Upsilon(\mathbf{c}_1) + \Upsilon(\mathbf{c}_2), \qquad (3.17)$$

where $\mathbf{c}_1 \oplus \mathbf{c}_2 \in \mathbb{R}^n$.

From the list of valid functions given in the previous section, the entropy, log of energy, and ℓ^1 norm are additive cost functions of the form given in Equation (3.17). These functions will therefore be useful in the best basis search because they will allow exact comparisons of the costs in the wavelet packet tree. By slightly altering the form of the ℓ^p norms, they can also be used as additive cost functions by letting

$$\tilde{\Upsilon}_p(\mathbf{x}) \quad = \quad (\|\mathbf{x}\|_p)^p = \sum_{i=1}^{n} |x_i|^p. \tag{3.18}$$

These functions then become information costs as described in [12] which must be minimized for $p < 2$ and maximized for $p > 2$.

Another issue which must be addressed in order to effectively use the functions in Sections 3.3.1–3.3.3 is the normalization of the wavelet packet coefficients. For the entropy cost, this corresponds to ensuring that the coefficients are positive, less than 1, and sum to 1. One possible normalization proposed in [12] is

$$\tilde{\mathcal{H}}_E(\mathbf{x}) \quad = \quad -\sum_{i=1}^{n} \frac{|x_i|^2}{\|\mathbf{x}\|^2} \log \left\{ \frac{|x_i|^2}{\|\mathbf{x}\|^2} \right\}, \tag{3.19}$$

but this does not lead to an additive cost function. Normalization issues for the entropy cost function and other costs are addressed in [13, 14], and we refer the reader to these sources for further information. In the next section, we will show how these cost functions are useful in the best basis search.

3.4 Best Basis Search

As discussed in Chapter 2, the wavelet packet tree provides an overcomplete representation of a given signal, and for a discrete signal of size N, it can be shown that this tree consists of more than $2^{\frac{N}{2}}$ possible bases. Our ultimate goal is to find an efficient method to "prune" a wavelet packet tree in order to find the "best" basis representation of a signal in some predetermined sense. This is essentially an optimization problem, requiring a cost function which is defined by a specific goal. For example, to achieve good compression, the functions discussed in Section 3.3 will be used, but other costs can (and will) be considered for other optimization criteria.

Wickerhauser and Coifman [12] propose a method for pruning the wavelet packet tree in only $O(N \log N)$ operations, and their search method will be adopted here for the best basis determination. The pruning algorithm recursively compares the cost of a parent node in the tree with the total cost of its two children. In order to minimize the overall cost function, the coefficients with the minimum cost replace those of the parent node. This approach can be altered to maximize a cost function by choosing the maximum cost of the parent and children nodes. The process is repeated at all levels starting from the bottom of the tree and traversing upwards until the top is reached. The coefficients at the top node then correspond to the "best" coefficients, and their associated cost minimizes (or maximizes) the given cost function. Figure 3.5 illustrates the mechanics of this algorithm. At each node, the numbers in top boxes represent the cost of the coefficients at that node, while the numbers in the bottom boxes represent the optimal costs of all the nodes below the parent. The nodes corresponding to the best basis are the ellipses with the brick texture, and the arrows show which coefficients and costs are propagated up the tree.

Note that the comparisons between the parent and children nodes are of the form previously shown in Equations (3.15) and (3.16). To ensure that the order of the underlying majorization is preserved, we will primarily focus on additive cost functions in conjunction with this pruning algorithm. In addition, we will see that some of the non-additive cost functions in Section 3.3 also work fairly well. We test the performance of the costs on a family of signals similar to the one shown in Figure 3.6(a). These signals were constructed by passing white noise through a filter to generate different realizations of a signal with the same underlying power spectrum.

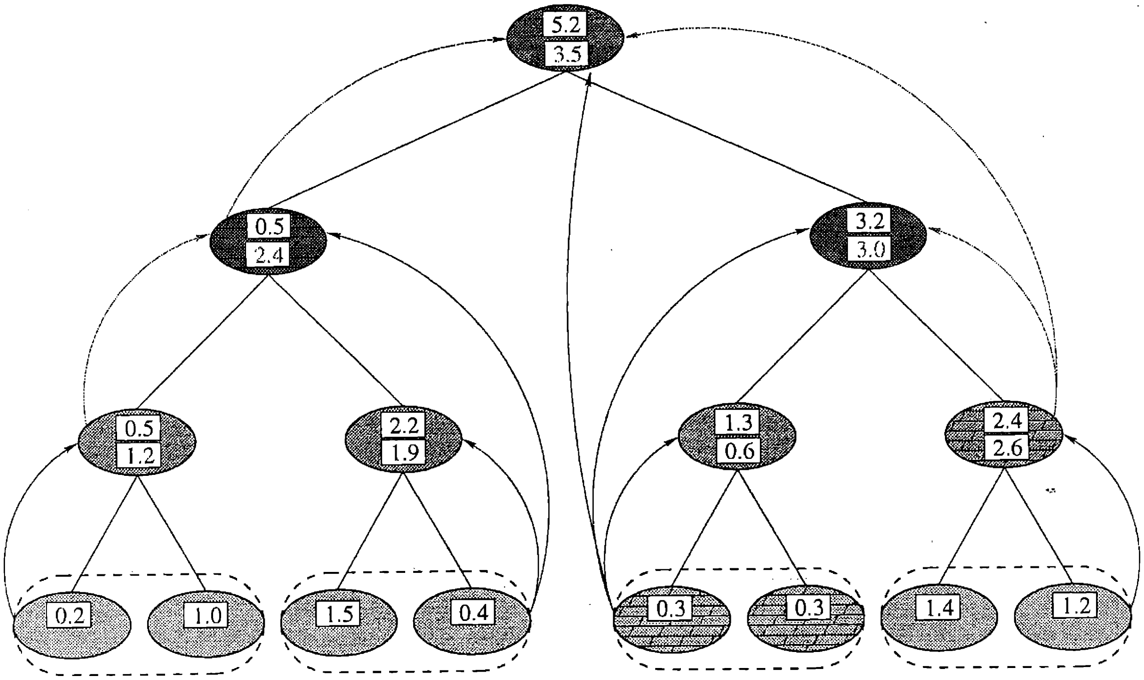


Figure 3.5: Efficient method for finding the best basis given an additive cost function.

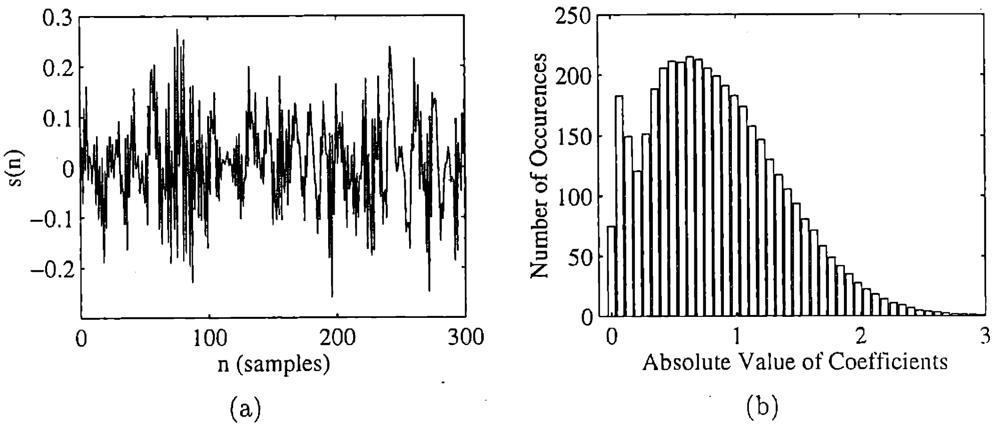


Figure 3.6: (a) Average signal used to test the performance of the best basis algorithm for different cost functions. (b) Average histogram of the magnitude of the coefficients corresponding to the test signal.

The histogram in Figure 3.6(b) shows the magnitude of the signal coefficients averaged over 50 different noise realizations. The performances for the information cost functions are shown in Figure 3.7, while the other cost functions from Section 3.3 are shown in Figure 3.8.

The results in Figure 3.7 correspond to additive cost functions, and the results are all very similar. Note that the histograms in Figures 3.7(a) and (b) show surprisingly good results, even though the associated cost function is neither Schur-convex nor Schur-concave. The threshold T, however, was judiciously chosen to achieve the rapid decay of the histogram, and the cost function was minimized for the small threshold $T = 0.05$ and maximized for the large threshold $T = 2.5$. In practice, though, there is no robust choice for the threshold, and hence, as predicted earlier, this cost does not consistently preserve the order of majorization. The remaining information cost functions generated very similar results. Figure 3.8(d) shows the ℓ^1 cost which is also additive and which in the previous section, was shown to maximize the rate of decay in the exponential distribution.

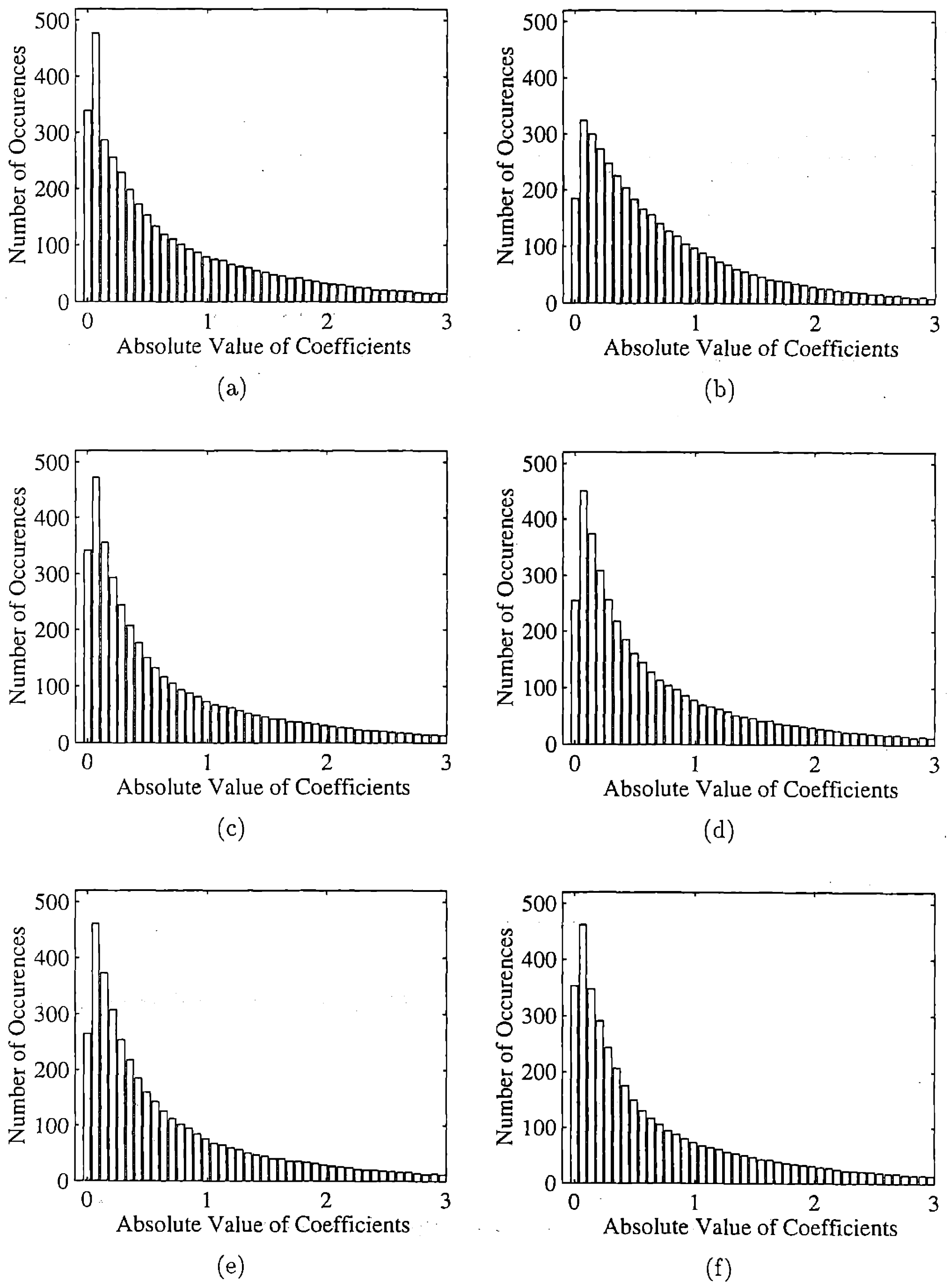
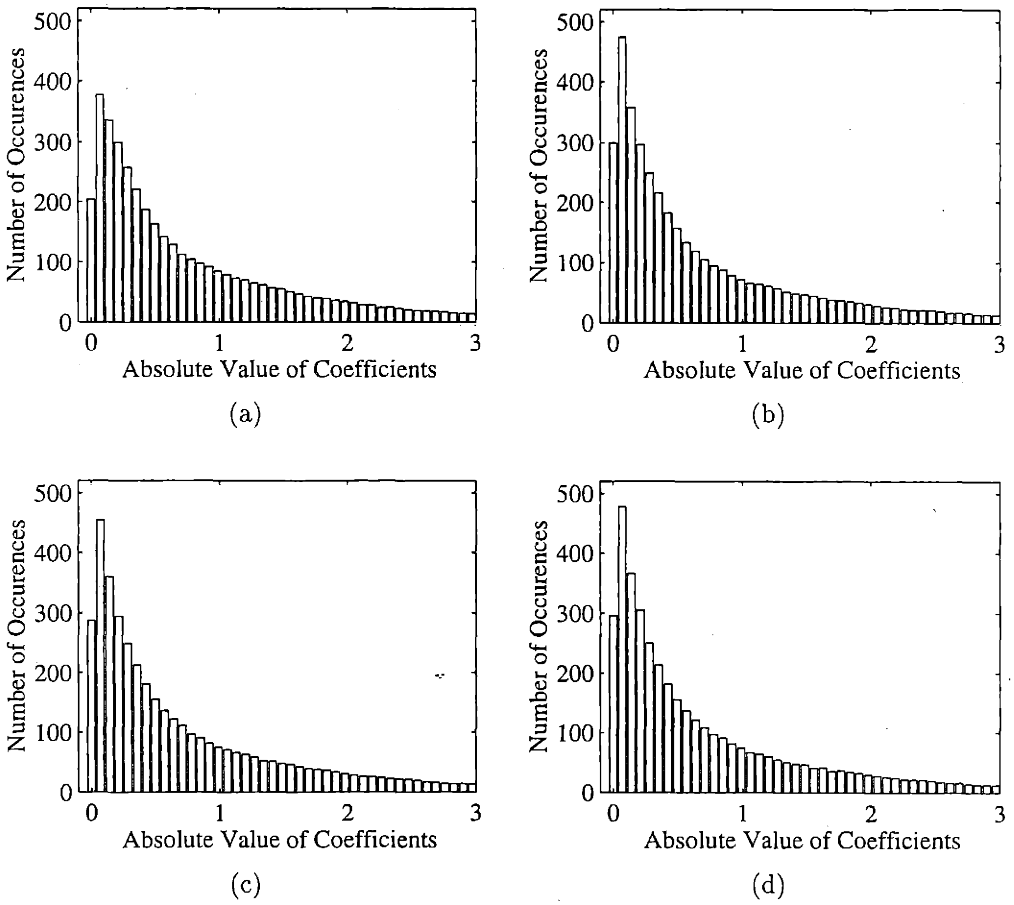


Figure 3.7: Histograms of the wavelet coefficients that optimize each cost function. (a) Number above the threshold $T = 0.05$. (b) Number above the threshold $T = 2.5$. (c) ℓ^p norm for $p = 0.2$. (d) ℓ^p norm for $p = 2.5$. (e) Entropy cost function. (f) Log of the energy.



(a) (b) (c) (d)

Figure 3.8: Histograms of other cost functions proposed in Section 3.3. (a) Gini coefficient. (b) Schutz coefficient. (c) Lorenz total bow distance. (d) ℓ^p norm for $p = 1$.

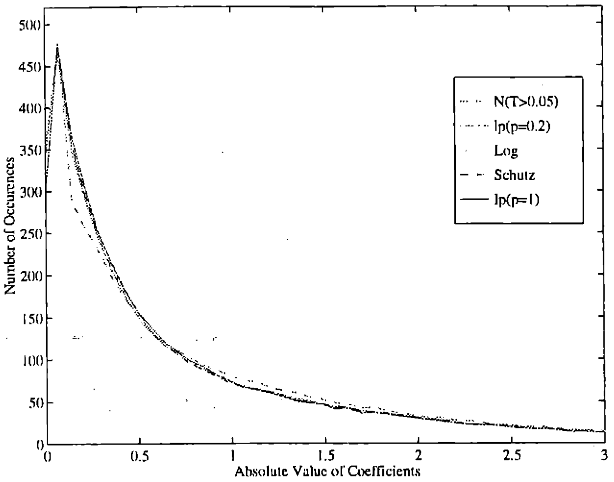


Figure 3.9: Comparison of the five best histograms shown in Figures 3.7 and 3.8.

In the previous section, we also discussed some cost functions based on the Lorenz curve which were not additive. The histograms corresponding to these cost functions are shown in Figures 3.8(a)–(c). Even though they are not additive, they still give generally good results. Their performance in the best basis algorithm seems to imply that $\Upsilon(\mathbf{c}_1) + \Upsilon(\mathbf{c}_2)$ is a Schur-convex function in $\mathbb{R}^n$, even though we could not guarantee this fact. If this is true, then the comparisons $\Upsilon(\mathbf{p}) \leq \Upsilon(\mathbf{c}_1) + \Upsilon(\mathbf{c}_2)$ will preserve the order of an underlying majorization. To compare all of the results, we show the histograms with the fastest decay in Figure 3.9, plotted as a continuous line graph for visual clarity. This shows that each of the proposed cost functions yield similar performances with respect to compression.

In the next chapter, we will show why finding a compressed representation of a signal is useful in the denoising problem, and we will introduce two more cost functions which find the "best" representation among the wavelet packet coefficients with the goal of minimizing the reconstruction error.

Chapter 4

Denoising Techniques Using Wavelet Packets

In the previous two chapters, we presented the tools necessary to first obtain an overcomplete representation of a signal and then to choose the "best" representation that satisfies the goal of compression. In this chapter, we use these tools to reconstruct a signal from its noisy observations. One of the most studied denoising problems, and the one we consider here, is that of an unknown discrete-time signal corrupted by additive noise,

$$x(m) \quad = \quad s(m) + v(m), \quad m = 0, 1, \cdots, N-1. \tag{4.1}$$

The goal is to remove the unwanted noise using a single realization of the noisy signal, consisting of N coefficients. For simplicity, the noise is assumed to be white and Gaussian, with known variance σ^2. For notational convenience, the observed samples, $\{x(m)\}$ for $m = 0, 1, \cdots, N-1$, are represented by the column vector $\mathbf{x}$,

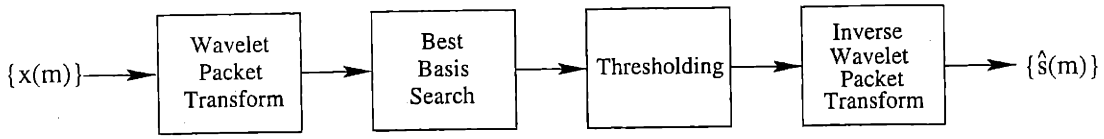


Figure 4.1: General denoising strategy using wavelet packets and a thresholding procedure.

and the underlying signal $\{s(m)\}$ and the set of noise samples $\{v(m)\}$ are represented by corresponding vectors $\mathbf{s}$ and $\mathbf{v}$, respectively. Equation (4.1) can then be rewritten as the following vector equation,

$$\mathbf{x} \ = \ \mathbf{s}+\mathbf{v}, \tag{4.2}$$

where the covariance matrix of the noise is

$$\mathbf{K}_{\mathbf{v}} \ = \ \mathrm{E}\{\mathbf{v}\mathbf{v}^T\} = \sigma^2\mathbf{I}. \tag{4.3}$$

This is a classical estimation problem, and given the statistics for $\mathbf{s}$ and $\mathbf{v}$, an optimal minimum mean-squared error solution can be found. In this analysis of the problem, however, no statistics or model for the underlying signal are assumed. The solution, instead, lies in the overcomplete representation of signals provided by wavelet packets. The topic of this chapter will be to find the best signal representation to effectively remove the noise.

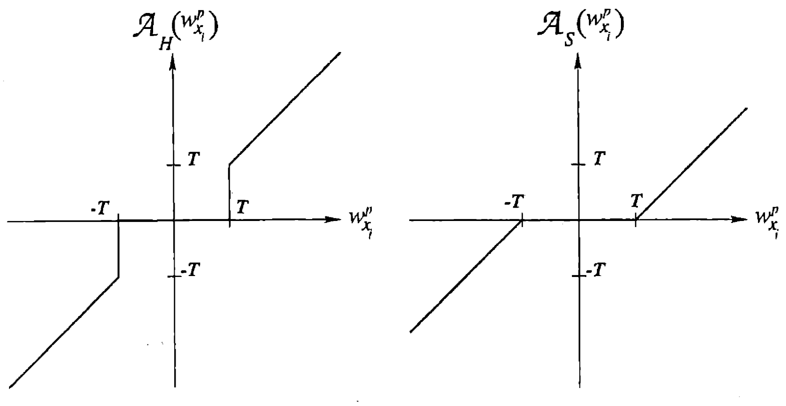
Figure 4.1 shows the general denoising strategy that will be followed. The observed signal $\{x(m)\}$ is first decomposed into a set of wavelet packet coefficients, via a linear transformation. The matrix $\boldsymbol{W}^p_x$ is such a transformation and corresponds to the wavelet packet basis $\mathcal{B}^p = \{2^{(j-l)/2}\psi^l_m(2^{j-l}t-k)|k \in \mathbb{Z}, \{(l,m)|I_{l,m} \in p\}\}$, as discussed in Chapter 2. By applying the transformation to the observed signal and using linearity, the following decomposition is obtained,

$$\begin{aligned} \boldsymbol{W}_x^p \mathrm{x} &= \boldsymbol{W}_x^p \mathrm{s} + \boldsymbol{W}_x^p \mathrm{v} \qquad (4.4) \\ \boldsymbol{w}_x^p &= \boldsymbol{w}_s^p + \boldsymbol{w}_v^p, \qquad (4.5) \end{aligned}$$

where $\boldsymbol{w}_x^p$ is the vector of wavelet packet coefficients. The coefficients resulting from the transformation are thus partitioned into signal and noise components. Since we have limited the wavelet packet decompositions to orthogonal transformations (*i.e.* $\boldsymbol{W}_x^{pT}\boldsymbol{W}_x^p = \boldsymbol{W}_x^p\boldsymbol{W}_x^{pT} = \mathbf{I}$), the noise statistics will remain invariant in the new basis,

$$\begin{aligned} \boldsymbol{w}_v^p &= \boldsymbol{W}_x^p \mathrm{v} \\ \mathrm{K}_{\boldsymbol{w}_v^p} &= \mathrm{E}\{\boldsymbol{w}_v^p \boldsymbol{w}_v^{pT}\} = \mathrm{E}\{\boldsymbol{W}_x^p \mathrm{v}\mathrm{v}^T \boldsymbol{W}_x^{pT}\} \\ &= \boldsymbol{W}_x^p \mathrm{K}_{\mathrm{v}} \boldsymbol{W}_x^{pT} = \sigma^2 \boldsymbol{W}_x^p \boldsymbol{W}_x^{pT} = \sigma^2 \mathbf{I}. \qquad (4.6) \end{aligned}$$

By applying a transformation to the observed signal, as shown in Equation (4.5), the problem is identical to the original denoising problem. In this chapter, we will show that the properties of wavelet packets can be exploited to obtain significant reductions in the level of noise, while at the same time preserving the quality of the underlying signal. A fundamental step in the denoising process is to remove noisy coefficients, accomplished by discarding coefficients below a given threshold. Two common thresholding strategies will be discussed in Section 4.1. Another important step in the denoising process is to find a "best" representation of a signal. We will address two different goals that define the meaning of "best". One goal is to find the most compressed representation and is discussed in Section 4.2, and the other goal is to minimize the reconstruction error and is discussed in Section 4.3. In Section 4.4, we provide some results and compare the two methods proposed in this chapter.



(a) (b)

Figure 4.2: Two prominent thresholding strategies. (a) Hard thresholding. (b) Soft thresholding.

4.1 Thresholding Techniques

Choosing a "good" thresholding strategy is an important factor in the denoising process. There are many possible strategies for thresholding "undesired" small coefficients, but two of the simplest and most prominent methods are shown in Figure 4.2. Hard thresholding shown in Figure 4.2(a) removes coefficients with magnitude less than or equal to T and retains coefficients with magnitude greater than T. Soft thresholding shown in Figure 4.2(b) not only removes the small coefficients, but also shrinks the magnitude of the large coefficients by T.

To see how soft thresholding works, recall that the wavelet packet coefficients of the observed signal are partitioned into signal plus noise coefficients,

$$\boldsymbol{w}_x^p = \boldsymbol{w}_s^p + \boldsymbol{w}_v^p.$$

The underlying signal can be perfectly reconstructed if the noise is subtracted exactly, or

$$\hat{\mathrm{s}} \;=\; \mathrm{s} = (\boldsymbol{W}_x^p)^{-1}(\boldsymbol{w}_x^p - \boldsymbol{w}_v^p)\,. \tag{4.7}$$

This method, however, is unreasonable since $\boldsymbol{w}_v^p$ is unknown. A reasonable alternative is to estimate the value of each of the coefficients $\{w_{v_i}^p\}$ by some value T. The sign of this estimate is assumed to be the same as the noisy coefficients $\{w_{x_i}^p\}$, resulting in the following estimation strategy for each signal coefficient,

$$\hat{w}_{s_i}^p \;=\; w_{x_i}^p - T\mathrm{sgn}(w_{x_i}^p) \quad \text{where} \tag{4.8}$$

$$\mathrm{sgn}(w_{x_i}^p) \;=\; \begin{cases} 1 & w_{x_i}^p \geq 0 \\ -1 & w_{x_i}^p < 0 \end{cases}\,. \tag{4.9}$$

This process, therefore, shrinks the magnitude of the noisy coefficients $\{w_{x_i}^p\}$ by an amount T. We assume, however, that the amount of shrinkage is never sufficient to change the sign of the noisy coefficient, and as a result, we set all coefficients with magnitude less than T to zero. The estimation process described above is exactly the soft thresholding procedure shown in Figure 4.2(b) and expressed as,

$$\hat{w}_{s_i}^p \;=\; \mathcal{A}_S(w_{x_i}^p) = \begin{cases} w_{x_i}^p - T & w_{x_i}^p > T \\ 0 & \left|w_{x_i}^p\right| \leq T \\ w_{x_i}^p + T & w_{x_i}^p < -T \end{cases}\,. \tag{4.10}$$

An alternative approach is to assume that large coefficients (*i.e.* above the threshold T) are not affected by the noise. This is not an unreasonable assumption since the larger coefficients are percentage-wise less-affected by the noise than the smaller coefficients. This type of estimation is the hard thresholding strategy given in Figure 4.2(a) and expressed as,

$$\hat{w}^p_{s_i} \quad = \quad \mathcal{A}_H(w^p_{x_i}) = \begin{cases} w^p_{x_i} & w^p_{x_i} > T \\ 0 & \left|w^p_{x_i}\right| \leq T \\ w^p_{x_i} & w^p_{x_i} < -T \end{cases} . \tag{4.11}$$

This technique is sometimes more advantageous than soft thresholding (especially with small signal-to-noise ratios) because it does not remove as much energy from the signal (*i.e.* reconstructions using soft thresholding tend to have amplitudes smaller than the original signal). For the analyses considered here, we use the hard thresholding strategy for simplicity, realizing that the results can be extended to other thresholding methods.

Both thresholding methods shown in Figure 4.2 discard coefficients at a level T, and therefore, choosing a "good" value for the threshold is an important issue to address. The goal in thresholding is to exclude coefficients which are potentially purely noise. This equivalently ensures that coefficients above the threshold will contain at least some signal information. In the denoising problem considered here, the noise coefficients, $\{w^p_{v_i}\}$, are N independent Gaussian random variables with variance σ^2. It can be shown that the supremum of $\{|w^p_{v_i}|^2\}_{1\leq i\leq N}$ is given by $2\sigma^2 \log N$ [15]. As a result, we choose the threshold

$$T = \sqrt{2\sigma^2 \log N}, \tag{4.12}$$

so that

$$\sup_i \mid < \mathbf{v}, \boldsymbol{W}^p_{x_i} > \mid \leq T \quad \textit{almost surely (a.s.)}, \tag{4.13}$$

and this asymptotically guarantees that all coefficients which are purely noise will be removed. To ensure that no coefficients below T contain important signal information, we again have to find a signal representation with very few large magnitude coefficients. This implies that finding the best compressed representation of a signal is a valid approach to denoising, and the search for this type of representation is the topic of the next section.

4.2 Compression-Based Denoising

The main goal of Chapter 3 was to determine appropriate cost functions to find parsimonious signal representations. These functions will prove to be extremely useful here. The wavelet packet transformation $\boldsymbol{W}_x^p$ yields an equivalent denoising problem in form (*i.e.* $\boldsymbol{w}_x^p = \boldsymbol{w}_s^p + \boldsymbol{w}_v^p$) as the original problem, but now, the properties of wavelet packets can be exploited. An important property of wavelet packets is that they can be designed to efficiently represent signals in a given class of functions. The set of coefficients $\{w_{x_i}^p\}$ for all $p \in \mathcal{P}$ will typically consist of a few large magnitude coefficients and many small magnitude coefficients, depending on the type of wavelet chosen. Wavelet packets, therefore, provide some compression properties at the onset, and then, the cost functions from Chapter 3 can be used to find an even more compressed representation via the best basis search algorithm discussed in Section 3.4.

The resulting best basis will then place most of the signal information in the largest coefficients, and since the smaller coefficients contribute less information, they can be discarded with little change to the original signal. At the same time, discarding small noisy coefficients tends to improve the signal quality, and this is the essence of the thresholding idea presented in the previous section. Equation (4.6) shows that the noise statistics are invariant under an orthogonal transformation such as the wavelet packet decomposition. This means that the noise coefficients are spread according to the normal distribution, and thus, their effect on the underlying signal coefficients is statistically identical. Since the underlying signal coefficients tend to be more concentrated, thresholding the smaller coefficients, improves the signal quality by primarily removing noise. This reasoning shows that compression and denoising are complementary goals.

The procedure for denoising was previously shown in Figure 4.1. For the goal of compression, the box labeled best basis search will use the best basis algorithm with the cost functions given in Section 3.3. To see the effectiveness of this procedure, we consider an example. The original signal, shown in Figure 4.3(a), consists of sinusoids with two discontinuities. The noisy versions of the underlying signal are shown in Figures 4.3(b) and (c) with signal-to-noise ratios of 5 dB and −5 dB respectively, where SNR is a measure based on the energy of the signal and the noise variance, or

$$\mathrm{SNR} \;=\; 10\log_{10}\left(\frac{\sum_{m=1}^{N}|s(m)|^2}{\sigma^2}\right). \tag{4.14}$$

Reconstructions of the noisy signals from Figures 4.3(b) and (c) are shown in Figures 4.4 and 4.5 respectively for four different cost functions. All of the reconstructions are fairly similar for a given SNR. In Figure 4.4, the "log of energy" cost function gives the poorest reconstruction by smoothing-out the discontinuity on the right. The results in Figure 4.5 are consistently worse but not unreasonable given the level of noise. More reconstructions are provided in Section 4.4 along with some performance measures. While compression is a reasonable goal and provides good results in the denoising problem, it is not an optimal strategy with respect to the thresholding rule. In the next section, we look for a cost function that will yield reconstructions that are optimal in terms of minimizing the mean-squared error.

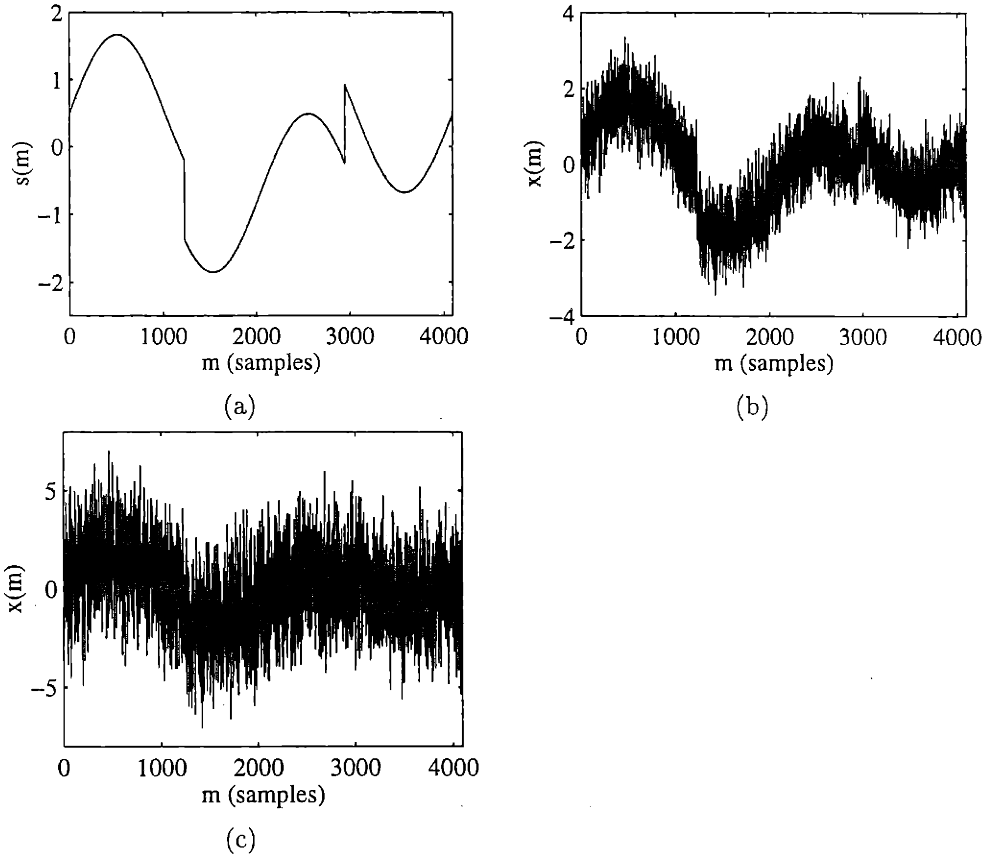


Figure 4.3: (a) Underlying signal $\{s(m)\}$ called HeaviSine. (b) Noisy observations $\{x(m)\}$ with SNR = 5 dB. (c) Noisy observations $\{x(m)\}$ with SNR = −5 dB.

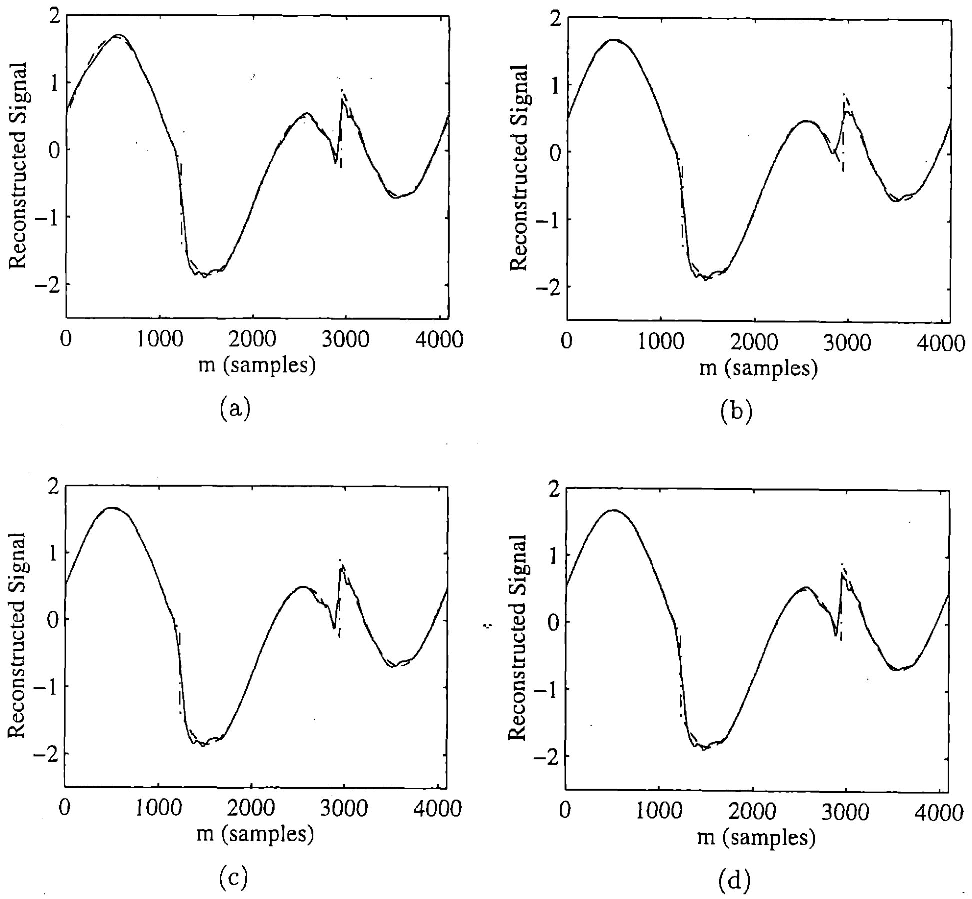
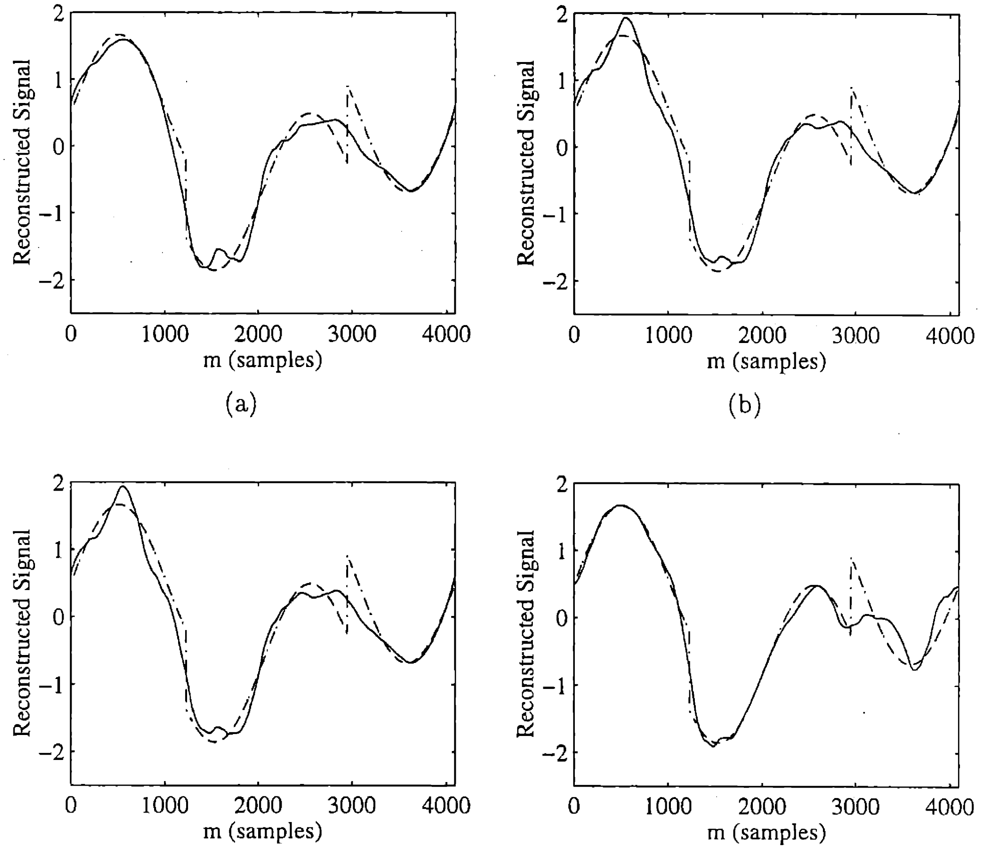


Figure 4.4: Reconstructions of the noisy signal shown in Figure 4.3(b) for four different cost functions. (a) Entropy. (b) Log of energy. (c) ℓ^1 norm. (d) Number of coefficients above $T = 0.55$.



(a)

(b)

(c)

(d)

Figure 4.5: Reconstructions of the noisy signal shown in Figure 4.3(c) for four different cost functions. (a) Entropy. (b) Log of energy. (c) ℓ^1 norm. (d) Number of coefficients above $T = 0.55$.

4.3 Minimal Reconstruction Error-Based Denoising

In this section, we propose a denoising scheme that minimizes the reconstruction error of the underlying signal, following the methodology of [16]. Specifically, we seek an appropriate additive functional $\mathcal{C}(\cdot)$ that can be used to minimize the mean-squared error $\mathrm{E}\{||\mathbf{e}||^2\}$ of the noise removal algorithm, where the error $\mathbf{e}$ is defined to be $\mathbf{s} - \hat{\mathbf{s}}$. To do this effectively, the total additive cost, $\sum_i \mathcal{C}(w^p_{x_i})$, must approximate the mean-squared error. In Section 4.3.1, we follow a simple-minded approach to determine the risk incurred by the thresholding procedure. For expediency, we restrict the thresholding rule to hard thresholding, previously shown in Figure 4.2(a), and defined in Equation (4.11). This approach, however, leads to a biased estimator of the risk. In Section 4.3.2, we show that an unbiased estimator exists by calling upon the Stein unbiased risk estimator, and in Section 4.3.3, we compare the two risks by evaluating the bias term.

4.3.1 Biased Risk Estimator

In this section, we follow a simple-minded approach to determine the risk incurred by the hard thresholding rule defined in Equation (4.11), and we ultimately seek an additive functional that will estimate the reconstruction error. We restrict our study to white Gaussian noise in this case, since the threshold T was chosen using this assumption. To determine the risk, we compute the expected loss due to thresholding at a level T. Hard thresholding has an associated quadratic loss which depends on T

and the underlying signal coefficient w_{s_i}, or

$$\mathcal{L}\{\mathcal{A}_H(w_{x_i}), w_{s_i}, T\} = \left(w_{s_i} - \mathcal{A}_H(w_{x_i})\right)^2 . \tag{4.15}$$

When applied to the wavelet coefficients in a basis $\mathcal{B}^p = \{\boldsymbol{W}^p_{x_i}\}$, the mean value of the loss is the estimation error or risk in $||\mathrm{s} - \hat{\mathrm{s}}||$, or

$$\begin{aligned} \mathrm{E}\left\{\mathcal{L}\left\{\mathcal{A}_H(\boldsymbol{w}^p_x), \mathrm{s}, T\right\}\right\} &= \mathcal{R}(\mathrm{s}, T) \\ &= \mathrm{E}\left\{||\mathrm{s} - \boldsymbol{W}^p_x \mathcal{A}_H(\boldsymbol{w}^p_x)||^2\right\} . \end{aligned} \tag{4.16}$$

Let $\{s(m)\}$ be decomposed into the same wavelet packet basis as the noisy signal $\{x(m)\}$ with coefficients $\{w^p_{s_i} =< \mathrm{s}, \boldsymbol{W}^p_{x_i} >\}$. The corresponding vector of signal coefficients is represented by $\boldsymbol{w}^p_s$ and ordered in a manner equivalent to $\boldsymbol{w}^p_x$. By decomposing $\{s(m)\}$ into the same basis as $\{x(m)\}$, we have partitioned each of the coefficients $\{w^p_{x_i}\}$ into signal and noise components (*i.e.* $w^p_{x_i} = w^p_{s_i} + w^p_{v_i}$). Since we only consider orthogonal bases here, the risk can then be expressed in terms of the basis coefficients, or

$$\begin{aligned} \mathcal{R}(\mathrm{s}, T) &= \mathrm{E}\{||\boldsymbol{W}^p_x \boldsymbol{w}^p_s - \boldsymbol{W}^p_x \mathcal{A}_H(\boldsymbol{w}^p_x)||^2\} \\ &= \sum_{i=1}^{N} \mathrm{E}\left\{\left|w^p_{s_i} - \mathcal{A}_H(w^p_{x_i})\right|^2\right\} . \end{aligned} \tag{4.17}$$

In order to define an estimator, we must analyze two specific cases:

Case 1: If $|w^p_{x_i}|^2 \leq T^2$ with the hard thresholding strategy, the coefficient $w^p_{x_i}$ is set to zero. This contributes the value of $|w^p_{s_i}|^2$ to the total risk. Since

$$\mathrm{E}\left\{\left|w^p_{x_i}\right|^2\right\} = \left|w^p_{s_i}\right|^2 + \sigma^2, \tag{4.18}$$

we estimate $\left|w^p_{s_i}\right|^2$ by $\left|w^p_{x_i}\right|^2 - \sigma^2$.

Case 2: If $|w^p_{x_i}|^2 > T^2$, the coefficient $w^p_{x_i}$ is left unchanged, yielding a mean-square error that is on average equal to the noise variance σ^2.

The total approximation error can thus be estimated by

$$\mathcal{R}_B(\mathrm{s}, T) = \sum_{i=1}^{N} \mathcal{C}\left(\left|w^p_{x_i}\right|^2\right), \tag{4.19}$$

where

$$\mathcal{C}(u) = \begin{cases} u - \sigma^2 & \text{if } u \le T^2 \\ \sigma^2 & \text{if } u > T^2 \end{cases}. \tag{4.20}$$

Examining Equations (4.19) and (4.20), we see that $\mathcal{R}_B(\mathrm{s}, T)$ is an additive cost function, and as a result, we can use this estimator to search for the basis which minimizes $\mathcal{R}_B(\mathrm{s}, T)$ among a collection $\{\mathcal{B}^p\}_{p\in\mathcal{P}}$ of orthonormal bases. We use the symbol $\mathcal{R}_B(\mathrm{s}, T)$ to denote that this estimator is biased, a fact which will be shown in the following section.

4.3.2 Unbiased Risk Estimator

In the following theorem, we show that the estimator $\mathcal{R}_B(\mathrm{s}, T)$ is suboptimal in the sense that it is biased. We proceed to compute the true risk $\mathcal{R}(\mathrm{s}, T) = \mathrm{E}\{||\mathrm{s} - \hat{\mathrm{s}}||^2\}$ and derive the bias using the Stein unbiased risk estimator [17]. Section 4.3.3 will show that this bias term has a minimal effect on the optimality of the search, if a threshold T is judiciously chosen.

Theorem 4 *Let $\{\boldsymbol{W}^p_{x_i}\}_{1\leq i\leq N}$ be an orthonormal basis of the observation space. If the coefficients $\{v(m)\}$ are zero mean, uncorrelated Gaussian random variables with variance σ^2, the bias of the estimator $\mathcal{R}_B(\mathrm{s},T)$ with respect to $\mathcal{R}(\mathrm{s},T)$ is*

$$\begin{aligned} \mu &= \mathcal{R}(\mathrm{s},T) - \mathrm{E}\{\mathcal{R}_B(\mathrm{s},T)\} \\ &= 2T\sigma^2\sum_{i=1}^{N}\left[\phi(T- <\mathrm{s},\boldsymbol{W}^p_{x_i}>) + \phi(-T- <\mathrm{s},\boldsymbol{W}^p_{x_i}>)\right] \end{aligned} \qquad (4.21)$$

with

$$\phi(u) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{u^2}{2\sigma^2}}.$$

Proof: We drop the superscript p for clarity. Define

$$\begin{aligned} \mathcal{A}_H(w_{x_i}) &= w_{x_i}\mathcal{I}_{\{|w_{x_i}|>T\}} \\ g_T(w_{x_i}) &= -w_{x_i}\mathcal{I}_{\{|w_{x_i}|\leq T\}}, \end{aligned}$$

where $\mathcal{I}_{\{\cdot\}}$ is an indicator function constrained by its argument and where the noisy coefficient w_{x_i} has a normal distribution, $w_{x_i} \sim \mathcal{N}\left(w_{s_i},\sigma^2\right)$. We can then write

$$\mathcal{A}_H(w_{x_i}) = w_{x_i} + g_T(w_{x_i}),$$

to obtain the following,

$$\begin{aligned} &\mathrm{E}\left\{\sum_{i=1}^{N}[\mathcal{A}_H(w_{x_i}) - w_{s_i}]^2\right\} \\ &= \sum_{i=1}^{N}\mathrm{E}\{[(w_{x_i} - w_{s_i}) + g_T(w_{x_i})]^2\} \\ &= \sum_{i=1}^{N}(\mathrm{E}\{(w_{v_i})^2\} + 2\mathrm{E}\{w_{v_i}g_T(w_{x_i})\} + \mathrm{E}\{g_T^2(w_{x_i})\}). \end{aligned} \qquad (4.22)$$

Using the property described in [17],

$$\begin{aligned}\mathrm{E}\{w_{v_i} g_T(w_{x_i})\} &= \int w_{v_i} g_T(w_{v_i} + w_{s_i})\phi(w_{v_i})dw_{v_i} \\ &= -\sigma^2 \int g_T(w_{v_i} + w_{s_i})\phi'(w_{v_i})dw_{v_i} \\ &= \sigma^2 \int g'_T(w_{v_i} + w_{s_i})\phi(w_{v_i})dw_{v_i},\end{aligned}$$

where "′" denotes appropriate differentiation. Calling upon derivatives in the generalized sense, one can write,

$$\frac{d}{dw_{x_i}}\mathcal{I}_{\{|w_{x_i}|\leq T\}} = \delta(w_{x_i} + T) - \delta(w_{x_i} - T),$$

with $\delta(\cdot)$ denoting the Dirac impulse. We can then use it to derive

$$\int g'_T(w_{v_i} + w_{s_i})\phi(w_{v_i})dw_{v_i} = \\ -\int \mathcal{I}_{\{|w_{x_i}|\leq T\}}\phi(w_{v_i})dw_{v_i} + T\left(\phi(T - w_{s_i}) + \phi(-T - w_{s_i})\right).$$

Substituting the above expressions back into Equation (4.22), we obtain,

$$\mathrm{E}\left\{\sum_{i=1}^{N}[\mathcal{A}_H(w_{x_i}) - w_{s_i}]^2\right\} = \mathrm{E}\left\{\mathcal{R}_B(\mathrm{s}, T)\right\} + 2T\sigma^2\sum_{i=1}^{N}\left[\phi(T - w_{s_i}) + \phi(-T - w_{s_i})\right].$$

■

This theorem proves that the expected value of the suboptimal estimator $\mathcal{R}_B(\mathrm{s}, T)$ is a lower bound on the mean-squared error. The estimator is biased because we have assumed that the magnitude of the signal components are always above T in Equations (4.19) and (4.20). Since we did not account for the errors due to an erroneous decision, we see that a coefficient composed of both signal and noise components may be present below the threshold T.

4.3.3 Risk Comparison

The risk associated with the simple thresholding rule $\mathcal{A}_H(\cdot)$ is clearly different from the optimal or unbiased risk, and the significance of this difference will be dependent upon T and $\{s(m)\}$. Heuristically, this difference is due to the naive and perhaps optimistic rule which attributes any coefficient below T to noise and any coefficient above T to the underlying signal. In short, a noisy signal coefficient can be less than or equal to T depending on its local energy and how it is modified by the noise. Therefore, the nature of the underlying signal in the presence of noise at a level around the threshold T is very relevant. Recall that T is solely determined by the noise variance and the observation interval N. A Bayesian-like approach would lead us to assume some prior knowledge about $\{s(m)\}$, in order to evaluate the significance of the bias in the "suboptimal" decision rule.

To obtain a more quantitative characterization of the bias term, we assign a prior probability density $f(w^p_{s_i})$ to the signal coefficients. Using this information, we can show that the bias term is strongly dependent on the statistical nature of the signal. This analysis will also shed some light on the search for the optimal threshold T as the signal statistics vary.

Proposition 1 *Assume a probability density $f(w^p_{s_i})$ of the form*

$$f(w^p_{s_i}) \quad = \quad \epsilon f_1(w^p_{s_i}) + (1-\epsilon) f_2(w^p_{s_i}),$$

where $f_1(w^p_{s_i})$ is absolutely continuous and $f_2(w^p_{s_i})$ has a finite or countably infinite number of singularities (i.e. $f_2(w^p_{s_i}) = \sum_{k=0}^{\infty} p_k \delta(w^p_{s_i} - \nu_k)$). *The expected value of the* *bias term μ is given by,*

$$\begin{aligned} \mathrm{E}_s\{\mu\} &= 2T\sigma^2 N \left[\epsilon \sum_{j=0}^{\infty} \frac{\sigma^{2j}}{(2j)!} 1 \cdot 3 \cdots (2j-1) \left[f_1^{(2j)}(T) + f_1^{(2j)}(-T) \right] \right. \\ & \left. +(1-\epsilon) \sum_{k=0}^{\infty} p_k \left[\phi(T-\nu_k) + \phi(-T-\nu_k) \right] \right]. \qquad (4.23) \end{aligned}$$

Proof: We assume that the wavelet coefficients of the underlying signal are identically distributed. The expected value of the bias term is then given by,

$$\mathrm{E}_s\{\mu\} = 2T\sigma^2 \sum_{i=1}^{N} \int \left[\phi(T - w^p_{s_i}) + \phi(-T - w^p_{s_i}) \right] f(w^p_{s_i}) dw^p_{s_i}. \qquad (4.24)$$

We only consider densities of the following form, where $f(x)$ is the distribution for any $w^p_{s_i}$,

$$f(x) = \epsilon f_1(x) + (1-\epsilon) f_2(x).$$

In particular, $f_1(x)$ is absolutely continuous, and $f_2(x)$ has a finite or countably infinite number of singularities. Since $f_1(x)$ is analytic, it can be represented by a Taylor Series expansion, and $f_2(x)$ can be represented by

$$\begin{aligned} f_2(x) &= \sum_{k=0}^{\infty} p_k \delta(x - \nu_k) \\ \text{where} \quad & \sum_{k=0}^{\infty} p_k = 1. \end{aligned}$$

As a result, $\mathrm{E}_s\{\mu\}$ can be separated into two expressions, one that is dependent on

$f_1(x)$ and the other dependent on $f_2(x)$, or

$$\begin{aligned} \mathrm{E}_s\{\mu\} &= 2T\sigma^2 \sum_{i=1}^{N} \left[\epsilon \int \left[\phi(T - w^p_{s_i}) + \phi(-T - w^p_{s_i})\right] f_1(w^p_{s_i}) dw^p_{s_i} \right. \\ & \left. +(1-\epsilon) \int \left[\phi(T - w^p_{s_i}) + \phi(-T - w^p_{s_i})\right] f_2(w^p_{s_i}) dw^p_{s_i} \right]. \qquad (4.25) \end{aligned}$$

Given the similarity of the two terms $\phi(\cdot)$ in the first integral of Equation (4.25), we only evaluate the first term. Letting $\tau_i = T - w^p_{s_i}$, we obtain the Taylor series expansion of $f_1(T - \tau_i)$ around T,

$$\int \phi(\tau_i) f_1(T - \tau_i) d\tau_i = \sum_{j=0}^{\infty} \int \frac{(-\tau_i)^j}{j!} f_1^{(j)}(T) \phi(\tau_i) d\tau_i. \qquad (4.26)$$

This last expression is the sum of scaled moments of the Gaussian function, which are known to be [18]

$$m_j = \begin{cases} 1 \cdot 3 \cdots (j-1)\sigma^j & j \text{ even}, \\ 0 & j \text{ odd}. \end{cases} \qquad (4.27)$$

The other term in the first integral of Equation (4.25) leads to a similar expression. Evaluating the second integral for an arbitrary $x = w^p_{s_i}$, gives

$$\begin{aligned} & \int \left[\phi(T - x) + \phi(-T - x)\right] f_2(x) dx \\ &= \int \left[\phi(T - x) + \phi(-T - x)\right] \left[\sum_{k=0}^{\infty} p_k \delta(x - \nu_k) \right] dx \\ &= \int \sum_{k=0}^{\infty} p_k \left[\phi(T - \nu_k)\delta(x - \nu_k) + \phi(-T - \nu_k)\delta(x - \nu_k)\right] dx \\ &= \sum_{k=0}^{\infty} p_k \left[\phi(T - \nu_k) + \phi(-T - \nu_k)\right]. \qquad (4.28) \end{aligned}$$

Combining the results of Equations (4.25)–(4.28), we obtain an expression which proves the proposition. ∎

Consequently, Equation (4.23) shows that the bias term of the suboptimal risk is strongly dependent on T. This implies that the overall minimum of the true risk will be dependent on the *a priori* probability density $f(\cdot)$. The mode of the $\mathrm{E}_s(\mu)$ will indeed determine the extremum point, and when combined with $\mathcal{R}_B(\mathbf{s}, T)$ will result *a posteriorily* in a minimum at a corresponding "optimal" threshold T. Note also that the bias term in Equation (4.21) assumes prior knowledge of the signal coefficients, and as a result, no true unbiased estimator can be achieved in practice. This difficulty, however, can be partially lifted by picking the Maximum Likelihood Estimate (in this case, the noisy coefficient) to obtain an upper bound on the bias.

4.4 Results

4.4.1 A One-dimensional Example

In this example, we analyze the two risks $\mathcal{R}_B(\mathbf{s}, T)$ and $\mathcal{R}(\mathbf{s}, T)$, which we subsequently refer to as the biased and unbiased costs, respectively. The entropy cost function first introduced in Section 3.3 and described in [19] will be used for comparison. In this analysis, white Gaussian noise with variance σ^2 is added to a known signal at a specified SNR level, using the definition of SNR defined earlier in Equation (4.14). Using one of the three costs under consideration, a best basis is obtained for the noisy signal by minimizing the associated cost in a dictionary of possible bases, and the thresholding rule defined in Equation (4.11) is applied to the coefficients before reconstruction. We focus on the performance of three synthetic signals: Ramp,

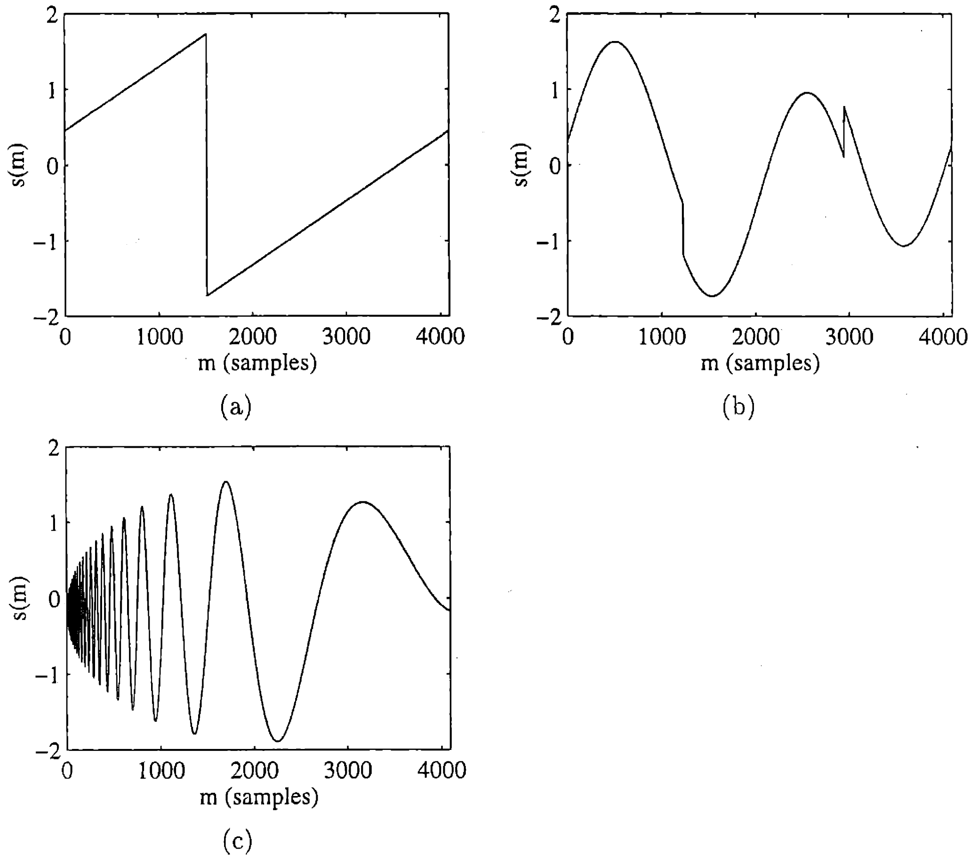


Figure 4.6: Synthetic signals used to illustrate the performances of the proposed cost functions. (a) Ramp Signal. (b) HeaviSine Signal. (c) Doppler Signal.

HeaviSine, and Doppler, shown in Figure 4.6.

To illustrate the performance quality of the prescribed denoising scheme, a sample reconstruction of the Ramp signal for each cost is shown in Figure 4.7 with an SNR level of 5 dB. While the noise is not completely removed and ringing is exhibited, the basic shape of the original signal is retained, and the discontinuity is preserved. While Figure 4.7 is adequate to give the "flavor" of a typical reconstruction, it does not provide a quantitative measure of performance. To compare the performances of the estimators, an average risk was computed for 500 different noise realizations at

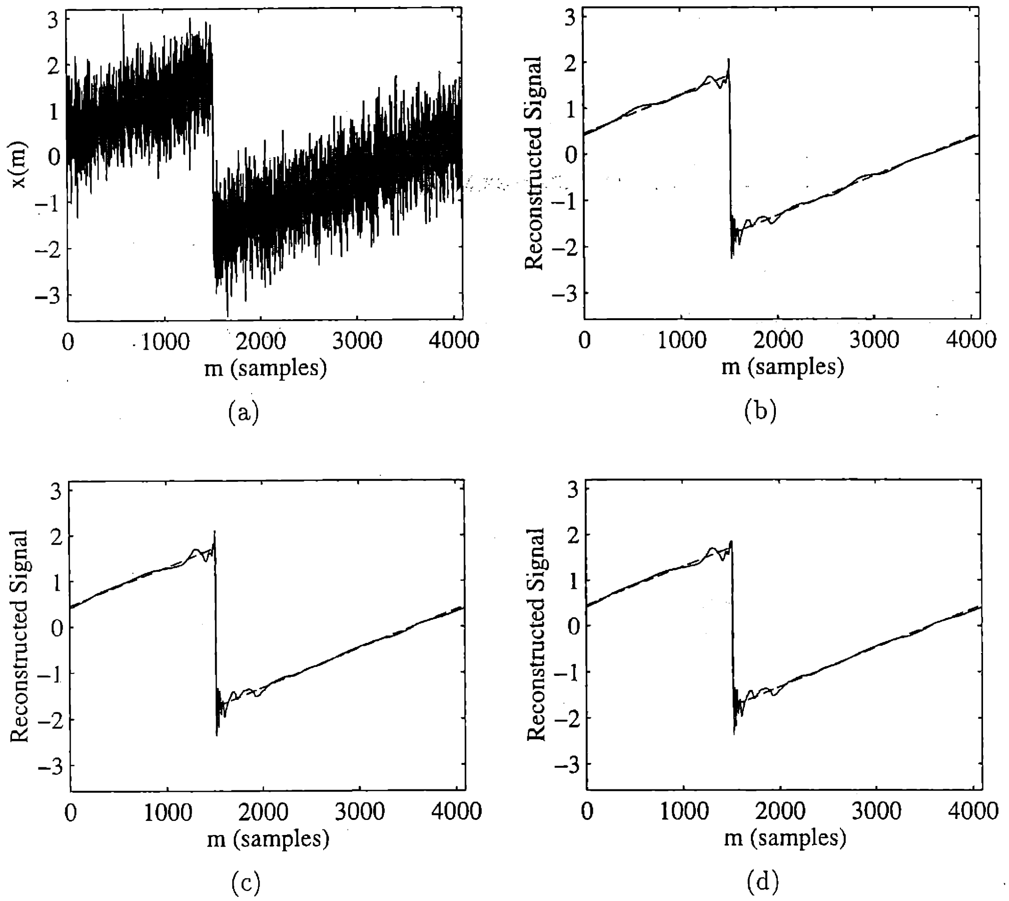


Figure 4.7: (a) Noisy Ramp signal at an SNR level of 5 dB. (b) Sample reconstruction with the entropy cost function. (c) Sample reconstruction with the biased cost function. (d) Sample reconstruction with the unbiased cost function.

100 SNR levels. We specifically computed

$$\bar{\mathcal{R}}(\mathrm{s}, T) = \frac{1}{M} \sum_{j=1}^{M} \left[\frac{||\mathrm{s} - \hat{\mathrm{s}}_j||^2}{||\mathrm{s}||^2} \right], \tag{4.29}$$

where j is the index of the realization number and M is the total number of realizations. Note that the risk is normalized by $||\mathrm{s}||^2$ to allow comparisons between the risks corresponding to signals with different energies. Figure 4.8 shows the results for the three signals considered here.

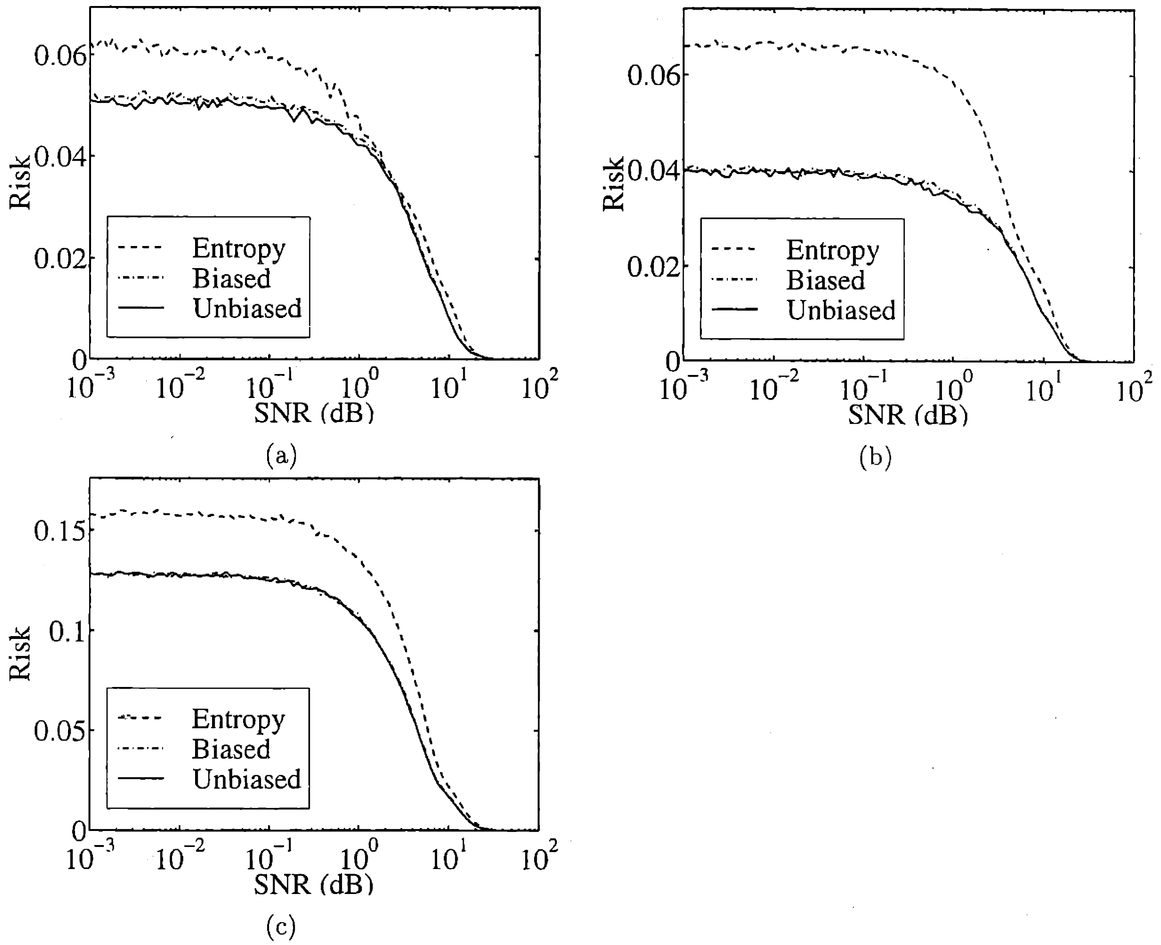


Figure 4.8: Performance curves for the Ramp, HeaviSine, and Doppler signals as a function of SNR. Each plot shows results for the three cost functions considered in this example. (a) Average risks for the Ramp signal. (b) Average risks for the HeaviSine signal. (c) Average risks for the Doppler signal.

The risk curves shown in Figure 4.8 are all fairly similar. As expected, the risks associated with the biased and unbiased costs are smaller than the risk associated with the entropy cost. This shows why one might prefer to use the biased and unbiased costs over a compression-based cost, like entropy. Both the biased and unbiased costs have a "built-in" measure of performance, namely the mean-squared error. By minimizing this cost, we know that we will obtain the "best" signal estimate associated with our performance measure. Compression-based costs, however, do not possess an inherent indicator of performance, and therefore, minimizing these costs do not directly imply any degree of quality.

Figure 4.8 also shows that the biased and unbiased risk results are almost indistinguishable, which indicates that the biased risk is a good approximation to the unbiased risk, in this case. In general, we prefer to use the biased risk, since it is much simpler to compute. Also, the true unbiased risk is not achievable in practice, since the bias term in Equation (4.21) directly assumes *a priori* knowledge of the signal coefficients. An upper bound on the bias term can, however, be obtained by picking the Maximum Likelihood Estimate (MLE). In this case, the MLE of the bias term is achieved by replacing the signal coefficient $w^p_{s_i}$ by the noisy signal coefficient $w^p_{x_i}$.

4.4.2 A Quasi-two-dimensional Example

To provide more results using the proposed denoising algorithms, we consider a quasi-two-dimensional example. For this example, an object is centered at a point (x_o, y_o) with a radius that varies with the angular value θ. For N equally spaced values of θ, we measure the distance from the origin to a point on the object contour corresponding to the value of θ, as shown in Figure 4.9. In this case, we assume that the discrete set of angles $\{\theta(m)\}$ are measured perfectly for all values of m, while the radial

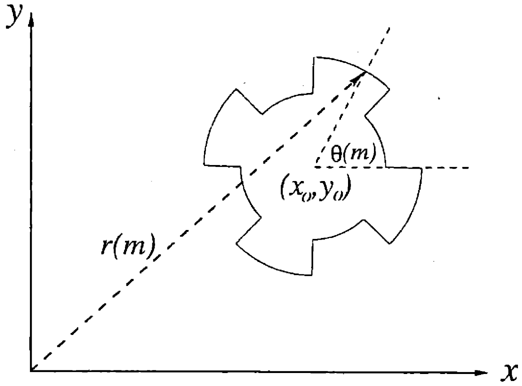


Figure 4.9: Method for obtaining the radial measurements $r(m)$.

measurements $\{r(m)\}$ cannot be determined precisely, or

$$\begin{aligned} x(m) &= r(m) + v(m) \\ \theta(m) &= \frac{2\pi m}{N}, \qquad m = 0, 1, \cdots, N-1 \end{aligned}$$

and $v(m) \sim \mathcal{N}(0, \sigma^2)$. As a result, we obtain the same denoising problem as before, where $\{x(m)\}$ is the noisy signal of interest, with now, the complete signal being viewed as a two-dimensional image with additive radial noise.

The same reconstruction scheme may be applied to this problem by processing the radial measurement. As an example, consider an object similar to the one shown in Figure 4.9 centered at $(x_o, y_o) = (0, 0)$ with an inner radius of 1 and an outer radius of 1.5. Figure 4.10 shows the noisy signal along with the three sample reconstructions for an SNR level of 5 dB. Note that the results using the biased and unbiased costs exhibit less smoothing around the sharp discontinuities in the radial measurement than the entropy cost function.

This concept can be extended to more complex scenarios, such as the recon-

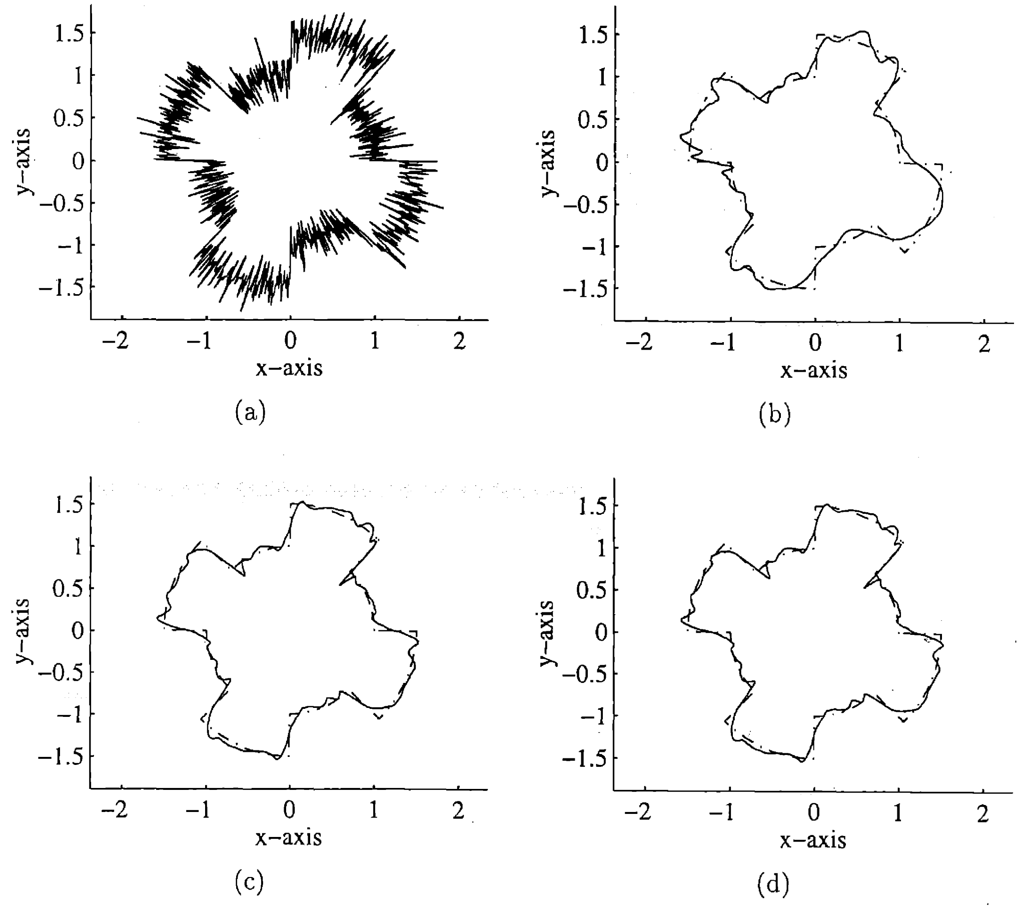


Figure 4.10: (a) Noisy radial measurement at an SNR level of 5 dB. (b) Sample reconstruction with the entropy cost function. (c) Sample reconstruction with the biased cost function. (d) Sample reconstruction with the unbiased cost function.

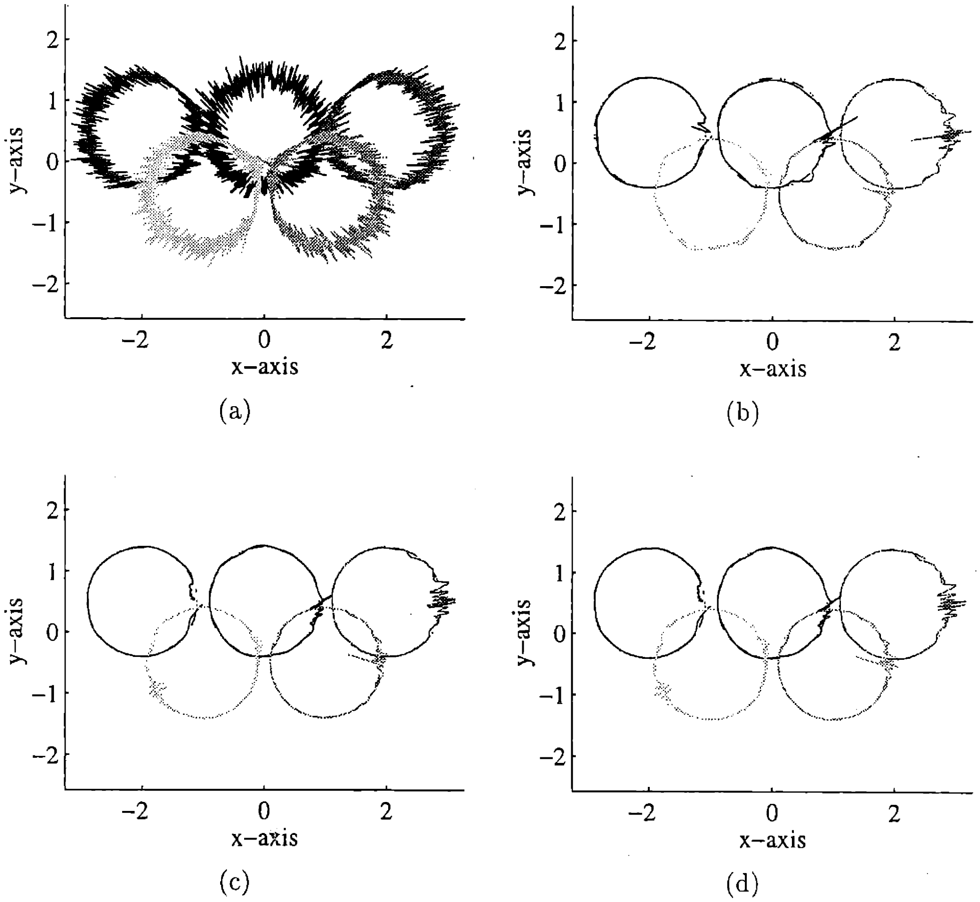


Figure 4.11: (a) Noisy radial measurement at an SNR level of 10 dB. (b) Sample reconstruction with the entropy cost function. (c) Sample reconstruction with the biased cost function. (d) Sample reconstruction with the unbiased cost function.

structions shown in Figure 4.11. In this example, there are five circles with different centers, and by concatenating the one-dimensional radial measurements from each circle, we obtain a signal that can be processed in the same way as before. While this example is contrived, there are variations on this theme that have real applications. For example, object profiles used for image recognition are often noisy. This inherent noise may be removed if an appropriate radial measure can be obtained for these objects.

Chapter 5

Applications to High-Resolution Radar

In this chapter, we show that some of the previously discussed denoising techniques may be applied to high-resolution radar (HRR) signals. In particular, we propose an algorithm that identifies the spatial orientation of targets, using one-dimensional radar returns. These returns essentially provide a "fingerprint" of a target by conveying important information about its overall dimension along with the configuration and shape of significant scatterers. Since HRR returns contain important information about a target, researchers have believed for many years that HRR data could be successfully used for automatic target recognition (ATR). No significant advances, however, have been made in this area due to the difficult nature of the problem.

The ATR problem is plagued by many factors which make it difficult to develop both efficient and robust algorithms. The noisy environment is one of the most significant problems which affects this detection and estimation problem. The radar returns not only contain the target of interest, but also noise added from background clutter such as the ground, trees, buildings, and other objects. Another factor that threatens the performance of ATR systems is the variability in the position and shape of the targets of interest. From scene to scene, targets are oriented in different spatial configurations that can significantly alter the radar returns. Even dents and other abrasions on a target's body can alter the surface reflectivity enough to affect the radar returns. All of these factors are essentially noise sources that introduce a variable of uncertainty in HRR signals.

An ATR system must therefore account for all of these factors. These systems, however, are typically so complex that they cannot operate in real time. In some of the earliest approaches to ATR, the authors in [20, 21] and later [22] attempt to implement simpler systems using correlation and nearest neighbor methods. These approaches are fast and efficient but often lead to large misclassification errors in noisy environments. Recent work by [23] has focused on Bayesian approaches to the ATR problem. These methods are certainly more robust than the simpler correlation algorithms, but they require impractical computation times. In designing an ATR system, we therefore search for a middle ground that will suggest an algorithm which is both robust and computationally efficient. Our approach is to develop a database of signals that represent N known targets. These signals are ideally picked to be highly correlated with the returns for a matching target and uncorrelated with returns from all other targets. In order to develop a robust ATR system, this database is designed to provide an overcomplete representation of a given set of returns.

In Section 5.1, we describe the high-resolution radar data that will be used to evaluate the performance of our algorithm. In Section 5.2, we present the general ATR problem and then specialize our study to the ground targets, which we consider in our experiments. In Section 5.3, we propose an algorithm that can be used for automatic target recognition. We first describe the methodology for building the database, and then, present an efficient search algorithm to identify the most likely candidate target based on a given radar return. In Section 5.4, we conclude with some results to show the performance of the proposed algorithm.

5.1 Description of High-Resolution Radar Data

We previously mentioned that background clutter is a significant problem associated with HRR data. Another serious problem, however, is the lack of consistent real data for evaluation purposes, and this scarcity of data makes it difficult to test algorithms in order to assess real world performance. To address this problem, Wright Laboratories and DEMACO, Inc. have collaborated to produce a modeling tool called Xpatch, used to simulate radar returns for a target model. We exclusively use synthetic data obtained from Xpatch in order to test the algorithm that we propose. Before introducing this tool, we discuss real HRR data and two models that have been proposed in [23].

5.1.1 Two Models

Radar systems have been used for many years as a method of remote surveillance. Traditional systems transmit a modulated waveform typically composed of a sequence of pulses, and an antenna is then used to "listen" for a return signal. "Significant" returns indicate that an object is present, and these returns are modified versions of the transmitted signal, varying according to the range and velocity of a target. For many years, researchers believed that shorter pulses were the only method for achieving improved range resolution of targets, but they later discovered that range resolution was also related to the frequency bandwidth. This led to "chirp" signals and other waveforms in which the carrier frequency of the transmitted signal was varied over a range of frequencies. These ideas and new technologies such as wide-bandwidth microwave components, high-speed digital processing, and digitally controlled frequency sources have made high-resolution radar possible [24].

High-resolution radar returns are useful for recognition purposes because they provide "fingerprints" of a target. The electromagnetic waves reflected from an object of interest are altered according to the reflectivity density of the object's surface. Important features of the target, therefore, manifest themselves as large peaks in the returned signal. The locations of these significant scatterers vary, however, as a target is illuminated from different angles. In addition, various surfaces of the target have different reflectivity densities, and reflections from these surfaces may add destructively at one location and constructively at other locations. The returns therefore tend to vary dramatically with the aspect angle, which makes the ATR problem extremely challenging.

Two models of HRR returns have been proposed by Jacobs, et.al., [23]. Using their notation, the radar return can be represented by the sum of an unknown signal and noise, or

$$r(t) \quad = \quad s(t;\theta,a) + w(t). \tag{5.1}$$

This problem is the same type considered in Chapter 4, but in this case, the signal $s(t;\theta,a)$ is dependent on the angle θ at which the target is illuminated and the target type a. The deterministic model which they propose assumes that $s(t;\theta,a)$ is deterministic for a fixed target and angle, while the stochastic model assumes that

$s(t;\theta,a)$ is a random process. For the deterministic model, a sequence of HRR returns is then given by the following system of equations,

$$r_k(t) \quad = \quad s(t;\theta_k,a)+w_k(t), \tag{5.2}$$

where k is the waveform index. The sequence of profiles $\{r_k(t)\}$ is therefore random because of the additive noise, but the underlying signal is known exactly once complete information about the aspect angle and target type is provided.

The stochastic model for HRR data represents the reflectivity density of a target by a glint plus diffusion component. At a given location ρ, the reflectivity density is given by,

$$c(\rho) \quad = \quad c_g(\rho)e^{j\psi(\rho)}+c_s(\rho). \tag{5.3}$$

The glint component $c_g(\rho)$ is a deterministic function of position and represents the reflective amplitude of the object's surface. The phase term $\psi(\rho)$ has a distribution that depends on the incident angle of the radar wave. The diffusion component $c_s(\rho)$ is modeled as a complex Gaussian random process that is spatially white, or

$$\mathrm{E}\{\mathrm{c_s}(\rho)\mathrm{c_s^*}(\rho')\} \quad = \quad S(\rho)\delta(\rho-\rho'). \tag{5.4}$$

This model and other similar stochastic models, however, result in more complicated algorithms. In this treatment of the problem, we use the deterministic model exclusively since it closely resembles the denoising problem previously studied.

5.1.2 The Xpatch Simulator

One problem with using HRR data for ATR is that real data is not widely available. We are therefore led to use synthetic data obtained through simulation. The modeling tool Xpatch has received recent attention as a valid alternative to real data. This computer package simulates the radar return by a Shooting and Bouncing Ray (SBR) technique. The user first creates a CAD model of a target, composed entirely of flat triangular facets. These facets provide an approximation to the continuous surface of a real object and designate how radar waves will bounce off the surface. Once a model is created, Xpatch simulates the radar return at different specified aspect angles.

Using the SBR technique, a grid of rays are focused on the target in a given direction, and the rays are traced as they strike various surfaces of the target. The user can specify different reflectivities for the target's surfaces, which affect the amplitude of the reflected rays. This tool can also be used to simulate both horizontal and vertical polarizations at the transmitter and the receiver. Figure 5.1 shows an example of an electromagnetic wave with vertical polarization. By convention, the polarization is determined by the direction of the electric field [25]. By choosing different polarizations, various features of the target may become accentuated. The user can also specify the number of bounces to count in each simulation, as well as many other options. To calculate the estimated radar return, Xpatch then performs a physical optics integration at the point where a ray exits the target. In the algorithm proposed here, we strictly use range profiles obtained from the Xpatch simulations through the inverse Fourier transform.

The data considered throughout was produced by Xpatch for a set of CAD models designed to provide realistic data to universities. These models are part of the University Research Initiative Synthetic Dataset (URISD) and are shown in

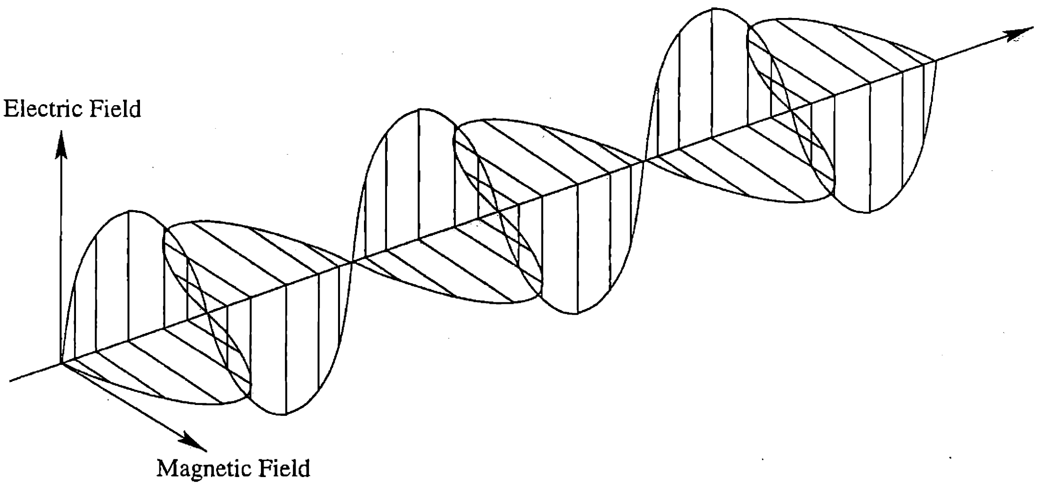


Figure 5.1: Representation of an electromagnetic wave.

Figure 5.2. The dataset includes HRR returns for three frequency bands: UHF, L, and X, but we only focus on the UHF data in the testing process. The dataset includes simulated returns at three different depression (or elevation) angles, 10°, 25°, and 40°, and at 1201 different azimuthal angles spaced 0.3° apart. Figure 5.3 shows the magnitude of four HRR returns obtained at different azimuthal angles for the model of the fire truck. The variation in both amplitude and shape is dramatic for large angular separation, as expected. These variations contribute significantly to the difficulty of the ATR problem, which we address in the next section.

5.2 Problem Statement

Using a sequence of HRR profiles, we wish to identify a target and estimate its position in real time. The position of a target is generally known, however, and modern radar systems are designed to track multiple targets effectively. If these assumptions are

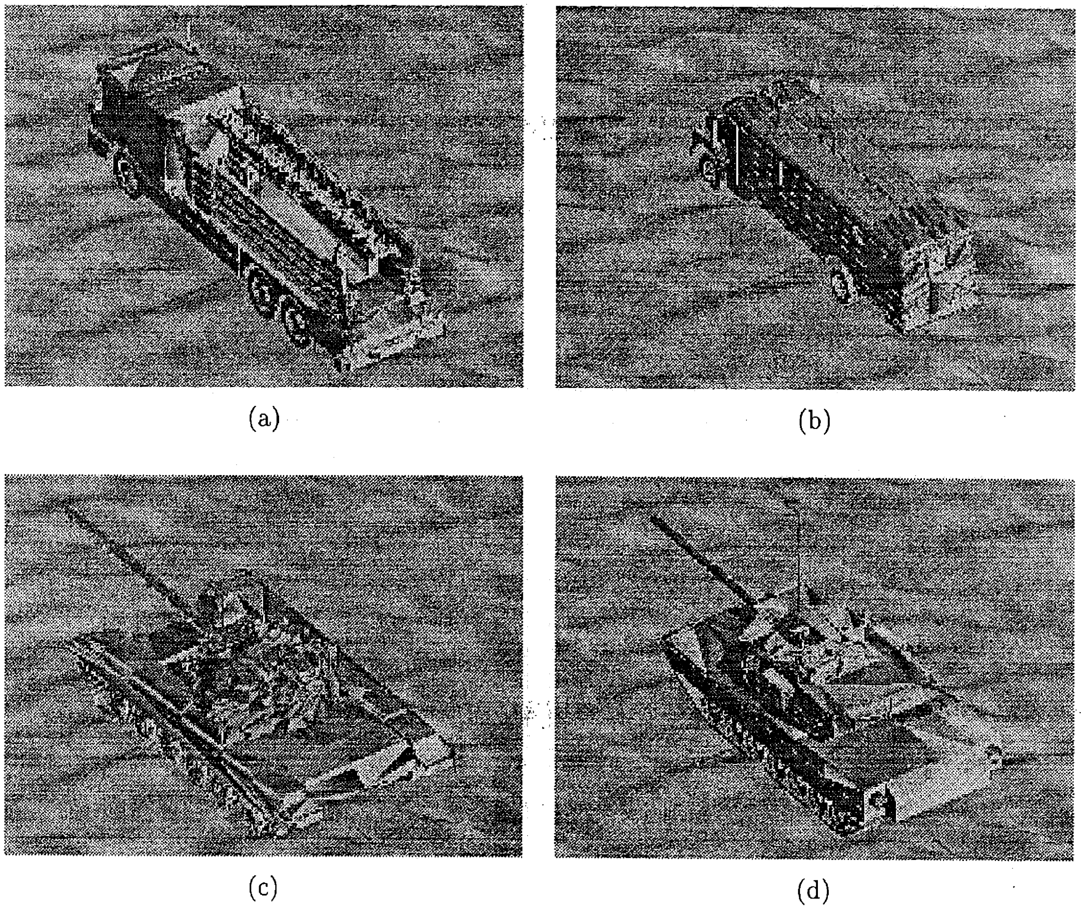
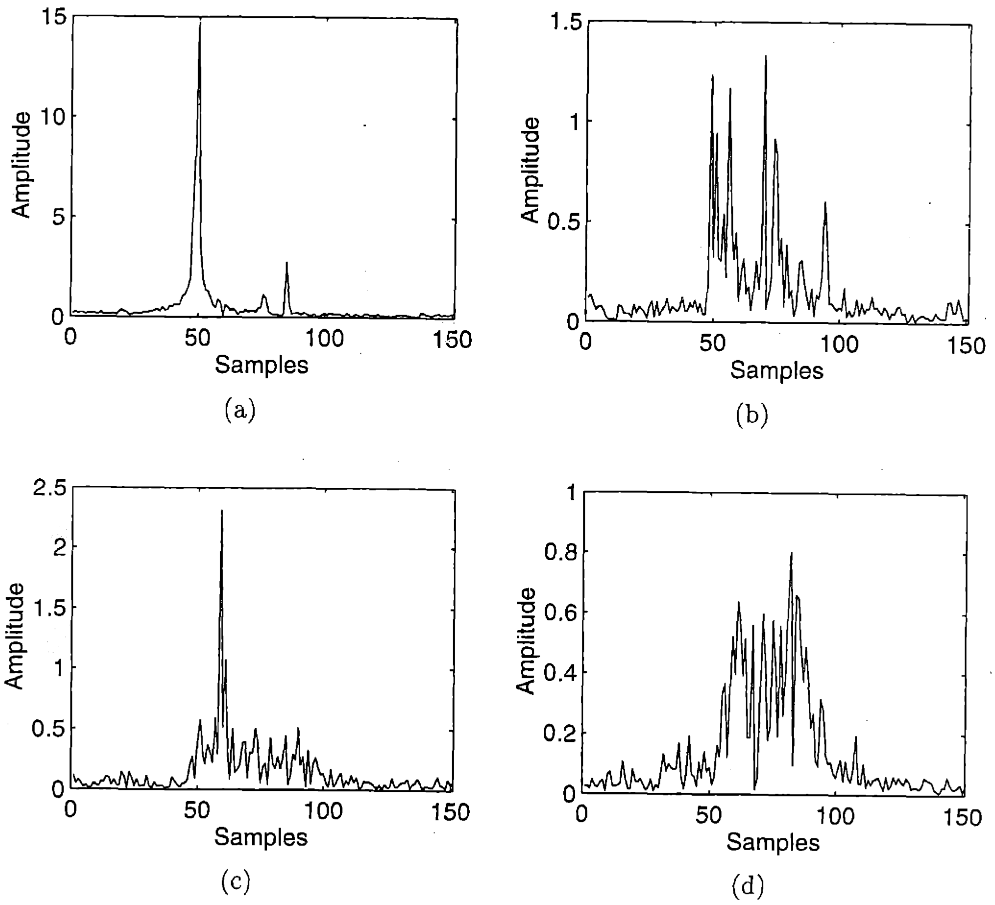


Figure 5.2: Models of the targets used in the dataset. (a) Fire truck (b) School bus (c) Tank 1 (d) Tank 2



(a)

(b)

(c)

(d)

Figure 5.3: Magnitude of HRR returns at different azimuthal angles for the fire truck at a depression angle of 10°. (a) 0° azimuth. (b) 15° azimuth. (c) 30° azimuth. (d) 45° azimuth.

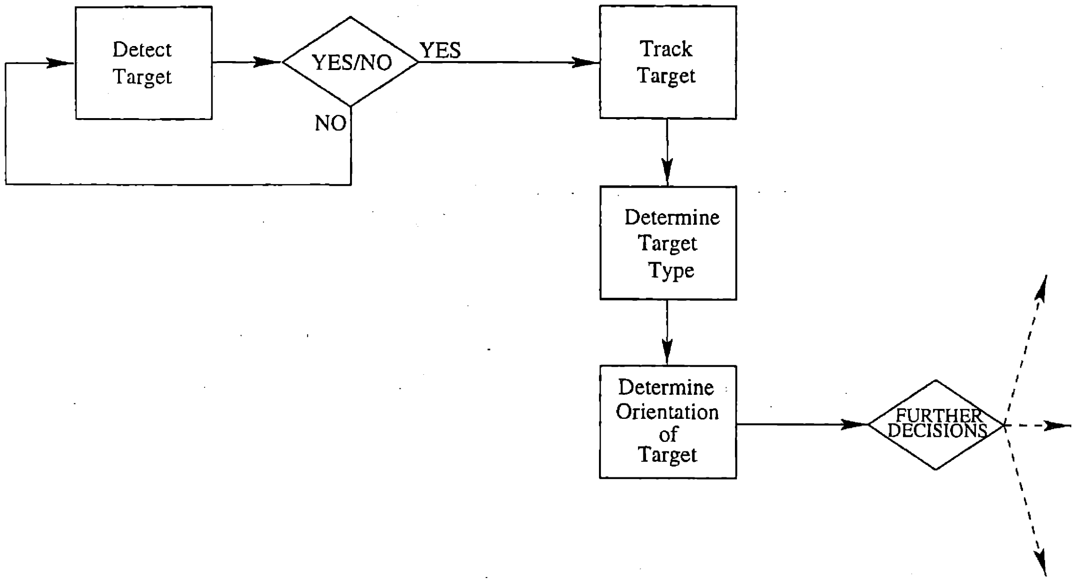


Figure 5.4: Detection and estimation procedure involved in the ATR problem.

satisfied, then we are assured that the HRR returns represent the target of interest and not some other object. The main goal in the ATR problem is then to determine the identity of a target and its orientation in space. Figure 5.4 shows the entire detection and estimation system, with the last two blocks being the object of our study herein. To estimate a target's orientation, both the azimuthal angle ϕ and depression angle θ must be determined as shown in Figure 5.5. The figure assumes a body-centered frame of reference for the target. Of the class of possible targets, we only consider ground targets here, which thus restricts the value of θ to the range $0° \leq \theta \leq 90°$. We also assume that the depression angle can be estimated fairly accurately based on the radar line of sight and the altitude measurements of the aircraft. We are therefore primarily interested in estimating the azimuthal angle ϕ at which the target is illuminated.

Recent research concerning the use of HRR profiles for ATR has been performed in [23]. They approach the problem from a Bayesian point-of-view, where the

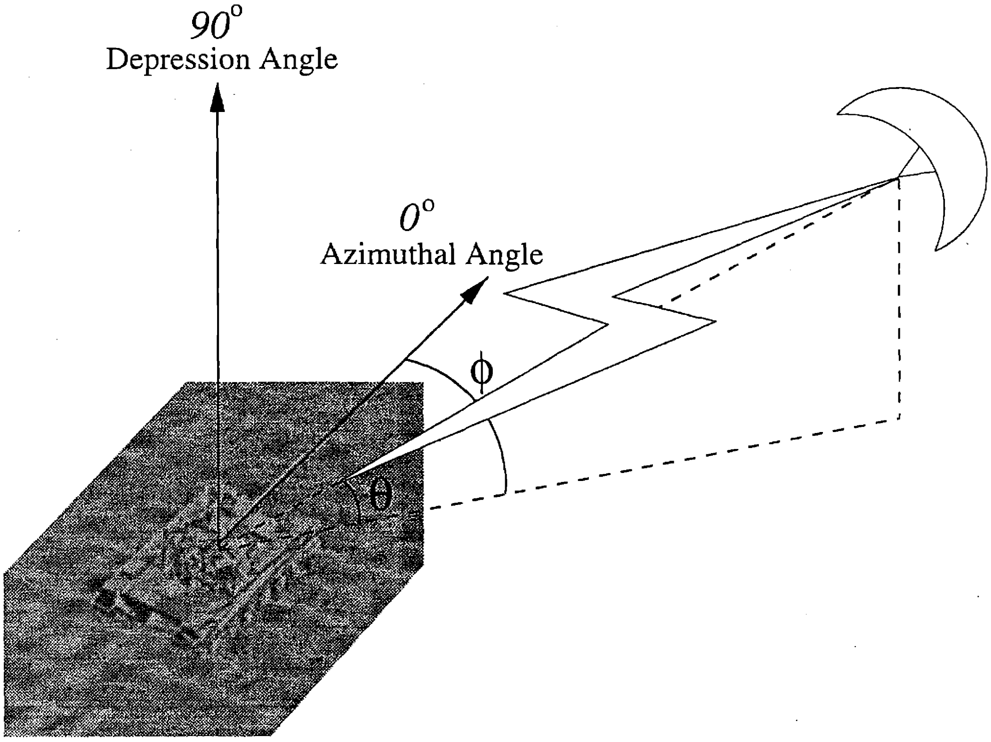


Figure 5.5: Shows how a target's orientation is measured in terms of the depression angle θ and the azimuthal angle ϕ.

dynamics of the target are estimated to induce a prior on a sequence of orientations. They also introduce the possibility of more than one target in a particular scene, and use the jump-diffusion algorithm described in [26] to continually estimate the number of targets to track. All of these estimation procedures combined, however, are computationally intensive, and consequently, decisions cannot be performed in real time. A simpler approach to the problem was originally proposed by [20, 21], where discrimination of different orientation angles or target types is performed using a correlation procedure or a nearest neighbor technique. In order to separate two classes $\{\mathbf{a}_i^1\}$ and $\{\mathbf{a}_i^2\}$ using a correlation procedure, one must find an n-dimensional weighting vector $\mathbf{w}$ and threshold T which satisfy

$$\mathbf{w}^*\mathbf{a}_i^1 + T > 0 \qquad \text{for all} \quad i = 1, 2, \cdots, N_1 \tag{5.5}$$

$$\mathbf{w}^*\mathbf{a}_i^2 + T < 0 \qquad \text{for all} \quad i = 1, 2, \cdots, N_2, \tag{5.6}$$

(using the notation of [21]). If $\mathbf{w}$ and T can be found to satisfy the equations above, then the two classes $\{\mathbf{a}_i^1\}$ and $\{\mathbf{a}_i^2\}$ are linearly separable, and the decision rule that separates the classes is a hyperplane in n-dimensional space. The nearest neighbor decision rule, on the other hand, bases its classification on the closest Euclidean distance, or

$$\mathbf{x} \in C_r \qquad \text{if} \quad ||\mathbf{x} - \mathbf{a}_j^r|| = \min_{i,k} ||\mathbf{x} - \mathbf{a}_i^k||. \tag{5.7}$$

Equation (5.7) above indicates that $\mathbf{x}$ is a member of a class C_r, if and only if $\mathbf{x}$ is "closest" to one of the elements of C_r. Both of these methods suggest that ATR can be performed by comparing an HRR return to a set of signals in a database using either a correlation procedure or a nearest neighbor technique. These procedures, however, lead to large misclassification errors in noisy environments, and some improvements must therefore be made.

In the next section, we propose an algorithm that provides some improvements over the previous work in [20, 21]. We are primarily concerned with developing a database of HRR profiles because we believe that this is a valid and efficient solution to the ATR problem. This database is constructed from a sequence of profiles which represent a particular target. Our goal is to find a representation that elicits the most information possible from the profiles. Once the database is in place, the ATR problem reduces to a table lookup, which can be rapidly performed. This eliminates the numerical computations involved in extrapolating the target trajectories as proposed in [23]. In the next section, we discuss the issues involved in establishing and searching the database.

5.3 Algorithm for Automatic Target Recognition

In this section, we propose an algorithm that can be used to identify targets and their spatial orientations. Figure 5.6(a) and (b) shows the two main parts of this algorithm. In the first part, we discuss the construction of the database containing signals representative of various targets at different aspect angles. We will show that an overcomplete representation of a set of HRR profiles leads to a more robust algorithm. The second part of the algorithm shown in Figure 5.6(b) and discussed in Section 5.3.2 is the database search procedure. Preprocessing is first performed on a received HRR profile in order to remove noise induced on the transmitted and received signals. One of the denoising procedures discussed in Chapter 4 may be used in this phase of the search algorithm. The next step in the algorithm is to search the database using the denoised signal. We show in Section 5.3.2 that the search for the most likely target and orientation may be efficiently performed in the overcomplete representation that we propose.

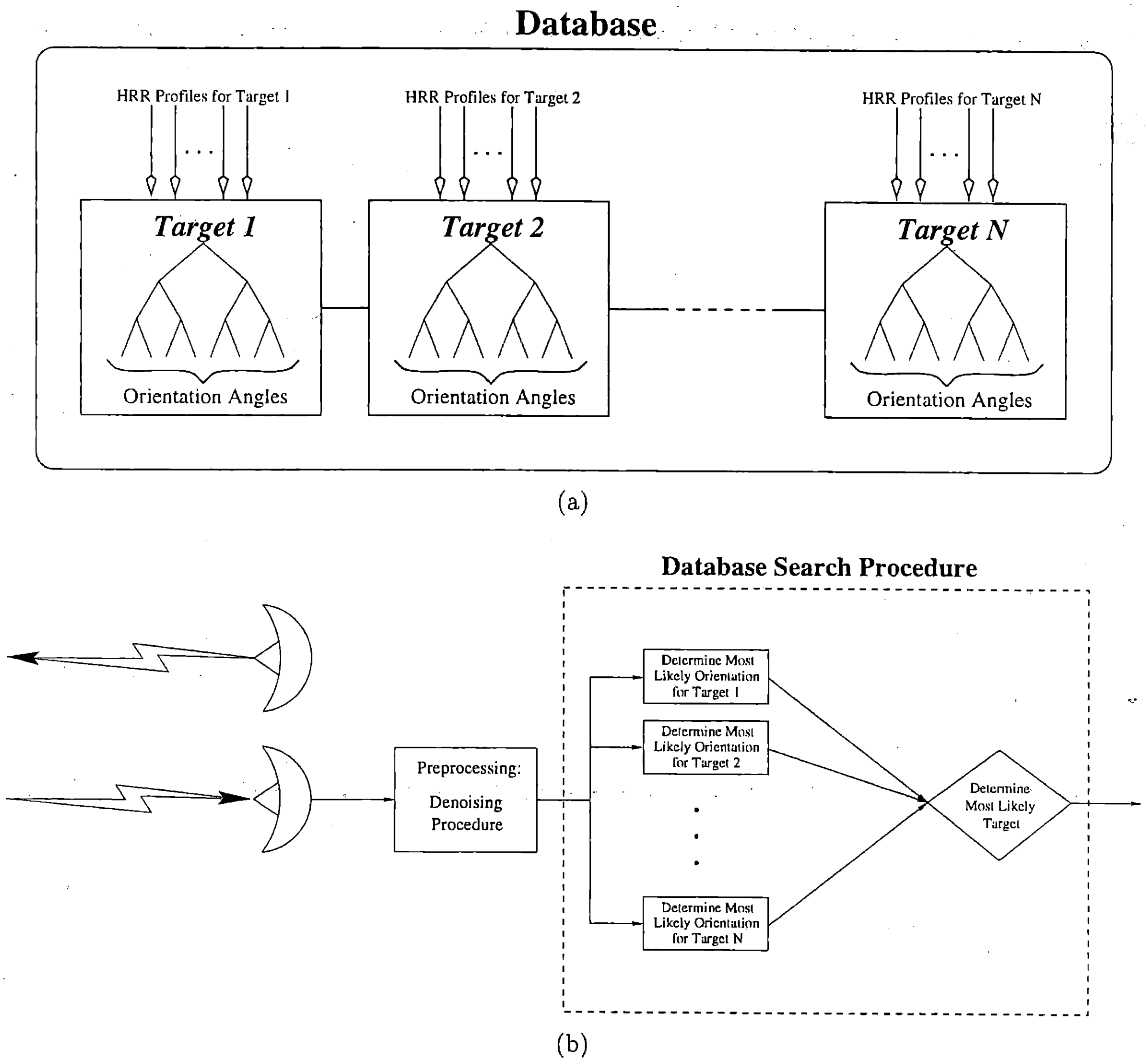

Database
HRR Profiles for Target 1
HRR Profiles for Target 2
HRR Profiles for Target N
Target 1
Target 2
Target N
Orientation Angles
Orientation Angles
Orientation Angles
(a)
Database Search Procedure
Preprocessing:
Denoising Procedure
Determine Most Likely Orientation for Target 1
Determine Most Likely Orientation for Target 2
Determine Most Likely Orientation for Target N
Determine Most Likely Target
(b)

Figure 5.6: Shows the two parts of the proposed algorithm. (a) Creating the database. (b) Searching the database.

5.3.1 Hierarchical Structure of Database

In creating the database shown in Figure 5.6(a), our goal is to incorporate maximal information about a target into a series of representative signals, where each signal corresponds to a particular sector of radar returns in the azimuthal direction. Since the HRR profiles are highly variable as a function of the aspect angle, we would like to include only the essential features in the representative signal. For a given sector, we search for an underlying signal that is highly correlated with all HRR returns in the sector. We cast this search into a denoising problem, where the underlying signal contains the important information about a sector, and the fluctuations among signals in the sector are modeled as noise.

For a sector of $(N-1)\Delta\phi$ degrees, there are N HRR returns from a given target,

$$\begin{aligned}\mathbf{x_1} &= \mathbf{s}+\mathbf{v_1}\\ \mathbf{x_2} &= \mathbf{s}+\mathbf{v_2}\\ &\vdots\\ \mathbf{x_N} &= \mathbf{s}+\mathbf{v_N},\end{aligned}$$

where each $\mathbf{x_i}$ is the sum of an underlying signal $\mathbf{s}$ and noise $\mathbf{v_i}$. We assume that $\mathbf{v_i}$ is white and Gaussian with $\mathbf{v_i} \sim \mathcal{N}(\mathbf{0}, \sigma_i^2\mathbf{I})$. In order to merge the information contained in all of these signals, we average $\mathbf{x_1}, \mathbf{x_2}, \cdots, \mathbf{x_N}$, to yield

$$\mathbf{x_{avg}} = \frac{\mathbf{x_1}+\mathbf{x_2}+\cdots+\mathbf{x_N}}{N} = \mathbf{s}+\mathbf{v_{avg}}.$$

We also assume that $\mathbf{v_i}$ is uncorrelated with $\mathbf{v_j}$ for all $i \neq j$, and consequently, $\mathbf{v_{avg}}$ is white Gaussian noise with $\mathbf{v_{avg}} \sim \mathcal{N}(\mathbf{0}, \frac{1}{N^2}\sum_{i=1}^{N}\sigma_i^2\mathbf{I})$. We let the aggregate signal $\mathbf{x}_{\mathbf{avg}}$ be the representative signal for the sector because it contains the essential features in the sector.

Given the method described above, we can use the representative signals in each sector to create a database. It is in general difficult to partition the sectors, since some partitions may mask the prominent features of the target. To remedy this problem, we propose an overcomplete representation of the HRR returns that is not as dependent on the selected partition. Figure 5.7 shows such a hierarchical representation of sectors, where a large sector is split into successively smaller sectors. With this representation, if an important feature is split among sectors in a lower level of the tree, then the entire feature will be represented in a sector located at a higher level on the tree. The signal at the top of the tree, $\mathbf{S}_1$, is an average over all azimuthal angles from 0° to 360°. The signal $\mathbf{S}_2$ is an average of only the profiles between 0° and 180°, while $\mathbf{S}_3$ is an average between 180° and 360°. This division process continues down the tree until the desired azimuthal resolution is obtained at the lowest level. This hierarchical approach to the aggregate signals provides an overcomplete representation of the HRR returns for a single target and helps create a robust ATR algorithm. The method for searching this overcomplete database is the topic of the next section.

5.3.2 Searching the Database

To search the database, we first perform a correlation procedure with each signal in the sector tree, where the tree is constructed by the method shown in Figure 5.7. This generates a tree of "costs" similar to the costs associated with the best basis search discussed in Section 3.4. Three such "cost" functions are considered below. Once a statistics tree has been constructed, it must be pruned to find the most likely

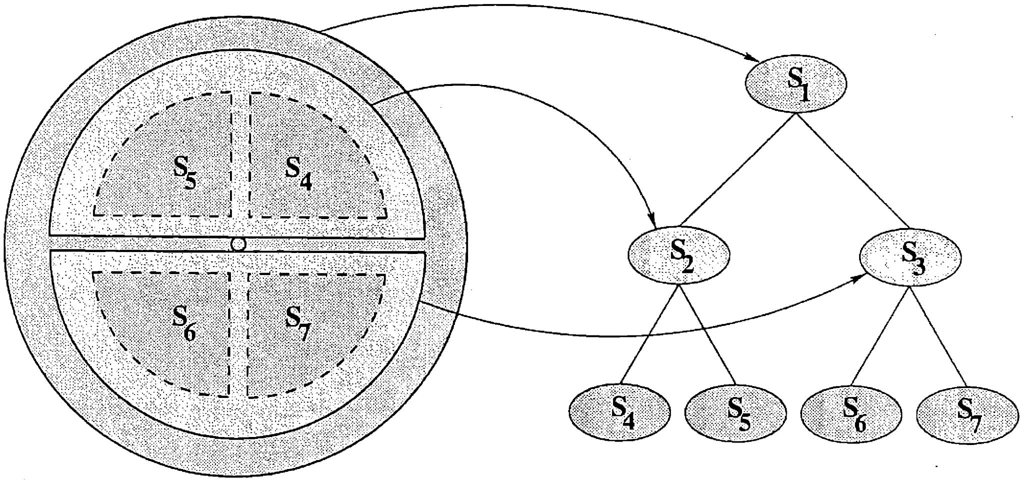


Figure 5.7: Hierarchical representation of sectors in the azimuthal direction.

orientation angle and target type. An efficient algorithm for pruning the tree is also discussed.

Cost Functions

Once a target is detected, we obtain an HRR return $\mathbf{y}$, and we correlate $\mathbf{y}$ with all of the signals $\mathbf{S_i}$ in the sector tree. For the correlation operation, we choose the maximum likelihood statistic,

$$\begin{aligned} \ell_{1,i} &= \mathbf{S}_i^* \mathbf{y} - \frac{1}{2}\mathbf{S}_i^* \mathbf{S}_i & (5.8) \\ \text{for } H_i &: \mathbf{y} = \mathbf{S}_i + \mathbf{v}, & (5.9) \end{aligned}$$

where $\mathbf{v} \sim \mathcal{N}(\mathbf{0}, \sigma_v^2 \mathbf{I})$. This statistic arises from the log-likelihood function. Under hypothesis H_i, the mean of $\mathbf{y}$ is $\mathbf{S}_i$, and the variance is σ_v^2. The log-likelihood function for hypothesis H_i is then given by,

$$\mathcal{L}_i = -\frac{n}{2}\log\left\{2\pi\sigma_v^2\right\} - \frac{1}{2\sigma_v^2}\left[\mathbf{y}^*\mathbf{y} - 2\mathbf{S}_i^*\mathbf{y} + \mathbf{S}_i^*\mathbf{S}_i\right]. \tag{5.10}$$

Since the constant term $-\frac{n}{2}\log\left\{2\pi\sigma_v^2\right\}$ and the scale factor $\frac{1}{2\sigma_v^2}$ are the same for all H_i, they do not provide any additional information in the discrimination process. In addition, the observed return $\mathbf{y}$ is the same for all hypotheses and may be removed. This results in the statistic given in Equation (5.8), which must be maximized to find the most likely hypothesis.

We also note that the term in brackets in Equation (5.10) is precisely the Euclidean distance between $\mathbf{y}$ and $\mathbf{S}_i$. The nearest neighbor method, therefore, is equivalent to the maximum likelihood statistic. This statistic is given by

$$\ell_{2,i} = (\mathbf{y} - \mathbf{S}_i)^*(\mathbf{y} - \mathbf{S}_i), \tag{5.11}$$

and must be minimized to find the most likely hypothesis.

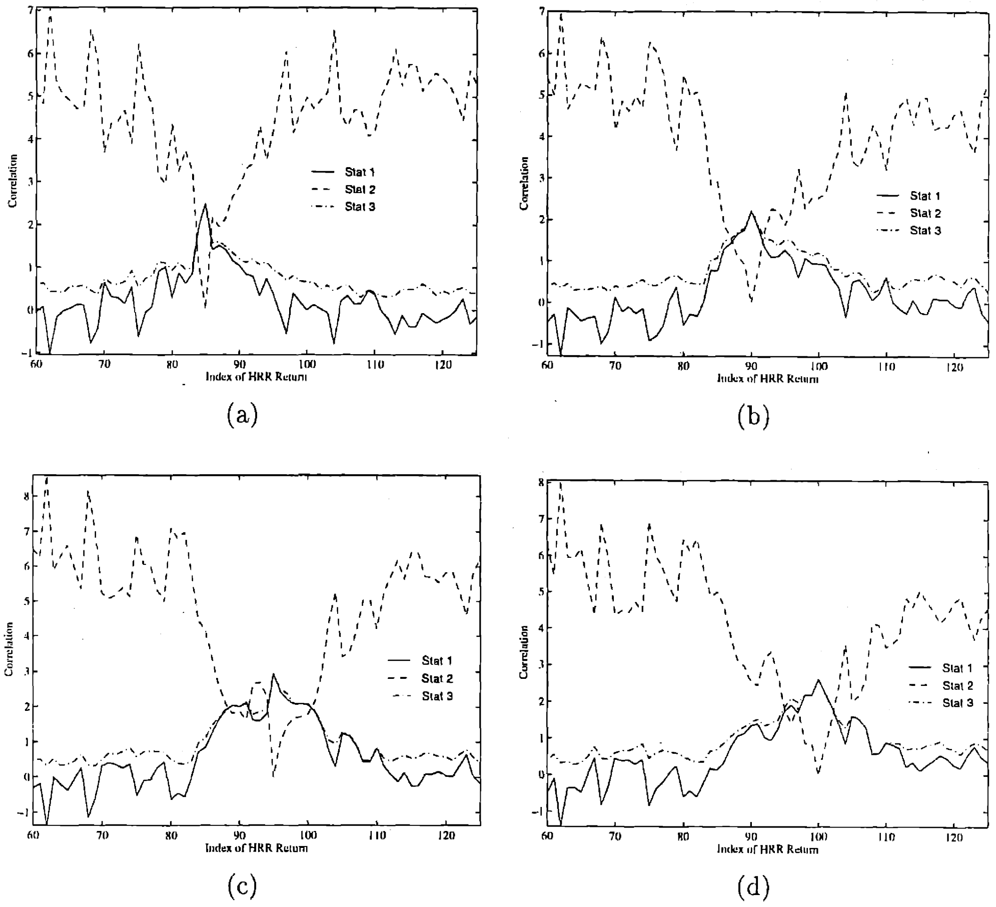
We consider a final statistic that allows a gain factor in each hypothesis, or

$$H_i \quad : \quad \mathbf{y} = a\mathbf{S}_i + \mathbf{v}. \tag{5.12}$$

This gain may be estimated as $\hat{a} = \frac{\mathbf{S}_i^*\mathbf{y}}{\mathbf{S}_i^*\mathbf{S}_i}$, resulting in the statistic

$$\begin{aligned} \ell_{3,i} &= \hat{a}\mathbf{S}_i^*\mathbf{y} - \frac{1}{2}\hat{a}^2\mathbf{S}_i^*\mathbf{S}_i & (5.13) \\ &= \frac{1}{2}\frac{(\mathbf{S}_i^*\mathbf{y})^2}{\mathbf{S}_i^*\mathbf{S}_i}, & (5.14) \end{aligned}$$

which must be maximized.



(a) (b)

(c) (d)

Figure 5.8: Shows that the statistics $\{\ell_{1,i}\}$, $\{\ell_{2,i}\}$, and $\{\ell_{3,i}\}$ discriminate the true HRR return. (a) True Return is $R(85)$. (b) True Return is $R(90)$. (c) True Return is $R(95)$. (d) True Return is $R(100)$.

In order to see how well these statistics discriminate different radar returns, we chose four different radar returns and correlated them with all other returns for a particular target. Figure 5.8 shows the results of this procedure for the fire truck at depression angle of 10°. In each case, the maximum correlation for the statistics $\{\ell_{1,i}\}$ and $\{\ell_{3,i}\}$ and the minimum correlation for $\{\ell_{2,i}\}$ are achieved at the true radar return. We therefore see that these statistics can be effectively used for discrimination purposes.

Optimal Search Algorithm

Given a tree of correlations, we now propose a method to determine the most likely azimuthal angle. One possible solution is to traverse the tree from top to bottom, making binary choices at each node of the tree. This method, however, can lead to large errors in estimating the azimuthal angle if an incorrect decision is made at the top of the tree. We, therefore, propose to find the path in the tree with the largest additive correlation, using an iterative search reminiscent of the best basis search discussed in Section 3.4. Figure 5.9 shows how this search can be performed in $O(N \log N)$ operations. In this example, the numbers in black correspond to the actual correlations at each node of the tree. At a given level, the maximum correlation of the two children nodes is added to the correlation of the parent node. This operation is performed until the top of the tree is reached. The correlation obtained at the top is the total correlation of the best path, and the best path is found by tracing the arrows back down the tree. This procedure may be used for the statistics $\{\ell_{1,i}\}$ and $\{\ell_{3,i}\}$ to maximize the correlation. To minimize the $\{\ell_{2,i}\}$ statistic, we choose the minimum correlation of the two children nodes at each level.

Using the above method, we can search a sector tree and find the best azimuthal angle for a given target, but now, we would like to extend the algorithm to include target identification. To address this issue, a sector tree is constructed for each target in a class of possible targets, and each tree is searched to determine the path with the maximum (or minimum, depending upon the statistic used) additive correlation. After an exhaustive search over all possible targets, we choose the target whose sector tree yields the largest (or smallest) correlation. This simple method, however, sometimes leads to errors in target identification because it does not account for the energy differences in the returned signals of different targets. Normalizing the radar returns to have unit energy is therefore necessary before any comparisons can

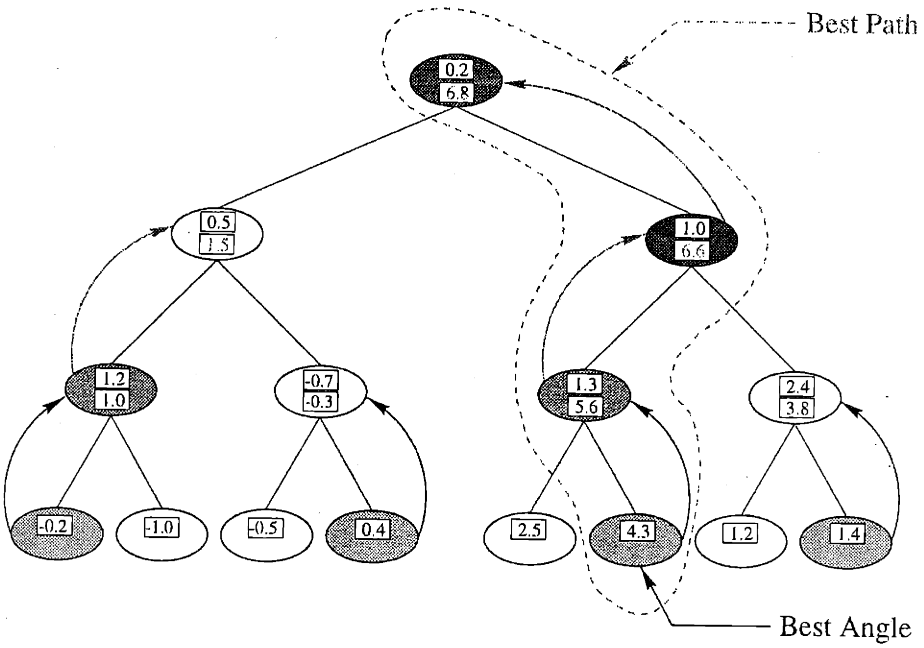


Figure 5.9: Efficient search algorithm for the path of largest correlation.

be performed. The additive cost structure which we propose also allows different polarizations to be used in making decisions about orientation and target type. Separate databases can be created for different polarizations, and returns obtained with these polarizations can then be used to search the appropriate database. The statistics trees for the polarizations are then added together, and the search algorithm is performed on this aggregate tree. We demonstrate some of these ideas in the next section via several experiments.

5.4 Results

To test the proposed algorithm, a sector tree was created for each of the four targets in the dataset at the depression angles: 10°, 25°, and 40°. In this analysis, we incorporate 256 azimuthal angles between 0° and 307.2° spaced 1.2° apart. The representative signal at the top of the tree contains an average of all 256 angles, while successive levels average $256(2^{-j})$ signals for $j \in \{0, \cdots, J\}$. In this example, we choose $J = 8$ so that the lowest level of the tree has a resolution of 1.2°. In general, the depth of the tree is dependent upon the required accuracy in the angular estimation and upon memory constraints.

Our database consists of 12 different sector trees corresponding to the 4 targets and the 3 depression angles. In this analysis, we limit the database search to the most likely azimuthal orientation, assuming that both the target and depression angle are known[1]. To test the algorithm, we use the full set of HRR returns spaced 0.3° apart. Since our HRR data is limited, this allows us to test the algorithm with HRR returns that are not in the database, as well as some that are. When testing returns not in the database, the maximum acceptable error is 0.9°, and for returns in the database, the only acceptable angular error is 0°.

Figure 5.10 gives the results for Tank 2 (shown previously in Figure 5.2(d)) at all three depression angles. The x-axis shows the angle of the HRR return that was used to search the database, while the y-axis shows the magnitude of the angular error as a result of the search. All of the plots provide results for HRR returns in the region 45° to 135°. The errors which resulted are shown in Figures 5.10(a),

[1] The depression angle is, in general, known fairly accurately from airplane measurements and the radar line of sight.

	Depression Angle		
Object	10°	25°	40°
Fire Truck	7.52%	4.69%	1.86%
School Bus	8.50%	7.13%	2.64%
Tank 1	3.03%	3.61%	0.78%
Tank 2	2.05%	1.37%	1.07%

Table 5.1: Provides the percentage misclassification error in the azimuthal direction for all targets and depression angles.

(c), and (e), with the largest errors occurring in Figures 5.10(a) and (c). These two errors occur, however, because the algorithm picked returns that corresponded to another axis of symmetry. By accounting for this symmetry, we obtain the graphs in Figures 5.10(b), (d), and (f). These figures show that the majority of errors lie at or below 0.9°, which is an acceptable error, given the resolution of the database.

Figure 5.11 shows similar results for Tank 1, shown previously in Figure 5.2(c). Even after removing the symmetry, Figures 5.11(b), (d), and (f) still show large errors that are not accounted for, but in general, the results are fairly accurate. To quantify the errors made by the algorithm, we compute the percentage of angular misclassifications made for each of the targets and depression angles. An error is considered to be anything above 0.9°. The results are shown in Table 5.1, and tend to indicate that the algorithm generally performs better for larger depression angles. Note that for these results, no axis of symmetry was taken into account.

We now consider the effect of noise on the performance of the algorithm. Noise was added to the HRR returns for SNR levels ranging from 0 dB to 15 dB. The HRR returns between 0° and 48° were used for the fire truck at a depression angle of 25°. Only returns included in the database were chosen, which means that the returns

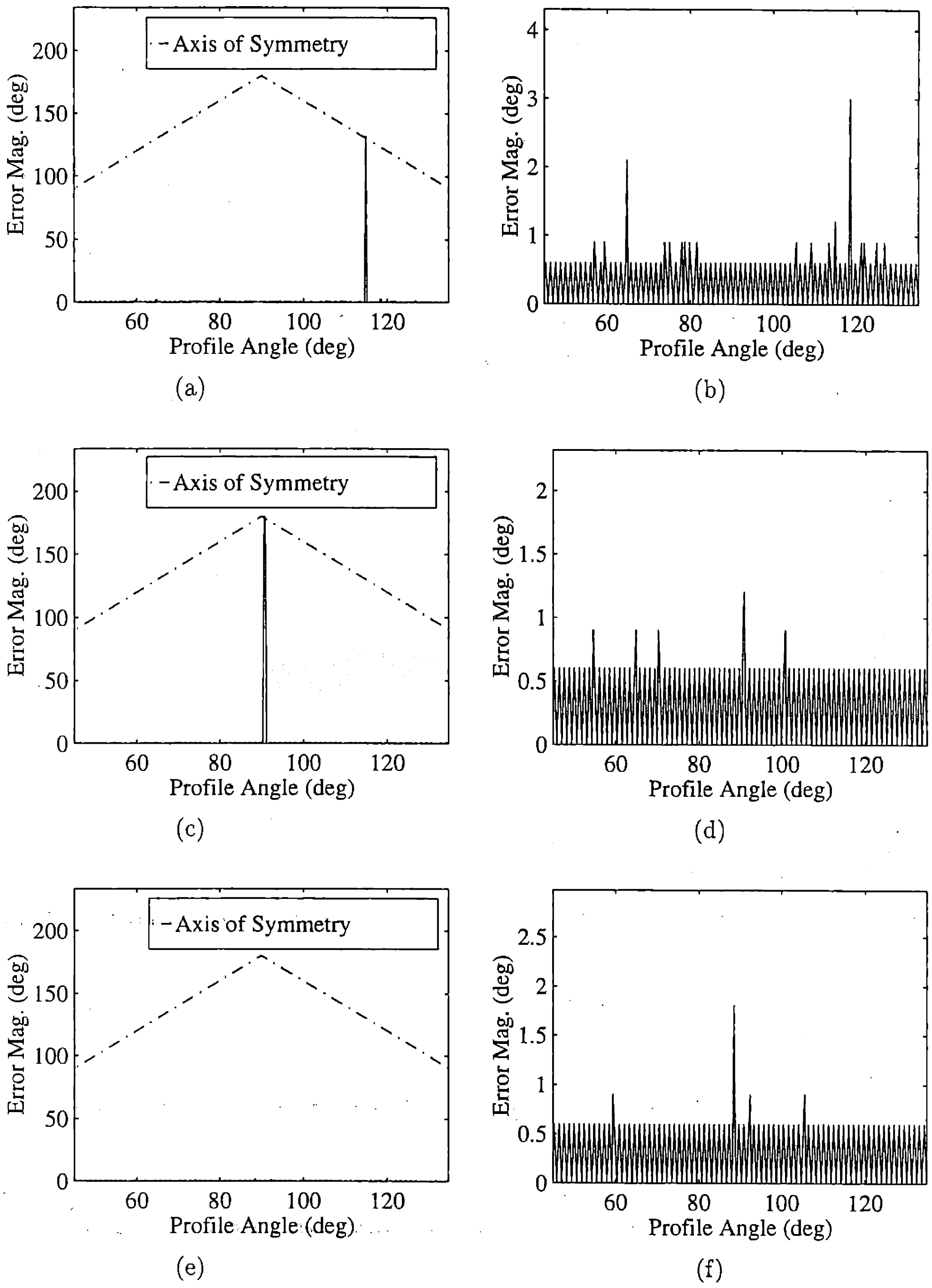


Figure 5.10: Plots of the errors that occur in searching the database using the HRR returns of Tank 2 at three different depression angles. (a) Tank 2 at a 10° depression angle. (b) Accounts for the axis of symmetry (10° depression). (c) Tank 2 at a 25° depression angle. (d) Accounts for the axis of symmetry (25° depression). (e) Tank 2 at a 40° depression angle. (f) Accounts for the axis of symmetry (40° depression).

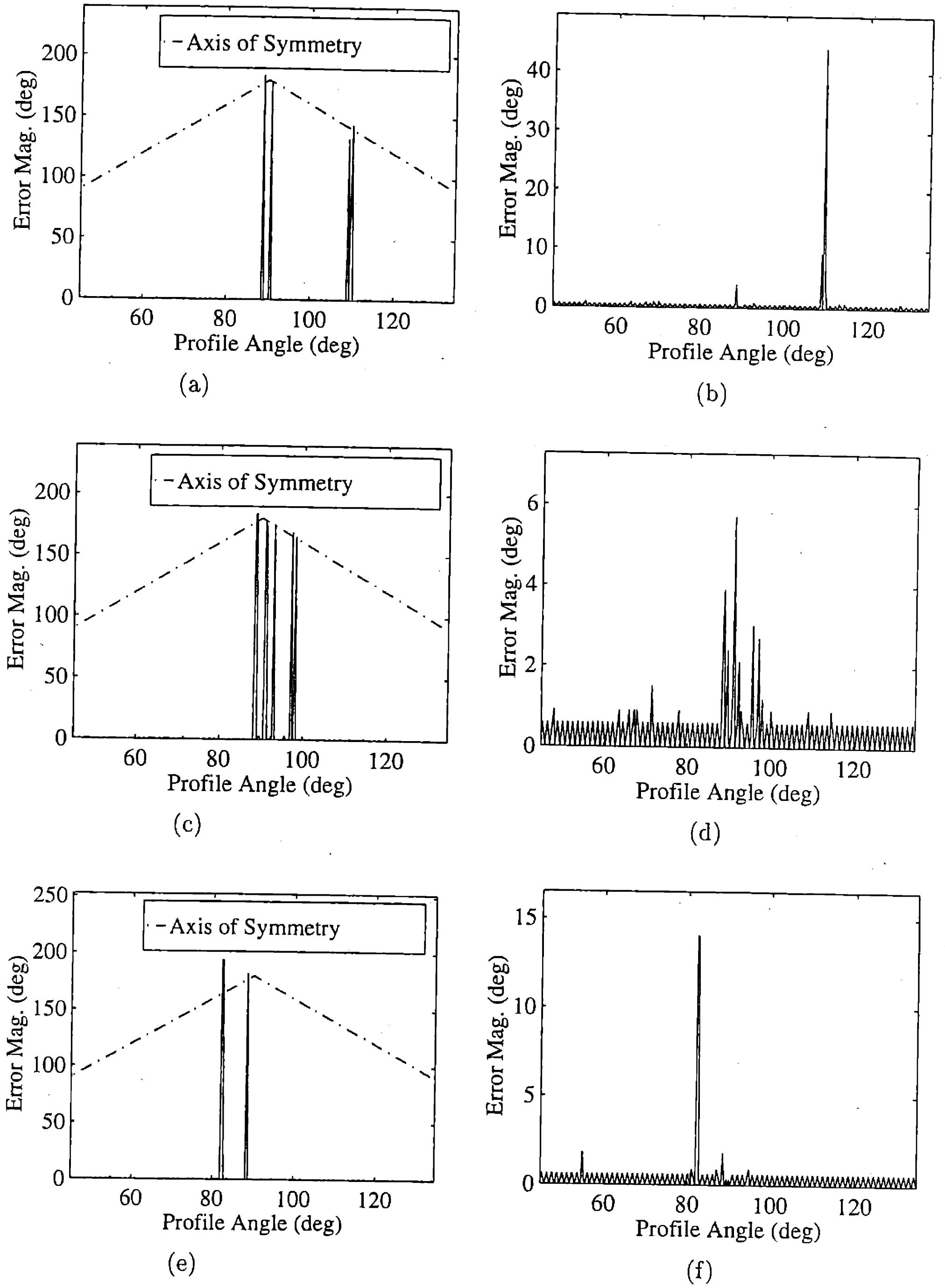


Figure 5.11: Plots of the errors that occur in searching the database using the HRR returns of Tank 1 at three different depression angles. (a) Tank 1 at a 10° depression angle. (b) Accounts for the axis of symmetry (10° depression). (c) Tank 1 at a 25° depression angle. (d) Accounts for the axis of symmetry (25° depression). (e) Tank 1 at a 40° depression angle. (f) Accounts for the axis of symmetry (40° depression).

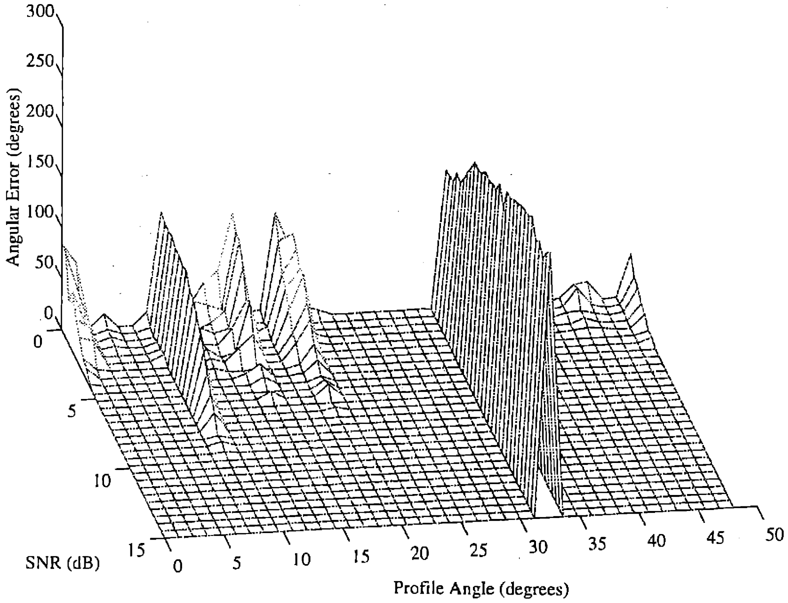


Figure 5.12: Plot of angular error for the fire truck at a depression angle of 25° as a function of SNR and the angle of the HRR profile used to search the database.

which we used were spaced 1.2° apart. The noisy returns were first denoised by using the entropy cost function to obtain the most compressed signal representation followed by thresholding at a level $T = \sqrt{2\sigma^2 \log N}$. The denoised signal was then used to search the database in order to determine the most likely azimuthal angle for the fire truck at a depression angle of 25°. The results are shown in Figure 5.12. In general, the errors are small for SNR levels above 5 dB. The HRR profile at approximately 32° is, however, consistently incorrect. This error is most likely due to an axis of symmetry that we have not accounted for in this analysis. These results are, however, promising because they indicate that for higher levels of SNR, the angular error remains within reason.

Chapter 6

Conclusions

In this chapter, we provide a summary of the results discussed in this thesis, along with some issues that suggest future research. Our primary approach has focused on wavelet packets as a tool for finding a signal representation that satisfies a specific goal. For the goal of compression, we introduced the theory of majorization. Majorization provides a framework for discriminating two vectors based on the distribution of their components. In this framework, we searched for cost functionals that could be used to preserve the ordering of an underlying majorization. These functions were then optimized using the Best Basis Search algorithm proposed in [19].

The search for the best compressed representation was then applied to the denoising problem. Noisy coefficients below a specified threshold T were discarded, in order to remove a significant portion of the noise while preserving the quality of the underlying signal. Compression-based denoising, however, did not lead to a measure of the reconstruction quality. To address this problem, we proposed the minimal reconstruction error criterion, which simultaneously accounted for the thresholding rule and the noise statistics with the goal of minimizing the error in reconstructing the underlying signal.

In the final topic of this thesis, we presented an algorithm for performing automatic target recognition using high-resolution radar data. This algorithm was presented in the framework of a database search. The methodology for both constructing and searching the database was presented, and the algorithm was shown to be both robust and computationally efficient.

We list below some natural extensions to the results previously presented. We suggest specific improvements that could enhance both the denoising techniques and the algorithm for ATR.

Denoising Techniques

- Any enhancement to the denoising problem should certainly begin by improving the tools used in the process. Designing better wavelets and proposing new overcomplete dictionaries of bases are therefore important topics to address.
- The purpose of Chapter 3 was to introduce a framework for viewing the functionals which measure compression. By finding an appropriate measure for majorization, we will, in essence, find a good measure of compression.
- Thresholding is also another important issue which must be addressed. The thresholding strategy discussed here assumes white Gaussian noise, but what if other types of noise are present? The thresholding strategy must be changed accordingly. Some work has been done in [27] concerning this, by treating the case of long-tailed noise. This leads to a robust wavelet thresholding technique.

- When different thresholding strategies are used, we must also determine the corresponding minimal reconstruction error cost function. In fact, a more general cost function which is parameterized by the thresholding strategy would prove extremely useful.

Algorithm for ATR

- As discussed previously, one primary problem with the analysis of ATR systems is the lack of data. Generating synthetic data via Xpatch and obtaining real data are therefore two important goals for the purposes of evaluation.

- Finding a better model as well as further understanding the current models of HRR signals may in fact lead to a better method for obtaining the representative signals in the database.

- Some alternative methods to the averaging technique may also improve the performance of the algorithm. Averaging is an intuitive scheme for combining information, but in some cases, it may, in fact, mask important features of a target. Some improvements may also be found by slightly overlapping the averaged sectors in order to incorporate even more redundancy in the database.

- An appropriate sampling of the azimuthal space must be determined for the correlation procedure to effectively discriminate returns at different orientation angles.

- The performance of the system is significantly affected by the chosen correlation statistics. Finding better statistics is an important step in improving its performance.

- We would also like to investigate the estimations of target type and azimuthal angle via multiple returns. Since the target will not change and the azimuthal angle will only vary slightly from one return to the next, the number of errors will be reduced by using several returns in the estimation procedure.

Bibliography

[1] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions on Pattern Recognition and Machine Intelligence*, vol. 11, pp. 674–693, July 1989.

[2] Y. Meyer, *Ondelettes et Opèrateurs.* Paris: Hermann, 1990. In two volumes.

[3] G. Strang and T. Nguyen, *Wavelets and Filter Banks.* Wellesley MA: Wellesley-Cambridge Press, 1996.

[4] M. Vetterli and J. Kovačević, *Wavlets and Subband Coding.* Englewood Cliffs, New Jersey: Prentice Hall, 1995.

[5] M. J. T. Smith and T. P. Barnwell, III, "Exact reconstruction techniques for tree-structured subband coders.," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, pp. 431–441, June 1986.

[6] I. Daubechies, *Ten Lectures on Wavelets.* Philadelphia, PA: SIAM, 1992.

[7] A. Cohen and I. Daubechies, "On the instability of arbitrary biorthogonal wavelet packets," *SIAM Journal on Mathematical Analysis*, vol. 24, pp. 1340–1354, September 1993.

[8] B. Jawerth and W. Sweldens, "An overview of wavelet based multiresolution analyses," *SIAM Review*, vol. 36, pp. 377–412, September 1994.

[9] G. Hardy, J. Littlewood, and G. Pólya, *Inequalities.* Cambridge Press, second ed., 1934.

[10] A. W. Marshall and I. Olkin, *Inequalities: Theory and Majorization and Its Applications.* New York: Academic Press, 1979.

[11] J. E. Pečarić, F. Proschan, and Y. L. Tong, *Convex Functions, Partial Orderings, and Statistical Applications.* San Diego, CA: Academic Press, 1992.

[12] M. V. Wickerhauser, "Lectures on wavelet packet algorithms," tech. rep., Washington University, St. Louis, Missouri, November 1991.

[13] J. B. Buckheit and D. L. Donoho, "Wavelab and reproducible research," in *Wavelets and Statistics* (A. Antoniadis and G. Oppenheim, eds.), Lecture Notes in Statistics, pp. 55–81, New York: Springer Verlag, 1995.

[14] C. Taswell, "Top-down and bottom-up tree search algorithms for selecting bases in wavelet packet transforms," in *Wavelets and Statistics* (A. Antoniadis and G. Oppenheim, eds.), Lecture Notes in Statistics, pp. 345–359, New York: Springer Verlag, 1995.

[15] B. Gnedenko, "Sur la Distribution Limite du Terme Maximum d'une Serie Aleatoire," *Annals of Mathematics*, vol. 44, pp. 423–453, July 1943.

[16] H. Krim, D. S. Tucker, S. Mallat, and D. Donoho, "On denoising and best signal representation." To be submitted, 1997.

[17] C. Stein, "Estimation of the Mean of a Multivariate Normal Distribution," *The Annals of Statistics*, vol. 9, no. 6, pp. 1135–1151, 1981.

[18] A. Papoulis, *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill Book Company, second ed., 1984.

[19] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Transactions on Information Theory*, vol. IT-38, pp. 713–718, Mar. 1992.

[20] L. J. White and A. A. Ksienski, "Aircraft identification using a bilinear surface representation of radar data," *Pattern Recognition*, vol. 6, pp. 35–45, 1974.

[21] A. A. Ksienski, Y.-T. Lin, and L. J. White, "Low-frequency approach to target identification," *Proceedings of the IEEE*, vol. 63, pp. 1651–1660, December 1975.

[22] J. S. Chen and E. K. Walton, "Comparison of two target classification techniques," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-22, pp. 15–21, January 1986.

[23] S. Jacobs, J. O'Sullivan, M. Faisal, and D. Snyder, "Automatic target recognition system using high resolution radar.," in *SPIE Proceedings*, vol. 2845, (Denver, Colorado), The International Society for Optical Engineering., August 1996.

[24] D. R. Wehner, *High Resolution Radar.* Norwood, MA: Artech House, 1987.

[25] H. W. Cole, *Understanding Radar.* Boston: Blackwell Scientific Publications, second ed., 1992.

[26] M. I. Miller, U. Grenander, J. A. O'Sullivan, and D. L. Snyder, "Automatic target recognition organized via jump-diffusion algorithms," *IEEE Transactions on Image Processing*, vol. 6, pp. 157–174, January 1997.

[27] I. C. Schick and H. Krim, "Robust wavelet thresholding for noise suppression," Tech. Rep. LIDS-P-2375, Massachusetts Institute of Technology, December 1996.