February, 1997

LIDS-TH 2384

Research Supported By:

DARPA F49620-93-1-0604
BU GC123919NGD

Multiscale Geometric Feature Extraction And Object Recognition

Jaggi, S

Multiscale Geometric Feature Extraction and Object Recognition

by

Seema Jaggi

B.S. Electrical Engineering
Purdue University, 1990

S.M. Electrical Engineering and Computer Science
Massachusetts Institute of Technology, 1992

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

Doctor of Philosophy

in

Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

February 1997

Author_____
Department of Electrical Engineering and Computer Science
January 31, 1997

Certified by_____
Alan S. Willsky
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Certified by_____
W. Clement Karl
Assistant Professor of Electrical, Computer, and Systems Engineering
Boston University
Thesis Supervisor

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

LIDS-TH-2384

Multiscale Geometric Feature Extraction and Object Recognition

by

Seema Jaggi

Submitted to the Department of Electrical Engineering and Computer Science
on January 31, 1997, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

This thesis describes a new adaptive approximation technique called high resolution pursuit (HRP), and demonstrates how HRP can be used to extract features that are suitable for object recognition. Recently, adaptive approximation techniques have become popular for obtaining representations of large classes of signals. These techniques include method of frames, matching pursuit, and basis pursuit. In this work, HRP is developed as an alternative to existing function approximation techniques. Existing techniques do not always efficiently yield representations that are sparse and physically interpretable. HRP is an enhanced version of the matching pursuit algorithm that overcomes its shortcomings by emphasizing local fit over global fit at each stage. Furthermore, the HRP algorithm has the same order of computational complexity as matching pursuit.

To demonstrate the utility of HRP for feature extraction, we develop a technique based on HRP features for the recognition of airplanes from silhouettes. Features extracted by HRP are shown to be robust to boundary perturbations, scale variations, small orientation variations, and variations due to occlusion. Furthermore, the HRP-based technique is shown to surpass the traditional Fourier descriptor techniques in the presence of occlusion.

Thesis Supervisor: Alan S. Willsky
Title: Professor of Electrical Engineering and Computer Science

Thesis Supervisor: W. Clement Karl
Title: Assistant Professor of Electrical, Computer, and Systems Engineering
Boston University

Acknowledgments

I have been privileged to work with Alan Willsky and Clem Karl for the past six years. Alan brings a great deal of enthusiasm and insight to every problem. I am very grateful to Alan for making the Stochastic Systems Group a great place to work and providing me with funding, as well as a few lessons in oenology. I have shared many long and fruitful discussions with Clem, who always provided me with clear and insightful comments and somehow understood my most muddled ideas. In particular, Clem has provided many useful suggestions for my writing. Thanks also to Stephane Mallat for his guidance in this work, and to Eric Grimson and Ron Chaney for some helpful discussions.

In my time here at MIT, I have been fortunate to have interacted with many extraordinary and interesting people. I would like to acknowledge my long line of officemates : Rachel for being a great friend; Paul for some incredibly long discussions about some incredibly minute points; Mickey for being two of the coolest guys I ever met; Lori for sharing many laughs; Eric for providing a never-ending supply of bad puns; Ilya for demonstrating his ability to survive on only chocolate; and Dewey for demonstrating his ability to survive on only fried food. To the other members of the SSG, Bill, Cedric, Mike S., Terrence, Andrew, Charlie, Austin, Hamid, and Jun, thanks for providing many sanity-saving distractions in 35-437. Thanks to Ben for being my morning (lunchtime ?) coffee break partner. Thanks to all my friends at LIDS, especially Stefano, Francesca, and Aradhana. To Sarita and Prasanna, thanks for sharing many good meals (and a few really gory movies) with me.

The two people who deserve the most credit for where I am today are my parents. Mom and Dad, thanks for believing in me. It seems that all your *Satyanarayan poojas* paid off. To my sisters, Preeti and Anju, thanks for your friendship. Thanks also to Weesie and Charlie Daniel for their encouragement. Finally, thanks to my husband, Mike, for his support. Ironically, I am looking forward to spending less time with Mike now that my thesis is finished.

अपने पुजनीय माताजी और

आदरणीय पिताजी को समर्पण

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis describes high resolution pursuit, a new adaptive approximation technique, and demonstrates how high resolution pursuit can be used to extract features which are suitable for object recognition. The high resolution pursuit algorithm, like other adaptive approximation techniques, is a method to decompose a function as the weighted sum of elements from a redundant dictionary. The high resolution pursuit decomposition also yields a set of geometric features which are ideal for object recognition. Thus, this thesis contributes to the areas of adaptive approximation and object recognition.

1.1 Adaptive Approximation Motivation

Recently, adaptive approximation techniques have become popular for obtaining parsimonious representations of large classes of signals. In these adaptive approximation techniques, the goal is to find a representation of a function f as a weighted sum of

elements from an overcomplete dictionary. That is, f is represented as

$$f = \sum_{\gamma \in \Gamma} \lambda_\gamma g_\gamma \tag{1.1}$$

where the set $\{g_\gamma | \gamma \in \Gamma\}$ is a redundant dictionary spanning the space of possible functions. If very few of the coefficients λ in (1.1) are non-zero, then the decomposition is said to be sparse.

Redundant dictionaries are used since they are flexible enough to match many of the important structures of the function. As a result of this flexibility, decompositions over redundant dictionaries may have very few non-zero coefficients (i.e. sparse representations are possible). Of course, the decomposition problem would be simpler (trivial) if orthogonal dictionaries were used. The main drawback of orthogonal dictionaries is that they may not be flexible enough to include the different types of structures present in a function, and therefore it may not be possible to obtain sparse decompositions. For example, suppose f was the sum of a sinusoid and a single Dirac. Two possible orthogonal dictionaries are the Fourier and Dirac dictionaries. A decomposition of f over either the Fourier dictionary or Dirac dictionary would not yield a sparse description of the function since neither dictionary contains all the structures of the function. Borrowing from the analogy used in [26], these dictionaries are like a language with a very limited vocabulary. It is still possible to express all ideas using this limited vocabulary. However, full sentences must be used to replace unavailable words. A better dictionary for decomposing f would be a redundant dictionary which contains both the Fourier and Dirac bases.

Since many possible representations of f of the form of (1.1) exist in a redundant dictionary, the price of using a flexible redundant dictionary is the ambiguity associated with determining the "optimal" decomposition. Several methods have been suggested for determining the "optimal" decomposition. These include the method

of frames [9], best orthogonal basis [8], matching pursuit [26], and, most recently, basis pursuit [6].

Intuitively, the "optimal" decomposition should have the following characteristics :

- **Sparsity** A sparse representation is one in which a minimum number of dictionary elements are used to represent any function. In particular, if a function is synthesized as the sum of dictionary elements, the "optimal" adaptive approximation representation would be precisely those elements used to construct the function.

- **Super-Resolution** A closely related concept is that of resolution. The decomposition of a function which is the sum of two closely spaced dictionary elements should show the presence of both elements.

- **Physical Interpretation** A physically interpretable decomposition is one in which each term of (1.1) relates to the geometric (e.g. size and location) characteristics of the function.

- **Hierarchy** The "optimal" decomposition should have a corresponding hierarchy so that a few elements of sum in (1.1) yield a coarse approximation to the function.

- **Stability** In the case where f is corrupted by noise, the "optimal" decomposition should be stable so that small perturbations in the underlying signal do not drastically change the representation.

- **Computationally Tractability** To be practical, the decomposition should be obtained in a manner that is computationally tractable.

Existing adaptive approximation techniques do not always yield representations with all of these desired characteristics. The method of frames [9] chooses the de-

composition which minimizes the ℓ^2 norm of the coefficients, λ_γ, in (1.1). As a result, the method of frames tends towards solutions which are not sparsity-preserving. In contrast, basis pursuit [6] chooses the decomposition which minimizes the ℓ^1 norm of the coefficients in (1.1), a criterion that has been shown to yield sparse representations. However, to find this optimal solution, the minimization problem is translated to an equivalent large scale linear program, which is know to be computationally complex. Thus, basis pursuit produces representations which preserve sparsity and resolve closely spaced features, but is computationally complex. Finally, matching pursuit [26] addresses the sparsity issue directly by building the decomposition in (1.1) up one element at a time. At each step, the dictionary element chosen is the one which yields a maximum reduction in residual power. However, the major drawback of matching pursuit is that it is unable to resolve closely spaced elements. The greedy nature of matching pursuit can in fact lead to the introduction of artifacts which in turn lead to the extraction of features which are not physically meaningful. Both matching pursuit and basis pursuit will be further explored in Chapter 2.

In light of the desired representation characteristics outlined above, high resolution pursuit is developed in this thesis as an alternative to existing function approximation techniques. High resolution pursuit is an enhanced version of the matching pursuit algorithm which overcomes the shortcomings of the original matching pursuit algorithm by emphasizing local fit over global fit without significantly increasing the computational complexity over that of matching pursuit. In particular, high resolution pursuit decompositions frequently rival those of basis pursuit in terms of sparsity and resolution, but require much less computation.

1.2 Silhouette Recognition Motivation

Humans have the ability to analyze visual scenes, localize and recognize objects, and make decisions based on our visual observations. A computer vision system that could emulate these tasks, which are simple for a human, would have numerous industrial, military, and medical applications. Model-based object recognition [3,15] from images is currently the focus of much work in computer vision and is performed by comparing features extracted from a given data image to features extracted from a predefined set of model images and determining which model the data image most closely resembles.

One specific computer vision problem that will be of interest in this work is the recognition of airplanes from silhouettes. Since humans have the ability to recognize objects from only silhouette or boundary curves, there is a widely-held intuition in computer vision [3, 13, 16, 19, 24, 30, 40] that these object boundaries contain much of the significant information required to recognize objects. For a computer vision algorithm for recognizing objects from silhouettes to be of practical importance, the algorithm must perform well under a number of possible variations in the image, such as scale variation, orientation variation, boundary perturbations, and variation due to occlusion. Further, features should be quickly computable.

Previous work in model-based object recognition from silhouettes has focused on feature matching where the features were of two distinct types : global and local. Global features are those constructed from the entire object; some examples of global features are Fourier descriptors and moments (e.g. area). Some relevant benefits of global features are the following. First, object recognition is carried out simply by matching these global features. That is, it is clear which model feature to compare to which image feature so that a correspondence problem does not arise. Second, one can envision a clear hierarchy in the object recognition phase. For example, in recognition based on moments, a match of the lower order moments would be more

significantly weighted than a match of the higher order moments. On the other hand, global features also have the following drawbacks. First, object recognition based on global features does not handle occlusion well. Second, the *entire* set of features (i.e. the overall representation) may be drastically changed by a local change in the underlying object.

In contrast, local features, as the name implies, are determined by local image properties. Some commonly used local features include edges [20], vertices [7, 32], and curvature extrema [31]. Again, there are both benefits and drawbacks to using local features. Object recognition based on local features performs well in cases of partial occlusion since only a subset of the object features are changed by a local change in the object. On the other hand, determining the correspondence between model features and object features may present a large computational burden. In addition, local features may not be not intrinsically geometric. Certain geometric properties of the object may be deduced from an aggregation of local features, but the individual features do not yield size, orientation or elongation information. Geometry is instead introduced by imposing geometrically based constraints in the correspondence search [21]. There is generally no hierarchy among the set of local features (i.e. there is no way to determine which features are more important than others). Finally, local features may not be robust to errors introduced by unstructured noise. For example, the presence of noise in an image may greatly distort the edge locations.

In this thesis, high resolution pursuit is proposed as a means of extracting features that are robust to the presence of boundary perturbations and variations due to occlusion. The features extracted by high resolution pursuit quantify the size and location of subparts of the object, and therefore represent a new class of features for object recognition. This new class of features are robust to both boundary perturbations and perturbations due to occlusion.

1.3 Contributions

This thesis contributes to the areas of adaptive approximation and object recognition. The characteristics of the "optimal" adaptive approximation decomposition are desirable qualities in feature extraction for object recognition as well. Thus, the parameters of the elements extracted by high resolution pursuit are attractive as candidates features for object recognition.

First, this thesis contributes to the area of adaptive approximation. The technique developed in this work, high resolution pursuit, is a new adaptive approximation technique. The decompositions obtained using high resolution pursuit are sparsity-preserving, exhibit super-resolution, and have an inherent hierarchy. Further, high resolution pursuit produces decompositions which are stable in the presence of small perturbations in the underlying function and does so in a computationally tractable way. This thesis contains a thorough investigation of high resolution pursuit as a general adaptive approximation procedure. Two distinct interpretations of the high resolution pursuit algorithm are described. The first description gives a geometric interpretation of the high resolution pursuit algorithm, while the second gives a constrained maximization interpretation. In addition, the convergence properties and computational complexity of the high resolution pursuit algorithm are studied. Finally, the high resolution pursuit algorithm is demonstrated on several 1D signals using the cubic b-spline and wavelet packet dictionaries.

Second, this thesis contributes to the area of object recognition from silhouettes. In the object recognition context, the elements extracted by high resolution pursuit are a new class of features that describe geometric (i.e. size and location) properties of subparts of the object. The features based on elements extracted by high resolution pursuit exhibit some qualities typically associated with global features and some qualities typically associated with local features. Like global features, the features

based on high resolution pursuit are robust to variations due to noise and exhibit a hierarchy. Like local features, the features based on high resolution pursuit are robust to variations due to occlusion. Since a small number of high resolution pursuit features can be used to describe an object, there is only a small correspondence problem which must be solved in comparing data and model silhouettes. As we demonstrate in Chapter 4, this new class of features can be used to recognize objects in the presence of scale variation, orientation variation, and boundary noise, as well as occlusion. This is an improvement on existing techniques which are often not robust to both boundary noise and occlusion.

1.4 Organization

The organization of the thesis is as follows. Chapter 2 describes two techniques from adaptive approximation, matching pursuit and basis pursuit. This chapter also includes a description of the cubic b-spline dictionary which will be useful in this work. In Chapter 3, the high resolution pursuit algorithm is developed and demonstrated. Chapter 3 also presents a geometric interpretation and a constrained maximization interpretation of the high resolution pursuit algorithm. Chapter 4 demonstrates how the elements extract by the high resolution pursuit algorithm can be used for the recognition of airplanes from silhouettes. Finally, Chapter 5 summarizes the conclusions of this work and presents some ideas for future research.

Chapter 2

Background:Adaptive Approximation

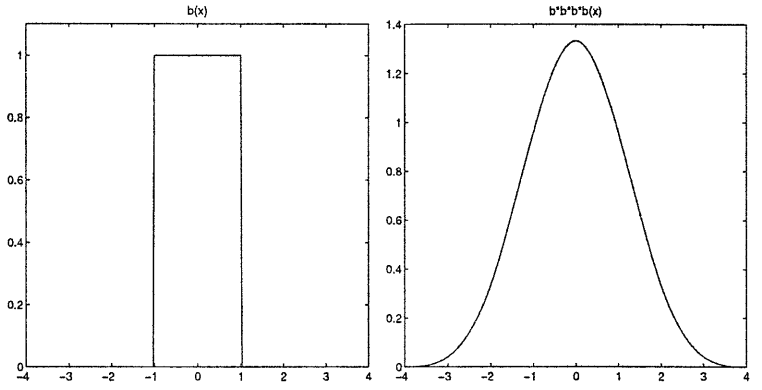
Recently, adaptive approximation techniques have become popular for obtaining representations of large classes of functions. In adaptive approximation, the goal is to find the representation of a function f as a weighted sum of elements from a redundant (overcomplete) dictionary. That is, f is represented as

$$f = \sum_{\gamma \in \Gamma} \lambda_\gamma g_\gamma \tag{2.1}$$

where the set $\{g_\gamma | \gamma \in \Gamma\}$ is a dictionary which spans the space of possible functions but is redundant. Many possible representations of f exist in this redundant dictionary. Several methods have been suggested to find the "optimal" representation of the form of (2.1). These methods include the method of frames [9], best orthogonal basis [8], matching pursuit [26], and, most recently, basis pursuit [6]. Matching pursuit (MP) and basis pursuit (BP) will be particularly relevant in this thesis. In this chapter, we summarize some relevant topics from adaptive approximation techniques.

2.1 Dictionaries

An important issue in using these adaptive approximation techniques is the choice of the dictionary. Adaptive approximation techniques are used to determine the "optimal" decomposition of f of the form (2.1) from a redundant dictionary. Redundant dictionaries are used since they can easily accommodate elements that have a wide range of time-frequency characteristics and match the important structures of the function to be decomposed. Of course, the decomposition problem would be simpler (trivial) if orthogonal dictionaries were used. The main drawback of orthogonal dictionaries is that they may not be flexible enough to include the different types of structures present in a function. For example, suppose f was the sum of a sinusoid and a single Dirac. Two possible orthogonal dictionaries are the Fourier and Dirac dictionaries. A decomposition of f over either the Fourier dictionary or Dirac dictionary would not yield a sparse description of the function since neither dictionary contains all the structures of the function. Borrowing from the analogy used in [26], these dictionaries are like a language with a very limited vocabulary. It is still possible to express all ideas using this limited vocabulary. However, full sentences must be used to replace unavailable words. A better dictionary for decomposing f would be a redundant dictionary which contains both the Fourier and Dirac bases. To summarize, the dictionary should be chosen to contain elements which match the important structures of the function to be decomposed and often redundant dictionaries are required to fulfill this requirement. Two redundant dictionaries that will be particularly useful in this thesis are the cubic b-spline dictionary and the wavelet packet dictionary, though many others can be imagined.



(a) A box spline, $b(x)$.

(b) A cubic box spline, $g_u(x) = b * b * b * b(x)$.

Figure 2-1: Box Splines.

2.1.1 The Cubic B-spline Dictionary

The cubic b-spline dictionary consists of cubic b-splines over an appropriate range of scales and translations. This section will describe the cubic b-spline dictionary structure and highlight properties that will be important in this thesis.

A cubic b-spline is a box spline convolved with itself three times. Figure 2-1 shows a box spline $b(x)$ and the resulting cubic b-spline $g_u(x)$. The analytic forms for $b(x)$ and $g_u(x)$ are

$$b(x) = \begin{cases} 1 & -1 \leq x \leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{2.2}$$

$$g_u(x) = b * b * b * b(x) \tag{2.3}$$

$$= \begin{cases} \frac{1}{24}(x^3 + 12x^2 + 48x + 64) & \text{for } -4 \leq x \leq -2 \\ \frac{1}{24}(-3x^3 - 12x^2 + 32) & \text{for } -2 \leq x \leq 0 \\ \frac{1}{24}(3x^3 - 12x^2 + 32) & \text{for } 0 \leq x \leq 2 \\ \frac{1}{24}(-x^3 + 12x^2 - 48x + 64) & \text{for } 2 \leq x \leq 4 \end{cases} \tag{2.4}$$

Let $g(x) = g_u(x)/\|g_u(x)\|$. A cubic b-spline at scale j and translation t will be denoted $g_{j,t}(x)$ and is given by

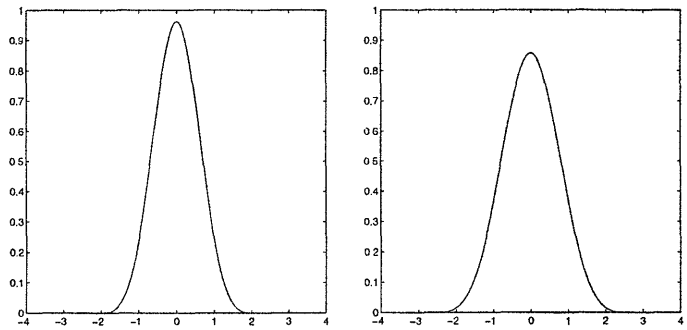
$$g_{j,t}(x) = \sqrt{2^j} g(2^j(x - t)). \tag{2.5}$$

As $j \rightarrow \infty$, the cubic b-splines become finer in scale and approach Diracs. In this work, we will often use γ to denote the pair (j, t) so that $g_\gamma(x) = g_{j,t}(x)$. The cubic b-spline dictionary then contains a set of functions $g_{j,t}(x)$ over an appropriate range of scales and translations. For example, Figure 2-2 shows some elements at different scales from a cubic b-spline dictionary. Note that the elements of the dictionary are normalized so that $\|g_{j,t}\| = 1$ for all j and all t.

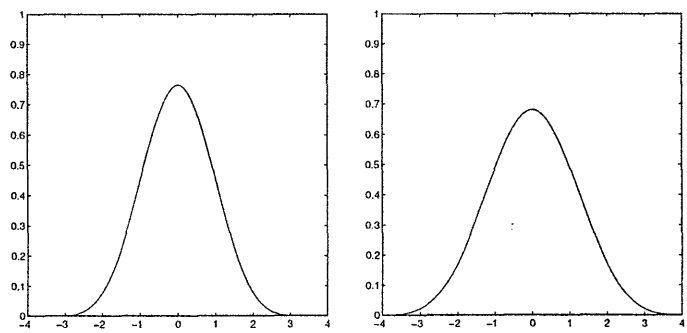
The following property of cubic b-splines will be conceptually important for the work in this thesis. Any cubic b-spline may be written as the sum of finer scale cubic b-splines which are also dictionary elements. For example, $g_{j,t}$ may be written as the weighted sum of finer scale cubic b-splines which are all at the same scale, $j + k$; that is,

$$g_{j,t} = \sum_{i=1}^{L} c_i g_{j+k,t_i}. \tag{2.6}$$

Note that the sum in (2.6) is unique and finite. This is illustrated in Figure 2-3 for $k = 1$ and $k = 2$. Following this idea, and for convenience later, let us define for each element in the cubic b-spline dictionary, g_γ, an associated set of indices, $I_\gamma(k)$. The functions which are indexed by $I_\gamma(k)$ are the dictionary elements at the finer scale



(a) $j = j_0, t = 0.$ (b) $j = j_0 - 1/3, t = 0.$



(c) $j = j_0 - 2/3, t = 0.$ (d) $j = j_0 - 1, t = 0.$

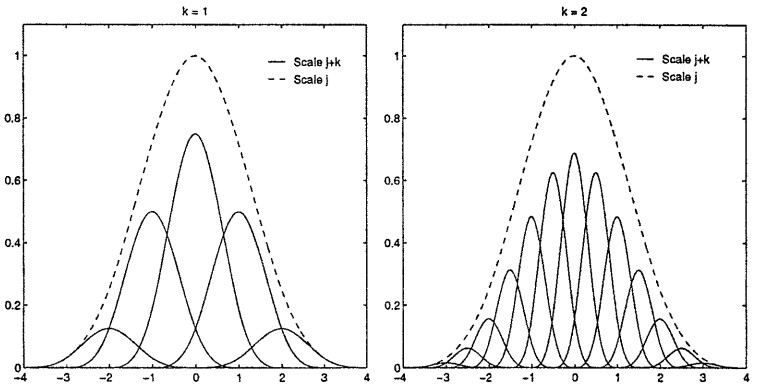
Figure 2-2: Some samples of cubic b-splines, $g_{j,t}$, for several values of j.

$j + k$ which when properly weighted and summed yield g_γ plus γ itself. That is,

$$I_\gamma(k) = \left\{ (j+k, t_i) \,\middle|\, g_\gamma = \sum_{i=1}^{L} c_i g_{j+k,t_i} \right\} \bigcup \{\gamma\}. \tag{2.7}$$

The index γ has been included in the family $I_\gamma(k)$ because it will be useful in Section 3.3. Thus, (2.6) can be written equivalently as

$$g_\gamma = \sum_{i \in I_\gamma(k)/\gamma} c_i g_i. \tag{2.8}$$



(a) $k = 1$. (b) $k = 2$.

Figure 2-3: Weighted sum of cubic b-splines at scale $j + k$ yields a cubic b-spline at scale j.

2.1.2 The Wavelet Packet Dictionary

The wavelet packet dictionary is a redundant dictionary consisting of the functions used to generate the wavelet packet decomposition. This section highlights the structure of the wavelet packet dictionary that will be important in this thesis. More detailed descriptions of wavelet and wavelet packet transforms may be found in [22,36].

The wavelet packet decomposition is an extension of the wavelet decomposition. As described in [22, 29, 33], the wavelet transform of a function is the projection of that function onto translated, scaled versions of a mother wavelet, $\psi(x)$. Let

$$\psi_{j,t}(x) = \sqrt{2^j}\psi(2^j(x-t)). \tag{2.9}$$

The set of wavelet functions at a given scale j and translates $t = 2^{-j}l$, with $l \in \mathbf{Z}$ forms an orthogonal basis for a space that is denoted $\mathbf{W}_j$. The scaling function that

accompanies $\psi(x)$ is denoted $\phi(x)$. Scaled and translated version of $\phi(x)$ are denoted

$$\phi_{j,t}(x) = \sqrt{2^j}\phi(2^j(x-t)). \tag{2.10}$$

The set of scaling functions at a given scale j and translates $t = 2^{-j}l$, with $l \in \mathbf{Z}$ forms an orthogonal basis for a space that is denoted $\mathbf{V}_j$. The space $\mathbf{W}_j$ contains high frequency elements, the space $\mathbf{V}_j$ contains low frequency elements, and the two spaces are orthogonal to one another. Linear combinations of the scaling functions at scale j yield the wavelet and scaling functions at the next coarser scale, $j-1$. These linear combinations are specified by the conjugate mirror filters [1] h_1 and h_2. That is,

$$\phi_{j-1,0}(x) = \sum_{n=-\infty}^{+\infty} h_1[n]\phi_{j,2^{-j}n}(x) \tag{2.11}$$

$$\psi_{j-1,0}(x) = \sum_{n=-\infty}^{+\infty} h_2[n]\phi_{j,2^{-j}n}(x). \tag{2.12}$$

The set of coarser functions $\{\psi_{j-1,2^{-(j-1)}l}\}_{l\in\mathbf{Z}}$ and $\{\phi_{j-1,2^{-(j-1)}l}\}_{l\in\mathbf{Z}}$ are bases for $\mathbf{W}_{j-1}$ and $\mathbf{V}_{j-1}$, respectively. The spaces $\mathbf{W}_{j-1}$ and $\mathbf{V}_{j-1}$ are contained in $\mathbf{V}_j$ and thus orthogonal to $\mathbf{W}_j$. The wavelet transform is constructed by projecting onto spaces that are formed by repeatedly dividing the low frequency spaces $\mathbf{V}_j$. As a result, the wavelet transform yields poor frequency resolution for high frequencies. In contrast, the wavelet packet transform is constructed by dividing $\mathbf{W}_j$ as well as $\mathbf{V}_j$. Generalizing the wavelet notation, the wavelet packet decomposition of a function is the projection of that function onto a set of spaces $\mathbf{W}_{j,\omega}$ where j is scale and ω is a frequency index. Each space $\mathbf{W}_{j,\omega}$ has a corresponding orthogonal basis $\{\psi_{j,t,\omega}(x)\}_{t=2^{-j}l|l\in\mathbf{Z}}$. Linear combinations (specified by h_1 and h_2) of the basis functions of the space $\mathbf{W}_{j,\omega}$

[1] We have used h_1 and h_2 to refer to the conjugate mirror filters which are usually [22] referred to as h and g. This notation was used to avoid confusion with our dictionary elements g.

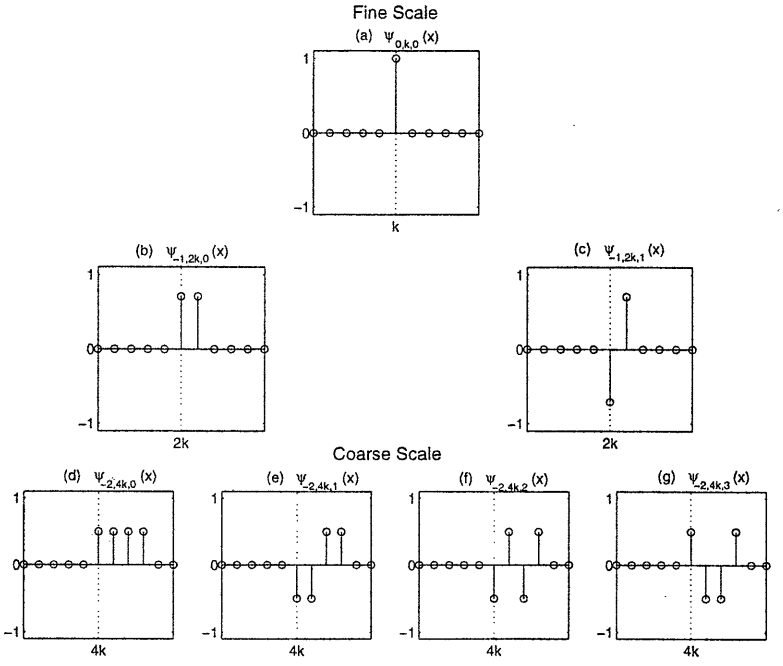


Figure 2-4: Representative elements of Haar wavelet packet dictionary. Note that the finest resolution has been designated $j = 0$.

yield the basis functions of the spaces $\mathbf{W}_{j-1,2\omega}$ and $\mathbf{W}_{j-1,2\omega+1}$. That is,

$$\psi_{j-1,0,2\omega}(x) = \sum_{n=-\infty}^{+\infty} h_1[n]\psi_{j,2^{-j}n,\omega}(x) \tag{2.13}$$

$$\psi_{j-1,0,2\omega+1}(x) = \sum_{n=-\infty}^{+\infty} h_2[n]\psi_{j,2^{-j}n,\omega}(x). \tag{2.14}$$

Figure 2-4 shows sample elements from a wavelet packet dictionary based on the Haar wavelet.

In this work, we will denote elements of the wavelet packet dictionary $g_\gamma = \psi_{j,t,\omega}$ where $\gamma = (j, t, \omega)$ is now a joint index over scale, translation, and frequency. This is in contrast to the cubic b-spline dictionary which was indexed only by scale and translation. Also in this work, it will be convenient to use the time-frequency plane to represent the support of wavelet packet dictionary elements. Figure 2-5 shows

the time-frequency plane representation of elements from a wavelet packet dictionary based on the Haar wavelet; wavelet packet dictionaries based on other wavelets exhibit similar time-frequency behavior. Each rectangle in the time-frequency plane represents the area over which the energy of the function is concentrated. The scale of the element determines the size of the rectangle and the translation and frequency of the element determine the location in the time-frequency plane. Note that Figure 2-5 shows that the wavelet packet transform divides the frequency axis into intervals of different sizes in a way that does not depend on the frequency. This is in contrast to the wavelet transform which divided the frequency axis into large frequency intervals at high frequencies and small frequency intervals at low frequencies.

Note the following important properties of the wavelet packet dictionary. First, the wavelet packet dictionary is redundant. The collection of functions of the same scale (size) is a basis for $\mathbb{R}^P$, where P is the length of f [22]. The entire dictionary is a collection of bases and is therefore redundant. Second, equations (2.13) and (2.14) imply that each dictionary element is the weighted sum of dictionary elements at a finer scale. Thus, just as was the case for the cubic b-spline dictionary, it is possible to define for each element, g_γ, an associated set of indices $I_\gamma(k)$ where the functions indexed by $I_\gamma(k)$ are the function g_γ and the dictionary elements at scale $j+k$ which when properly summed yield g_γ. That is,

$$I_\gamma(k) = \left\{ (j+k, t_i, \omega/2) \,\middle|\, g_\gamma = \sum_{i=1}^{L} c_i g_{j+k,t_i,\omega/2} \right\} \bigcup \{\gamma\} \qquad \text{if } \omega \text{ even} \quad (2.15)$$

$$I_\gamma(k) = \left\{ (j+k, t_i, (\omega-1)/2) \,\middle|\, g_\gamma = \sum_{i=1}^{L} c_i g_{j+k,t_i,(\omega-1)/2} \right\} \bigcup \{\gamma\} \quad \text{if } \omega \text{ odd} \quad (2.16)$$

Again, γ has been appended to $I_\gamma(k)$ for convenience later. This definition for $I_\gamma(k)$ is analogous to the definition for $I_\gamma(k)$ for the cubic b-spline dictionary given in (2.7). This property is will be important when we define the new similarity measure in

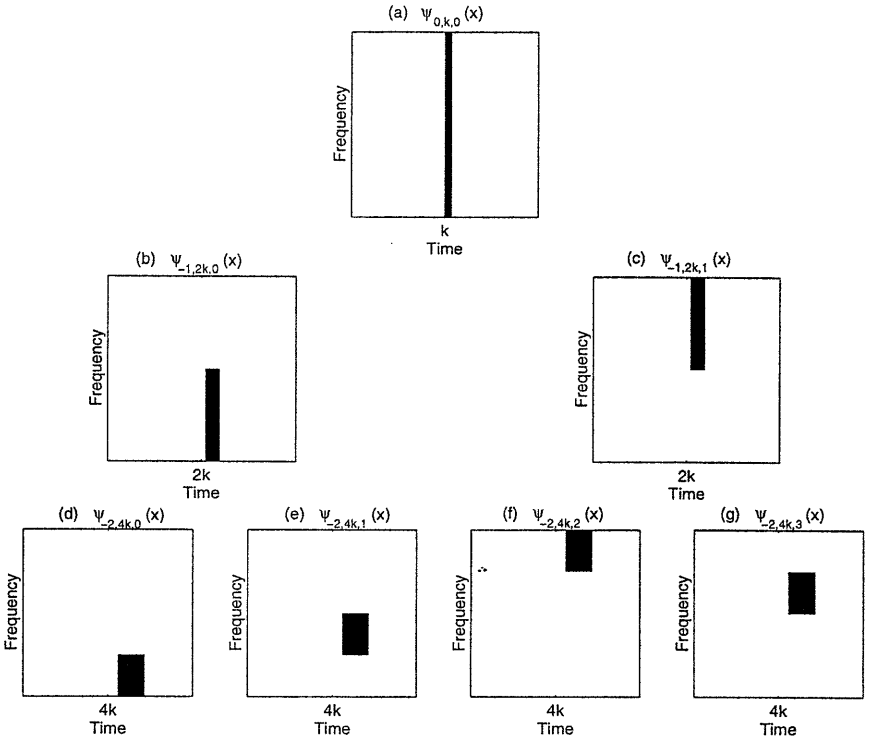


Figure 2-5: Time Frequency representations of the Haar Wavelet Packet Dictionary. Each rectangle represents the area over which the energy of the function is concentrated. For each dictionary element, scale determines the size of the rectangle and translation and frequency determine the location.

Section 3.1.

2.2 Matching Pursuit

Matching pursuit (MP) is an iterative, adaptive algorithm for signal decomposition [26]. The original signal is decomposed as the sum of the most contributive elements from a dictionary set. In MP, elements are chosen one at a time. At each iteration, the most contributive element is defined to be the one which minimizes the L^2 norm of the fit error, or equivalently, has a maximum inner product with the last residual. The underlying goal of the matching pursuit algorithm is to extract local

physical features of the signal. For certain basic examples, MP yields un-intuitive results because of the greedy nature of the algorithm. In this section, we describe the matching pursuit algorithm for signal decomposition and provide examples where this algorithm fails to extract local physical features of the signal.

The matching pursuit algorithm is defined as follows. Let f be the function to be decomposed and $\mathcal{D} = \{g_\gamma | \gamma \in \Gamma\}$ be the set of dictionary vectors with $\|g_\gamma\| = 1$. The first element $g_{\gamma_0} \in \mathcal{D}$ is chosen such that

$$g_{\gamma_0} = \arg\max_{g_\gamma} | < f, g_\gamma > |. \tag{2.17}$$

The function f is then decomposed as

$$f =< f, g_{\gamma_0} > g_{\gamma_0} + Rf \tag{2.18}$$

where Rf is the residual. It is easy to show that g_{γ_0}, the element with the largest inner product with f, also yields the smallest residual energy. An energy conservation equation follows :

$$\|f\|^2 = | < f, g_{\gamma_0} > |^2 + \|Rf\|^2. \tag{2.19}$$

Subsequent elements are chosen similarly to be the best fit to the previous residual. That is,

$$g_{\gamma_n} = \arg\max_{g_\gamma} | < R^n f, g_\gamma > | \tag{2.20}$$

where $R^n f$ is the n-th residual. This yields a cumulative decomposition of

$$f = \sum_{n=0}^{m-1} < R^n f, g_{\gamma_n} > g_{\gamma_n} + R^m f \tag{2.21}$$

and a cumulative energy equation of

$$\|f\|^2 = \sum_{n=0}^{m-1} | < R^n f, g_{\gamma_n} > |^2 + \|R^m f\|^2. \tag{2.22}$$

Further, Mallat and Zhang [26] show that if f and $g_\gamma \in \mathcal{D}$ are discrete functions with a finite number of samples, then $\|R^n f\|$ decays exponentially.

In the matching pursuit algorithm, the inner product is used as the measure of correlation or similarity between a function and elements of the dictionary. It is well known [14] that the inner product may not be a good measure of similarity between two signals. As we show in Section 2.2.2, the inner product-based similarity measure makes MP a greedy algorithm which yields un-intuitive decompositions for some simple examples.

2.2.1 Computational Complexity

The MP algorithm can be efficiently implemented so that the number of operations required for each iteration is proportional to the size of the dictionary. Let Q be the number of elements in the dictionary. To begin, compute and store $< f, g_\gamma >$ for all $g_\gamma \in \mathcal{D}$ and $< g_\gamma, g_\beta >$ for all $g_\gamma, g_\beta \in \mathcal{D}$. This requires $\mathcal{O}(Q^2 P)$ (where P is the length of the signal) operations, although this initial computation can be significantly reduced by taking advantage of specific dictionary structures. These initial computations reduce the computational requirements for subsequent iterations. Since

$$R^{n+1} f = R^n f - < R^n f, g_{\gamma_n} > g_{\gamma_n}, \tag{2.23}$$

we know that

$$< R^{n+1} f, g_\gamma > = < R^n f, g_\gamma > - < R^n f, g_{\gamma_n} >< g_{\gamma_n}, g_\gamma > . \tag{2.24}$$

Each term on the right hand side of (2.24) has been stored at the previous iteration; therefore, the computation of $< R^{n+1}f, g_\gamma >$ for all $g_\gamma \in \mathcal{D}$ requires $\mathcal{O}(2Q)$ operations. In [26], the authors show that computational complexity may be further reduced by performing MP at each step over a subdictionary and then refining the estimate using a Newton search strategy. Thus, there exist implementations of MP where the computational burden is reasonable.

2.2.2 Drawbacks of MP

The MP approach works well for many types of signals. It has been shown to be especially useful for extracting structure from signals which consist of components with widely varying time-frequency localizations [26]. MP is a greedy algorithm in the sense that the element chosen at each step is the one which absorbs the most remaining energy in the signal. In practice, this results in an algorithm that sacrifices local fit for global fit, as illustrated by the following example.

The twin peaks function, f, illustrated in Figure 2-6, is the sum of two cubic b-splines at the same scale but different, nearby translates. Let the dictionary $\mathcal{D}$ consist of cubic b-splines at a wide range of translates and scales, including those used to construct f. This dictionary is well suited for the signal under consideration. For the twin peaks example, the first element chosen by MP is one which does not match either of the two functions which are the true components of f. This is illustrated in Figure 2-6 which shows the original function and the first element chosen by MP, g_{γ_0}. We can gain insight into the behavior of MP for the twin peaks example by studying the projection graph which is defined to be the inner product of f with each dictionary element as a function of the scale and translation of the dictionary element. A contour plot of the projection graph is shown in Figure 2-7. The proximity of the two components of f leads to a maximum of the similarity function (the

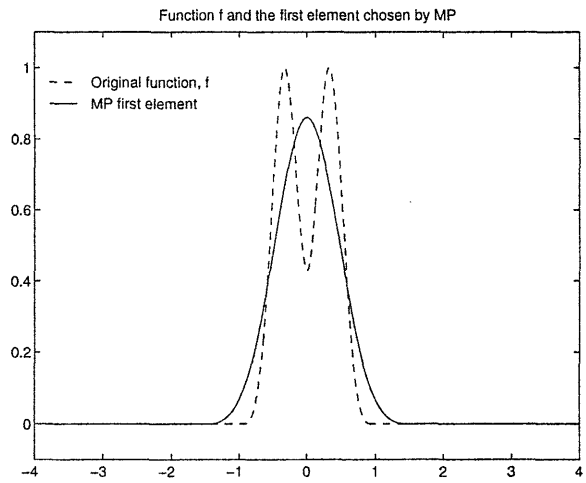


Figure 2-6: The twin peaks function and first element chosen by MP.

inner product) which is not at the correct translation and scale of either element. The first MP residual is shown in Figure 2-8. The residual has a large negative component at $t = 0$ where the original function was positive. Thus, instead of finding significant features of the signal, MP has effectively introduced new artifacts which the algorithm will have to account for by fitting additional elements. This problem is further compounded as subsequent elements are chosen by MP in an effort to correct the initial mistake. Figure 2-9 shows the first ten elements chosen by MP to represent f. Here, note that the elements chosen by MP do not correspond to the physical features of the function. In fact, many of these are "non-features" which only serve to correct mistakes from previous stages.

2.3 Basis Pursuit

The basis pursuit (BP) principle [6] is to find the decomposition given in (2.1) which minimizes the ℓ^1-norm of the coefficients λ_n. The examples presented in [6] indicate that basis pursuit yields decompositions which are sparse and show super-resolution.

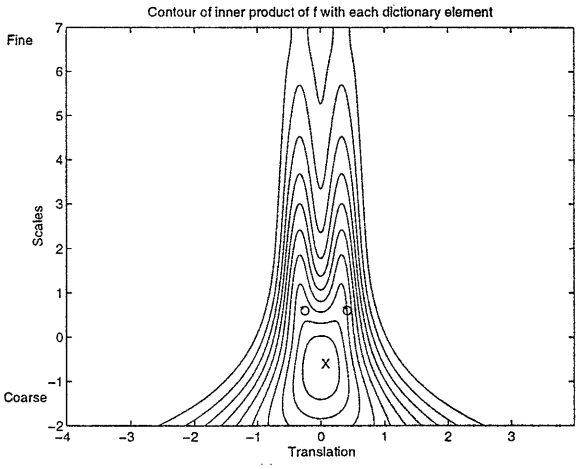


Figure 2-7: The projection graph is the inner product of the function with each dictionary element which is indexed by scale and translation. This figure shows the contour of the projection graph. X marks maximum inner product. O marks location of true elements of function.

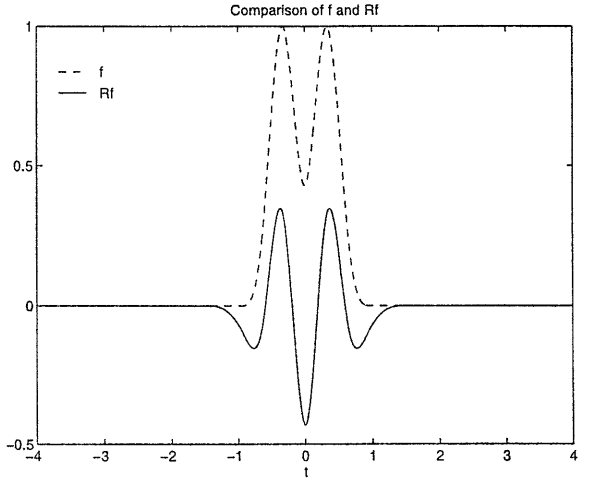


Figure 2-8: First residual generated by MP.

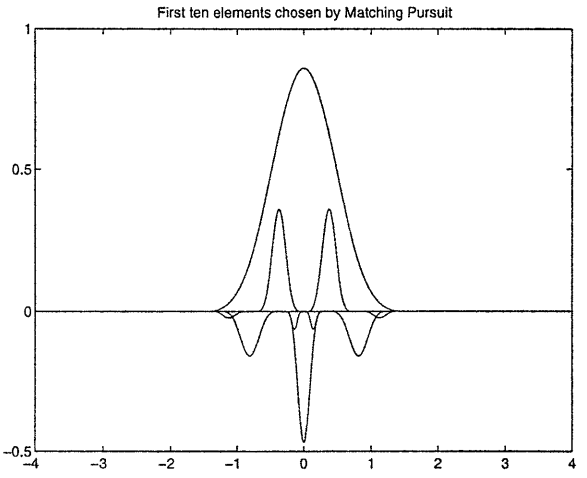


Figure 2-9: The first ten elements picked by MP.

Thus, in general, BP does not exhibit problems highlighted by the twin peaks example. An important drawback in the implementation of BP is that of computational complexity. Since basis pursuit decompositions are based on solving a large-scale optimization problem, there exist examples where the decomposition may not be completed in a reasonable amount of time, as stressed in [6].

To implement the basis pursuit principle, an equivalent linear programming problem is solved. The basis pursuit principle is to find the decomposition of the form (2.1), which minimizes the ℓ^1-norm of the coefficients λ_n. Equivalently, in matrix notation, the basis pursuit problem statement is

$$\min \|\Lambda\|_1 \text{ subject to } f = \Lambda^T G \tag{2.25}$$

where G is a matrix of all the vectors in $\mathcal{D}$, $f \in \Re^p$, $\Lambda \in \Re^q$, and $G \in \Re^{p \times q}$. The solution to (2.25) is found by formulating and solving an equivalent standard form linear program.

Two algorithms from linear programming are proposed in [6] to implement the basis pursuit principle : the simplex method and interior point methods. For a signal

of length P and a dictionary of Q elements, the BP principle implemented using the simplex method requires an average of $\mathcal{O}(Q^2P)$ calculations, though it could require as many as $\mathcal{O}(2^P - 1)\mathcal{O}(QP)$ calculations. The complexity of interior point methods depends on the implementation. Interior point methods are typically polynomial in Q and P [11, 17]. Thus, the implementation of BP is computationally complex, as compared to MP.

Chapter 3

High Resolution Pursuit

The objective of high resolution pursuit (HRP) is to combine the computational speed of MP and the super-resolution of BP. The HRP algorithm, developed in this chapter, is similar in structure to the MP algorithm and in fact has the same computational complexity as MP. In contrast to MP, HRP employs a similarity measure which emphasizes local fit over global fit, and is thus able to achieve super-resolution similar to that exhibited by BP.

The organization of this chapter is as follows. Section 3.1 describes the HRP similarity measure and the basic algorithm. In Section 3.2, two interpretations of the HRP algorithm are described. We show a geometric interpretation as well as a constrained maximization interpretation of the HRP algorithm. In Section 3.3, the convergence properties of HRP are discussed. Next, Section 3.4 describes the HRP computational complexity. Then, the HRP algorithm is demonstrated in Section 3.5.

3.1 The HRP Algorithm

Let us begin by developing our intuition about the MP similarity measure using cubic b-spline dictionaries. For the case of cubic b-spline dictionaries, the inner product (the MP similarity measure) of f with dictionary element g_γ can be shown to be a weighted average of the inner products of f with finer scale dictionary elements. Recall the notation introduced in Section 2.1.1, where elements of the cubic b-spline dictionary at scale j and translation t are denoted $g_{j,t}$, or, equivalently, g_γ where γ is a joint index over scale and translation. Any cubic b-spline may be written as the sum of finer scale cubic b-splines composing g_γ which are also dictionary elements, as expressed in (2.8). Since $g_{j,t}$ may be represented as the weighted sum of cubic b-splines that are k scales finer, the inner product $< f, g_{j,t} >$ may also be expressed in terms of finer scale inner products,

$$< f, g_{j,t} >= \sum_{i=1}^{L} c_i < f, g_{j+k,t_i} >, \tag{3.1}$$

or, equivalently,

$$< f, g_\gamma >= \sum_{i \in I_\gamma(k)} c_i < f, g_i > \tag{3.2}$$

where $I_\gamma(k)$ is as defined in (2.7). In other words, the inner product of f and g_γ may be interpreted as the weighted average of the inner products of f with high resolution dictionary elements.

This interpretation of the MP similarity measure yields intuition about what form a new, more locally-sensitive similarity measure might take. Even though each of the high resolution correlations in (3.1) (i.e. $\{< f, g_i >\}_{i \in I_\gamma(k)}$) is sensitive to local structure, the (weighted) averaging process of (3.1) renders $< f, g_\gamma >$ relatively insensitive to local structure. One can imagine that some other combination of the high resolution correlations, $\{< f, g_i >\}_{i \in I_\gamma(k)}$, might yield a new measure of similarity between

f and g_γ, which is more sensitive to local mismatch. Intuitively, this new similarity measure should be dominated by worst local fit. For example, the minimum of $\{< f, g_i >\}_{i \in I_\gamma(k)}$ is dominated by worst local fit.

The similarity measure we propose is essentially the minimum over $\{| < f, g_i > |\}$ for all $i \in I_\gamma(k)$. Our new similarity measure, $S(f, g_\gamma)$, is given by

$$S(f, g_\gamma) = m(f, g_\gamma) s(f, g_\gamma) \tag{3.3}$$

$$s(f, g_\gamma) = \min_{i \in I_\gamma(k)} \frac{| < f, g_i > |}{| < g_i, g_\gamma > |} \tag{3.4}$$

$$m(f, g_\gamma) = \begin{cases} +1 & \text{if } \frac{<f, g_i>}{<g_i, g_\gamma>} > 0 \text{ for all } i \in I_\gamma(k) \\ -1 & \text{if } \frac{<f, g_i>}{<g_i, g_\gamma>} < 0 \text{ for all } i \in I_\gamma(k) \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

where $I_\gamma(k)$ is given in (2.7) for cubic b-spline dictionaries. In this work, we will sometimes refer to the elements indexed by I_γ as the subfamily associated with dictionary element g_γ. The denominator of $s(f, g_\gamma)$ is a normalization factor which yields $S(g_\gamma, g_\gamma) = 1$. The term $m(f, g_\gamma)$ is included to assure that oscillatory functions yield similarity measures of zero with coarse scale dictionary elements. That is, if the scale of g_γ is coarser than the scale of the oscillations of f, then $m(f, g_\gamma)$ will force $S(f, g_\gamma)$ to be zero. This implies that for an oscillatory f the coarsest cubic b-spline element to yield a non-zero similarity measure would be the one whose scale matches the scale of the oscillations of f.

The HRP algorithm follows essentially the same procedure as the MP algorithm. In HRP, the elements are chosen one at a time. At each iteration, the element chosen by HRP is the one which maximizes the HRP similarity measure with the previous

residual. That is, the element chosen at the n-th step, g_{γ_n} is given by

$$g_{\gamma_n} = \arg\max_{\gamma\in\Gamma} |S(R^n f, g_\gamma)| \tag{3.6}$$

where the similarity between the n-th residual, $R^n f$, and a dictionary element, g_γ, is given by $S(R^n f, g_\gamma) = m(R^n f, g_\gamma)s(R^n f, g_\gamma)$ as defined in (3.4) and (3.5). In the HRP algorithm, the $n+1$-st residual is then generated as

$$R^{n+1} f = R^n f - S(R^n f, g_{\gamma_n}) g_{\gamma_n}. \tag{3.7}$$

One further note about the parameter k which essentially controls the depth of the resolution of the HRP algorithm. The HRP decomposition will change as a function of k, as will be illustrated in Section 3.5. When k is set to zero, the HRP decomposition will be identical to the MP decomposition. At the other extreme when k is very large, the fine scale elements of $I_\gamma(k)$ will approach Diracs and the HRP decomposition will be highly sensitive to noise in the signal. For our work k has been chosen empirically. In general, k should be regarded as a means to incorporate prior knowledge.

Although our discussion has concentrated on cubic b-spline dictionaries, the HRP algorithm and the intuition behind it extend in a straightforward way to other dictionaries where coarse scale elements may be written as the weighted sum of fine scale elements. The HRP algorithm in these more general dictionaries is exactly the same as before, but with the set $I_\gamma(k)$ defined to be analogous to (2.7). That is, the functions indexed by $I_\gamma(k)$ are g_γ and a set of functions k scales finer which yield g_γ when properly weighted and summed. For example, in the wavelet packet dictionary which has the property that coarse scale elements are the sum of fine scale elements, the analogous definition for $I_\gamma(k)$ is given in (2.15) and (2.16).

A related pointed that we should mention is that we have *chosen* to use a particular

structure for $I_\gamma(k)$, namely that $I_\gamma(k)$ indexes the set of functions k scales finer which can be summed to yield g_γ. Many other choices of I_γ might also be appropriate. For example, one could imagine using combinations of finer scale cubic b-splines that are not all at the same scale which also sum to g_γ. In some cases, this may be a way to incorporate prior knowledge.

Finally, note that the HRP algorithm is not limited dictionaries where coarse scale elements may be constructed as the weighted sum of finer scale elements. For dictionaries where it is not possible to represent coarse scale elements exactly as the sum of finer scale elements, it would be necessary to specify for each dictionary element g_γ a local family I_γ which consists of finer scale functions which somehow capture the local structure of g_γ.

3.2 HRP Interpretations

In this section, two interpretations of the HRP algorithm are described. First, for the case where the functions under consideration are 2-D vectors, the HRP similarity measure may be interpreted geometrically as an oblique projection of f onto g_γ and the magnitude of the oblique projection is determined by the associated set $I_\gamma(k)$. This is in contrast to MP where the similarity measure is an orthogonal projection of f onto g_γ. Second, we show that the element which maximizes the HRP similarity measure, $|S(R^{n-1}f, g_\gamma)|$, is the same one which solves a *constrained maximization* of $\|R^n f - R^{n-1}f\|$. This mirrors the development of MP in Section 2.2 where we noted that the element which maximized the inner product similarity measure, $| < R^{n-1}f, g_\gamma > |$, is the same one which solves an *unconstrained minimization* of $\|R^n f\|$.

3.2.1 Geometric Interpretation

In the following discussion, we use a vector example to illustrate the differences between MP and HRP and to show a geometric interpretation of the HRP similarity measure. The goal of the following discussion is to illustrate two points. First, HRP is better than MP at preserving sparsity. That is, given a function which may be synthesized as the sum of n dictionary elements, often the MP algorithm yields a decomposition of $m > n$ elements. On the other hand, by emphasizing local over global fit, HRP yields decompositions which preserve sparsity. Second, the HRP similarity measure, $S(f, g_\gamma)$, is a projection of f onto g_γ. The magnitude of this projection is determined by the associated subfamily I_γ.

This discussion will be based on the following simple, two-dimensional vector example. Consider the decomposition of the vector $\mathbf{f} \in I\!R^2$ over the dictionary consisting of $\{\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3\}$, where $\|\mathbf{g}_\gamma\| = 1$ for $\gamma = 1, 2, 3$. For this example, we use boldface notation to emphasize that $\mathbf{f}$ and $\mathbf{g}_\gamma$ are vectors. These vectors are illustrated in Figure 3-1 and are given by

$$\mathbf{g}_1 = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{3.8}$$

$$\mathbf{g}_2 = \begin{bmatrix} 0 & 1 \end{bmatrix} \tag{3.9}$$

$$\mathbf{g}_3 = \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix} \tag{3.10}$$

$$\mathbf{f} = \begin{bmatrix} \frac{1}{2} & \frac{3}{4} \end{bmatrix}. \tag{3.11}$$

The following scale structure, which will be useful for applying HRP to this example, is imposed in this vector dictionary. The elements $\mathbf{g}_1$ and $\mathbf{g}_2$ are finest scale elements, and are similar to Diracs, which were the finest scale elements in the cubic b-spline dictionary. The element $\mathbf{g}_3$ is a coarse scale element. In this vector dictionary, the coarse scale element ($\mathbf{g}_3$) may be constructed as the weighted sum of fine scale

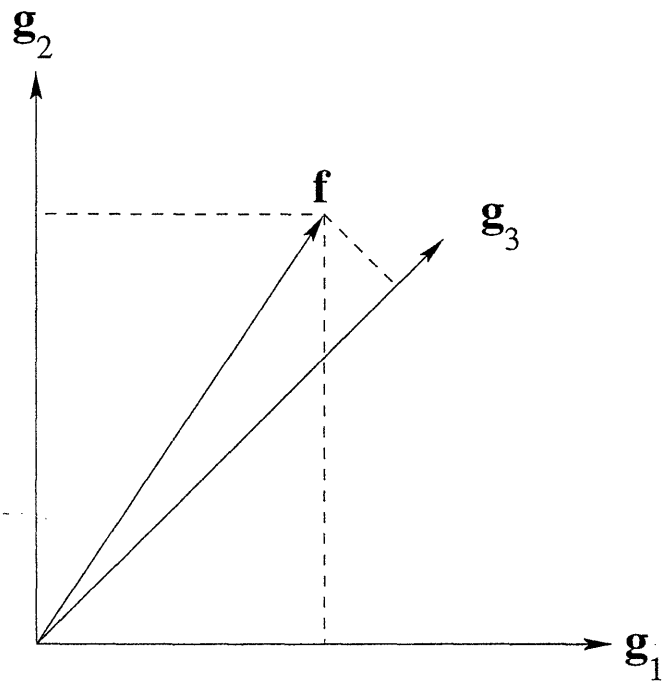


Figure 3-1: The function $\mathbf{f}$ and the dictionary $\{\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3\}$.

elements ($\mathbf{g}_1$ and $\mathbf{g}_2$). Thus, the scale structure of this simple vector dictionary is similar to the scale structure of the cubic b-spline dictionary.

We begin by examining the performance of MP. The first element chosen by MP is the dictionary element which maximizes the inner product $| < \mathbf{f}, \mathbf{g}_\gamma > |$. The projections of $\mathbf{f}$ onto each of the dictionary elements are shown by the dashed lines in Figure 3-1 and are also given in Table 3.1. Therefore, the first element chosen by MP is $\mathbf{g}_3$. The resulting residual is shown in Figure 3-2. To achieve an exact decomposition, a total of three elements must be used : the second element chosen by MP is $\mathbf{g}_2$, and the third element is $\mathbf{g}_1$.

For this example, the MP algorithm does not preserve sparsity. That is, the MP decomposition requires three elements to decompose a function that could have been represented with two. In effect, MP has introduced an artifact in $\mathbf{Rf}$. That is, $\mathbf{Rf}$ has

$< \mathbf{f}, \mathbf{g}_1 >$	0.5
$< \mathbf{f}, \mathbf{g}_2 >$	0.75
$< \mathbf{f}, \mathbf{g}_3 >$	0.8839

Table 3.1: Projection of $\mathbf{f}$ onto each dictionary element

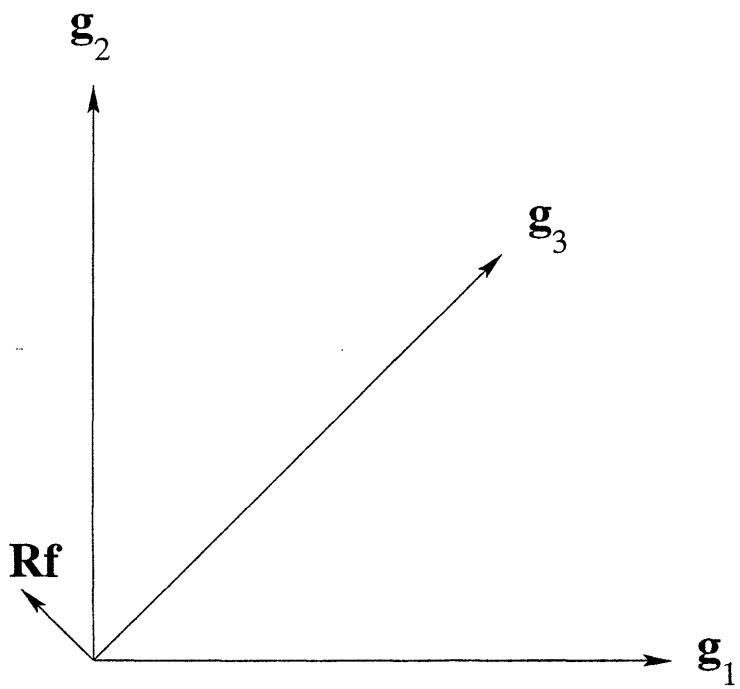


Figure 3-2: The MP residual $\mathbf{Rf}$ and the dictionary $\{\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3\}$.

a negative component in the direction of $\mathbf{g}_1$ which did not exist in $\mathbf{f}$, and which will have to be corrected for in subsequent stages. This is a parallel to the introduction of artifacts illustrated in the twin peaks example of Section 2.2.2.

In contrast, consider the performance of HRP. To apply HRP, we must specify a set I_γ associated with each dictionary element. Since we have been given a scale structure that is similar to that of the cubic b-spline dictionary, we can define the sets I_γ to have a similar structure to that in the cubic b-spline dictionary. That is, the associated subfamily for a finest scale element is just itself, and the associated subfamily for a coarse scale element is the element itself plus the set of finer scale elements which when properly weighted and summed yield the coarse element. This implies $I_1 = \{1\}$, $I_2 = \{2\}$, and $I_3 = \{1, 2, 3\}$. Applying (3.3),

$$S(\mathbf{f}, \mathbf{g}_1) = < \mathbf{f}, \mathbf{g}_1 > \tag{3.12}$$

$$S(\mathbf{f}, \mathbf{g}_2) = < \mathbf{f}, \mathbf{g}_2 > \tag{3.13}$$

$$S(\mathbf{f}, \mathbf{g}_3) = \min\left(\left|\frac{< \mathbf{f}, \mathbf{g}_1 >}{< \mathbf{g}_3, \mathbf{g}_1 >}\right|, \left|\frac{< \mathbf{f}, \mathbf{g}_2 >}{< \mathbf{g}_3, \mathbf{g}_2 >}\right|, | < \mathbf{f}, \mathbf{g}_3 > |\right) \tag{3.14}$$

It is instructive to show how $S(\mathbf{f}, \mathbf{g}_\gamma)$ for $\gamma = 1, 2, 3$ may be determined graphically. Clearly, $S(\mathbf{f}, \mathbf{g}_1)$, $S(\mathbf{f}, \mathbf{g}_2)$, and the last term in on the right-hand side of (3.14) can be determined graphically. The first two terms in (3.14) may be determined graphically as follows. Let $\mathbf{r} = \alpha \mathbf{g}_3$ denote the vector which satisfies

$$< \mathbf{r}, \mathbf{g}_1 > = < \mathbf{f}, \mathbf{g}_1 > \tag{3.15}$$

for some value of α. This equality is shown in Figure 3-3. Equivalently,

$$< \mathbf{f}, \mathbf{g}_1 > = \alpha < \mathbf{g}_3, \mathbf{g}_1 > \tag{3.16}$$

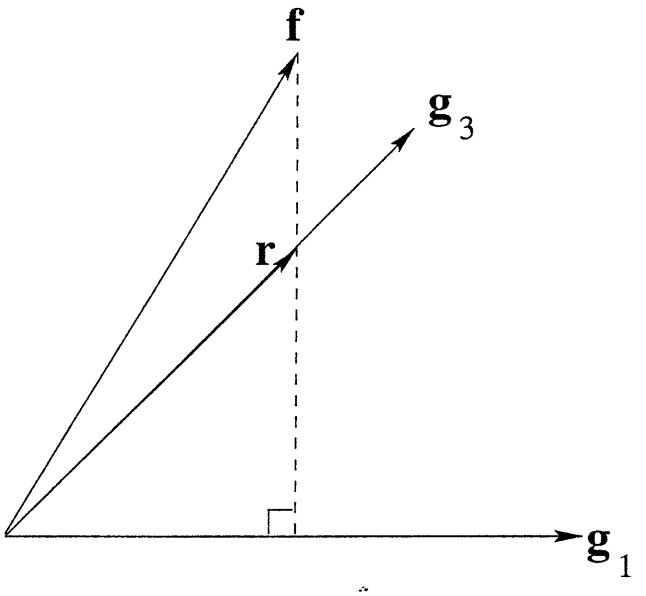


Figure 3-3: A geometric interpretation of the HRP similarity measure.

Since $\|\mathbf{g}_3\| = 1$, it follows that $\|\mathbf{r}\| = \left|\frac{<\mathbf{f},\mathbf{g}_1>}{<\mathbf{g}_3,\mathbf{g}_1>}\right|$ which is precisely the first term in (3.14). This term is the magnitude of the oblique projection of $\mathbf{f}$ onto $\mathbf{g}_3$ generated by the orthogonal projection of $\mathbf{f}$ onto $\mathbf{g}_1$. Similarly, the second term in (3.14) is equal to the magnitude of the oblique projection of $\mathbf{f}$ onto $\mathbf{g}_3$ generated by the orthogonal projection of $\mathbf{f}$ onto $\mathbf{g}_2$. Thus, $S(\mathbf{f}, \mathbf{g}_3)$ is the minimum of the three projections of $\mathbf{f}$ onto $\mathbf{g}_3$ shown in Figure 3-4.

The first element chosen by HRP is the one which maximizes $|S(\mathbf{f}, \mathbf{g}_\gamma)|$. For the example illustrated in Figure 3-1, the values of $|S(\mathbf{f}, \mathbf{g}_\gamma)|$ are listed in the first column of Table 3.2. Thus, the first element chosen by HRP is $\mathbf{g}_2$. The resulting residual, $\mathbf{Rf}$, is $[\ \frac{1}{2}\ \ 0\ ]$. The values of $|S(\mathbf{Rf}, \mathbf{g}_\gamma)|$ are listed in the second column of Table 3.2, indicating that the second element chosen by HRP is $\mathbf{g}_1$. Thus, an exact decomposition is achieved with two elements and sparsity of the synthesis is preserved.

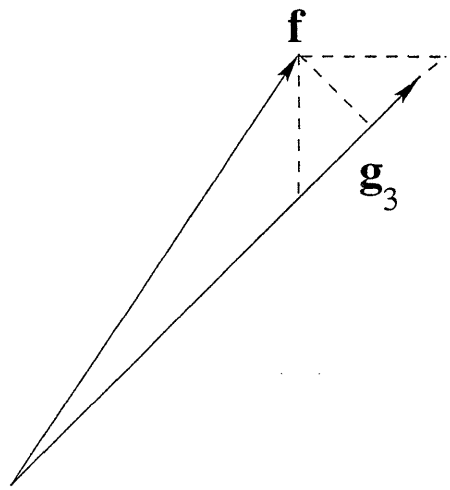


Figure 3-4: Projections of $\mathbf{f}$ onto $\mathbf{g}_3$

γ	$S(\mathbf{f}, \mathbf{g}_\gamma)$	$S(\mathbf{Rf}, \mathbf{g}_\gamma)$
1	0.5	0.5
2	0.75	0
3	0.7071	0

Table 3.2: HRP similarity of $\mathbf{f}$ and $\mathbf{Rf}$ with each dictionary element.

3.2.2 A Constrained Maximization Interpretation

Additional insight may be gained through the following alternative interpretation of the HRP algorithm. This section shows that the element which maximizes the HRP similarity measure, $|S(R^{n-1}f, g_\gamma)|$, is the same one which solves a *constrained maximization* of $\|R^n f - R^{n-1}f\|$. This discussion mirrors the development of MP in Section 2.2 which noted that the element which maximized the inner product similarity measure, $| < R^{n-1}f, g_\gamma > |$ was the same one which solves an *unconstrained minimization* of $\|R^n f\|$.

Let us begin by introducing some notation and mathematically describing the interpretation of HRP as a constrained maximization. In the interest of cleaner notation, we will restrict our discussion to the case of the first element chosen by HRP. That is, we will show that the element which maximizes $|S(Rf, g_\gamma)|$ is the same one which solves a constrained maximization of $\|Rf - f\|$. The discussion that follows immediately extends to subsequent stages. Suppose that the first residual generated by any dictionary element, g_γ, is denoted $R_\gamma f$ and is of the form

$$R_\gamma f = f - C(f, g_\gamma) g_\gamma \tag{3.17}$$

where $C(f, g_\gamma)$ is an unknown scalar. It follows that

$$\|R_\gamma f - f\| = |C(f, g_\gamma)|. \tag{3.18}$$

Now consider the maximization of $\|R_\gamma f - f\|$ or equivalently $|C(f, g_\gamma)|$ under the following constraints :

$$| < R_\gamma f, g_i > | \;\leq\; | < f, g_i > | \qquad \text{for all } i \in I_\gamma(k) \tag{3.19}$$

$$\text{sign}(< R_\gamma f, g_i >) \;=\; \text{sign}(< f, g_i >) \qquad \text{for all } i \in I_\gamma(k). \tag{3.20}$$

We will show that for any $R_\gamma f$ to meet constraints (3.19) and (3.20) $|C(f, g_\gamma)|$ must be less than or equal to $|S(f, g_\gamma)|$ as defined in (3.3). Also, the maximum value of any particular $R_\gamma f$ which meets the constraints occurs when $C(f, g_\gamma) = S(f, g_\gamma)$. Further, we will show that the dictionary element which solves this constrained maximization problem is the same as the first element chosen by HRP. In equation form,

$$g_{\gamma_0} = \arg\max_{\gamma \in \Gamma} |S(f, g_\gamma)| \tag{3.21}$$

$$= \arg\max_{\gamma \in \Gamma} \|R_\gamma - f\| \text{ subject to (3.19) and (3.20).} \tag{3.22}$$

Before showing that the element which solves the constrained maximization is the same as the first element chosen by HRP, we will discuss the constraints in (3.19) and (3.20). The constraint in (3.19) captures the idea that the magnitude of the residual projection onto any element should decrease both globally and locally. In other words, if g_γ is well matched to f, then the projection of the residual onto g_γ should decrease, and the projection of the residual onto all the local structures which make up g_γ (i.e. g_i for $i \in I_\gamma(k)$) should decrease. The constraint in (3.20) is motivated by the shortcomings of the MP algorithm. Recall that in the MP decomposition of the twin peaks example, Rf had a large negative component at $t = 0$ where the original function was positive. This negative component was not a feature of the function, but rather an artifact introduced by MP. Thus, constraint (3.20) steers the decomposition away from including artifacts such as those introduced by MP in the twin peaks example. It is important to note that the two constraints effectively balance one another and together imply that the projection onto all local structures of g_γ must decrease, but not so much that the sign of the projections onto local structures change.

We can also illustrate the effect of these constraints in terms of the simple example shown in Figure 3-1. Recall that for this example the function **f** is to be decomposed

over a dictionary set $\{\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3\}$, $\mathbf{g}_1$ and $\mathbf{g}_2$ are finest scale elements (i.e. they have no associated subfamilies), and $\mathbf{g}_3$ has an associated subfamily indexed by $I_3 = \{1, 2, 3\}$. Figure 3-5 shows the residuals generated for dictionary element $\mathbf{g}_3$ by three different values of $C(\mathbf{f}, \mathbf{g}_3)$. Figure 3-5a shows the residual that results when $C(\mathbf{f}, \mathbf{g}_3) = 1.5$. In this case,

$$\mathbf{R}_3\mathbf{f} = [\ -0.5607 \quad -0.3107\ ]. \tag{3.23}$$

This residual violates both constraints (3.19) and (3.20), since $| < \mathbf{R}_3\mathbf{f}, \mathbf{g}_1 > | = 0.5607$ while $| < \mathbf{f}, \mathbf{g}_1 > |$ is only 0.5 and since both $< \mathbf{R}_3\mathbf{f}, \mathbf{g}_1 >$ and $| < \mathbf{R}_3\mathbf{f}, \mathbf{g}_2 > |$ are negative while $< \mathbf{f}, \mathbf{g}_1 >$ and $< \mathbf{f}, \mathbf{g}_2 >$ are positive. Figure 3-5b shows the residual that results when $C(\mathbf{f}, \mathbf{g}_3) = 1.0$. In this case,

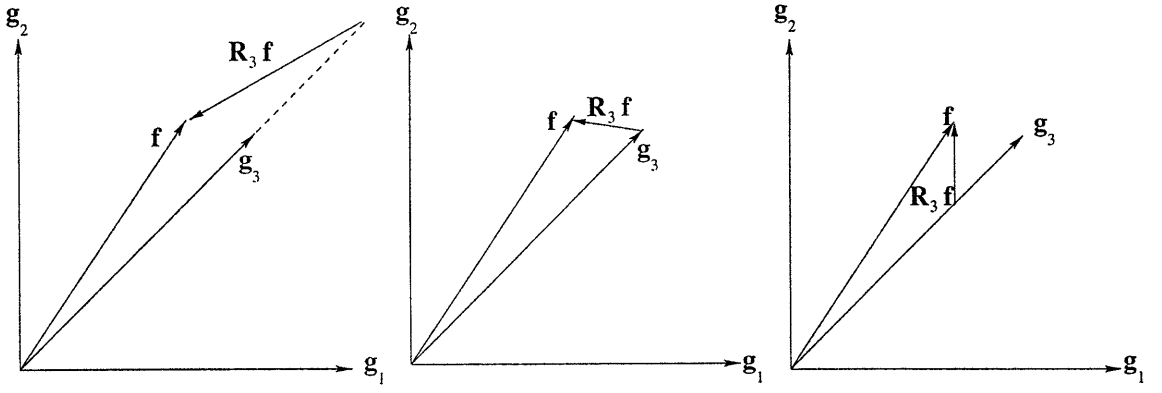
$$\mathbf{R}_3\mathbf{f} = [\ -0.2071 \quad 0.0429\ ]. \tag{3.24}$$

This residual violates only constraint (3.20), since $< \mathbf{R}_3\mathbf{f}, \mathbf{g}_1 >$ is negative while $< \mathbf{f}, \mathbf{g}_1 >$ is positive. Finally, Figure 3-5c shows the residual that results when $C(\mathbf{f}, \mathbf{g}_3) = .7071$ which is precisely the value of $S(\mathbf{f}, \mathbf{g}_3)$ in Table 3.2. In this case, the residual is given $\mathbf{R}_3\mathbf{f} = [\ 0 \quad 0.25\ ]$. This residual does not violate either constraint. In fact, this value of $C(\mathbf{f}, \mathbf{g}_3)$ is the maximum weight that can be given to $\mathbf{g}_3$ and still produce a residual which does not violate the constraints.

Returning to the general case, we now show that the element which maximizes $|S(f, g_\gamma)|$ is the same as the one which maximizes $\|R_\gamma f - f\|$ subject to (3.19) and (3.20). Assume for now that

$$\frac{< f, g_i >}{< g_\gamma, g_i >} > 0 \qquad \text{for all } g_i \in I_\gamma(k). \tag{3.25}$$

Some simple algebraic manipulation yields a sequence of alternative forms for con-



(a) $\mathbf{R_3 f}$ for $C(\mathbf{f}, \mathbf{g}_3) = 1.5$. (b) $\mathbf{R_3 f}$ for $C(\mathbf{f}, \mathbf{g}_3) = 1.0$. (c) $\mathbf{R_3 f}$ for $C(\mathbf{f}, \mathbf{g}_3) = 0.7071$.

Figure 3-5: A geometric interpretation of the constraints.

straint (3.19) as follows

$$| < R_\gamma f, g_i > | \leq | < f, g_i > | \tag{3.26}$$

$$| < f, g_i > - C(f, g_\gamma) < g_\gamma, g_i > | \leq | < f, g_i > | \tag{3.27}$$

$$\left| 1 - C(f, g_\gamma) \frac{< g_\gamma, g_i >}{< f, g_i >} \right| \leq 1 \tag{3.28}$$

$$0 \leq C(f, g_\gamma) \leq \frac{2 < f, g_i >}{< g_\gamma, g_i >} \tag{3.29}$$

where the last line follows because of (3.25). Similarly, alternative forms for constraint (3.20) are obtained as follows

$$\text{sign}(< Rf, g_i >) = \text{sign}(< f, g_i >) \tag{3.30}$$

$$< Rf, g_i >< f, g_i > \geq 0 \tag{3.31}$$

$$(< f, g_i > - C(f, g_\gamma) < g_i, g_i >) < f, g_i > \geq 0 \tag{3.32}$$

$$C(f, g_\gamma) \leq \frac{< f, g_i >}{< g_\gamma, g_i >} \tag{3.33}$$

where the last line follows because of (3.25). Combining (3.29) and (3.33), we have

$$0 \leq C(f, g_\gamma) \leq \frac{< f, g_i >}{< g_\gamma, g_i >}. \tag{3.34}$$

The same derivation can be followed through for the case where $\frac{<f,g_i>}{<g_\gamma,g_i>} < 0$ for all $g_i \in I_\gamma(k)$ yielding

$$0 \geq C(f, g_\gamma) \geq \frac{< f, g_i >}{< g_\gamma, g_i >}. \tag{3.35}$$

For the case where the ratio $\frac{<f,g_i>}{<g_\gamma,g_i>}$ does not have the same sign for all $g_i \in I_\gamma(k)$, the only value of $C(f, g_\gamma)$ which meets both constraints for all elements in the subfamily I_γ is zero. Thus, for each dictionary element, $C(f, g_\gamma) = S(f, g_\gamma)$ as defined in (3.3) maximizes $\|R_\gamma f - f\|$ under constraints (3.19) and (3.20). Further, the single dictionary element which maximizes $\|R_\gamma f - f\|$ under constraints (3.19) and (3.20) is the same one which maximizes $|S(f, g_\gamma)|$.

3.3 HRP Convergence Properties

In this section, the properties of the HRP algorithm for finite discrete functions $f[t]$ for $0 < t \leq P$ are studied. The main result of this section shows that if the dictionary Γ is complete then the HRP algorithm produces residuals whose norms decay exponentially.

To prove the exponential convergence of the norm of the residuals produced by HRP, the following lemma is needed. This lemma proves that at each step the similarity function must be bounded below by a fraction of the energy of the current residual. A crucial element of this proof is the assumption that the dictionary contains all elements $g_\gamma[t]$ of the form :

$$g_\gamma[t] = \delta[t - r] \text{ for } 0 < r \leq P \tag{3.36}$$

where

$$\delta[t] = \begin{cases} 1 & \text{for } t = 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.37}$$

Note that by definition $S(f, \delta[t-r]) = f[r]$.

Lemma 1 *For a dictionary Γ which contains elements of the form given in (3.36),*

$$|S(R^n f, g_{\gamma_n})| \geq \frac{1}{\sqrt{P}} \|R^n f\|. \tag{3.38}$$

Proof : The similarity function will always be greater than the value of $R^n f$ at any particular point. That is,

$$|S(R^n f, g_{\gamma_n})| \geq |R^n f[r]| \quad \text{for any } r. \tag{3.39}$$

This follows because, by definition,

$$g_{\gamma_n} = \arg\sup_{\gamma \in \Gamma} |S(R^n f, g_\gamma)|, \tag{3.40}$$

and $\delta[t-r] \in \Gamma$ and $S(R^n f, \delta[t-r]) = R^n f[r]$. This implies

$$|S(R^n f, g_{\gamma_n})| \geq \sup_r |R^n f[r]|. \tag{3.41}$$

Further,

$$\|R^n f\|^2 = \sum_{r=1}^{P} |R^n f[r]|^2 \tag{3.42}$$

$$\|R^n f\|^2 \leq P(\sup_r |R^n f[r]|)^2 \tag{3.43}$$

which implies

$$\sup_r |R^n f[r]| \geq \frac{1}{\sqrt{P}} \|R^n f\|. \tag{3.44}$$

It follows that,

$$|S(R^n f, g_{\gamma_n})| \geq \frac{1}{\sqrt{P}} \|R^n f\|. \tag{3.45}$$

□

The following theorem shows that for a complete dictionary which contains elements of the form given in (3.36), the HRP algorithm yields residuals whose energies decay exponentially.

Theorem 1 *For a dictionary Γ which contains elements of the form given in (3.36),*

$$\|R^{n+1} f\| \leq (1 - \frac{1}{P})^{1/2} \|R^n f\|. \tag{3.46}$$

Proof : Note that

$$\|R^{n+1} f\|^2 = \|R^n f\|^2 - 2S(R^n f, g_{\gamma_n}) < R^n f, g_{\gamma_n} > + S^2(R^n f, g_{\gamma_n}). \tag{3.47}$$

From the definition of the similarity function, we know

$$| < R^n f, g_{\gamma_n} > | \geq S(R^n f, g_{\gamma_n}) \tag{3.48}$$

$$\text{sign}(< R^n f, g_{\gamma_n} >) = \text{sign}(S(R^n f, g_{\gamma_n})). \tag{3.49}$$

This implies

$$\|R^{n+1} f\|^2 \leq \|R^n f\|^2 - S^2(R^n f, g_{\gamma_n}). \tag{3.50}$$

Lemma 1 then implies

$$\|R^{n+1}f\|^2 \leq \|R^n f\|^2 - \frac{1}{P}\|R^n f\|^2 \tag{3.51}$$

$$= \|R^n f\|^2(1 - \frac{1}{P}). \tag{3.52}$$

□

3.4 HRP Computational Complexity

The HRP algorithm may be efficiently implemented so that the number of operations required for each iteration is proportional to the size of the dictionary. Further, the computational complexity of HRP may be reduced by using a subdictionary constructed by sampling the scale/shift space. In this section, we discuss the computational complexity of HRP for the cubic b-spline and wavelet packet dictionaries.

We begin with the cubic b-spline dictionary. Recall that the notation for the dictionary is $\{g_\gamma|\gamma \in \Gamma\}$. Suppose we construct a reduced dictionary $\{g_\gamma|\gamma \in \Gamma_R\}$. For the cubic b-spline dictionary, the reduced dictionary has scales j which are integers in the range $0 \leq j \leq \log_2(P)$, where P is the length of the signal, and 2^j evenly spaced translations. This reduced dictionary has a total of $C = 2P - 1$ elements. Let H be the set of functions which form the subfamilies for all elements of the reduced dictionary, $H = \{g_i\}$ for $i \in I_\gamma$ and $\gamma \in \Gamma_R$. The HRP algorithm is *initialized* by computing $< f, g_i >$ for all $g_i \in H$ and $< g_\gamma, g_i >$ for all $\gamma \in \Gamma$ and all $g_i \in H$. This initialization requires a one-time computation of $\mathcal{O}(P^2(\log_2(P))^2)$ operations using the FFT. The HRP similarity measure $S(f, g_\gamma)$ for $\gamma \in \Gamma_R$ may then be computed in $\mathcal{O}(KC)$ operations where K is the cardinality of the set $I_\gamma(k)$. The element which maximizes $|S(f, g_\gamma)|$ over the reduced dictionary is an approximation to the element which maximizes $|S(f, g_\gamma)|$ over the unreduced dictionary. The element which max-

imizes $|S(f, g_\gamma)|$ unreduced dictionary could then be found using a Newton search strategy. Let g_{γ_0} be the element which maximizes $|S(f, g_\gamma)|$ over the unreduced dictionary. Using (3.7), the inner products $< Rf, g_\gamma >$ for all $g_i \in H$ can be computed as

$$< Rf, g_\gamma > = < f, g_\gamma > - S(f, g_{\gamma_0}) < g_{\gamma_0}, g_\gamma > . \tag{3.53}$$

The inner products $< Rf, g_i >$ for all $g_i \in H$ can then be computed using (3.53). Since each of the terms on the right hand side of (3.53) has been previously stored, the calculation of $< Rf, g_i >$ for all $g_i \in H$ takes $\mathcal{O}(KC)$ operations. Extending this argument, we see that each iteration takes $\mathcal{O}(KC) = \mathcal{O}(2PK)$ operations. The number of iterations will typically be much smaller than P.

For the wavelet packet dictionary, the size of the reduced dictionary is $C = P\log_2(P)$. This reduced dictionary has scales j which are integers in the range $0 \leq j \leq \log_2(P)$, $2^{-j}P$ frequency bins for scale j, and 2^j evenly spaced translations for every scale and frequency bin. HRP using the wavelet packet dictionary can be initialized in $\mathcal{O}(P^2 \log_2(P))$ operations by computing $< f, g_i >$. Each iteration for HRP with the wavelet packet dictionary requires the computation of $S(R^n f, g_\gamma)$, the computation of $< g_{\gamma_n}, g_i >$, and the computation of $< Rf, g_i >$. This is a total of $\mathcal{O}(KC) = \mathcal{O}(KP\log_2(P))$ operations per iteration where K is the cardinality of the set $I_\gamma(k)$. Again, the number of iterations will be much smaller than P.

3.5 Demonstration of HRP

In this section, we show that HRP is able to extract signal structure using some simulated examples. As discussed in Chapter 2, the two dictionaries which are of particular interest for this work are cubic b-spline and wavelet packet dictionaries. In this section, we investigate the performance of HRP on some simulated examples for

both the cubic b-spline and wavelet packet dictionaries.

3.5.1 HRP with Cubic B-spline Dictionaries

In this section, the performance of HRP using cubic b-spline dictionaries is explored. We return to the twin peaks example introduced in Section 2.2. HRP is able to resolve the two closely spaced elements used to construct the function. We also investigate the performance of HRP on the so-called gong signal used in [6]. This is an example of the performance of HRP on a function that is not constructed as the sum of dictionary elements.

Twin Peaks Revisited

Recall the twin peaks example of Section 2.2 for which MP yielded unintuitive results. The twin peaks signal is constructed as the sum of two dictionary elements at scale 32 and translation $t = \pm 0.3281$. The contour plot of the HRP similarity function for fitting the first element is shown in Figure 3-6 and clearly shows two maxima at the scale and translations which correspond to the features of the original signal. This is in contrast to the analogous contour plot for MP, which was originally shown in Figure 2-7 and is repeated in Figure 3-7. As this figure shows, the MP similarity measure has a single maxima at scale 40 and translation $t = 0$.

The coherent structures of this signal are captured by the first two elements of the HRP approximation. The first ten elements of the HRP decomposition are shown in Figure 3-8. Since HRP chooses two reasonable elements in the first stages, subsequent elements serve to refine the fit rather than to correct mistakes from previous stages. One can imagine that, in a feature extraction setting, the first two elements would provide a good approximation to the signal and could be used as features of the signal.

As discussed earlier, the HRP decomposition will be affected by the depth at which

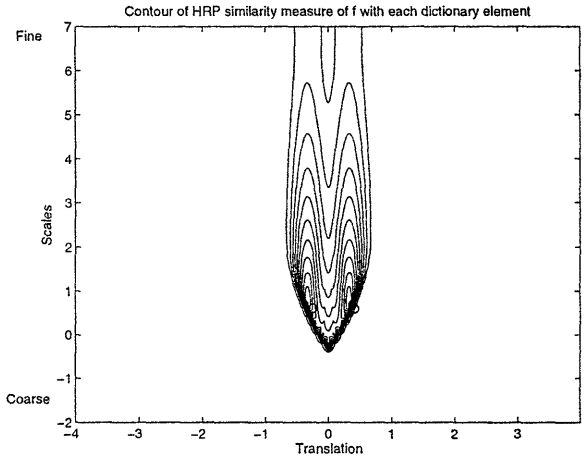


Figure 3-6: The HRP similarity graph is the HRP similarity measure between the function and each dictionary element which is indexed by scale and translation. This figure shows the contour of the HRP similarity graph. O marks location of true elements of the function which are the same as the maxima of the HRP similarity graph.

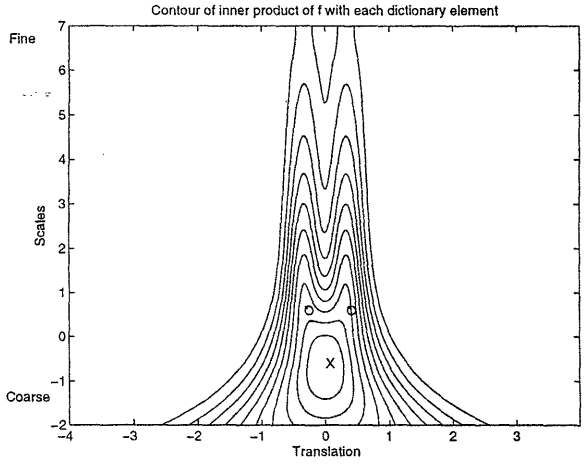


Figure 3-7: Contour of MP projection graph for the twin peaks example. This graph is also shown in Figure 2-7 and repeated here for comparison

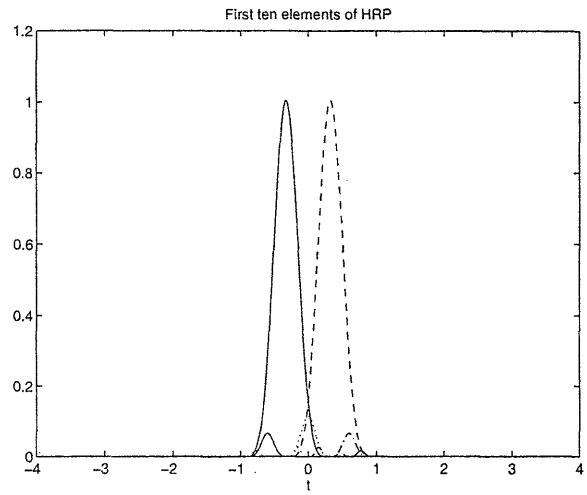


Figure 3-8: First ten elements for twin peaks example using HRP.

the family $I_\gamma(k)$ is constructed. Figure 3-9a-d show the coherent features of the HRP decomposition with depths zero, one, two and three, respectively. At a depth of zero, HRP reduces to MP and the signal is decomposed as a coarse scale feature plus a negatively weighted fine scale feature near the center. At a depth of one, HRP gives the decomposition in Figure 3-9b which may be interpreted as a coarse scale feature plus fine scale details at $t \approx \pm 0.25$. Finally, at a depth of two or higher, HRP gives the decomposition shown in Figure 3-9c-d, which is interpreted as the sum of two positively weighted fine scale features. In real data applications, the depth of $I_\gamma(k)$ may be used to incorporate prior knowledge into the decomposition.

Figure 3-10 compares the residual norms for MP and HRP for the twin peaks example up to 1024 elements. We can identify three distinct regions of convergence. In the first region, from approximately element 1 through 10, both algorithms generate residuals whose norms decay quickly and at a very similar rate, but the decompositions for the two algorithms look very different. MP is behaving in a greedy way by picking coarse features instead of fine features. On the other hand, HRP is behaving in a slightly less greedy way and only picks coarse features when they match the signal

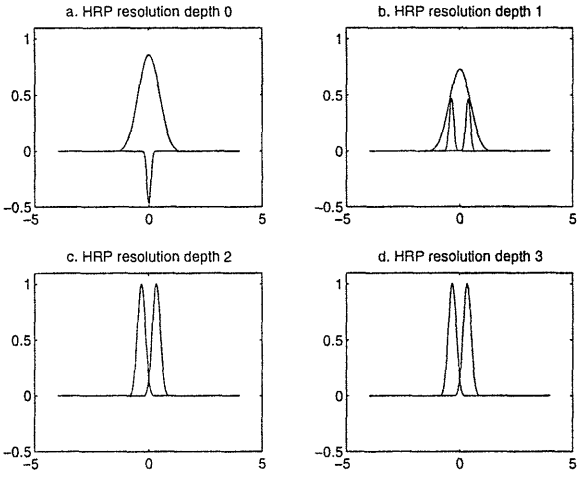


Figure 3-9: Changes in the HRP decomposition of the twin peaks signal as the resolution depth (i.e. the value of k) is changed. Each subfigure shows the first few elements of the HRP decomposition for a different value of k. (a) $k = 0$. (b) $k = 1$. (c) $k = 2$. (d) $k = 3$.

structure locally as well as globally. In the next region, from approximately element 10 to 200, the rate of decay of the residual norms for both decrease slightly from the rate in the first region. In this second region, the MP residual norms are slightly smaller than HRP residual norms. This is to be expected since the MP criterion is to minimize the norm of the residual at each step. The final region starts at approximately element 200. In this final region, the MP residuals continue to decay at an exponential rate, but the HRP residuals decay at a rate much faster than exponential. In this region, the residuals only have structure at the finest scale (i.e. Diracs). HRP will only extract Diracs at this stage; MP, on the other hand, will continue to extract coarser features. This behavior is simply an extension of the behavior shown in Figure 3-11 which shows that MP often extracts coarse scale structures from signals which have only fine scale structure, but HRP extracts fine scale structure. The implication of this behavior is that once HRP attains the Dirac extraction mode, the residual will converge to zero in N iterations, where N is the number of samples of the signal.

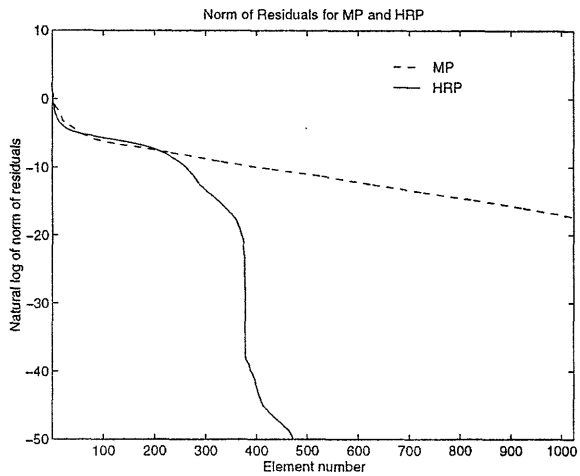


Figure 3-10: Comparison of MP and HRP residual norms for twin peaks example.

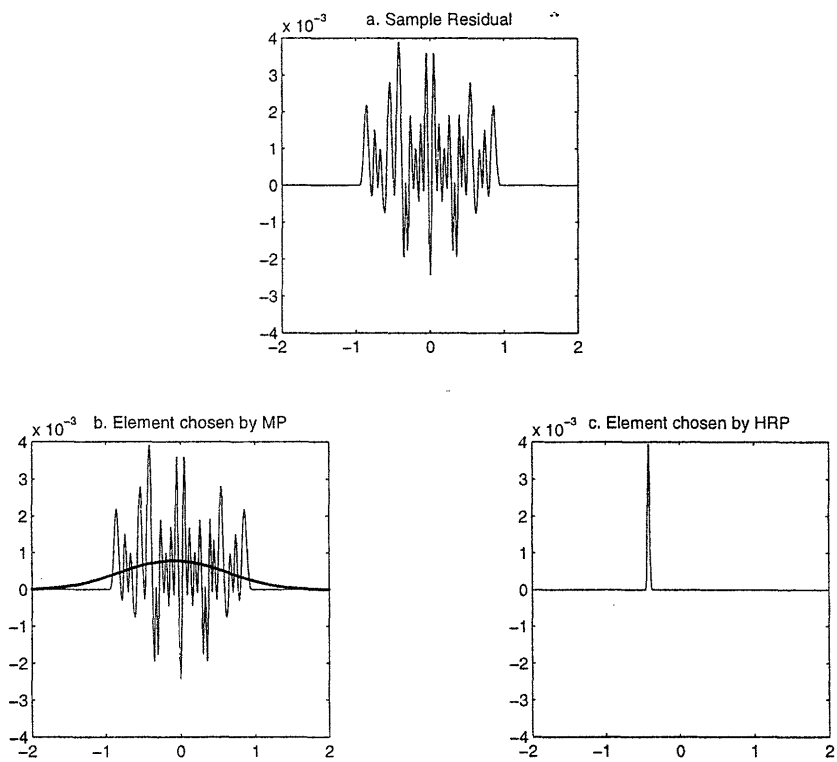


Figure 3-11: Comparison of MP and HRP on a residue with only fine scale structure. (a) Sample Residual. (b) MP chooses an element with coarse scale structure when the signal has only fine scale features. (c) HRP chooses an element with fine scale structure.

The Gong Signal

The next signal we consider is the one illustrated in Figure 3-12 which has a sharp attack at $x = -2$ followed by a slow decay. This is the envelope of the gong signal used in [6]. The ideal decomposition for this signal would capture the attack with elements well localized in time and would not place elements prior to the attack of the signal. We will compare the results of the MP algorithm with those of the HRP algorithm when $k = 2$. Figure 3-13 shows the first element picked by the two algorithms. As was with the case with the twin peaks example, the first element chosen by the MP algorithm is one which introduces an artifact (i.e. energy prior to the attack of the signal). This artifact will have to be corrected for in subsequent stages. In contrast, the first element chosen by HRP does not have a significant amount of energy prior to the attack of the signal. Figure 3-14 shows the first ten elements chosen by the two algorithms. Since the first element of the MP decomposition has significant energy prior to the attack of the signal, three of the subsequent nine elements are used to correct for this initial mistake. On the other hand, HRP captures the attack of the signal with elements well localized in time and does not place elements before the attack. Finally, Figure 3-15 shows the sum of the first ten elements for the two algorithms. Note that the MP approximation has some ringing prior to the attack of the gong signal and that the HRP algorithm has only a small error prior to the attack of the signal.

Figure 3-16 compares the norms of the MP and HRP residuals. Once again, three regions of convergence are evident. The first region, which extends from element 1 through 10, both algorithms decay at similar rates. However, while MP extracts elements which fit the signal well globally, HRP only extracts elements which fit well locally as well as globally. In the second region, from element 10 to 500, both algorithms show exponential convergence. In the final region, above element 500,

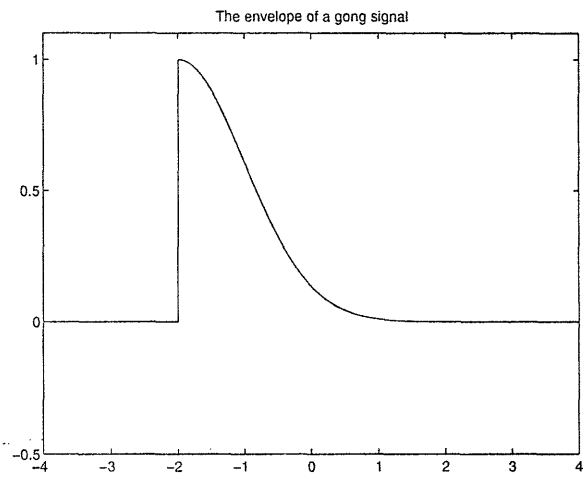


Figure 3-12: Envelope of the gong signal.

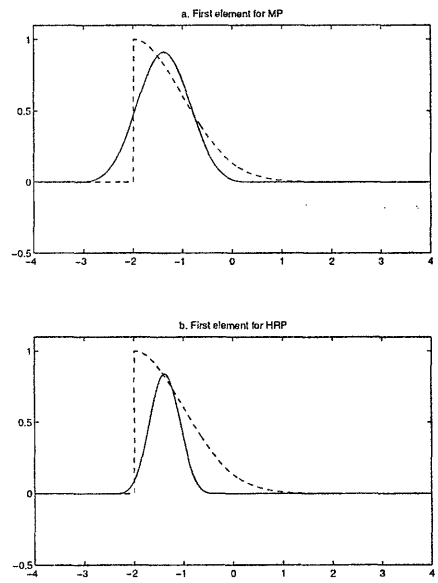


Figure 3-13: First element picked by MP and HRP for the gong example. The dashed line is the gong signal.

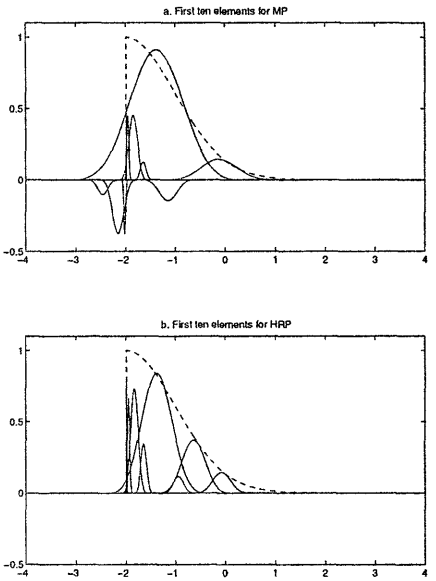


Figure 3-14: First ten elements picked by MP and HRP for the gong example. The dashed line is the gong signal.

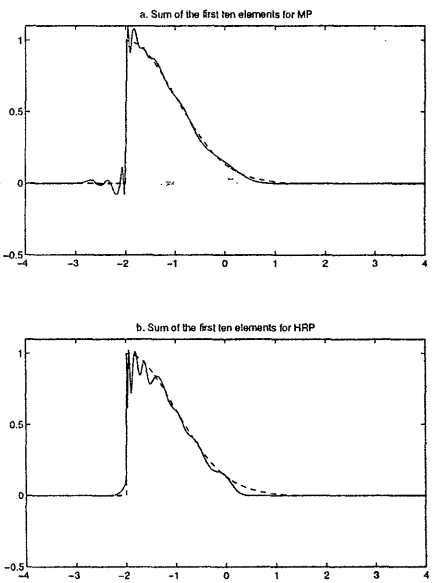


Figure 3-15: Sum of the first ten elements picked by MP and HRP for the gong example. The dashed line is the gong signal.

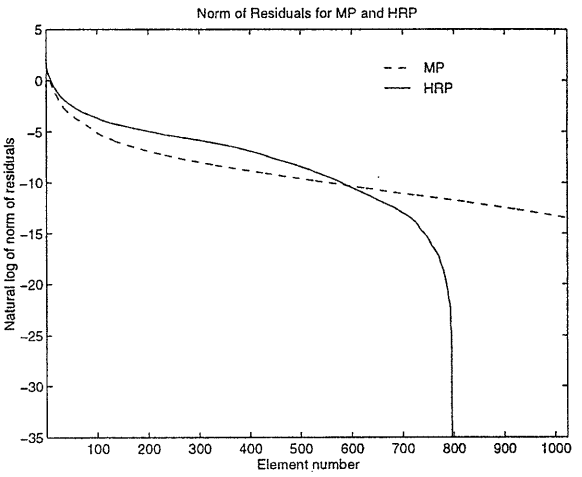


Figure 3-16: Comparison of MP and HRP residual norms for the gong example.

HRP shows a convergence rate much faster than exponential. Again, this results from the fact that HRP enters a mode where it extracts only Diracs.

3.5.2 HRP with Wavelet Packet Dictionaries

In this section, we show that HRP with wavelet packet dictionaries is able extract signal structure and able to resolve two elements which have the same scale and frequency characteristics but differ in translation. However, because of the particular structure we have chosen for the family I_γ, the HRP algorithm will *not* be able to resolve two elements which have the same scale and translation characteristics but differ in frequency. In this section, we highlight the strengths and weaknesses of HRP with wavelet packet dictionaries.

Super-Resolution Examples

Just as was the case for cubic b-spline dictionaries, HRP using wavelet packet dictionaries is able to resolve elements in translation. Figure 3-17 shows the signals f_1, f_2, and $f = f_1 + f_2$ and their corresponding ideal representations on a time-frequency

plane. The functions f_1 and f_2 have the same scale and frequency characteristics but differ in translation and are elements of a wavelet packet dictionary based on the Symmlet wavelet described in [6]. The signal f will be decomposed over the Symmlet wavelet packet dictionary which contains each of the elements used to construct the signal f. Thus, the signal and the dictionary are well-matched. This example is analogous to the twin peaks example which was the sum of two cubic b-spline elements at the same scale but different, nearby translates and was decomposed over the cubic b-spline dictionary. Figure 3-18 shows the time-frequency representations of the MP, BP, and HRP decompositions. The HRP decomposition uses $k = 2$. The MP decomposition is unable to resolve the two elements and gives the appearance of a complicated signal structure. On the other hand, both the BP and HRP decompositions are able to resolve the two elements and yield a sparse representation for this signal. Even though the BP and HRP decompositions are identical, HRP has an advantage over BP in terms of computational complexity.

Because of the particular structure we have chosen for the family $I_\gamma(k)$, the HRP algorithm will *not* be able to resolve two elements in frequency. In the wavelet packet dictionary, we can construct a signal that is the sum of two dictionary elements which share scale and translation characteristics but differ in frequency. This type of a construction was not possible in the cubic b-spline dictionary since all cubic b-spline dictionary elements share basically the same frequency characteristics. We construct another signal $f = f_1 + f_2$ where f_1 and f_2 are elements of the Symmlet wavelet packet dictionary as shown in Figure 3-19. Figure 3-19 also shows the ideal time-frequency plane representations for f_1, f_2 and f. For this example, the functions f_1 and f_2 have the same scale and translation characteristics but differ in frequency. The signal f will be decomposed over the Symmlet wavelet packet dictionary. Again, the signal and the dictionary are well-matched. Figure 3-20 shows the time-frequency representations

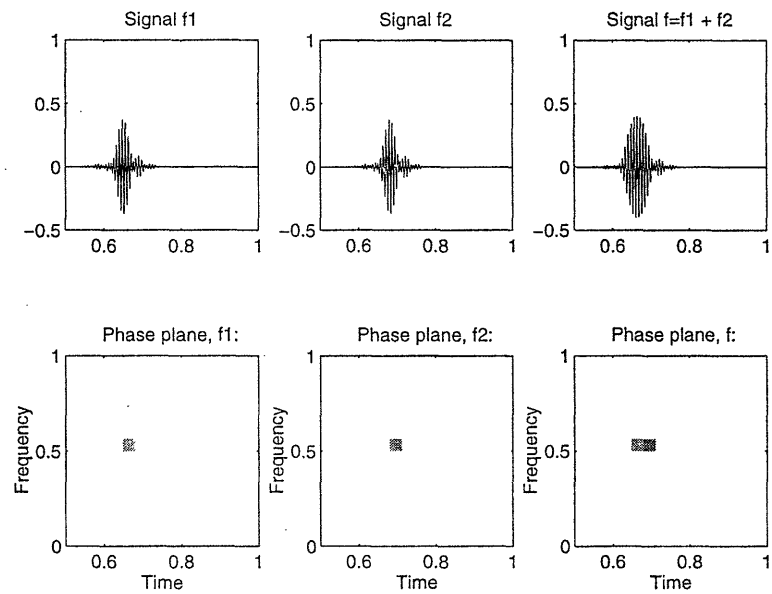


Figure 3-17: The signal f is the sum of f_1 and f_2 which are both elements of the Symmlet wavelet packet dictionary. The signals f_1 and f_2 have the same scale and frequency characteristics but differ in translation. The time-frequency plane representations of f_1, f_2, and f are also shown.

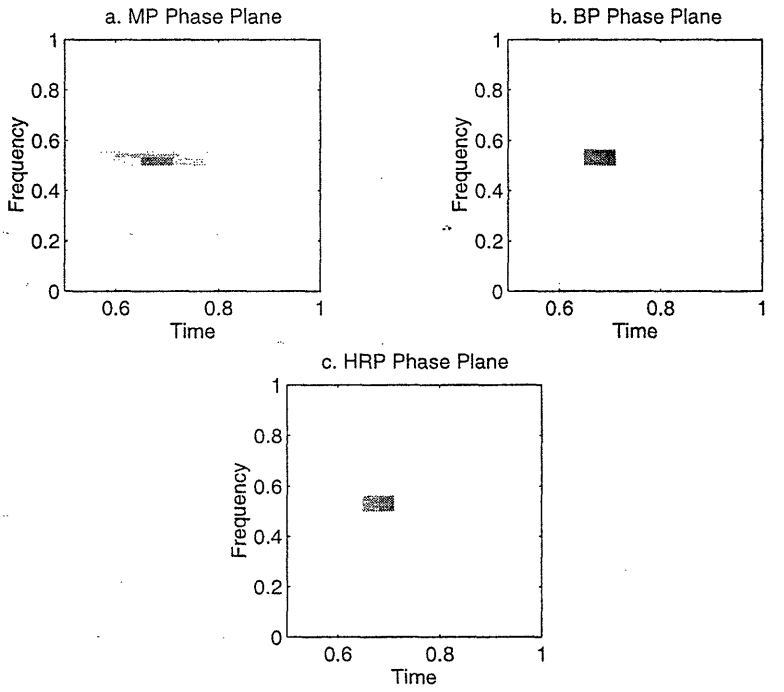


Figure 3-18: The time-frequency plane representations of the MP, BP, and HRP decompositions.

of the MP, BP, and HRP (with $k = 2$) decompositions. For this example, the both the MP and HRP decompositions are unable to resolve the two elements and give the appearance of a complicated signal structure. In contrast, BP is still able to resolve the two elements. The sub-optimal behavior of HRP for this example can be attributed to the structure we have chosen for the subfamily $I_\gamma(k)$. Compared to the time-frequency support of the function g_γ, the functions included in the subfamily $I_\gamma(k)$ have a narrower support in time and therefore a wider support in frequency. It follows that the HRP similarity measure based on this particular structure of $I_\gamma(k)$ is a minimum over inner products with elements with a wider support in frequency and therefore will not be able to distinguish two elements closely spaced in frequency. One can imagine, however, developing an HRP algorithm to resolve elements close in frequency where the family I_γ is defined to consist of elements with finer frequency resolution and therefore coarser time resolution. That is, the subfamilies in this case are, in a sense, the "duals" of those used in this section.

Finally, consider the signal f shown in Figure 3-21a which is the sum of four elements : a Dirac, a sinusoid, and 2 Symmlet wavelet packet dictionary elements which are closely spaced in translation. The signal is decomposed over the Symmlet wavelet packet dictionary and the results of the MP, BP, and HRP algorithms are shown in Figure 3-21b-d. The MP decomposition clearly resolves the Dirac and the sinusoid but is unable to resolve the other two elements. Both BP and HRP resolve all four elements, but HRP improves on the computation time of BP by a factor of four. The signal f is very similar to the carbon signal introduced in [6] which included two additional Symmlet wavelet packet elements that were closely spaced in frequency. We have chosen to delete those additional elements for this example since, as we have already pointed out, HRP, as we have configured it here, will not always be able to resolve elements closely spaced in frequency.

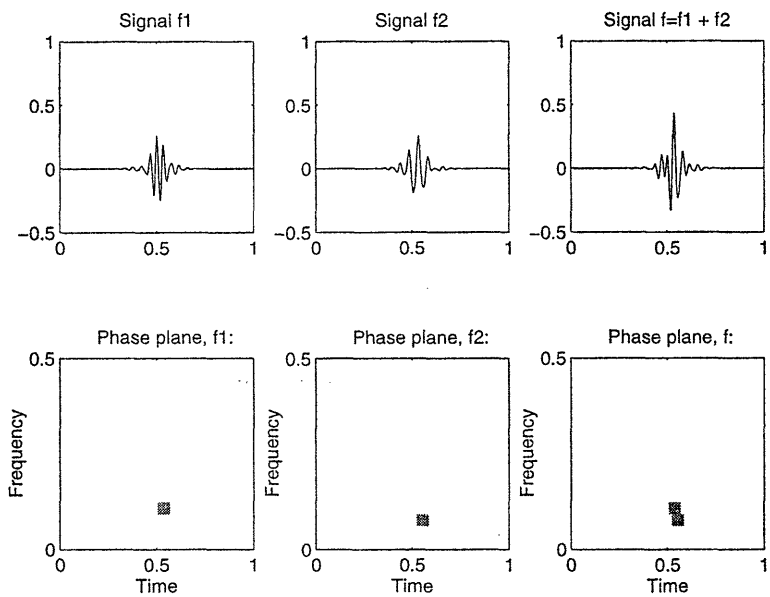


Figure 3-19: The signal f is the sum of f_1 and f_2 which are both elements of the Symmlet wavelet packet dictionary. The signals f_1 and f_2 have the same scale and translation characteristics but differ in frequency. The time-frequency plane representations of f_1, f_2, and f are also shown.

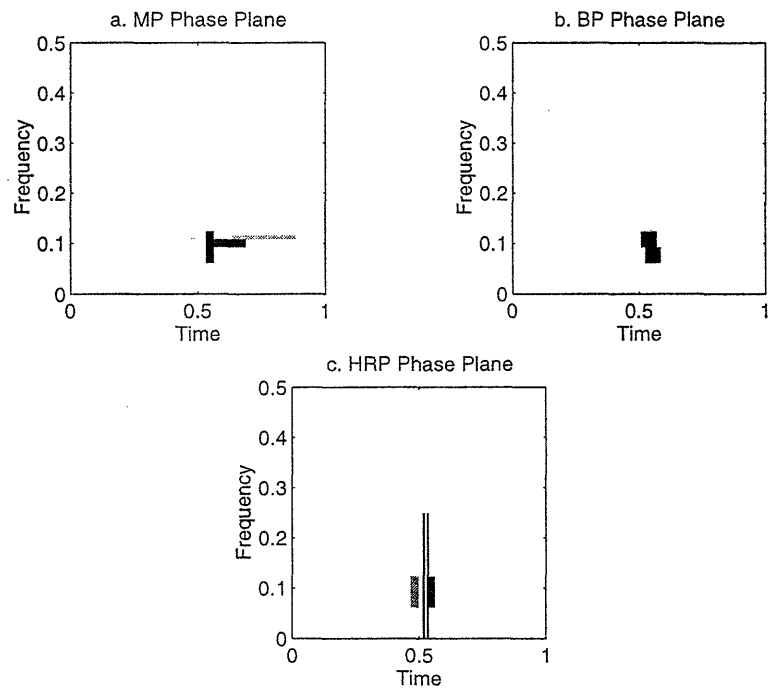


Figure 3-20: The time-frequency plane representations of the MP, BP, and HRP decompositions.

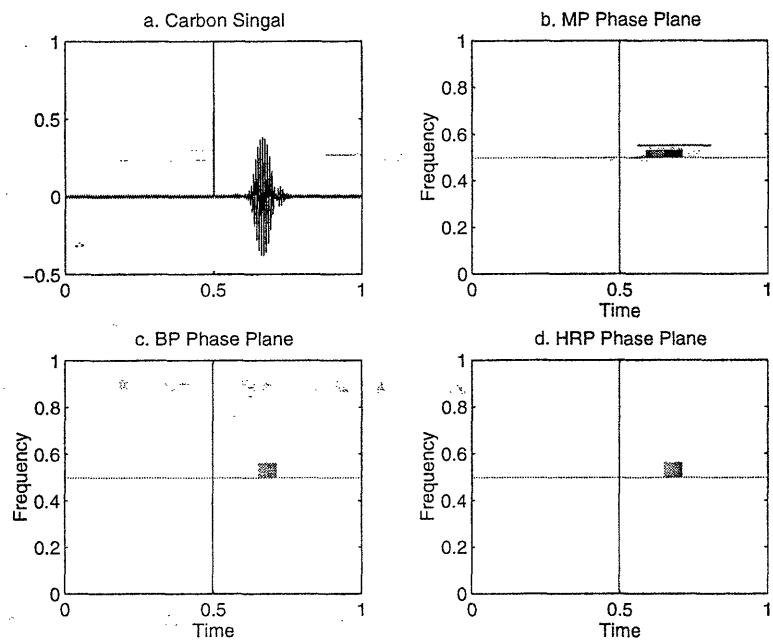


Figure 3-21: Results for the carbon signal. (a) The carbon signal which consists of the sum of four dictionary elements. (b) The MP decomposition. Note that nearby elements are blurred. (c) The BP decomposition. Note that all four elements are resolved. (d) The HRP decomposition. Again, all four elements are resolved.

The Gong Signal

Figure 3-22a shows a gong signal. As was mentioned in Section 3.5.1, this type of signal has a sharp attack followed by a slow envelope decay. Again, the ideal decomposition would capture the attack with elements well localized in time and would capture the correct frequency of the modulation. Further, the ideal decomposition would not introduce elements prior to the attack of the signal. That is, it would not introduce a pre-echo effect which is particularly disturbing for audio signals.

Figures 3-22b-d show the time-frequency plane results for MP, BP, and HRP, respectively. The partial reconstructions for three, five and ten elements for each of the three methods are shown in Figure 3-23. The signal was analyzed using a wavelet packet dictionary constructed from the Daubechies six tap wavelet [10]. MP captures the point of the attack and identifies the correct frequency, but introduces several elements prior to the attack of the signal which results in the addition of subsequent artifacts in the reconstruction. Although the elements before the attack have a small weight, they significantly impact the reconstruction. Thus, the MP reconstruction exhibits this pre-echo effect. BP performs very well since it captures the attack, does not place elements prior to the attack of the signal, and captures the correct frequency of the modulation. HRP captures the point of the attack and does not introduce elements prior to the attack of the signal. However, HRP does not do as well as BP in capturing the correct frequency of the modulation. Comparing the rates of decay of the three methods (see Figure 3-24), we see that BP decays at a rate slightly faster than HRP. In conclusion, HRP does not surpass BP in the quality of the decompositions. However, HRP provides decompositions without artifacts prior to the attack of the signal and does not require the intensive computation of BP.

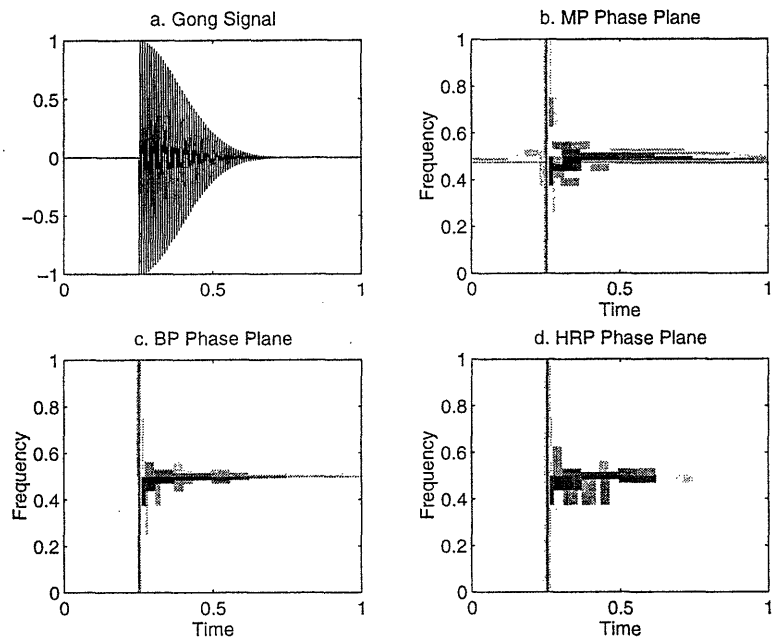


Figure 3-22: (a) The gong signal. (b) Time-Frequency plane for MP. (c) Time-frequency plane for BP. (d) Time-Frequency plane for HRP.

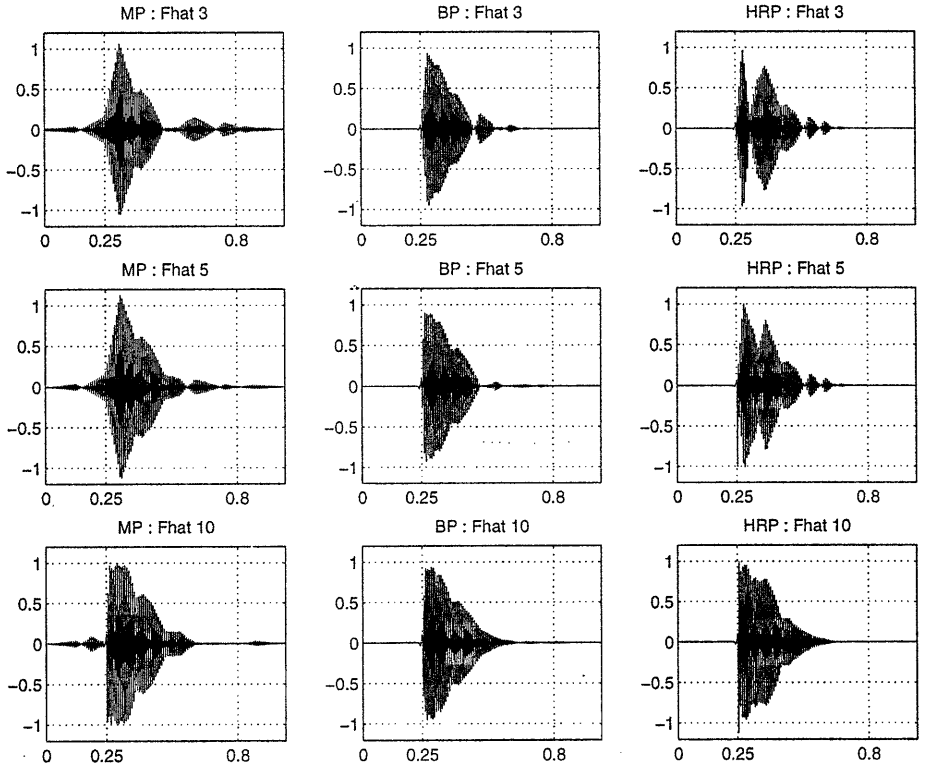


Figure 3-23: Partial reconstructions for MP, BP, and HRP with 3, 5 and 10 elements. In the MP reconstruction, we see the elements prior to the attack of the signal have a significant impact on the partial reconstruction.

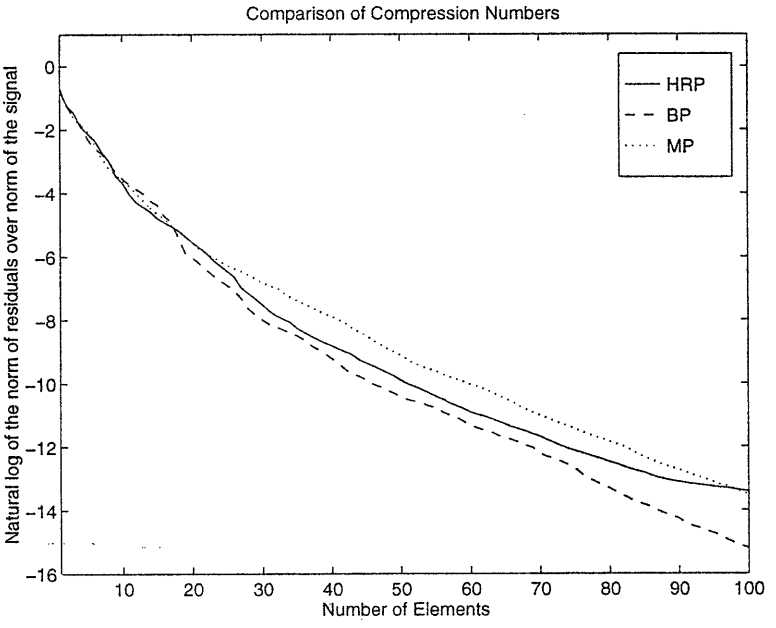


Figure 3-24: Rates of decay of the three methods.

Chapter 4

Recognition of Airplanes from Silhouettes

4.1 Introduction

Humans have the ability to analyze visual scenes, localize and recognize objects, and make decisions based on our visual observations. A computer vision system that could emulate these tasks, which are simple for a human, would have numerous industrial, military, and medical applications. Of course, computer vision is an immense field encompassing a vast array of research [2, 21, 28]. In this chapter, we investigate the application of HRP to one very specific computer vision problem, namely, model-based object recognition from silhouettes.

Model-based object recognition [3, 15] is performed by comparing features extracted from a given data image to features extracted from a predefined set of model images and determining which model the data image most closely resembles. Since humans have the ability to recognize objects from only silhouette or boundary data, there is a widely-held intuition in computer vision [3, 13, 16, 19, 24, 30, 40] that these

object boundaries contain much of the significant information required to recognize objects. Silhouettes may be generated from gray scale images using any one of a number of edge detectors [5, 12, 18, 27], segmentation algorithms [38], or active contour models [1][1].

For a computer vision algorithm for recognizing objects from silhouettes to be of practical importance, the algorithm must perform well under a number of possible variations in the image. Since it is not always feasible to control the spatial relationship between the object and the imaging senor, a computer vision system must be robust to variations in scale, orientation, and translation. Further, imaging conditions such as lighting, reflectance, and haze may result in object boundaries which appear perturbed from their original position. These imaging conditions or the positioning of the object in the scene may also cause portions of object boundaries to be occluded. That is, a portion of the object boundary may be missing entirely. In addition, the algorithm should be robust to variations within the object class[2].

A number of algorithms have been investigated to recognize objects from their silhouettes under scale variations, orientation variations, boundary perturbations, and occlusion. In [23,30], the authors use Fourier descriptors to recognize silhouettes. The benefits of these techniques based on Fourier descriptors is their small computational burden, robustness to scale and orientation variation, and their relative robustness to boundary perturbations. On the other hand, these techniques deteriorate rapidly when occlusions are present. In [24], the authors investigate a technique to classify partial boundaries based on Fourier descriptors. The basic idea behind the technique is to estimate the Fourier descriptors of the complete boundary by minimizing a cost function which is the sum of the least square fit to the Fourier descriptors of the

[1] Of course, this does not imply that recognition from silhouettes is applicable in all settings since it may not always be possible to binarize a scene to obtain a silhouette as pointed in [21].

[2] In the airplane data set that we investigate in this chapter, the presence or absence of under-wing stores (e.g. missiles) would be an example of such variations within the object class.

partial boundary and a compactness measure. The authors show that their technique is able to distinguish some rather broad classes of objects, rather than individual shapes. In contrast, the technique that we describe in this chapter will be able to distinguish individual shapes. In [25], the authors segment the object boundaries and compare two boundaries based on the ratio of the lengths of consecutive segments. This technique was demonstrated on occluded boundaries but no results were presented showing the performance under boundary perturbations. In [39], the authors present a technique based on the chord length distribution, i.e. the normalized set of chord lengths between all pairs of boundary points. The main drawback of the technique outlined in [39] is the computational complexity. Because of the large computational burden imposed by their algorithm, the authors of [39] test their method under boundary perturbations as well as noise, but only use a very small sample size.

In this chapter, we propose an algorithm based on HRP to recognize airplanes from their silhouettes. Since a number of researchers [3, 13, 23, 30] have based their recognition schemes on a 1D representation of the silhouette, we propose to use HRP to extract elements from a 1D representation of the silhouette, namely the centroidal distance profile [3, 13, 23]. The parameters of the HRP elements are organized into a feature vector. Recognition is performed using this feature vector and an M-ary hypothesis testing scheme. Our experimental results show that this approach is robust to boundary perturbations, scale variations, and small orientation variations. Further, our approach is also robust to variations due to occlusion. This is in contrast to the basic technique based on Fourier descriptors [23,30] which cannot accommodate both boundary perturbations and variations due to occlusion. Moreover, in contrast to the technique for recognition of partial boundaries based on Fourier descriptors in [24], the HRP based technique we describe in this chapter is able to distinguish individual shapes rather than broad classes of objects.

4.2 Background

In this section, we discuss methods for extracting 1D profiles from silhouettes, some models used to create noisy silhouettes, and a classical classification technique based on Fourier descriptors, and give a brief survey of the current literature.

The data set of interest for our work consists of the silhouettes of seventeen military airplanes shown in Figure 4-1. Similar data sets have been used in [4, 19, 23–25]. This data set is a fairly challenging one since it contains some planes which are very similar, for example, Planes # 2, 9, and 10 are similar to one another and Planes #4, 15 and 16 are similar to one another. Planes # 7, 12 and 17 are considered swept wing aircraft.

4.2.1 1D Profile Extraction

A number of methods for extracting 1D representations of silhouettes are considered in [3,23]. These include the centroidal distance profile, the complex coordinate profile, and the curvature profile. These techniques often depend on not just extracting boundary curves but extracting ordered points from boundary curves. Such a set of ordered points may be obtained as a byproduct of the curve extraction (e.g. active contours [1]) or from some other technique (e.g. chain codes [3]). The centroidal distance profile is the distance from the points along the object boundary to the object's centroid. That is, if $(x(m), y(m))$ are an ordered sequence of boundary points then the centroidal distance profile $f(m)$ is given by

$$f(m) = \sqrt{(x(m) - X_c)^2 + (y(m) - Y_c)^2} \tag{4.1}$$

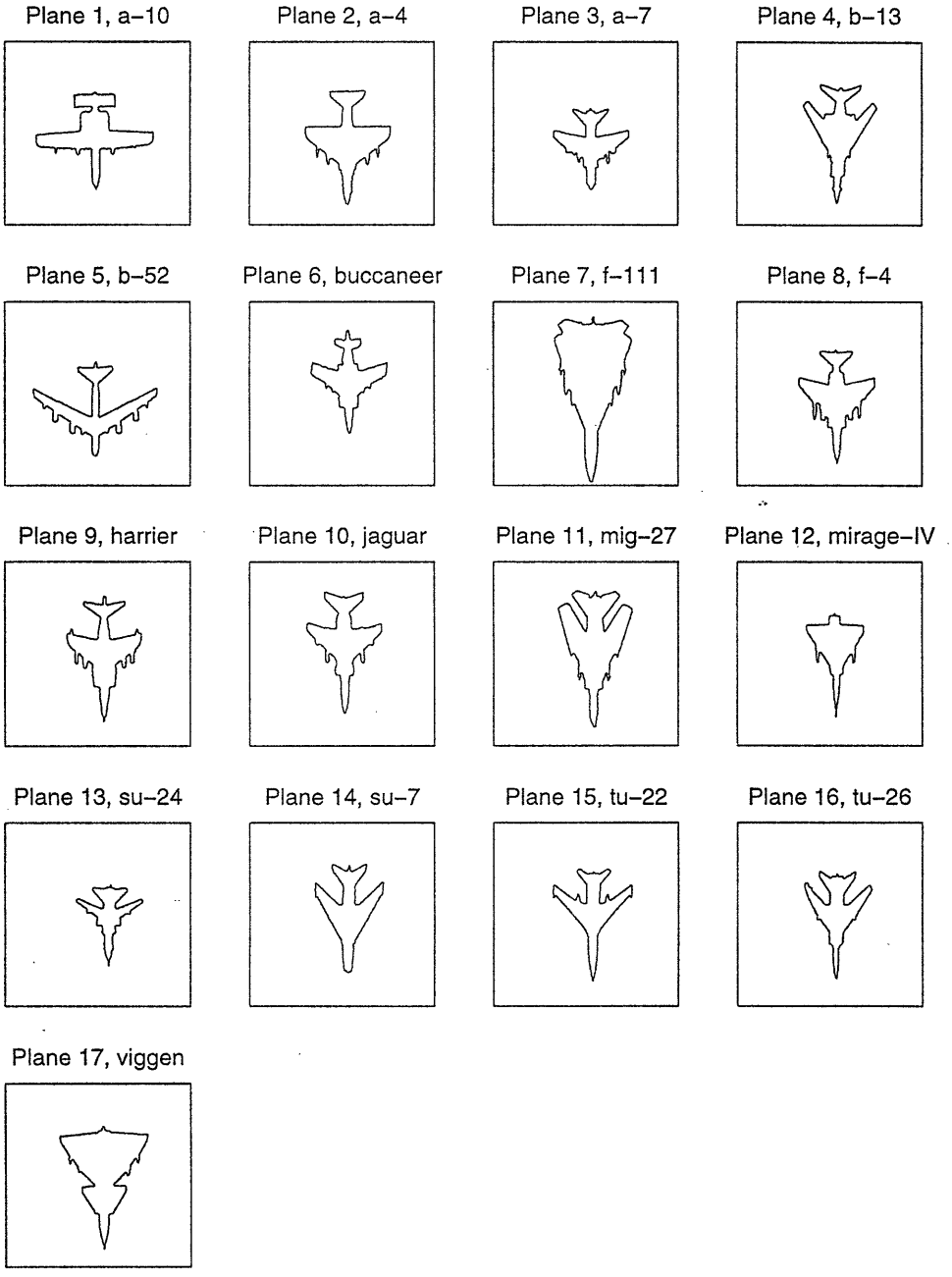


Figure 4-1: Silhouettes of 17 military airplanes which make up the data set.

where (X_c, Y_c) is the centroid of the object and is defined as

$$X_c = \frac{\sum_x \sum_y B(x,y)x}{\sum_x \sum_y B(x,y)} \tag{4.2}$$

$$Y_c = \frac{\sum_x \sum_y B(x,y)y}{\sum_x \sum_y B(x,y)} \tag{4.3}$$

where $B(x,y)$ is set to 1 for points inside the object boundary and 0 for points outside the object boundary [3]. The complex coordinate profile [23] is a complex representation of points along the object's boundary. If $(x(m), y(m))$ are an ordered sequence of boundary points then the complex coordinate profile $z(m)$ is given by

$$z(m) = (x(m) - X_c) + j(y(m) - Y_c) \tag{4.4}$$

where (X_c, Y_c) is the centroid of the object as defined in (4.2) and (4.3). Finally, the curvature profile, for boundary points $(x(m), y(m))$, is given by $c(m)$ [23] where

$$c(m) = \arctan\left(\frac{y(m) - y(m-w)}{x(m) - x(m-w)}\right) - \arctan\left(\frac{y(m-1) - y(m-w-1)}{x(m-1) - x(m-w-1)}\right) \tag{4.5}$$

where w is a constant.

For this work, we will use the centroidal distance function to represent the silhouettes in our airplane database. We have chosen to use the centroidal distance function since it is fairly robust to noise, as shown in [3,23], and it is well matched to the cubic b-spline dictionary which was investigated in Chapter 2.

There are two schools of thought on how to extract the centroidal distance profile from a silhouette : equiangular spacing and equidistant spacing. In [13], the authors calculate the centroidal distance profile at the points of intersection of the boundary with equiangularly spaced radius vectors, as illustrated in Figure 4-2. In [3, 23], the authors calculate the centroidal distance profile using points which are equidistant

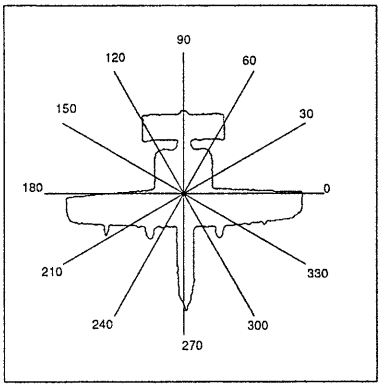


Figure 4-2: Equiangular spacing for extracting the centroidal profile. The numbers indicate the angle in degrees.

along boundary as illustrated in Figure 4-3.

Since equiangular spacing leads to non-uniform sampling of the boundary and ambiguity when the radial vector intersects the boundary more than once (e.g. the vector at 60° in Figure 4-2), we will use equidistant spacing to calculate the centroidal distance profile in this work. Figure 4-4 shows the centroidal distance profile, f, corresponding to equidistant spacing for Plane #1. This centroidal distance profile starts at the tip of the nose and proceeds in a counter clockwise direction. Note that the centroidal distance profile will be periodic. From this figure, we note that the physical features of the plane are clearly identifiable in the centroidal distant profile. The first peak in f (which is wrapped around the interval) corresponds to the nose of the aircraft. Similarly, the next peak corresponds to one wing, the third peak corresponds to the tail, and the last peak corresponds to the other wing. In this work, the first point in the centroidal profile will be the one corresponding to $\theta = -\pi/2$. Figure 4-5 shows the centroidal distance profile calculated at equidistant points extracted for each of the planes in the data set under consideration.

One drawback of the centroidal distance profile is that it essentially discards angular information. Clearly, it is not possible to reconstruct the original boundary from

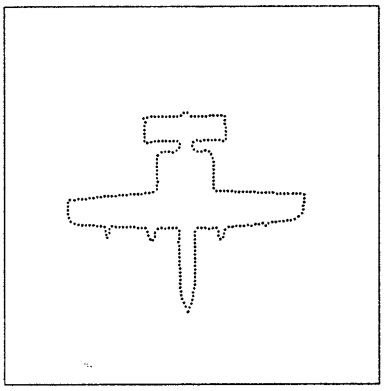


Figure 4-3: Equally spaced points along the boundary.

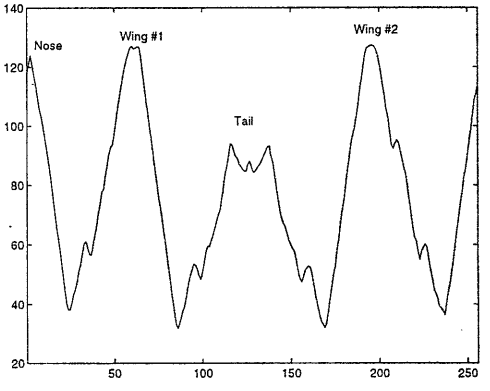


Figure 4-4: The centroidal distance profile, f, for Plane #1 calculated at equally spaced points along the boundary.

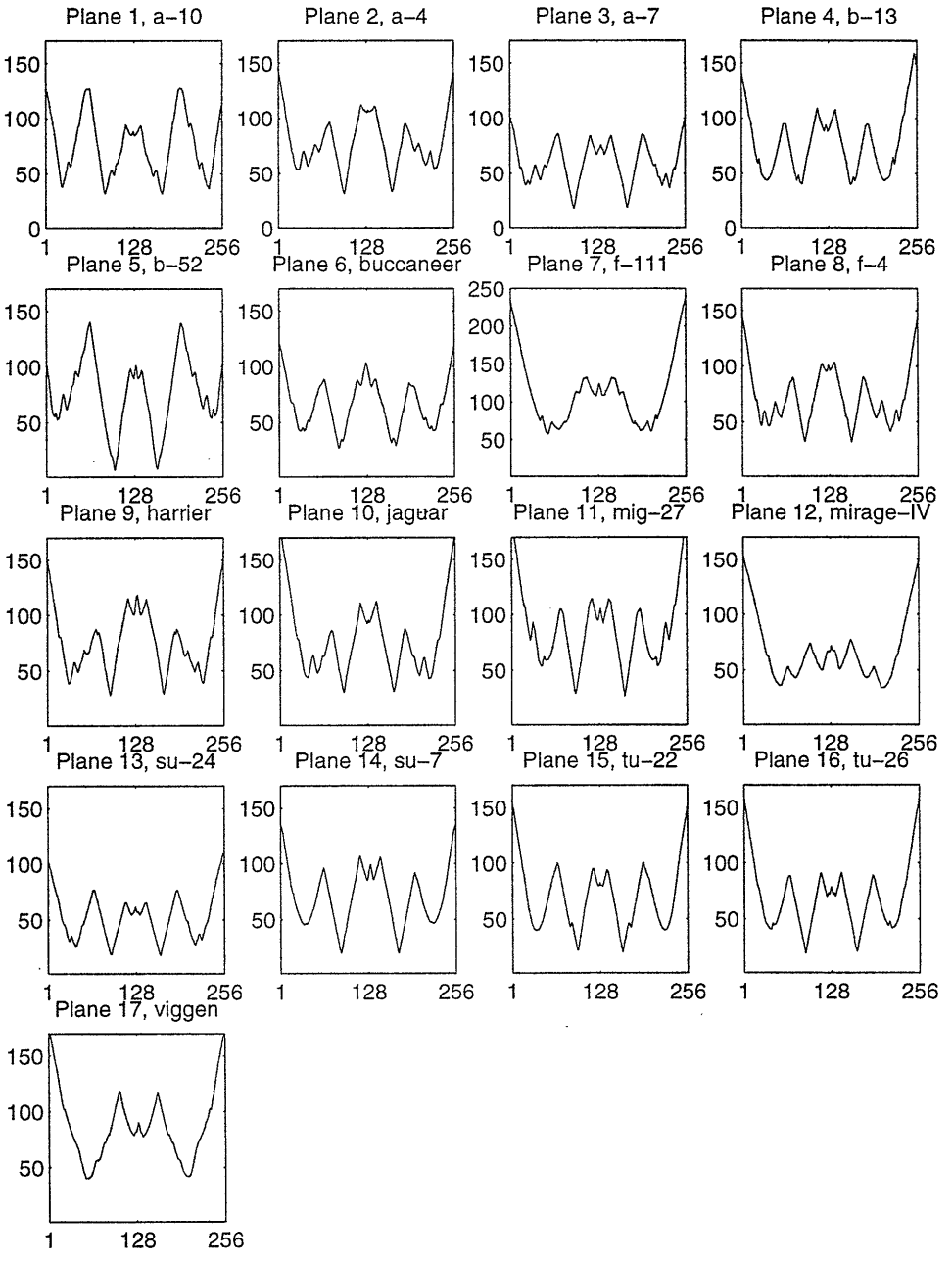


Figure 4-5: The 1D centroidal distance function extracted from each plane in the data set. The points used are equispaced along the boundary.

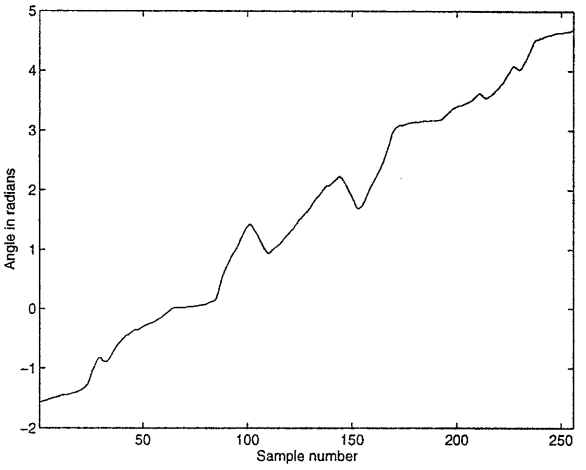


Figure 4-6: The angular profile accompanying the centroidal distance profile in Figure 4-4.

the centroidal distance profile alone. The additional information needed to construct the original boundary is the angular profile, $\theta(m)$. For boundary points $(x(m), y(m))$, the angular profile $\theta(m)$ is given by

$$\theta(m) = \arctan\left(\frac{y(m) - Y_c}{x(m) - X_c}\right). \tag{4.6}$$

The centroidal distance profile and the angular profile are just the representation of the boundary in polar coordinates centered at the centroid of the silhouette. As an example, Figure 4-6 shows the angular profile accompanying the centroidal profile shown in Figure 4-4. This angular information will be useful in Section 4.4.

4.2.2 Silhouette Variations

For an algorithm for recognizing objects from silhouettes to be of practical importance, the algorithm must perform well under a number of possible variations in the silhouette or boundary curve. The algorithm must be robust to variations in scale and orientation since it is not always possible to control the spatial relationship be-

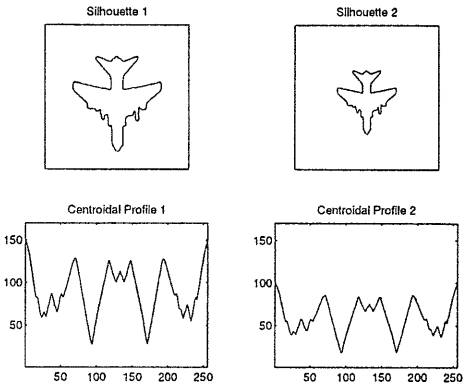


Figure 4-7: The centroidal distance profile of Plane #3 at different scales.

tween the object and the imaging system. Certain imaging conditions may cause the perturbations in the boundary or even occlusions. In this section we illustrate some of these possible variations, their effects on the centroidal distance function, and describe models used in the literature to create these silhouette variations.

The scale of the silhouette changes as a function of the distance between the object and the imaging system. The resulting centroidal distance profile shows a change in amplitude. Figure 4-7 shows the centroidal distance profile for Plane #3 at two different scales. The peaks in the centroidal distance profile corresponding to the nose, wings, and tail are still identifiable. Next, consider rotation of the object, where the axis of rotation is perpendicular to the imaging plane. Suppose the centroidal distance profile f is always extracted starting at fixed angular position, then this rotation results in a circular shift of the centroidal distance profile. Figure 4-8 shows Plane #4 at different rotations and the corresponding centroidal distance profiles, where the starting point for the centroidal profile is at $\theta = -\pi/2$.

Imaging conditions such as lighting, reflectance, and haze may cause perturbations in the boundary curve itself. We will model these boundary perturbations using the technique adopted in [3, 23, 39]. In [3, 23, 39], p percent of the boundary points are perturbed by Gaussian noise which is perpendicular to the boundary. That is, the

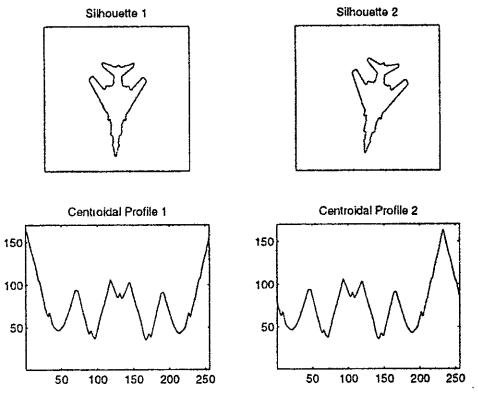


Figure 4-8: Plane #4 at two different rotations and the corresponding centroidal distance profiles.

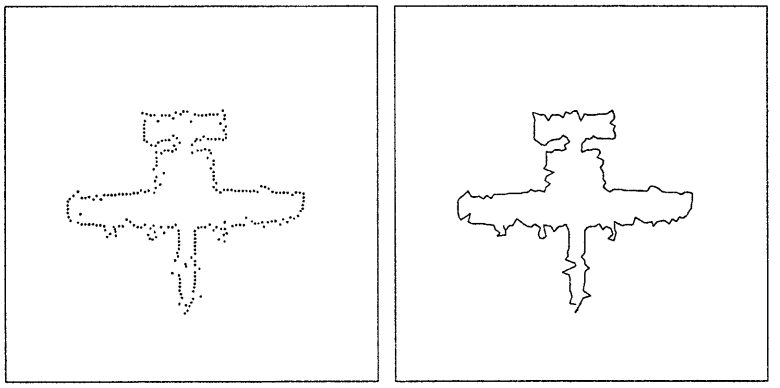
m-th boundary point goes from $(x(m), y(m))$ to $(x_n(m), y_n(m))$ where

$$x_n(m) = x(m) + dr\cos(\xi(m)) \tag{4.7}$$

$$y_n(m) = y(m) + dr\sin(\xi(m)) \tag{4.8}$$

where d is the distance between boundary points m and $m+1$, r is a random variable chosen from $N(0, s^2)$, and ξ is the angle between the x-axis and the direction normal to the boundary at point m. Figure 4-9a shows the points in Figure 4-3a perturbed according to (4.7) and (4.8) with $p = 40$ and $s = .9$. By connecting the perturbed points, we obtain the noisy boundary given in Figure 4-9b. The noisy 1D centroidal profile in Figure 4-10 is then generated from the unperturbed centroid and points that are *equally spaced* along the *noisy* boundary. This requires resampling the noisy boundary. This type of boundary perturbation also causes a change in the object's centroid. While other papers [3,23] have ignored this secondary effect, we will include it in our experimentation.

Occlusion may be caused by changes in lighting, reflectance, haze, or the presence of other objects in the image. In [25], occluded boundaries are created by replacing



(a) Noisy silhouette points.

(b) Noisy boundary formed by connecting the noisy silhouette points.

Figure 4-9: Generating Gaussian perturbations perpendicular to the object contour.

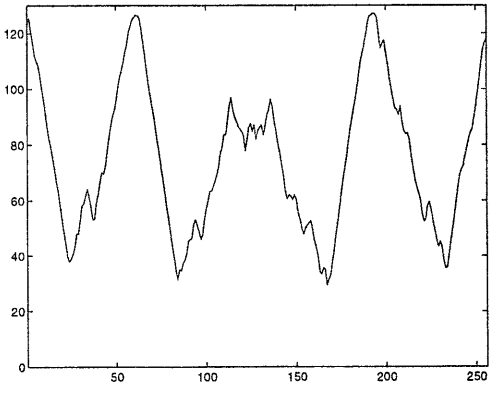


Figure 4-10: Noisy 1D centroidal distance function.

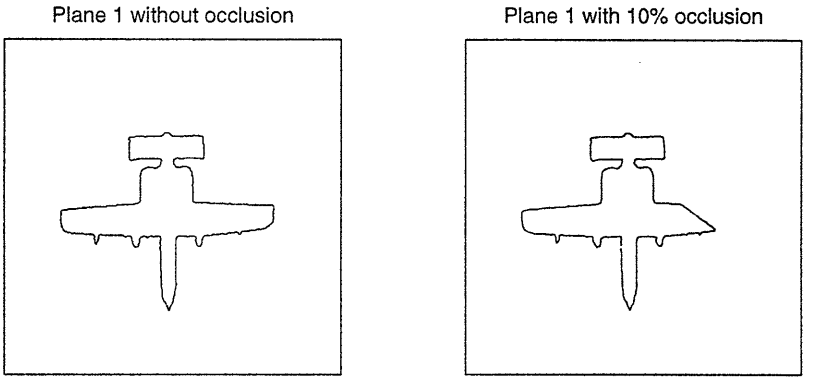


Figure 4-11: Plane #1 with and without occlusion. The plane on the right is generated with $q = 10\%$.

a random, consecutive set of q percent of the boundary points in a silhouette with a straight line. Figure 4-11 shows occluded version of Plane # 1 with $q = 0$ (i.e. no occlusion) and $q = 10$ and Figure 4-12 shows the corresponding centroidal distance profiles. Occlusion causes two types of distortion in the centroidal distance profile. First, since a portion of the boundary is missing more boundary points will be devoted to the remaining airplane features. As a result, the scale of the remaining features has changed. For example, in the occluded centroidal distance profile in Figure 4-12, since the first wing is occluded the second wing appears wider than in the unoccluded profile. Second, the occlusion also causes a shift in the centroid which leads to a significant warping in the centroidal distance profile.

4.2.3 Fourier Descriptors

One classical technique used to recognize silhouettes is based on Fourier descriptors [23,30,40]. The benefits of these techniques is their small computational burden, robustness to scale and orientation variation, and their relative robustness to boundary perturbations. On the other hand, these techniques deteriorate rapidly when occlusions are present.

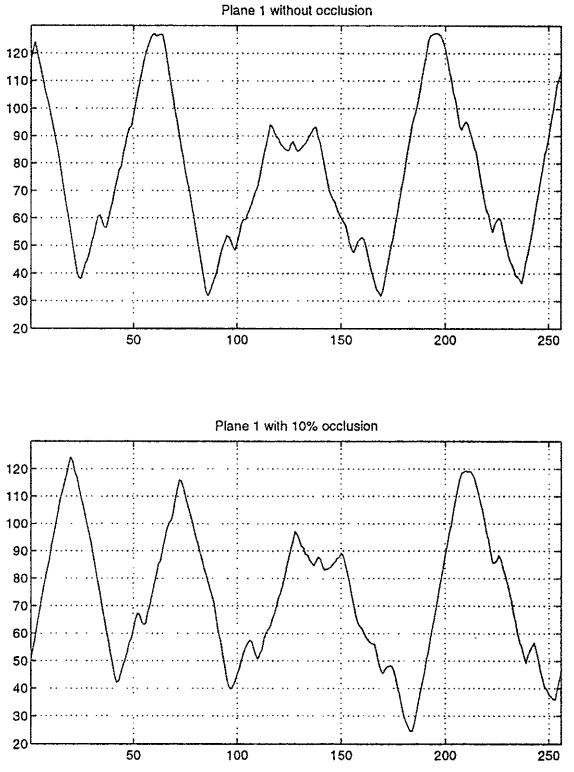


Figure 4-12: Corresponding centroidal distance profiles for Figure 4-11.

In [23], the authors construct a feature vector based on the normalized magnitude of the centroidal distance profile of a silhouette. If the centroidal distance profile of a silhouette has N samples and is given by f, and the discrete Fourier transform of f is given by F, then the feature vector $\mathbf{y}$ is given by

$$\mathbf{y} = \left[\frac{|F_1|}{|F_0|} \cdots \frac{|F_{N/2}|}{|F_0|}\right] \tag{4.9}$$

where F_i denotes the i-th component of F. The classification of silhouettes based on their Fourier descriptors is done using a K nearest neighbor procedure [14].

4.3 Detection Theory Background

As we have already mentioned, the basic approach in this work is to extract HRP features from a 1D representation of the airplane silhouettes and use these features to identify the plane. The search engine we will be using in this work is the M-ary hypothesis test and the generalized likelihood ratio test. In this section, we describe these two general concepts from detection theory.

4.3.1 Hypothesis Testing

The M-ary hypothesis testing problem is an extension of the binary hypothesis testing problem so we begin by considering the binary hypothesis test. Suppose we are given a vector of measurements $\mathbf{y}$ whose probabilistic behavior is determined by one of two hypotheses, H_0 and H_1. We are also given the following information regarding the measurements and hypotheses. First, we are given the probabilistic relationship between $\mathbf{y}$ and each of the hypotheses, i.e. $p_{\mathbf{y}|H_m}(\mathbf{Y}|H_m)$ for $m = \{0, 1\}$. Second, we are given P_m, the probability that hypothesis m is true. Third, we are given C_{mn}, the cost associated with choosing hypothesis m if hypothesis n is true. The decision

rule, $\mathcal{D}(\mathbf{Y})$ which minimizes the expected value of Bayesian risk [14, 37] is stated as follows:

$$\mathcal{D}(\mathbf{Y}) = \begin{cases} 1 & \text{if } \frac{p_{\mathbf{y}|H_1}(\mathbf{Y}|H_1)}{p_{\mathbf{y}|H_0}(\mathbf{Y}|H_0)} \geq \frac{(C_{10}-C_{00})P_0}{(C_{01}-C_{11})P_1} \\ 0 & \text{if } \frac{p_{\mathbf{y}|H_1}(\mathbf{Y}|H_1)}{p_{\mathbf{y}|H_0}(\mathbf{Y}|H_0)} < \frac{(C_{10}-C_{00})P_0}{(C_{01}-C_{11})P_1} \end{cases} \tag{4.10}$$

where the term on the left hand side of the inequality in (4.10) is referred to as the likelihood ratio. For the case where the cost associated with correct decisions is zero, the cost associated with incorrect decisions is one, and the hypotheses are equally likely, this decision rule reduces to

$$\mathcal{D}(\mathbf{Y}) = \begin{cases} 1 & \text{if } p_{\mathbf{y}|H_1}(\mathbf{Y}|H_1) \geq p_{\mathbf{y}|H_0}(\mathbf{Y}|H_0) \\ 0 & \text{if } p_{\mathbf{y}|H_1}(\mathbf{Y}|H_1) \geq p_{\mathbf{y}|H_0}(\mathbf{Y}|H_0) \end{cases} \tag{4.11}$$

That is, the choice which minimizes the Bayesian error criterion is the hypothesis with the maximum likelihood. In the M-ary hypothesis test, the set of possible hypotheses is given by $\{H_1, \ldots, H_M\}$. The Bayes minimum risk decision rule for the case where the cost associated with correct decisions is zero, the cost associated with incorrect decisions is one, and the hypotheses are equally likely is given by

$$\mathcal{D}(\mathbf{Y}) = m \text{ if } p_{\mathbf{y}|H_m}(\mathbf{Y}|H_m) \geq p_{\mathbf{y}|H_n}(\mathbf{Y}|H_n) \text{ for all } n \neq m. \tag{4.12}$$

Although we have shown the more general case with arbitrary values for P_m and C_{mn}, we will assume in the remainder of this chapter that the cost associated with correct decisions is zero, the cost associated with incorrect decisions is one, and the hypotheses are equally likely.

Reiterating, our proposed approach is to use the M-ary hypothesis test to determine which plane is most likely given a set of features based on the elements extracted by HRP from a noisy 1D representation of a silhouette. In this proposed approach, y will be a feature vector based on the HRP decomposition and each hypothesis H_m

will correspond to one of the planes in the database.

4.3.2 Generalized Likelihood Ratio Test

The hypothesis testing problem can also be extended to include more complex hypotheses, typically referred to as composite hypotheses [34]. A composite hypothesis is one which depends on another unknown value. That is, $p_{\mathbf{y}|H_m}(\mathbf{Y}|H_m)$ actually depends on some other unknown value β and should be rewritten as $p_{\mathbf{y}|H_m,\beta}(\mathbf{Y}|H_m,\beta)$. The parameter β is often referred to as a "nuisance" parameter since the objective is to choose a hypothesis, regardless of the actual value of β.

The generalized likelihood ratio test (GLRT) suggests that a logical approach to this problem is to replace the unknown parameter with its maximum likelihood estimate and continue with the hypothesis test previously outlined. In this case, the M-ary hypothesis testing decision rule becomes $\mathcal{D}(\mathbf{Y}) = m$ if

$$\frac{\max_\beta p_{\mathbf{y}|H_m,\beta}(\mathbf{Y}|H_m,\beta)}{\max_\beta p_{\mathbf{y}|H_n,\beta}(\mathbf{Y}|H_n,\beta)} \geq \gamma \text{ for all } n \neq m \tag{4.13}$$

where γ is some threshold. In the specific case where the cost associated with the correct decision is zero, the cost associated with an incorrect decision is one, and all hypotheses are equally likely, γ is equal to one.

The following examples serve to illustrate the generalized likelihood ratio test. Examples 2 and 3 will be particularly useful in the remainder of this chapter.

Example 1. One example often used to illustrate the GLRT is as follows. For this example, the measurements are a random vector $\mathbf{y}$ with N components. Suppose that under hypothesis H_0 the components of $\mathbf{y}$ are independent and identically distributed and have a Gaussian distribution with a mean of zero and a known variance σ. Under hypothesis H_1, each component of $\mathbf{y}$ has a Gaussian distribution with an unknown

mean β and a known variance σ. The maximum likelihood estimate of β is given by

$$\hat{\beta}_m = \frac{1}{N}\sum_{i=1}^{N} Y_i, \tag{4.14}$$

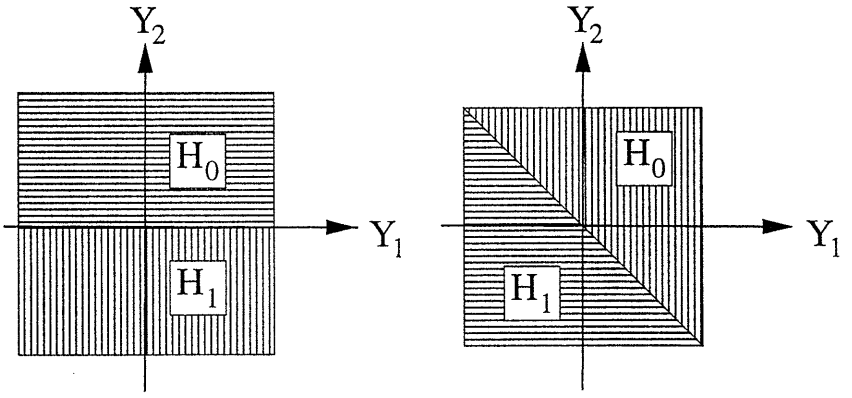
and the decision rule becomes

$$\mathcal{D}(\mathbf{Y}) = \begin{cases} 1 & \text{if } \frac{1}{2\sigma^2}\frac{1}{N}(\sum_{i=1}^{N} Y_i)^2 > \gamma \\ 0 & \text{if } \frac{1}{2\sigma^2}\frac{1}{N}(\sum_{i=1}^{N} Y_i)^2 < \gamma \end{cases} \tag{4.15}$$

where $\gamma > 1$.

Example 2. In this example, the measurements are given by the random vector $\mathbf{y}$ which has two components (i.e. $\mathbf{y} = [y_1 \quad y_2]$). Under hypothesis H_0, the components are independent and Gaussian with $m_0 = [\ 0 \quad 5\ ]$ and $\Lambda_y = \sigma^2 I$. Under hypothesis H_1, the components are independent and Gaussian with $m_1 = [\ 0 \quad -5\ ]$ and $\Lambda_y = \sigma^2 I$. The decision regions derived using the likelihood ratio test are shown in Figure 4-13a. Suppose that we are supplied with the additional information that the components of $\mathbf{y}$ may have been switched. That is, m_0 may actually be $[\ 5 \quad 0\ ]$ and m_1 may actually be $[\ -5 \quad 0\ ]$. This switching of components may be viewed as a nuisance parameter in the same way that the unknown mean β was in Example 1. We can formulate this problem in terms of two composite hypotheses $\mathbf{H}_0 = \{H_{0a}, H_{0b}\}$ and $\mathbf{H}_1 = \{H_{1a}, H_{1b}\}$, where each hypothesis implies the components of $\mathbf{y}$ are independent and Gaussian with the following means

$$\begin{aligned} H_{0a}: \quad m_{0a} &= [\ 0 \quad 5\ ] \\ H_{0b}: \quad m_{0b} &= [\ 5 \quad 0\ ] \\ H_{1a}: \quad m_{1a} &= [\ 0 \quad -5\ ] \\ H_{1b}: \quad m_{1b} &= [\ -5 \quad 0\ ] \end{aligned}$$



(a) The decision regions for Example 2 without switching.

(b) The decision regions for Example 2 with switching.

Figure 4-13: The decision regions for Example 2.

The decision regions using the generalized likelihood ratio test to account for switching are shown in Figure 4-13b. By using the GLRT to incorporate the additional information about switched components, the decision regions have been changed. Conceptually, this example is easily extended to incorporate the possibility of switched components when $\mathbf{y}$ has more than two components.

Example 3. For this example, let $\mathbf{y}$ be a random vector with four independent components and let the conditional probability for the i-th component of $\mathbf{y}$ given hypothesis m be denoted $p_{y_i|H_m}(Y_i|H_m)$. This implies that

$$p_{\mathbf{y}|H_m}(\mathbf{Y}|H_m) = \prod_{i=1}^{4} p_{y_i|H_m}(Y_i|H_m). \tag{4.16}$$

Suppose we are also given the additional information that one component of $\mathbf{y}$ has been corrupted, but we are not told which component. This corruption implies that y_i for some i is no longer described by the conditional density $p_{y_i|H_m}(Y_i|H_m)$, but rather by a conditional density that does not depend on the hypothesis. Once again,

the corruption of a component may be treated as a nuisance parameter just as the unknown mean was in Example 1. Using the GLRT, we obtain the following solution. For each hypothesis H_m, determine which component has most likely been corrupted. That is, find l_m^*

$$l_m^* = \arg\max_l p_{\mathbf{y}|H_m,l}(\mathbf{Y}|H_m, l) \tag{4.17}$$

where

$$p_{\mathbf{y}|H_m,l}(\mathbf{Y}|H_m, l) = \begin{cases} \prod_{i\in\{2,3,4\}} p_{y_i|H_m}(Y_i|H_m) & \text{if } l = 1 \\ \prod_{i\in\{1,3,4\}} p_{y_i|H_m}(Y_i|H_m) & \text{if } l = 2 \\ \prod_{i\in\{1,2,4\}} p_{y_i|H_m}(Y_i|H_m) & \text{if } l = 3 \\ \prod_{i\in\{1,2,3\}} p_{y_i|H_m}(Y_i|H_m) & \text{if } l = 4. \end{cases} \tag{4.18}$$

Then, determine the most likely hypothesis by excluding the component which is mostly likely corrupted for each hypothesis, as follows

$$H^* = \arg\max_{H_m} p_{\mathbf{y}|H_m,l}(\mathbf{Y}|H_m, l_m^*). \tag{4.19}$$

4.4 HRP-Based Recognition

Our proposed approach will be to use HRP to decompose a 1D representation of the airplane silhouette. There are several possible 1D representations for silhouettes considered in the literature which we described in Section 4.2.1 and we have chosen to use the centroidal distance profile. Each centroidal profile f will start at $\theta = -\pi/2$, where θ is the angular profile associated with f, and continue in a counter clockwise direction. We will then construct a feature vector from the parameters (i.e. scale, translation, and magnitude) of the elements of the HRP decomposition of the centroidal distance profile and use this feature vector for recognition. The recognition phase of our approach will be based on M-ary hypothesis testing and the generalized

ratio test. In this section, we describe the details of our proposed feature extraction and HRP-based recognition algorithm.

Some notation that will be useful in this section is as follows. The centroidal distance profile will be denoted f and the accompanying angular profile will be denoted θ. Centroidal profiles which are corrupted by noise or occlusion will be denoted f_n and the corresponding angular profile will be denoted θ_n.

4.4.1 Feature Extraction

There are a number of questions to address in order use HRP to decompose the centroidal distance representation of an airplane silhouette. We must decide the number of elements to extract, choose a dictionary that is appropriate for the centroidal distance profiles, and determine a value of the depth parameter in the HRP algorithm k which yields robust decompositions. In addition, we discuss how to incorporate some of the angular information which is essentially discarded in the centroidal profile representation of a silhouette. Finally, we discuss how to organize the parameters of the elements from the HRP decomposition into a feature vector that makes recognition using the M-ary hypothesis test feasible.

The following observations guide us in choosing the appropriate number of HRP elements to extract from the centroidal distance profiles. Figure 4-14 shows four noisy realizations of the profile of Plane #1 created using the boundary perturbation model described Section 4.2.2 with $p = 40\%$ and $s = 0.9$. Figure 4-15 shows the corresponding centroidal profiles for each of the noisy realizations of Figure 4-14. These profiles are extracted starting at the tip of the nose and proceeding counter clockwise. The physical features of the airplane are clearly visible in the features of the centroidal profile : the first peak in the profile corresponds to the plane's nose, the second corresponds to one wing, the third corresponds to the tail, and the fourth

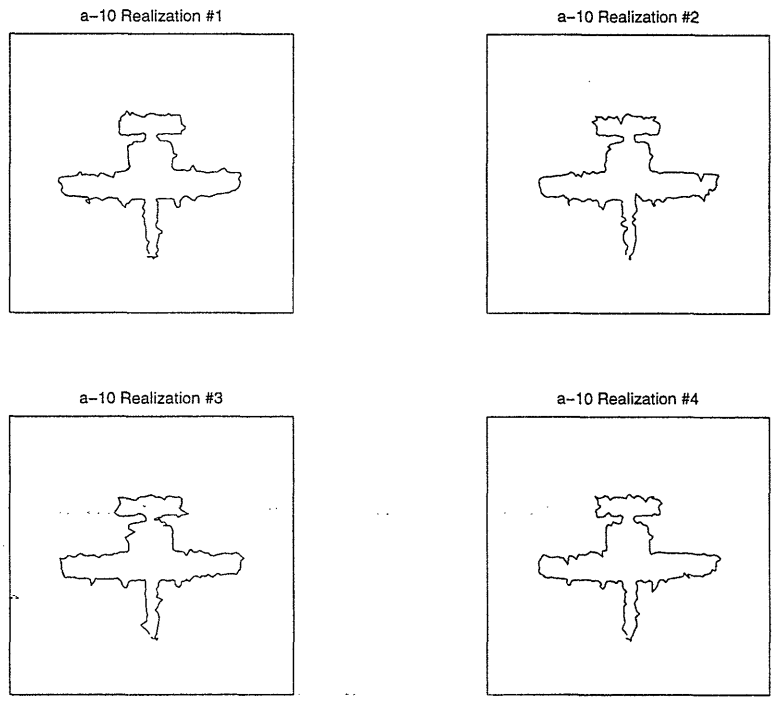


Figure 4-14: Four noisy realizations of Plane #1, generated using $p = 40\%$ and $s = 0.9$.

corresponds to the other wing. These four peaks are evident in each of the noisy profiles. Given that centroidal distance profiles for most planes in the data base exhibit four similar strong features, one logical approach would be to extract four elements from each profile. Although one can imagine developing a scheme to allow the number of elements extracted to adapt to the particular profile, we have chosen to extract four elements for this work. Future work may incorporate additional elements to increase the discriminating power of the algorithm. In addition, if HRP is tuned via the parameter k so that each HRP element corresponds to one feature (or peak) in the centroidal distance profile, then each of these four HRP elements will correspond to a physical feature of the plane, and the parameters of the HRP elements give a description of the physical features of the plane.

The HRP decomposition may be used with many dictionaries. For any applica-

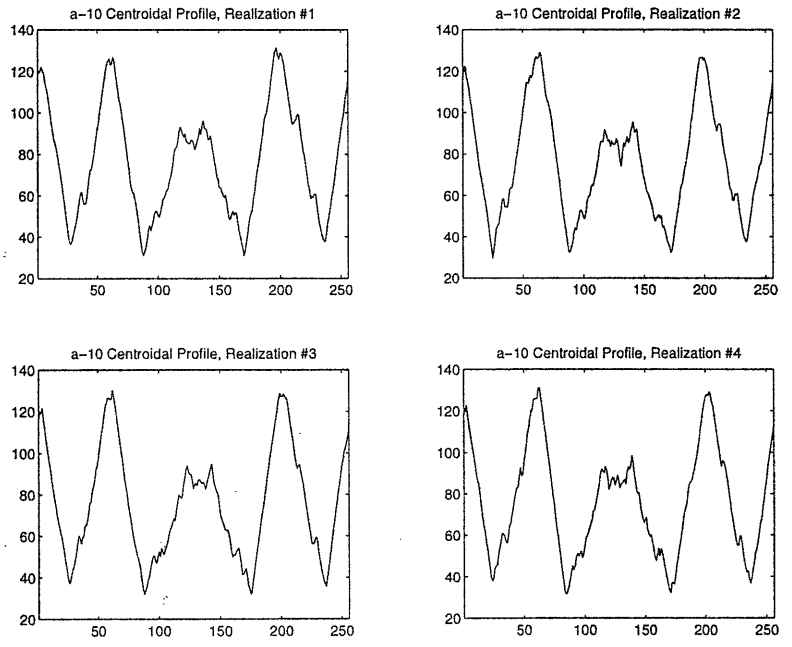


Figure 4-15: The centroidal distance functions corresponding to the noisy realizations of Plane #1 shown in Figure 4-14.

tion, the dictionary should be chosen so that it is well-matched to the signals under consideration. From Figure 4-15, we note that one possible dictionary is the cubic b-spline dictionary which was introduced in Chapter 2. This dictionary contains elements which are primitive enough to match the wide range of features in our data set. In our the experiments in the next section, we will use the cubic b-spline dictionary.

The value of the depth parameter k must be chosen to be robust to noise and yet yield a set of HRP elements from which it is possible to distinguish the planes in the data set from one another. One way to accomplish this is to choose k so that each HRP element corresponds to a feature (or peak) in the centroidal distance profile. In this case, each of HRP elements will correspond to a physical feature of the plane. However, choosing a value for k may be complicated since there is a wide range of variation in the centroidal profiles of the airplanes in the data set and the value of k must yield qualitatively accurate descriptions for all planes in the data

set. Consider the following decompositions of noisy centroidal profiles of Plane #1 from our data base, where noise has been added as described in Section 4.2.2 with $p = 40\%$ and $s = 0.9$. Figure 4-16 shows the decomposition of a noisy centroidal profile from Plane #1 with $k = 1$. Note that in this decomposition the elements from the HRP decomposition do not correspond to the peaks of the centroidal profile. As a result, the parameters of the HRP elements do not give a description of the physical features of the plane. Since our underlying intuition was to extract four features which corresponded to each of the physical features of the plane (i.e. the nose, the wings, and the tail), we conclude that this value of k is a poor choice. In comparison, Figure 4-17 shows the decomposition of the same noisy centroidal profile from Plane #1 with $k = 2$, with the elements labeled alphabetically from left to right. Qualitatively, we might say that elements a, b, and d give a good fit to the peaks in the centroidal distance function, but element c gives a poor fit since it is not well matched to the profile over the entire feature corresponding to the tail. Since element c is a poor fit, we conclude again that $k = 2$ is a poor choice. Now consider the HRP decomposition of the same noisy centroidal profile if we use $k = 3$ to extract two elements, then use $k = 2$ to extract a third element, and then use $k = 1$ to extract a fourth element. The results of this decomposition are shown in Figure 4-18. This decomposition yields the qualitative performance we desire, namely, that each element corresponds to a physical feature of the plane and accurately describes that feature. Through experimentation, we have found that using this same pattern for k (i.e. $k = 3$ for the first two elements, $k = 2$ for the third element, and $k = 1$ for the fourth element), in general, yields robust and accurate decompositions for the centroidal profiles from each of the planes in the data set. Further, our experimentation results show that this pattern of values for k yields qualitatively accurate descriptions in the presence of several different values of boundary noise and occlusional noise.

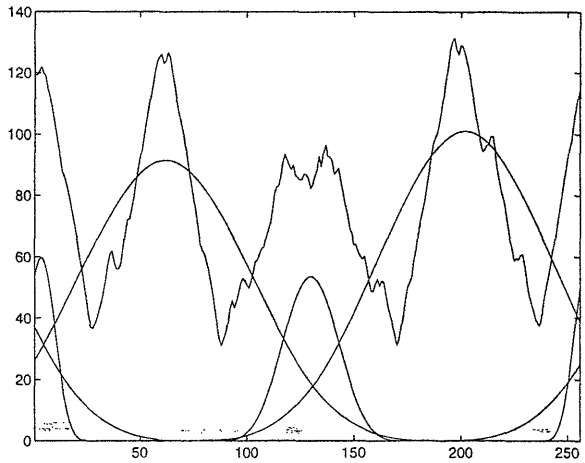


Figure 4-16: The HRP decomposition of a noisy version of the centroidal profile for Plane #1 using $k = 1$.

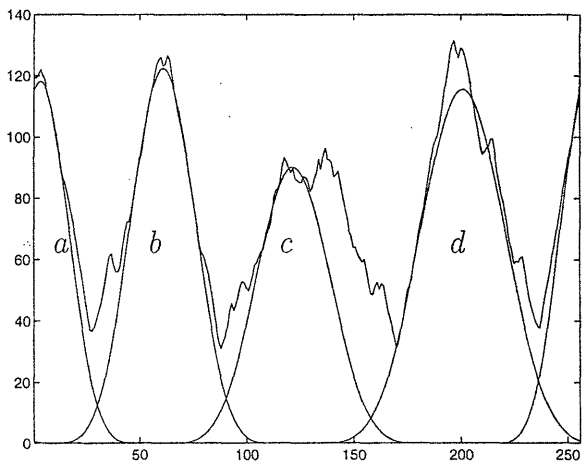


Figure 4-17: The HRP decomposition of a noisy version of the centroidal profile for Plane #1 using $k = 2$.

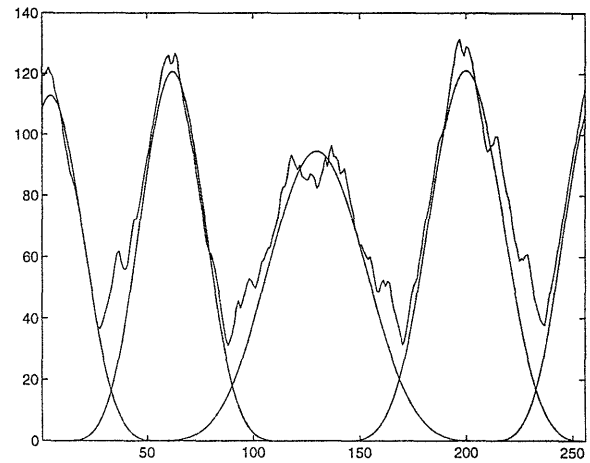


Figure 4-18: The HRP decomposition of a noisy version of the centroidal profile for Plane #1 using $k = 3$ for the first two elements, $k = 2$ for the third element, and $k = 1$ for the fourth element.

To summarize, our proposed approach is to use HRP to extract four elements from each centroidal distance profile using $k = 3$ for the first two elements, $k = 2$ for the third element, and $k = 4$ for the fourth element. We will then construct a feature vector derived from the three parameters (i.e. scale, translation, and magnitude) of each of the elements of the HRP decomposition. Thus, the feature vector that we will use for recognition consists of 12 components. We now discuss methods to derive the particular components of the feature vector from the parameters of the HRP elements and to do so in way that incorporates the angular information which is discarded when extracting the centroidal distance profile from the silhouette.

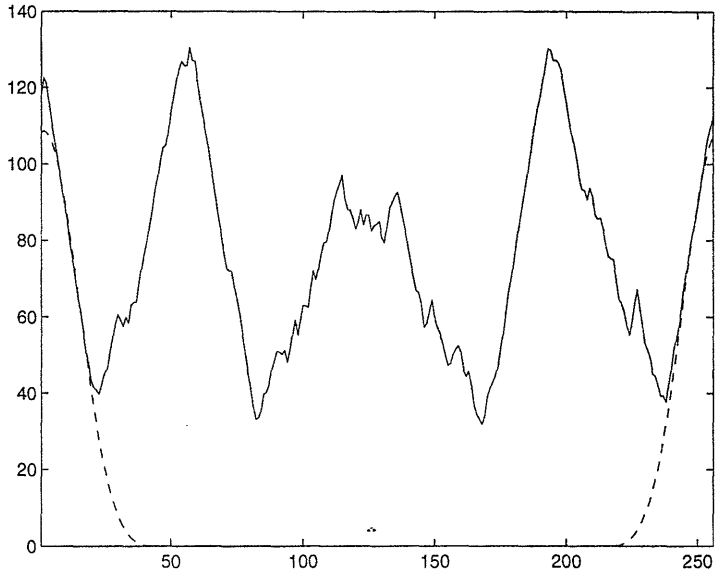
Incorporating Angular Information

As we pointed out in Section 4.2.1, the centroidal distance profile discards angular information. We now show how this angular information can be incorporated with the parameters of the HRP elements to create a feature vector which is robust to variations due to noise and occlusion.

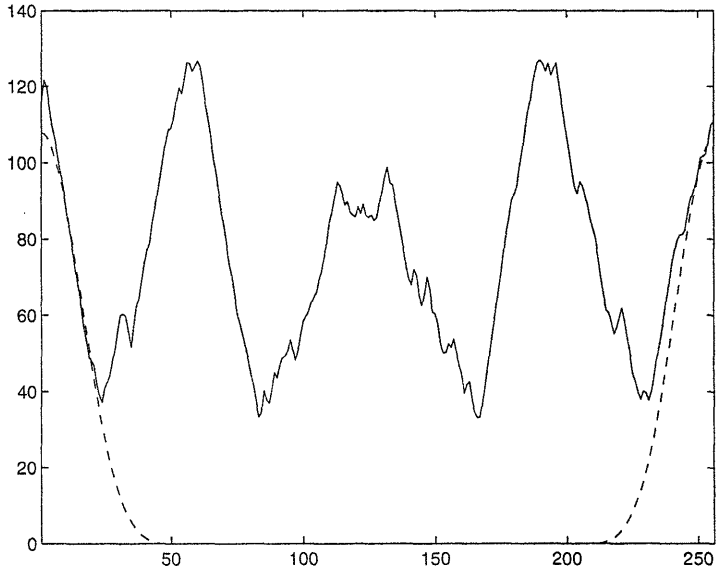
Since the centroidal distance function is periodic, the translation parameter of an element extracted by HRP can be misleading. To illustrate this, consider the motivational example illustrated in Figure 4-19 which shows two noisy centroidal distance profiles for Plane #1, f_{n_1} and f_{n_2}, and the HRP element which corresponds to the plane's nose for each profile. For f_{n_1}, the translation of the HRP element which corresponds to the nose is $t = 1$. In contrast, for f_{n_2}, the translation of the analogous element is $t = 256$. The translation parameter of the HRP element indicates that the nose features of the two centroidal profiles are far apart which is *not* the case. To overcome this ambiguity introduced by the translation parameter, we note that the translation parameter for each element extracted by HRP from the centroidal distance profile may be replaced by its angular equivalent. This is done simply by using the angular profile associated with each centroidal distance profile, like the one shown in Figure 4-6. That is, if $\theta(x)$ is the angular profile accompanying the centroidal profile $f_n(x)$, then each translation parameter t of each HRP element is replaced by $\theta(t)$.

In the presence of occlusion, the scale parameter of an HRP element can also be misleading. Consider a second motivational example. Suppose we are given an occluded silhouette of Plane #1 as shown in Figure 4-20 and the corresponding centroidal distance function as shown in Figure 4-21. Figure 4-21a also shows the HRP element which corresponds to the wing feature in the occluded profile. For comparison, Figure 4-21b shows the centroidal profile for an unoccluded version of Plane #1 and the HRP element which represents the wing feature. We note that the scale and translation parameters of the HRP element which represents the wing feature in the occluded profile are quite different from those of the HRP element which represents the wing feature in the unoccluded profile. The occlusion has caused this shift in parameters[3]. The mismatch in the translation parameters will be minimized by

[3]Note that, to a lesser degree, boundary noise can cause a similar shift in the scale and translation parameters.



(a) fn_1, a noisy centroidal profile from Plane #1 and the HRP element corresponding to the nose.



(b) fn_2, a noisy centroidal profile from Plane #1 and the HRP element corresponding to the nose.

Figure 4-19: A comparison of HRP elements corresponding to the nose of Plane #1.

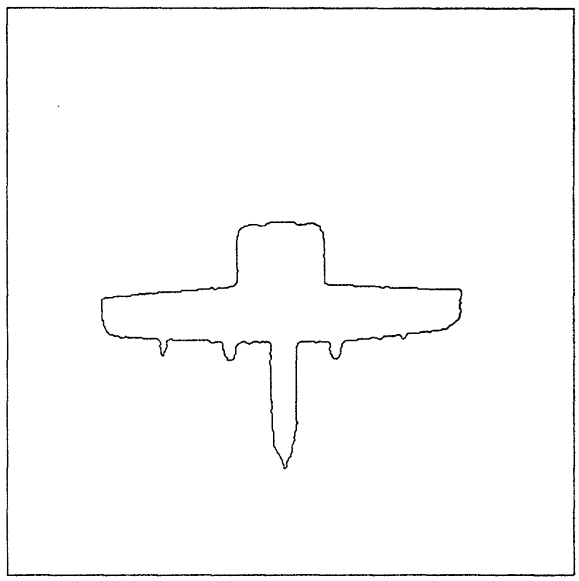
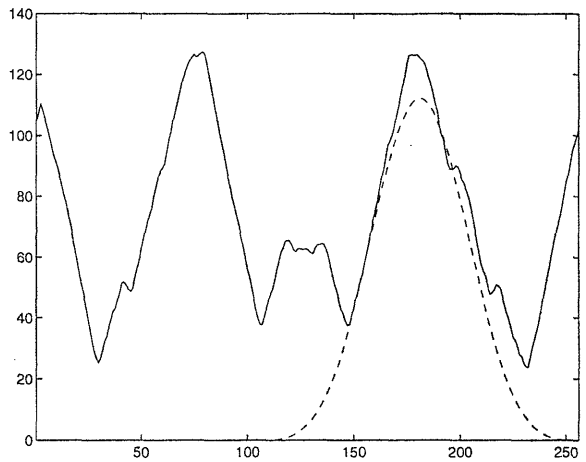


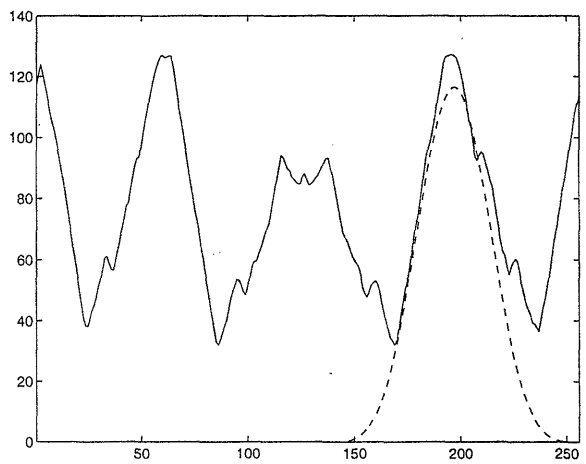
Figure 4-20: The boundary corresponding to Plane #1 with the tail occluded.

replacing t with $\theta(t)$ as described above. The remaining question is how should we account for the change in the scale parameter.

The following observation about the slope of the angular profile will be useful in understanding how to account for the change in the scale parameter. We observe that the occlusion has also caused a change in the slope of the angular profile over the support of the element corresponding to the wing. Figure 4-23a shows the entire angular profile of the occluded plane. Figure 4-23b shows the portion of the same angular profile over the support of the HRP wing element and the best fit line to this portion of the angular profile. The slope of this line is 0.0136. For comparison, Figure 4-22a shows the entire angular profile for the unoccluded plane, and Figure 4-22b shows the portion of the same angular profile over the support of the HRP wing element and the best fit line to this portion of the angular profile which has a slope of 0.0145. Thus, the occlusion has caused a small change in the slope of the angular profile over the support of the element corresponding to the wing. The question we

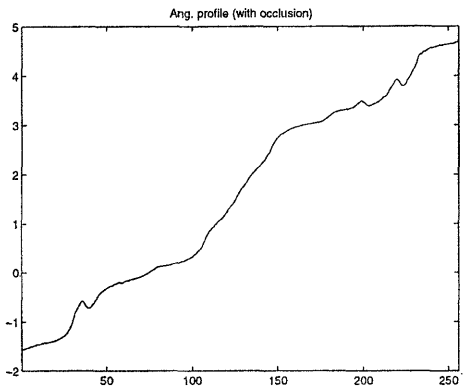


(a) The centroidal distance function for the occluded boundary and the HRP wing element.



(b) The centroidal distance function for the unoccluded boundary and the HRP wing element.

Figure 4-21: A comparison of HRP elements corresponding to the wing of Plane #1.



(a) Angular profile of the occluded version of Plane #1 shown in Figure 4-20.

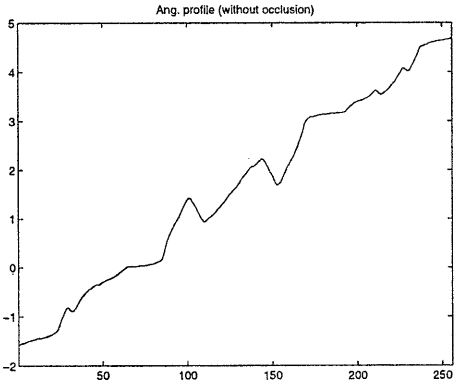
(b) Portion of the angular profile corresponding to the wing and the best fit line.

Figure 4-22: Angular profile for the occluded plane.

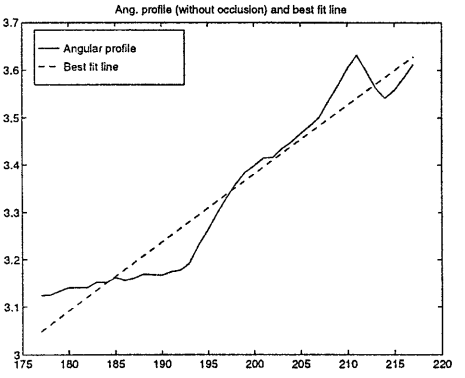
would like to answer is can this change in the local slope of the angular profile be used to account for the change in scale in the HRP wing element.

As we now show, it is possible to normalize the scale of the HRP elements based on the local slope of the angular profile. Recall that the centroidal distance profile plus the angular profile are a complete description of the curve describing the boundary of the silhouette. In fact, the centroidal distance profile, $f(m)$, and angular profile, $\theta(m)$, are the polar coordinates of the sampled boundary curve, for $m = 1 \ldots P$. That is, the boundary curve of a silhouette is described by the polar coordinates $(f(m), \theta(m))$. Suppose that $r(m)$ is a cubic b-spline extracted from $f(m)$ using HRP and that $a(m)$ is the best fit line to the angular profile $\theta(m)$ over the support of $r(m)$. Then the polar coordinates $(r(m), a(m))$ trace out an approximation to a portion of the boundary curve described by $(f(m), \theta(m))$.

Having established the relevance of curves described by $(r(m), a(m))$ where $r(m)$ is a cubic b-spline and $a(m)$ is a line, we now turn to a more abstract investiga-



(a) Angular profile of the unoccluded version of Plane #1.



(b) Portion of the angular profile (for the unoccluded plane) corresponding to the wing and the best fit line.

Figure 4-23: Angular profile for the unoccluded plane.

tion of these curves. Suppose that a curve is described by the polar coordinates $(r_0(m), a_0(m))$ where $r_0(m)$ is a cubic b-spline at scale j_0 and $a_0(m)$ is a line with slope s_0. Figure 4-24 illustrates $r_0(m)$, $a_0(m)$, and the curve described by the polar coordinates $(r_0(m), a_0(m))$. Actually, the curve shown in Figure 4-24c may be represented by many pairs of polar coordinates of the form $(r(m), a(m))$ where $r(m)$ is a cubic b-spline and $a(m)$ is a line. For example, one would expect that the polar coordinates $(r(m), a(m))$, where $r(m)$ is a cubic b-spline with half the support of the $r_0(m)$ (i.e. a cubic b-spline at scale $j_0 + 1$) and $a(m)$ is a straight line with double the slope of $a_0(m)$, would be one alternative representation of the the curve shown in Figure 4-24c. Generalizing this intuition, we note that the polar coordinates $(r(m), a(m))$ where $r(m)$ is a cubic b-spline at scale j and $a(m)$ is a line with slope s describe the curve shown in Figure 4-24c if

$$j = j_0 + \log_2\left(\frac{s}{s_0}\right). \tag{4.20}$$

This is the general relationship between the scale of $r(m)$ and the slope of $a(m)$ which we will use to normalize the scale of the HRP elements based on the local slope of the angular profile. Specifically, suppose that the scale of an HRP element extracted from a centroidal distance profile is j and the slope of the angular profile over the support of the element is s, then the slope-adjusted scale will be given by $S(j)$ where

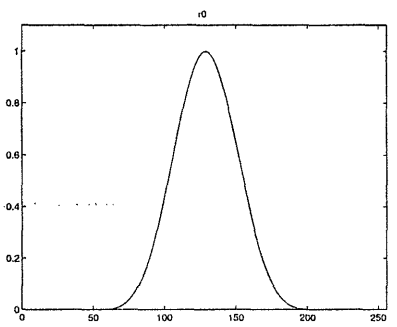
$$S(j) = j + \log_2\left(\frac{s_0}{s}\right) \tag{4.21}$$

where we have chosen to use a reference scale of $s_0 = 2\pi/P$ where P is the number of points in the centroidal profile. This value of s_0 is the average slope of the angular profile over the entire interval.

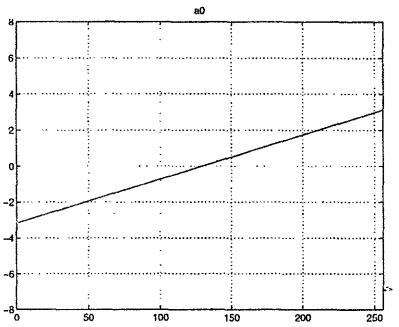
Feature Vector Construction

The feature vector describing a particular silhouette will be derived from the parameters of the four HRP elements extracted from the centroidal distance profile corresponding to the silhouette. Each HRP element has three parameters and therefore, the feature vector will consist of 12 components. The main issue that we must address are how to order the HRP parameters in this feature vector.

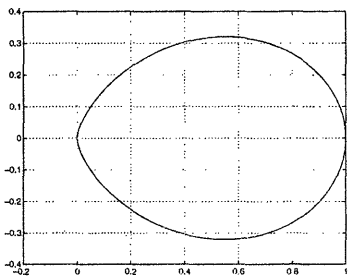
Again, the underlying intuition of our approach is to extract parameters from the centroidal distance profile which correspond to the physical features of the plane.



(a) $r_0(m)$



(b) $a_0(m)$



(c) The curve described by $(r_0(m), a_0(m))$

Figure 4-24: The functions $r_0(m)$ and $a_0(m)$ and the curve described by the polar coordinates (r_0, a_0).

Ideally, we would like to construct our feature vector $\mathbf{y}$ with the following structure

$$\mathbf{y} = \begin{bmatrix} S(j_{\text{nose}}) \\ \theta(t_{\text{nose}}) \\ v_{\text{nose}} \\ S(j_{\text{wing}_1}) \\ \theta(t_{\text{wing}_1}) - \theta(t_{\text{nose}}) \\ v_{\text{wing}_1} \\ S(j_{\text{tail}}) \\ \theta(t_{\text{tail}}) - \theta(t_{\text{nose}}) \\ v_{\text{tail}} \\ S(j_{\text{wing}_2}) \\ \theta(t_{\text{wing}_2}) - \theta(t_{\text{nose}}) \\ v_{\text{wing}_2} \end{bmatrix} \tag{4.22}$$

where j_{nose}, t_{nose}, and v_{nose} are the scale, translation, and magnitude, respectively, of the HRP element corresponding to the nose feature in the centroidal distance profile, etc. The HRP algorithm extracts elements from the centroidal distance profile one at a time, but this order does not generally correspond to physical features of the plane. That is, the algorithm will not always pick an element which corresponds to the nose feature first, the wing feature second, and so on. To construct the feature vector with the desired structure, we will use the translation parameters of the elements extracted by HRP. Suppose, for the time-being, that all the profiles are oriented to begin at the tip of the nose. Assuming that our profiles have length P, we could then take the first element to be the one which has a translation $t \in [1, .1P] \cup [.9P, P]$, and then order the remaining profiles according to translation. In this way, the first element is labeled the nose, the second is wing #1, then third is the tail, and the

fourth is wing #2. The feature vector is then constructed as in (4.22). Note that since we have assumed that all the profiles are oriented to begin at the tip of nose, $\theta(t_{\text{nose}}) \approx -\pi/2$ for all the planes.

This same ordering scheme can also be used to accommodate *small* errors in orientation. That is, if the silhouette has small error in orientation and the therefore the centroidal distance profile does not begin precisely at the tip of the nose, we can still take the first element to be the one which has a translation $t \in [1, .1P] \cup [.9P, P]$, and then order the remaining profiles according to translation. Again, the first element is labeled the nose, the second is wing #1, then third is the tail, and the fourth is wing #2. To make the feature vector $\mathbf{y}$ more robust, we set $\theta(t_{\text{nose}}) = -\pi/2$. That is, we have just adjusted for the small orientation error. The feature vector is then constructed as in (4.22).

Often it will be convenient for us to use the following notation for $\mathbf{y}$

$$\mathbf{y} = \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \\ \mathbf{e}_4 \end{bmatrix} \tag{4.23}$$

where $\mathbf{e_1}$ refers to the three parameters of the element corresponding to the nose, $\mathbf{e_2}$ refers to the three parameters of the element corresponding to the first wing, $\mathbf{e_3}$ refers to the three parameters of the element corresponding to the tail, and $\mathbf{e_4}$ refers to the three parameters of the element corresponding to the second wing. We will also occasionally refer to these as the elemental components of $\mathbf{y}$. In contrast, when we refer to the components of $\mathbf{y}$ we mean the individual parameters of the vector $\mathbf{y}$ as enumerated in (4.22).

4.4.2 M-ary Hypothesis Testing

The M-ary hypothesis test is used to recognize silhouettes based on the feature vector $\mathbf{y}$ constructed as indicated in (4.22). The main issue that must be addressed in applying the M-ary hypothesis testing approach to this problem is to choose appropriate models for the conditional distributions. In the recognition of aircraft from the data set shown in Section 4.2, we must also accommodate the possibility of switches in the elemental components of the feature vector and the possibility that portions of the data vector are corrupted because of occlusion.

Conditional Distribution Models

To simplify the recognition procedure, we choose to model the twelve components of the feature vector $\mathbf{y}$ as independent. This implies that

$$p_{\mathbf{y}|H_m}(\mathbf{Y}|H_m) = \prod_{i=1}^{12} p_{y_i|H_m}(Y_i|H_m). \tag{4.24}$$

Future work may develop a recognition scheme which takes advantage of dependencies in the feature vector $\mathbf{y}$.

The conditional distributions associated with components of the feature vector corresponding to slope-adjusted scale values and magnitudes will be modeled as Gaussian. That is, conditional densities of components #1,3,4,6,7,9,10,12 of the feature are modeled as Gaussian. The mean and variance for component i under hypothesis m will be denoted $E_{m,i}$ and $\sigma^2_{m,i}$, respectively. This implies that the form of each of these conditional densities is given by

$$p_{y_i|H_m}(Y_i|H_m) = \frac{1}{\sqrt{2\pi}\sigma_{m,i}} \exp\left(-\frac{(Y_i - E_{m,i})^2}{2\sigma^2_{m,i}}\right) \tag{4.25}$$

The parameters $E_{m,i}$ and $\sigma^2_{m,i}$ will be approximated as the sample means and variances

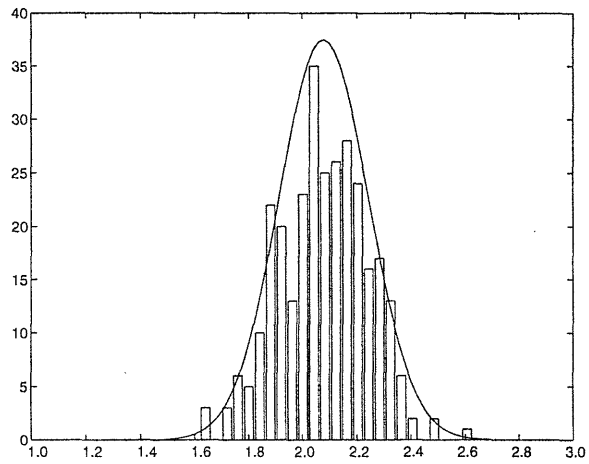


Figure 4-25: A histogram of training data for component #4 for Plane #1 and the corresponding Gaussian distribution model

from training data. For most of the aircraft, the Gaussian model provides a fairly accurate description of the scale and magnitude components of the feature vector, as illustrated in Figure 4-25, which shows a histogram of the training data for component #4 of realizations of $\mathbf{y}$ for Plane #1 from centroidal distance profiles which have been corrupted using the boundary perturbation model described in Section 4.2.2 with $p = 40\%$ and $s = 0.9$.

Components # 2, 5, 8, and 11 of the feature vector are angles and are not well modeled by the Gaussian distribution. One reason for this is the Gaussian distribution does not account for the periodicity inherent in an angular random variable. Components # 2, 5, 8, and 11 will instead be modeled by the "exponential cosine" distribution [35],

$$p_{y_i|H_m}(Y_i|H_m) = \frac{\exp(\alpha_{m,i}\cos(Y_i - E_{m,i}))}{2\pi I_0(\alpha_{m,i})} \tag{4.26}$$

where I_0 is a zeroth order modified Bessel function of the first kind, $E_{m,i}$ is the mean of component i under hypothesis m, and $\alpha_{m,i}$ is a parameter related to the variance of component i under hypothesis m. Figure 4-26 shows this exponential

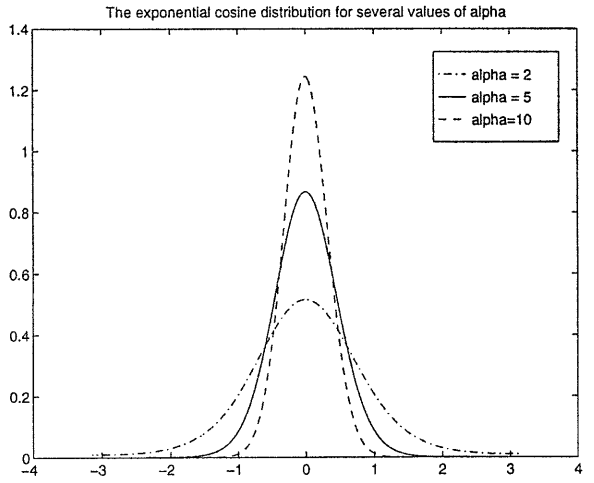


Figure 4-26: The exponential cosine distribution for several values of α.

cosine distribution for $E_{m,i} = 0$ and several values of α. For small values of α, the exponential cosine distribution looks very similar to a Gaussian distribution. The variance of this distribution is given by

$$\sigma^2 = \frac{\pi}{3} + 4\sum_{n=1}^{\infty} \frac{(-1)^n I_n(\alpha)}{n^2 I_0(\alpha)}. \tag{4.27}$$

Figure 4-27 shows the relationship between α and σ. In our M-ary hypothesis tester, the parameter $E_{m,i}$ will be approximated as the sample mean of y_i of the training data for plane m. The parameter $\alpha_{m,i}$ will be determined from the sample variance of y_i of the training data for plane m.

To summarize, using these models for the conditional probabilities, we choose among the seventeen hypothesis (where each hypothesis corresponds to a particular plane) which the most likely hypothesis H_4^* to be

$$H_4^* = \arg\max_{H_m} p_{\mathbf{y}|H_m}(\mathbf{Y}|H_m) \tag{4.28}$$

where we have included the subscript "4" to indicate that this is a decision based on

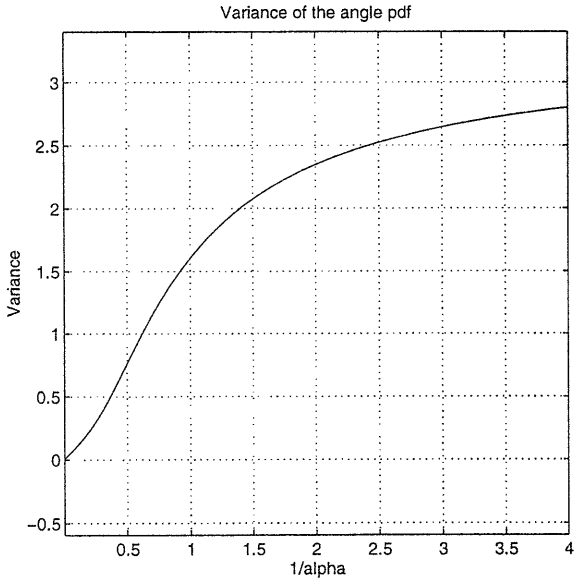


Figure 4-27: Variance of the exponential cosine distribution as a function of α.

the four elemental components in $\mathbf{y}$. Recalling our independence assumption we have

$$p_{\mathbf{y}|H_m}(\mathbf{Y}|H_m) = \prod_{i=1}^{12} p_{y_i|H_m}(Y_i|H_m) \tag{4.29}$$

where

$$p_{y_i|H_m}(Y_i|H_m) = \frac{\exp(\alpha_{m,i}\cos(Y_i - E_{m,i}))}{2\pi I_0(\alpha_{m,i})} \tag{4.30}$$

for $i = 2, 5, 8, 11$ and

$$p_{y_i|H_m}(Y_i|H_m) = \frac{1}{\sqrt{2\pi}\sigma_{m,i}} \exp\left(-\frac{(Y_i - E_{m,i})^2}{2\sigma_{m,i}^2}\right) \tag{4.31}$$

for $i = 1, 3, 4, 6, 7, 9, 10, 12$. Recall that if the underlying conditional distributions $p_{y_i|H_m}(Y_i|H_m)$ were all Gaussian, then maximizing $p_{\mathbf{y}|H_m}(\mathbf{Y}|H_m)$ is equivalent to minimizing a weighted distance. Since some of the underlying conditional distributions are exponential cosine instead of Gaussian, maximizing $p_{\mathbf{y}|H_m}(\mathbf{Y}|H_m)$ is not precisely

equivalent to minimizing a weighted distance. Intuitively, however, since the exponential cosine may be thought of as a Gaussian for large values of α, we can think of maximizing $p_{\mathbf{y}|H_m}(\mathbf{Y}|H_m)$ as being very close to minimizing a weighted distance. This basic procedure is amended in light of the following important caveats.

Swept Wing Aircraft

As mentioned in Section 4.2, Planes #7 and 17 are swept wing aircraft. Our training data indicate that we must use some caution in dealing with these aircraft. In fact, the training data indicate that the feature vectors extracted from these planes do not seem to be well modeled by the Gaussian distribution for the scale and magnitude components nor by the exponential cosine for the angular components. For example, Figure 4-28 shows the histogram for the training data for component #7 of Plane #7. Clearly, the data does not seem well matched to the Gaussian distribution. The main reason for such a wide variation in the parameters of the HRP elements is that for these planes HRP yields two, rather than four stable features. This fact is not surprising since the planes lack distinct wings, see Figure 4-5. As a result of the ordering procedure outlined in the previous section, the components of $\mathbf{y}$ for Planes #7 and 17 show a wide range of variation. Essentially, the labeling procedure of the previous section has assigned an incorrect label to the peaks in the centroidal distance profile. Suppose the *elemental* components of $\mathbf{y}$ are relabeled according to the magnitude parameter, v, instead of the translation parameter. That is, let the nose correspond to the element with the maximum magnitude (i.e. v), the tail correspond to the element with the next largest magnitude, and the wings correspond to the elements with the smallest magnitudes. Using this ordering scheme yields a feature vector whose components are well modeled by the Gaussian and exponential cosine distributions. For example, Figure 4-29 shows the histogram for component #

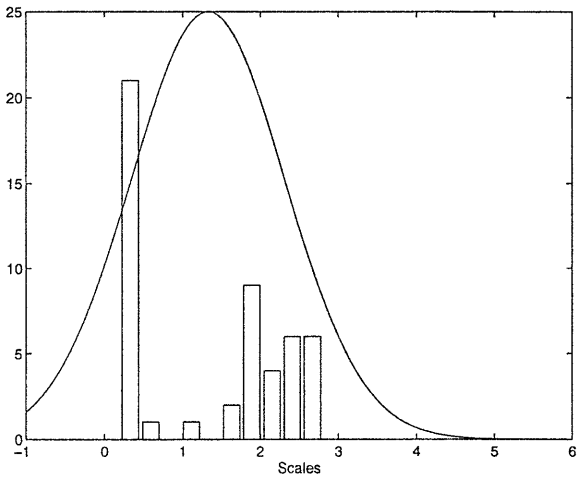


Figure 4-28: Histogram of training data for component #7 for Plane #7.

7 of Plane # 7 after this reordering. For these reasons, training data from Planes #7 and #17 will be reordered according to magnitude.

All of our testing data will be order according to translation as outlined in the previous section. This implies that in comparing features extracted from some unknown silhouette to models #7 and 17, we must treat the ordering of the HRP elements as a nuisance parameter and use the GLRT to find the optimal ordering. There are 24 possible orderings of the four elemental components of $\mathbf{y}$ and we use the GLRT to determine which one of these is most likely.

Occlusion

Further, as a result of occlusion, one or more of the elemental components of $\mathbf{y}$ may be corrupted. However, since the HRP algorithm is locally based, the remaining elemental components still give an accurate description of the physical features of the underlying plane. The GLRT is incorporated in our hypothesis testing procedure to account for the possibility of occlusion. This is a direction extension of Example # 3 from Section 4.3.2.

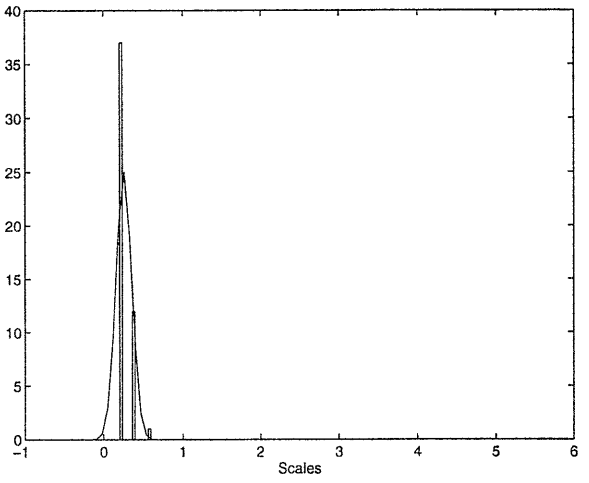


Figure 4-29: Histogram of training data for component #7 for Plane #7 after switching components.

If we know that precisely one elemental component has been corrupted, then the GLRT says for each hypothesis H_m determine which elemental component has most likely been corrupted. That is, find τ_m^*

$$\tau_m^* = \arg\max_{\tau} p_{\mathbf{y}|H_m,\tau}(\mathbf{Y}|H_m,\tau) \tag{4.32}$$

where

$$p_{\mathbf{y}|H_m,\tau}(\mathbf{Y}|H_m,\tau) = \begin{cases} \prod_{i\in\{2,3,4\}} p_{\mathbf{e}_i|H_m}(\mathbf{E_i}|H_m) & \text{if } \tau = 1 \\ \prod_{i\in\{1,3,4\}} p_{\mathbf{e}_i|H_m}(\mathbf{E_i}|H_m) & \text{if } \tau = 2 \\ \prod_{i\in\{1,2,4\}} p_{\mathbf{e}_i|H_m}(\mathbf{E_i}|H_m) & \text{if } \tau = 3 \\ \prod_{i\in\{1,2,3\}} p_{\mathbf{e}_i|H_m}(\mathbf{E_i}|H_m) & \text{if } \tau = 4 \end{cases} \tag{4.33}$$

where $\mathbf{E_i}$ refers to the elemental components of $\mathbf{Y}$. Then, determine the most likely hypothesis by excluding the component which is mostly likely corrupted for each hypothesis, as follows

$$H_3^* = \arg\max_{H_m} p_{\mathbf{y}|H_m,\tau}(\mathbf{Y}|H_m,\tau_m^*) \tag{4.34}$$

where we have included the subscript "3" to indicate that this is a decision based on three elemental components of $\mathbf{y}$.

Similarly if we know that precisely two elemental components have been corrupted, then the GLRT says for each hypothesis H_m determine which two elemental components are most likely to have been corrupted. That is, find δ_m^* where

$$\delta_m^* = \arg\max_{\delta} p_{\mathbf{y}|H_m,\delta}(\mathbf{Y}|H_m,\delta) \tag{4.35}$$

and

$$p_{\mathbf{y}|H_m,\delta}(\mathbf{Y}|H_m,\delta) = \begin{cases} \prod_{i\in\{1,2\}} p_{\mathbf{e}_i|H_m}(\mathbf{E_i}|H_m) & \text{if } \delta = 1 \\ \prod_{i\in\{1,3\}} p_{\mathbf{e}_i|H_m}(\mathbf{E_i}|H_m) & \text{if } \delta = 2 \\ \prod_{i\in\{1,4\}} p_{\mathbf{e}_i|H_m}(\mathbf{E_i}|H_m) & \text{if } \delta = 3 \\ \prod_{i\in\{2,3\}} p_{\mathbf{e}_i|H_m}(\mathbf{E_i}|H_m) & \text{if } \delta = 4 \\ \prod_{i\in\{2,4\}} p_{\mathbf{e}_i|H_m}(\mathbf{E_i}|H_m) & \text{if } \delta = 5 \\ \prod_{i\in\{3,4\}} p_{\mathbf{e}_i|H_m}(\mathbf{E_i}|H_m) & \text{if } \delta = 6 \end{cases} \tag{4.36}$$

Then, determine the most likely hypothesis by excluding the components which are mostly likely corrupted for each hypothesis, as follows

$$H_2^* = \arg\max_{H_m} p_{\mathbf{y}|H_m,\delta}(\mathbf{Y}|H_m,\delta_m^*) \tag{4.37}$$

where we have included the subscript "2" to indicate that this is a decision based on two elemental components of $\mathbf{y}$.

The question now arises if we are given a unknown a feature vector $\mathbf{y}$ extracted from centroidal profile a silhouette which may or may not be distorted due to occlusion, how does one determine if zero, one or two elemental components are corrupted and what is the most likely hypothesis given the data. In other words, if $H_4^* \neq H_3^* \neq H_2^*$, which hypothesis should we choose? We then choose among H_2^*, H_3^*

and H_4^* based on the likelihoods associated with each of these results. For convenience, let us define $p_{\mathbf{y}|H_2^*}(\mathbf{Y}|H_2^*)$, $p_{\mathbf{y}|H_3^*}(\mathbf{Y}|H_3^*)$, and $p_{\mathbf{y}|H_4^*}(\mathbf{Y}|H_4^*)$ as

$$p_{\mathbf{y}|H_2^*}(\mathbf{Y}|H_2^*) = \max_{H_m} p_{\mathbf{y}|H_m,\delta}(\mathbf{Y}|H_m,\delta_m^*) \tag{4.38}$$

$$p_{\mathbf{y}|H_3^*}(\mathbf{Y}|H_3^*) = \max_{H_m} p_{\mathbf{y}|H_m,\tau}(\mathbf{Y}|H_m,\tau_m^*) \tag{4.39}$$

$$p_{\mathbf{y}|H_4^*}(\mathbf{Y}|H_4^*) = \max_{H_m} p_{\mathbf{y}|H_m}(\mathbf{Y}|H_m). \tag{4.40}$$

Recall our intuition that the M-ary hypothesis test based on Gaussian and exponential cosine conditional densities is roughly equivalent to minimizing a weighted distance. Intuitively, since the distance between n elements will be larger than the distance between $n-1$ elements, we expect

$$p_{\mathbf{y}|H_2^*}(\mathbf{Y}|H_2^*) > p_{\mathbf{y}|H_3^*}(\mathbf{Y}|H_3^*) > p_{\mathbf{y}|H_4^*}(\mathbf{Y}|H_4^*). \tag{4.41}$$

The hypothesis we choose then is given by

$$H^* = \arg\max(p_{\mathbf{y}|H_4^*}(\mathbf{Y}|H_4^*), p_{\mathbf{y}|H_3^*}(\mathbf{Y}|H_3^*)/\mathcal{P}, p_{\mathbf{y}|H_2^*}(\mathbf{Y}|H_2^*)/\mathcal{P}^2) \tag{4.42}$$

where $\mathcal{P}$ is an empirically determined term to penalize the exclusion of signal features.

4.5 Experiments

In this section, we describe experiments to test our M-ary hypothesis plus GLRT approach. The results indicate that this approach is robust to boundary perturbations, scale variation, small orientation variation, and distortions due to occlusion.

In these experiments, we will be referring to training data sets and testing data sets which are disjoint (so we have not tested on the same data used in training).

Training data sets are used only to calculate sample means and variances for use in the appropriate conditional distributions for each element of the feature vector $\mathbf{y}$. Our proposed approach to analyze the training data will be to extract centroidal distance profiles from silhouettes and analyze those centroidal distance profiles using HRP to extract four elements. The HRP depth parameter, k, is set to 3 for the first two elements, 2 for the next element, and 1 for the last element. Then, we will organize the parameters of the HRP elements using the translation for all planes in the data set except Planes #7 and #17 which will be organized by the magnitude parameter.

The results we present in this section are the classification of the testing data sets. For the testing data set, our proposed approach will be to analyze centroidal distance profiles using HRP to extract four elements (using $k = 3$ for the first two elements, $k = 2$ for the third element, and $k = 1$ for the fourth element). Then, we organize the parameters of the HRP elements using translation for every profile and construct a feature vector $\mathbf{y}$ as described in (4.22). When comparing $\mathbf{y}$ to the models corresponding to Planes #7 and #17, we will use GLRT to account for possible switches of the elemental components of $\mathbf{y}$. Using this procedure we determine the best hypothesis using four, three and two element components of $\mathbf{y}$, i.e. H_4^*, H_3^*, H_2^*. The optimal hypothesis is then chosen according to (4.42).

For comparison, we also show results based on Fourier descriptors (FD) and the K nearest neighbor test with $K = 3$. This approach based on FD was described in [23] and was summarized in Section 4.2.3. Note that this approach uses $P/2$ features where P is the number of points in the centroidal distance profile. In the experiments we describe in this section $P = 256$ so the FD technique uses 128 features. In contrast, the HRP based technique uses a feature vector with only 12 components.

4.5.1 Boundary Perturbations

We construct training and testing data sets using the boundary perturbation model described in 4.2.2 for $p = 40\%$ and for several values of s. In this section, there is no scale variation, orientation variation, or variation due to occlusion in either the testing or training data sets. However, these testing data are classified using the M-ary hypothesis plus GLRT approach to account for the possibility of switching of the elemental components of $\mathbf{y}$ and the corruption of elemental components of $\mathbf{y}$ due to occlusion (i.e. while the data do not include object occlusion, the algorithm is not given this information and therefore must include the possibility of occlusion).

For the first experiment, both the training and testing data sets consisted of noisy profiles generated using $p = 40\%$ and $s = 0.9$. That is, the noise level was the same for both the training and testing data. Figure 4-30 shows the percent correct classification of the testing data as a function of plane number for both our HRP based technique and the FD nearest neighbor classifier. Figure 4-30 also shows the one standard deviation error bars for the estimate of percent of correct classification for each plane. Since the overall percent correct classification is 99%, these results indicate that both techniques perform very well. The HRP technique gives 99.42% correct classification overall and the FD technique gives 99.65% correct classification overall. Given that the standard deviation of the estimate of the percent correct classification is about 2.5%, we conclude that this difference in overall performance is not statistically significant.

Next, noisy profiles were generated using $p = 40\%$ and $s = 1.5$. Again, the noise level was the same for both the training and testing data sets. Figure 4-31 shows the percent correct classification of the testing data and one standard deviation error bars for the two methods for this case. The overall percent correct classification is 96.53% for the HRP based technique and 97.64% for the FD based technique. Again, given

that the standard deviation of the estimate of the percent correct classification 2.5%, we conclude that this difference in overall performance is not statistically significant.

Similarly, noisy profiles were generated using $p = 40\%$ and $s = 2.1$. The noise level was the same for both the training and testing data sets. Figure 4-32 shows the percent of correct classification of the testing data, as well as the one standard deviation error bars. The overall percent correct classification is 94.51% for the HRP based technique and 91.37% for the FD technique. These results show that while the FD technique shows deterioration for all the planes in the data set, the HRP technique has errors associated with only a few planes, namely, Planes #2, 4, 8 and 9, and has slightly better overall performance, at a modest level of statistical significance.

4.5.2 Rotational Noise

Next, we construct testing data which have rotational variation between $\pm 10^o$ plus boundary perturbations generated with $p = 40\%$ and $s = 0.9$, the smallest level of boundary perturbations. The training data used was generated without rotation and with boundary perturbations using $p = 40\%$ and $s = 0.9$. Thus, there is a mismatch between the training and testing data. Figure 4-33 shows the percent correct classification of the testing data for each plane and for both the HRP and FD techniques. In addition, Figure 4-33 shows the one standard deviation error bars for the estimate of percent of correct classification for each plane. Again, the difference in the overall performance is not statistically significant.

4.5.3 Scale Variation

Up until this point, we have not discussed the effect of scale on our M-ary hypothesis testing plus GLRT recognition scheme. Recall from Section 4.2.2 that scale variation in the silhouette leads to a change in amplitude in the centroidal distance profile. Also,

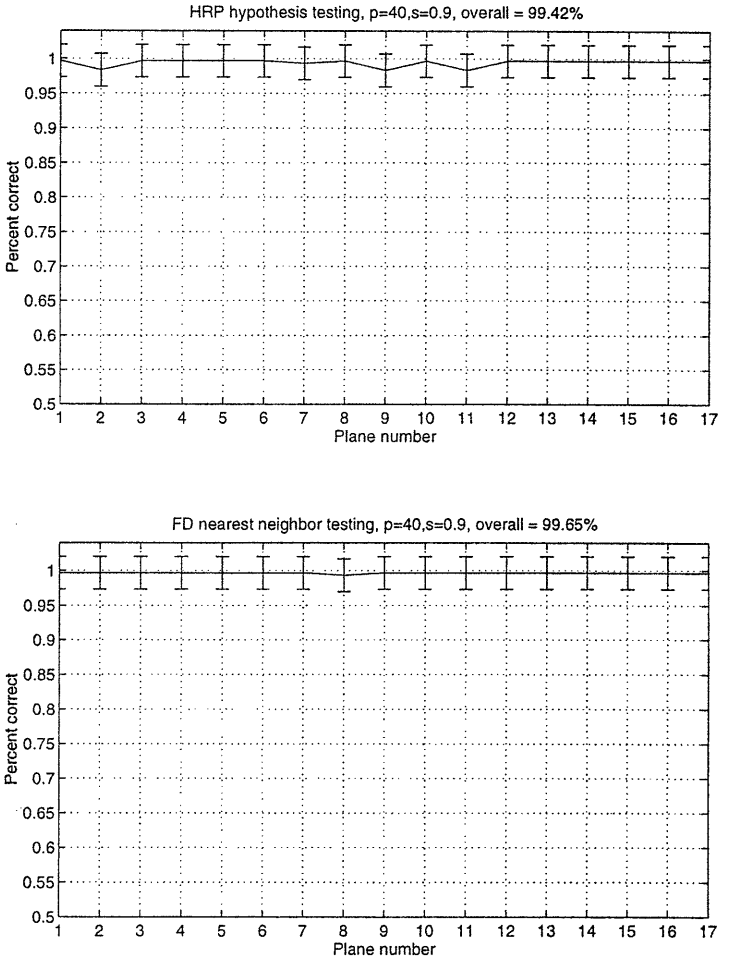


Figure 4-30: Classification results for noisy data generated using $p = 40\%$ and $s = 0.9$. For this experiment, the noise level in the training and testing data was the same and there was no scale or orientation variation. This figure shows the percent correct classification of the testing data for each plane and the one standard deviation error bars.

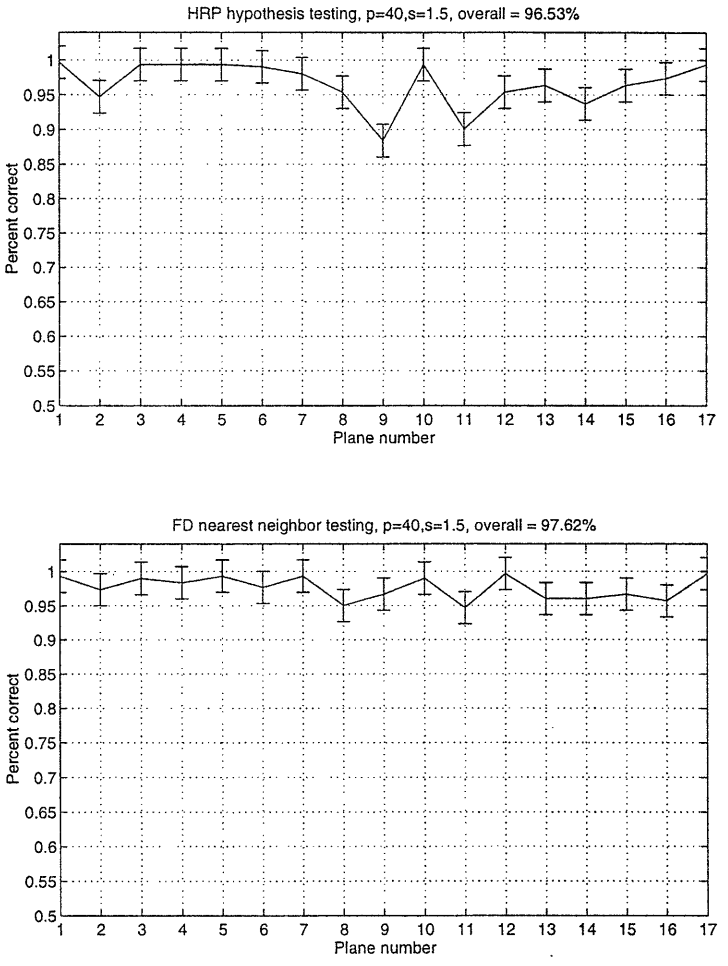


Figure 4-31: Classification results for noisy data generated using $p = 40\%$ and $s = 1.5$. For this experiment, the noise level in the training and testing data was the same and there was no scale or orientation variation. This figure shows the percent correct classification of the testing data for each plane and the one standard deviation error bars.

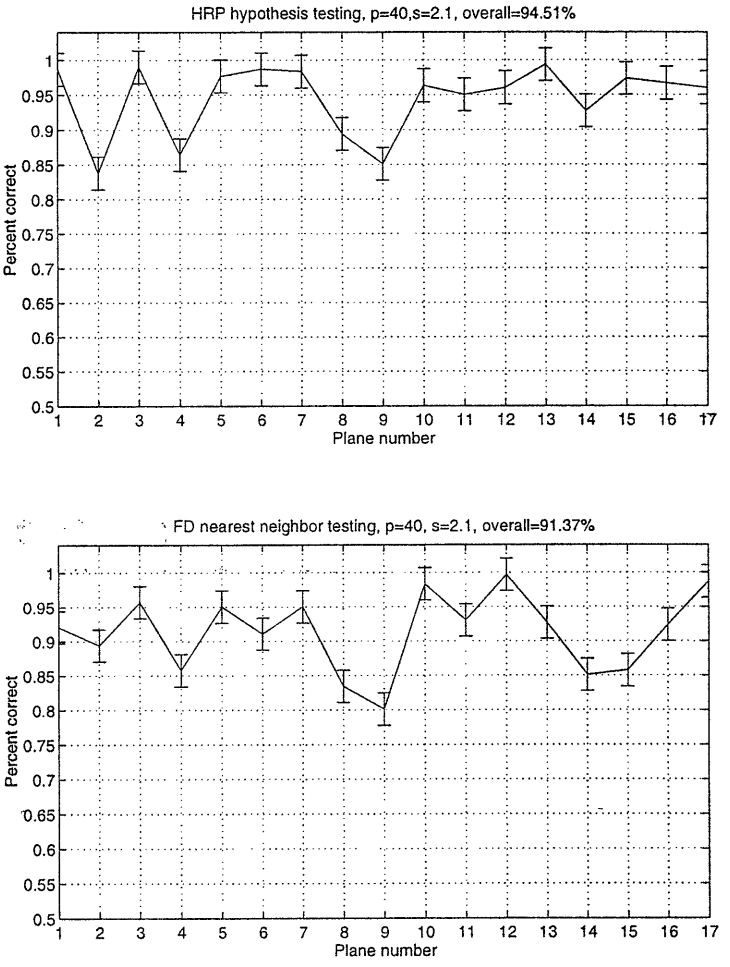


Figure 4-32: Classification results for noisy data generated using $p = 40\%$ and $s = 2.1$. For this experiment, the noise level in the training and testing data was the same and there was no scale or orientation variation. This figure shows the percent correct classification of the testing data for each plane and the one standard deviation error bars.

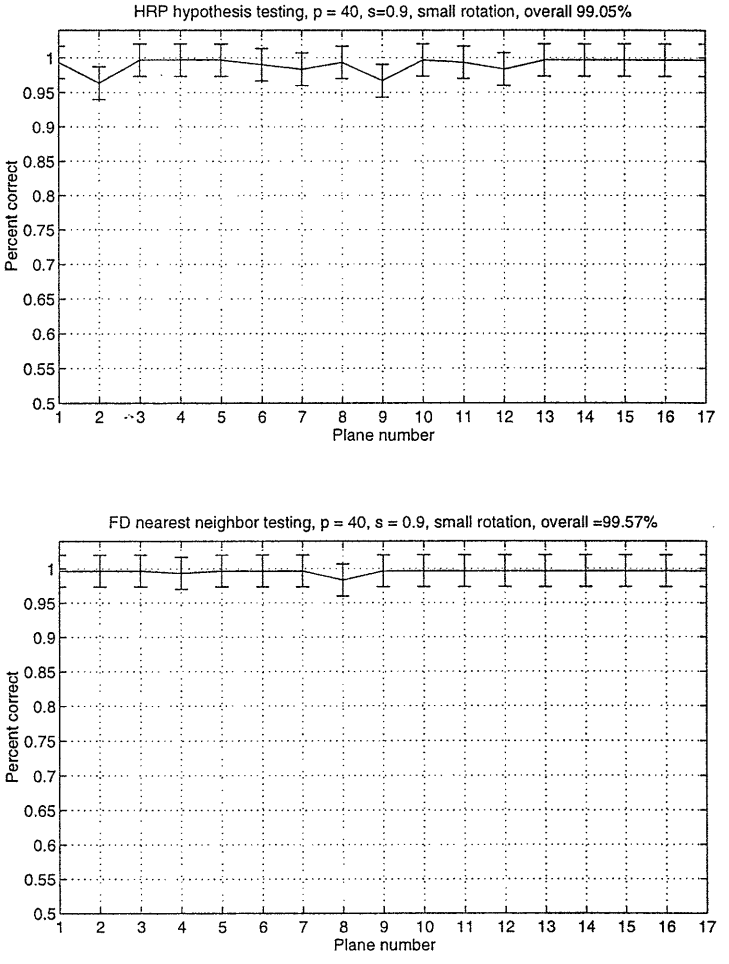


Figure 4-33: Classification results in the presence of rotational variation. The training data is generated with $p = 40\%$, $s = 0.9$ and no rotational variation. The testing data is generated with $p = 40\%$, $s = 0.9$ and rotational variation between $\pm 10^o$. This figure shows the percent correct classification of the testing data for each plane and the one standard deviation error bars.

recall from Chapter 3 that amplitude variation in a signal will only effect the magnitude parameter of the elements of the HRP decomposition of that signal. Certainly, the algorithm as we have described it will not be robust to variations in the scale of the silhouette. However, robust classification in the presence of scale variation of the silhouette can be accomplished with a very simple change in the basic algorithm. The basic M-ary hypothesis testing plus GLRT algorithm can be made insensitive to scale variation by normalizing each centroidal profile by its average value. That is, for both the training and testing data sets, each centroidal distance profile is normalized by its average value and the recognition proceeds as previously outlined. This insensitivity to scale variation is illustrated in the following experiment. For this experiment, noisy centroidal distance profiles are generated as described in Section 4.2.2 with $p = 40\%$ and $s = 0.9$ and normalized by their average value. Figure 4-34 shows the percent correct classification of the testing data for both the HRP technique and the FD technique. These results indicate that both techniques perform well (> 99 %) and the difference in the overall performance is not statistically significant.

4.5.4 Occlusion

Finally, we investigate the performance of the M-ary hypothesis plus GLRT recognition approach in the presence of occlusion. For this experiment, the training data set consists of centroidal profiles with the smallest level of boundary noise, i.e. $p = 40\%$ and $s = 0.9$. For the testing data set, we construct centroidal profiles corrupted by occlusion as described in [25] and summarized in Section 4.2.2 with $q = 10\%$. That is, for the testing data set 10% of the boundary is replaced by a straight line. Figure 4-35 shows the percent of correct classification for the testing data for both the HRP and FD techniques as well as the one standard deviation error bars. This figure shows that HRP does substantially better than the FD technique. The HRP

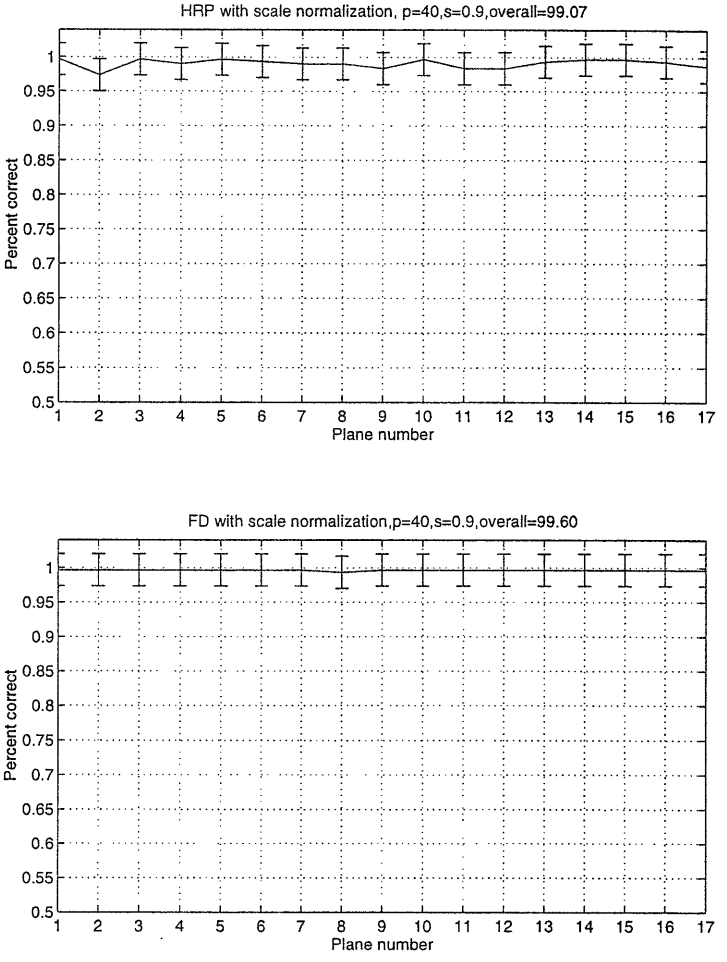


Figure 4-34: Classification results for the scale insensitive algorithm. Training and testing data generated using $p = 40\%$ and $s = 0.9$. Each profile normalized by its average value. Figure shows the percent correct classification of the the testing data for each plane and the one standard deviation error bars.

technique shows 94% correct classification while the FD technique shows only 88% correct classification, a statistically significant difference in which the classification errors have been cut in half.

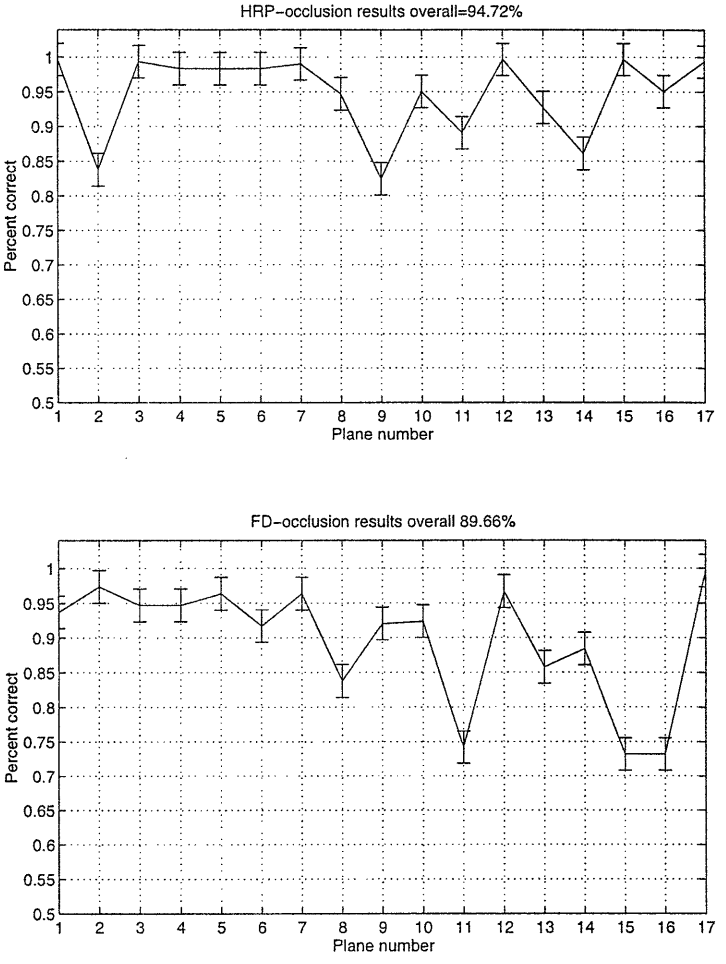


Figure 4-35: Classification results in the presence of occlusion. Training data consists of centroidal profiles with boundary noise using $p = 40\%$ and $s = 0.9$. Testing data consists of centroidal profiles with no boundary perturbations but with variation due to occlusion using $q = 10\%$. Figure shows the percent of correct classification for each plane and the one standard deviation error bars.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis, we have developed a new adaptive approximation technique, high resolution pursuit (HRP), and demonstrated how HRP can be used to extract features which are suitable for object recognition. In our investigation of HRP as an adaptive approximation technique, we have found that it yields decompositions which rival those obtained using basis pursuit (BP) in terms of super-resolution and sparsity, yet is less computationally intensive than BP. To investigate the performance of HRP as a technique for feature extraction, we demonstrated how HRP could be used to extract features from airplane silhouettes and how these features can be used to recognize airplane silhouettes in the presence of boundary perturbations and variations due to occlusion.

5.1.1 HRP for Adaptive Approximation

In this work, we developed a new algorithm for adaptive approximation, HRP. Like other techniques in adaptive approximation, HRP is used to decompose a function

over a redundant dictionary. The HRP algorithm is similar in structure to the matching pursuit (MP) algorithm. In contrast to MP, HRP employs a similarity measure which emphasizes local fit over global fit.

We can evaluate the HRP algorithm in terms of the six characteristics of the "optimal" decomposition which were outlined in Chapter 1. The HRP algorithm can be used to obtain decompositions which preserve sparsity and exhibit superresolution, as evidenced by the twin peaks example of Chapter 3. The first few elements extracted by HRP give a coarse approximation of the function. Thus, HRP decompositions have a corresponding hierarchy. It is this hierarchy which allows us to use only four elements for the recognition of airplanes from silhouettes in Chapter 4. Chapter 4 also demonstrates that HRP decompositions are stable in the presence of perturbations. Finally, Section 3.4 shows that the HRP algorithm can be efficiently implemented and has the the same computational complexity as the MP algorithm.

This thesis included an investigation of the behavior of HRP. During the course of our investigation of HRP in Chapter 3, we described two distinct interpretations of the HRP algorithm. In our geometric interpretation of HRP, we showed that the HRP similarity measure is an oblique projection of the function onto the dictionary elements, where the magnitude of this oblique projection is determined by the subfamily associated with the dictionary element. We also showed that the HRP algorithm can be interpreted as solving a constrained maximization of the difference between successive residuals. Further, we investigated the convergence properties of the HRP algorithm and were able to prove that for a discrete signal the norm of the residuals produced by the algorithm converge exponentially.

Finally, in Section 3.5 we compared decompositions obtained using HRP to those obtained using BP and MP. These examples employed both the cubic b-spline and wavelet packet dictionaries. These examples demonstrated that HRP decompositions

rival those of BP in terms of sparsity and super-resolution, yet the HRP algorithm is computationally much less intensive.

5.1.2 HRP for Feature Extraction

We have also illustrated that HRP can be used to extract features which are suitable for object recognition. The use of HRP for feature extraction and object recognition was illustrated using a data set of 17 airplanes in Chapter 4. In spite of the fact that several of these airplanes were very similar to one another, the HRP-based recognition scheme that we developed was able to distinguish the planes from one another in the presence of boundary perturbations, scale variation, small orientation variation, and variations due to occlusion.

In the object recognition context, the elements extracted by HRP are a new class of features that describe the geometric (i.e. size and location) properties of subparts of the object. These features based on the HRP decomposition are unique in that they exhibit some properties typically associated with global features and some properties typically associated with local features. As a result, these features based on the HRP decomposition are robust to both boundary noise and occlusion.

In developing an HRP-based scheme for object recognition of airplane silhouettes, we demonstrated how to choose an appropriate value for the HRP depth parameter and how to incorporate angular information which was discarded in the 1D representation of the silhouette. Further, we developed a recognition engine based on M-ary hypothesis testing and the generalized likelihood ratio. This entailed developing accurate conditional probability models for the features extracted from each plane.

In Chapter 4, we demonstrated that these HRP-based features were useful for recognizing airplanes from their silhouettes. The performance of the HRP-based approach was shown to be comparable to an approach based on Fourier descriptors

when the the silhouettes were corrupted by boundary perturbations, scale variation, and small orientation variation. In the presence of variations due to occlusion, the performance of the HRP-based method surpassed the performance of the method based on Fourier descriptors.

5.2 Future Work

As a general adaptive approximation technique, research on HRP may be extended in the following ways. First, although we have proved that the HRP algorithm converges for discrete signals, we have not proved that the HRP algorithm converges for continuous signals. Second, the 1D HRP algorithm developed in this thesis may be extended to 2D. Conceptually, the extension of HRP to 2D functions is straightforward. The basic idea will be similar to the development in 1D, where the correlation of a function with a coarse scale 2D dictionary function is given as the minimum of the set of correlations with finer scale dictionary elements. The exact form of subfamilies associated with the two-dimensional dictionary elements would have to be determined by the application. HRP in 2D requires the addition of eccentricity and rotation parameters.

Third, a multisignal version of HRP may be developed to determine similarities between sets of functions. For example, if we wish to determine the features which are similar in a set of functions $\{f_l\}$, one method might be to pick the first element as

$$g_{\gamma_0} = \arg \max_{g_\gamma} \min_{l} S(f_l, g_\gamma) \tag{5.1}$$

where $S(f_l, g_\gamma)$ is as defined in (3.3). This formulation yields elements which are present in each function f_l and therefore maybe useful in extracting similar features out of sets of functions. Such an approach may have significant applications in deter-

mining descriptions for classes of objects.

In relation to object recognition, the following future research based on HRP is recommended. First, a higher level of occlusion should be studied using the HRP-based recognition scheme outlined in Chapter 4. Higher levels of occlusion will significantly perturb the centroid of the object which will in turn effect the appearance of the centroidal distance profile. If we can adjust for the error in the centroid, the HRP-based recognition scheme described in Chapter 4 should work quite well even in the presence of higher levels of occlusion. Second, we have based our recognition on four HRP elements, but one can imagine a scheme where the number of elements extracted is allowed to adapt to the profile. The incorporation of additional HRP elements may increase the discriminating power of the algorithm. Finally, it would be interesting to apply 2D HRP features to the recognition of airplanes from their silhouettes or gray-scale images. Since silhouette recognition is inherently a 2D problem, it seems intuitive features based on 2D HRP elements would be a powerful recognition tool.

Bibliography

[1] Amir Amini, Terry Weymouth, and Ramesh Jain. Using dynamic programming for solving variational problems in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(9):855–867, 1990.

[2] Dana Ballard and Christopher Brown. *Computer Vision.* Prentice-Hall, Inc., 1982.

[3] G. N. Bebis and G. M. Papadourakis. Object recognition using invariant object boundary representations and neural network models. *Pattern Recognition*, 25(1), 1992.

[4] S. O. Belkasim, M. Shridhar, and M. Ahmadi. Pattern recognition with moment invariants: A comparative study and new results. *Pattern Recognition*, 24:1117–1138, 1991.

[5] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[6] Shaobing Chen and David Donoho. Atomic decomposition by basis pursuit. Technical report, Statistics Dept., Stanford University, May, 1995. Available via ftp at playfair.stanford.edu.

[7] David Clemens and David Jacobs. Model-group indexing for recognition. In *DARPA Image Understanding Workshop*, pages 604–613, 1989.

[8] R. R. Coifman and M. V. Wickerhauser. Entopy-based algorithms for best-basis selection. *IEEE Trans. Info. Theory*, 38:713–718, 1992.

[9] I. Daubechies. Time-frequency localization operators: a geometric phase space approach. *IEEE Trans. Info. Theory*, 34(4):605–612, 1988.

[10] I. Daubechies. *Ten Lectures on Wavelets.* SIAM, 1992.

[11] D. den Hertog. *Interior Point Approach to Linear, Quadratic and Convex Programming.* Kluwer Academic Publishers, 1994.

[12] Rachid Deriche. Fast algorithms for low-level vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1), 1990.

[13] Susan Dubois and Filson Glanz. An autoregressive model approach to two-dimensional shape classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8:55–66, January 1986.

[14] Richard Duda and Peter Hart. *Pattern Classification and Scene Analysis.* John Wiley and Sons, 1973.

[15] Dan Dudgeon and Richard Lacoss. An overview of automatic target recognition. *Lincoln Laboratory Journal*, 6(1):3–10, 1993.

[16] Gil Ettinger. Hierarchical object recognition using libraries of parameterized model sub-parts. Master's thesis, MIT, 1987.

[17] S-C. Fang and S. Puthenpura. *Linear Optimization and Extensions : Theory and Alogrithms.* Prentice Hall, 1993.

[18] Pascal Fua and Yvan Leclerc. Model driven edge detection. *Machine Vision and Applications*, 3:45–56, 1990.

[19] John Gorman, Robert Mitchell, and Frank Kuhl. Partial shape recognition using dynamic programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(2):257–266, March 1988.

[20] W. Eric L. Grimson. On the recognition of parameterized 2D objects. *International Journal of Computer Vision*, 3:353–372, 1989.

[21] W. Eric L. Grimson. *Object Recognition by Computer: the Role of Geometric Constraints.* MIT Press, 1990.

[22] B. Jawerth and W. Sweldens. An overview of wavelet based multiresolution analysis. *SIAM Review*, 36(3):377–412, September 1994.

[23] H. Kauppinen, Tapio Seppanen, and Matti Pietikainen. An experimental comparison of autoregressive and Fourier-based descriptors in 2d shape classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2), 1995.

[24] C. C. Lin and R. Chellappa. Clasification of partial 2-d shapes using fourier descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(5), 1987.

[25] H.-C. Liu and Mandyam D. Srinath. Partial shape classification using contour matching in distance transformation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(11), November 1990.

[26] S. Mallat and Z. Zhang. Matching pursuit with time-frequency dictionaries. *IEEE Trans. on Signal Processing*, Dec 1993.

[27] D. Marr and E. Hildreth. Theory of edge detectionb. *Proc. Royal Society London*, pages 187–217, 1980.

[28] David Marr. *Vision.* W. H. Freeman and Company, 1982.

[29] Yves Meyer. *Wavelets and Operators.* Cambridge University Press, 1992.

[30] Charles Richard and Hooshang Hemami. Identification of three-dimensional objects using fourier descriptors of the boundary curve. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-4(4), 1974.

[31] Whitman Richards, Benjamin Dawson, and Douglas Whittington. Encoding contour shape by curvature extrema. In Whitman Richards, editor, *Natural Computation.* MIT Press, 1988.

[32] Karen Sarachik. *An Analysis of the Effect of Gaussian Error in Object Recognition.* PhD thesis, MIT, 1994.

[33] Gilbert Strang. Wavelets and dilation equations: A brief introduction. *SIAM Review*, 31(4), December 1989.

[34] Harry Van Trees. *Detection, Estimation and Modulation Theory.* John Wiley and Sons, Inc., 1968.

[35] Andrew Viterbi. Phase-locked loop dynamics in the presence of noise by Fokker-Plank techniques. *Proceedings of the IEEE*, 1963.

[36] M. V. Wickerhauser. Lectures on wavelet packet algorithms. Technical report, Washington University, 1991.

[37] Alan Willsky and Jeff Shapiro. Class notes, 6.432. MIT, Spring 1991.

[38] Zhenyu Wu and Richard Leahy. An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11), 1993.

[39] Z. You and A. K. Jain. Performance evaluation of shape matching via chord length distribution. *Computer Vision, Graphics, and Image Processing*, 28, 1984.

[40] C. T. Zahn and R. Z. Roskies. Fourier descriptors for plane closed curves. *IEEE Transactions on Computers*, C-21(3), 1972.