Hierarchical Wavelet Representations of Ship Radar Returns

by J.S. Baras, S.I. Wolk

T.R. 93-100

ISR
INSTITUTE FOR SYSTEMS RESEARCH

*Sponsored by
the National Science Foundation
Engineering Research Center Program,
the University of Maryland,
Harvard University,
and Industry*

JB 93-11

Entitled:

Hierarchical Wavelet Representations of Ship Radar Returns

Authors:

(with Sheldon Wolk)

Submitted for:

publication to
IEEE Transactions on Signal Processing

March 1993

Hierarchical Wavelet Representations of Ship Radar Returns

John S. Baras* Sheldon I. Wolk†

March 19, 1993

Abstract

In this paper we investigate the problem of efficient representations of large databases of pulsed radar returns in order to economize memory requirements and minimize search time. We use synthetic radar returns from ships as the experimental data. We motivate wavelet multiresolution representations of such returns. We develop a novel algorithm for hierarchically organizing the database, which utilizes a multiresolution wavelet representation working in synergy with a Tree Structured Vector Quantizer (TSVQ), utilized in its clustering mode. The tree structure is induced by the multiresolution decomposition of the pulses. The TSVQ design algorithm is of the "greedy" type. We show by experimental results that the combined algorithm results in data search times that are logarithmic in the number of terminal tree nodes, with negligible performance degradation (as measured by distortion–entropy curves) from the full search vector quantization. Furthermore we show that the combined algorithm provides an efficient indexing scheme (with respect to variations in aspect, elevation and pulsewidth) for radar data which is equivalent to a multiresolution aspect graph or a reduced target model. Promising experimental results are reported using high quality synthetic data.

Key words: High resolution radar returns, multiresolution representations, wavelets, tree structured vector quantization, aspect graph, scale space diagram.

1 Introduction

High resolution ship radar returns contain in their structure substantial information about the target which can be used to better identify complex targets consisting of many

*Department of Electrical Engineering and Institute for Systems Research, University of Maryland at College Park, and AIMS, Inc.. Research supported by AIMS, Inc., 6159 Executive Blvd., Rockville, MD 20852.

†Code 5750, Tactical Electronic Warfare Division, Naval Research Laboratory, Washington, D.C.

scatterers. This applies to many forms of radar signatures, including the amplitude of pulsed radar (PR) returns, the phase of pulsed radar returns, Doppler radars (DR), synthetic aperture radar (SAR) returns, inverse synthetic aperture radar (ISAR) returns, millimeter–wave (MM–wave) radar returns. With the increasing resolution of modern radars it is *at least theoretically* possible to store many of the possible returns (i.e returns organized according to aspect, elevation, pulsewidth etc.) of a complex target and use them in the field for target identification. This is true for naval targets (e.g. ships, submarines, etc.) in particular. The advantage of the increasing radar resolution is the availability of more detailed information, and ultimately of *specific features*, characteristic of the radar return from a specific ship. The disadvantage is that these very detailed characteristics require an ever increasing amount of computer memory to be stored. The latter not only results in unfeasible memory requirements but it also slows down the search time in real field operations. It is therefore important to develop extremely efficient ways to compress the representations of high resolution data returns from real ships, and to design efficient coding schemes which operate in a hierarchical manner on the compressed representations to recover the ship identity. It is our contention that *multi-resolution representations* of the radar data, followed by properly designed hierarchical clustering are key means to achieve both objectives.

Wavelet theory [1]—[9] offers an attractive means for the development of such multiresolution representations. For instance wavelet theory was recently applied to the problem of compressing speech [10], and image signals [11], with very promising results. This success can be roughly explained by the fundamental property of wavelet representations of such signals (i.e. speech and image) to uncover the superposition of these signals in terms of different structures occurring on different time scales at different times (or spatial scales at different locations). Wavelet representations efficiently separate and sort these constituent structures of a complex signal. Wavelet analysis consists of a versatile collection of tools for the analysis and manipulation of signals such as sound and images [2], [8], [10], [11], [12]. Software packages have been developed recently [12], which provide the user with a collection of standard libraries of waveforms, which can be chosen to fit specific classes of signals. These libraries can be further equipped with fast numerical algorithms enabling real-time implementation of a variety of signal processing tasks such as data compression, extraction of parameters for recognition and diagnostics, transformation and manipulation of the underlying signals. Musical signals offer the best understood example. Indeed a musical note can be described by four basic parameters, intensity (or amplitude), frequency, time duration, time position. Wavelet packets are indexed by the same parameters, plus others corresponding to choice of library (we can think of a library as a musical instrument, i.e. the recipe used to generate all the waveforms, "notes", in the library). Sound signal analysis compares a sound with all elements of a given library, picks up large correlations (or notes which are closest to segments of the signal) and uses these to rebuild the signal with a minimal number of waveforms. The result provides an economical transcription, which if ordered by decreasing intensity sorts the main features in order of importance.

In this paper we describe our recent efforts to extend these results to cover representations of high resolution radar returns from ships. Although we anticipate that such extensions are possible for all types of radar returns described at the beginning of this section, we focus our efforts on high range-resolution radar returns. This is accomplished by developing hierarchical clustering schemes using wavelet representations which are motivated from the physics of radar scattering, in the same way that wavelet analysis of sound is motivated by the physics of sound and speech signal generation. Indeed, the operation and mathematics of pulsed radar fit nicely with the mathematics of wavelets as we shall demonstrate here. In addition we use sophisticated versions of Vector Quantization (VQ) [13] to further compress and cluster the wavelet representations of the radar signals, in a way that permits hierarchical search across resolutions and a progressive scheme for identification of the ship (target).

2 High Range–Resolution Radar Returns

High range-resolution radar returns can be described as complex valued signals of finite duration. The time duration of each return is controlled by a range gate which is assumed here to span the range extent of a distributed target such as a ship. We can represent these complex signals as two dimensional real-valued signals using two commonly used representations: (i) amplitude–phase representation $(A(r), \Phi(r))$, (ii) in-phase and quadrature components representation $(I(r), Q(r))$, where r represents the time delay (or range) variable within the pulse. For a complete characterization of a complex target one can store the whole set of these two dimensional functions (pulses) for all possible values of radar pulsewidth δ (i.e. different resolution), aspect α and elevation ϵ. Even if one quantizes the three–dimensional space of δ, α, ϵ the required storage is enormous and impractical for real applications. Our efforts to date have concentrated on amplitude representations only (that is one–dimensional signals). We shall treat the full two dimensional problem (e.g. amplitude–phase representations) elsewhere.

Given the amplitude of a high–range resolution radar return, several characteristics of the scatterer distribution of the target can be revealed. For instance in Figure 1 we show a typical pulse return from a synthetic radar return model from complex naval targets. Here we can see several prominent local maxima of the amplitude return, which correspond to dominant scatterers, or groups of small coherently additive dominant scatterers. Typical examples of dominant scatterers include flat plates, trihedrals or dihedrals. Since in a typical ship we can have thousands of scatterers, understanding dominant scatterers and reduction of detailed scattering models plays a significant role in our studies. We have substantial past experience in such issues as we have been studying modeling of scattering returns from complex targets for the last 15 years at the Tactical Electronic Warfare Division of the Naval Research Laboratory. We have developed, for instance, reduced scatterer ship models based on the physical principles of high resolution radar scattering.

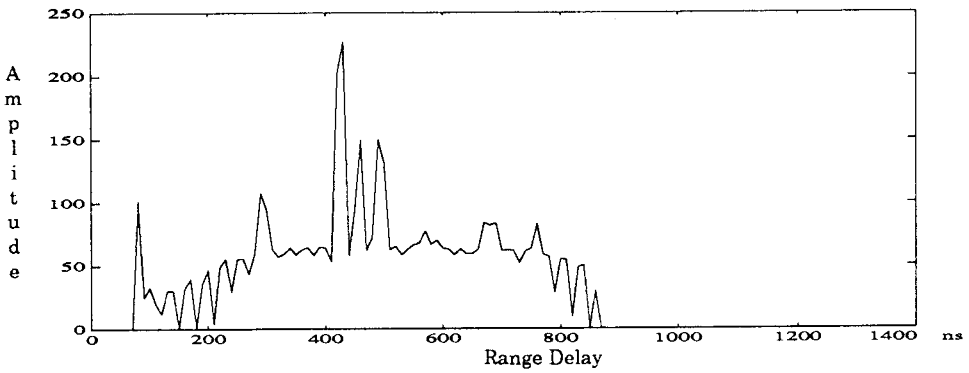

Figure 1: Typical amplitude of a ship radar pulse return

Varying the radar pulsewidth δ changes the resolution of the returned pulse, in the sense that more (narrow pulse) or less (wide pulse) details can be distinguished. Varying the view-point (i.e. the aspect, elevation (α, ϵ) pair) changes the shape of the returned pulse, because dominant scatterers have typically highly directive returns (in space). Therefore, understanding the view-point invariance of dominant scatterers also plays a fundamental role in our studies. A related point has to do with the variation of the radar return due to relative motion between the target (ship) and the radar. Since the target return is in reality the coherent sum of the returns from thousands of scatterers, relative motion (random or deterministic) introduces large variations in the pulse-to-pulse amplitude returns. Successful methods to provide effective compression of radar returns must address the substantial variability of the returns as well. It is clear that successful treatment of this variability is directly related to the view-point invariance of features in the return.

The fact that the radar return of a complex target is a rapidly fluctuating function of viewpoint should not be construed as an impenetrable barrier neither to a model data reduction nor to an efficient and hierarchical organization of a ship radar return database. It is well known that this behavior of the return from complex targets is principally due to the rapid phase variations that occur as scattering returns from individual scatterers combine. Our earlier studies on naval targets have demonstrated that the radar return from ships is primarily due to a small number of scatterers which have relatively broader (in aspect and elevation) reflectivity distributions, than the small scale rapid oscillations that are created by the rapid phase variations. These rapid phase variations are partly due to the inability of a radar sensor to determine accurately and in a stable manner phase centers for extended scatterers, and partly due to the ever existing relative motion (even small) between the radar sensor and the target scatterers. The underlying broader returns from a small set of significant (or persistent) scatterers manifests the significance of recovering this "slower variation" of the radar return as function of viewpoint and using it for recognition. The latter will describe the more salient features of the target signature.

As a consequence, some sort of averaging (or clustering) is necessary in representing the more meaningful, slower variation of the radar return (or the RCS) as aspect and elevation are changing. There are several ways that one can implement this smoothing, some are based on physical and some on mathematical arguments. For example, *stationary phase approximation* [15] can be utilized to smooth out these rapid variations, by essentially aggregating the returns from several scatterers. Other methods employed in modeling and data representation for the RCS of complex targets [15], describe the variation of some (empirical) statistic of the RCS fluctuations as a function of aspect (or more generally of view point as defined above). The preferred (due to its robustness) statistic is the median of the RCS computed over a small rectangle (in aspect and elevation coordinates) around the viewpoint [15]. Similarly, one can use the average or the median of the return pulse over a small neighborhood of the viewpoint. Based on these considerations it is physically meaningful to cluster the radar returns from various viewpoints into equivalence classes using a measure of similarity. The resulting quantization of the signal space (i.e. of the radar returns) characterizes the limits of discriminating between returns from different targets using information about the viewpoint; in essence if we insist on extremely fine quantization cells we are modeling the radar sensor noise and not the underlying complex target.

Such (aggregate or average) representations should be interpreted as a reduction in the number of scatterers "visible" from that particular viewpoint. Indeed, since we can identify scatterers with peaks in the amplitude of the RCS, any smoothing of the RCS fluctuations can be interpreted as a reduction (or aggregation) in the number of scatterers. The RCS of complex targets can be computed to a good approximation by considering a smaller set of *dominant scatterers*. Our earlier studies at the Naval Research Laboratory (see also [15]) have demonstrated that we can reduce the number of scatterers needed for the accurate simulation of radar returns from ships by a factor of a 100 (e.g. from 65,000 to 800, or from 250,000 to 2,000). These dominant scatterers are not always identifiable with concrete geometric structures on the ship. They can represent the aggregate return due to many smaller scatterers as well. The dominant scatterers are "visible" from wider viewpoint regions than geometric scattering will dictate. One can think of such models as synthetic aggregate models. However, the detailed characteristics of each ship were preserved and extensive experimentations have supported and validated these ideas. These so called "reduced profiles" can then be grouped into equivalence classes using appropriate distance measures. Mathematically, this is the same as performing some form of "clustering" or "centroid averaging" on these profiles of the returned radar pulses.

As we shall show in this paper, this approach of clustering the radar returns, can be performed in an efficient and physically meaningful manner. It is important, however, to combine this approach with a multiresolution representation of the radar pulse returns. Multiresolution representations of a ship radar return can of course be obtained by illuminating the ship with variable pulsewidth radar signals. In addition to providing information on the size and combination characteristics of different scatterers, such multiresolution representations of the return offer a natural aggregation scheme that can help in the clustering process just described. What is emerging from this combination is a powerful combination of multiresolution representations with hierarchical clustering, which permits efficient storage and speedy recovery of the information. In this approach *the clustering hierarchy is induced by the multiresolution representation.* This combination is the basis of the methodology we have developed.

3 Multiresolution Representations of Radar Returns

The data representing the amplitude radar returns of a complex target are the set $\{A(r), r \in [0,\ T];\ (\delta, \alpha, \epsilon) \in \mathcal{Q}_{\ni}\}$, where T is the length of the range gate and $\mathcal{Q}_{\ni}$ is a discrete, finite grid in $[\delta_{min}, \delta_{max}] \times [0, 2\pi) \times [0, \pi/2)$. Current representations of these sets employ statistical or *ad hoc* methods to reduce the amount of memory required for storing this set. However such methods as discrete cosine transform, subband decomposition or Karhunen-Loeve transform, cannot achieve a reduction of the total amount of bits needed of more than 10:1; and such compression rates are not enough for the efficient storage of high resolution radar returns. In addition we need to organize the data so as to achieve fast search times when using the database for classification or discrimination purposes.

Higher compression rates can only be achieved by developing coding techniques that are adapted to the information content of the signals, and to their physical nature. This requires the organization of the radar return data in such a way as to represent and separate the important features of the return. For the amplitude of ship radar pulse returns, this means the location of the local maxima and their geometrical characteristics. In particular we need ways to describe how these local maxima coalesce as the radar pulsewidth δ increases. This corresponds to changes in the return pulse as the resolution varies and corresponds to the physical property of coherently combining the returns from dominant scatterers as a function of range extent. Therefore we need coding schemes that efficiently preserve the most important discontinuities of the pulse in a multi-resolution scheme. These discontinuities, or more accurately the high curvature points of the return pulse, correspond to dominant scatterers (or groups of scatterers). The representation must be such that at coarse resolution (i.e. large δ) the local maxima of the return correspond to the larger structures of the ship, while at finer resolutions (i.e. small δ) the local maxima of the return correspond to the finer structures of the ship. For one-dimensional signals, such as the amplitude of radar pulse returns, these maxima and minima are really (trivial) edges. The significance and completeness of multi-edge representations of signals is well known in various contexts. In images the completeness of such multi-scale representations using edges has been conjectured by [21]. In addition we need to further compress these representations by taking into account the variability of these scatterer combinations with the view-point (i.e. the aspect and elevation angles).

Experiments with variable pulsewidths, in order to obtain a multiresolution representation of the ship is not a very practical solution. One of the objectives of the work reported here is to show, by experiments with synthetic data, that our methodology provides a substitute which involves only processing the high resolution radar returns. This result in itself is an important finding.

The radar return from a complex target, like a ship, in high range-resolution, can be represented as [25]

$$r = H_{IF} \star h_s \star p \tag{3.1}$$

where r is the received complex pulse, h_s the ship complex impulse response (representing the reflectivity of the ship), H_{IF} is the Intermediate Frequency (IF) impulse response, p is the complex envelope of the transmit pulse. Typically the radar will employ a matched filter in the receiver, and we are interested in the variability of pulsewidth from a minimum of 10 ns. At a bow-on aspect the ship is approximately 300 ft long, resulting in a range extended pulse return of approximately 600 ns plus a pulsewidth. To capture safely all ship pulses we used a range gate of 128 bins corresponding to a returned signal time duration of 1280 ns. At the finer resolution of 10 ns, and sampling at the corresponding rate produces 2^7 samples.

The NRL Code 5750 digital simulation model is a flexible tool for experimentation, and it has been used as the basic data generation source for the studies reported here. This model has been validated against field returns and provides high accuracy simulations. The digitally simulated ship model consists of over 400 scatterers of a variety of types, including flat plates, point scatterers and dihedrals. These scatterers are distributed in both range and space in accordance with their actual locations on a ship. To each of these scatterers we associate an amplitude and a phase due to its physical configuration, and an amplitude and a phase due to its location in space relative to the radar. As a result the ship impulse response h_s can be represented as

$$h_s(t) = \sum_i A_i e^{j\phi_i} e^{j\pi f \Delta_i}(1 + \Gamma_i)\delta(t - \Delta_i) \tag{3.2}$$

where A_i is the amplitude of the return from the i^{th} scatterer, ϕ_i is the phase of the return from the i^{th} scatterer, f is the carrier frequency, Δ_i is the distance of the i^{th} scatterer from a reference point; we have chosen t to represent range delay as usual in (3.2). The complex multipath factor Γ_i accounts for the signal return path that is reflected off the water's surface. Thus multipath reflections are also included in (3.2).

This representation is an idealization. In reality scatterers move as the ship moves in response to sea waves. This relative motion of the scatterers affects the signal return due to a number of phenomena, including: (1) phase changes due to changes in geometry, (2) amplitude and phase changes due to changes in the multipath contributions, and (3) amplitude and phase changes due to blockages. Since the ship motion is stochastically driven, and the multipath reflections are stochastic, the impulse response (3.2) is itself stochastic. The common model used for the general impulse response of a distributed target is that of a complex Gaussian random process [15]. The latter model can be useful in further analytical investigations of the results presented here. However the digital simulation model used accurately captures this randomness with the careful selection of the scatterers, their location, type and size.

We also maintain pulse–to–pulse independence, by selecting the radar pulse repetition interval (PRI) to be long enough with respect to the correlation time of an individual scatterer. Since we are interested in characterizations on a per pulse basis we designed the experiments so as to ensure that pulses are well spread with respect to a ship roll period. In our experiments we have used PRIs with values in the interval 80 ms to 1s.

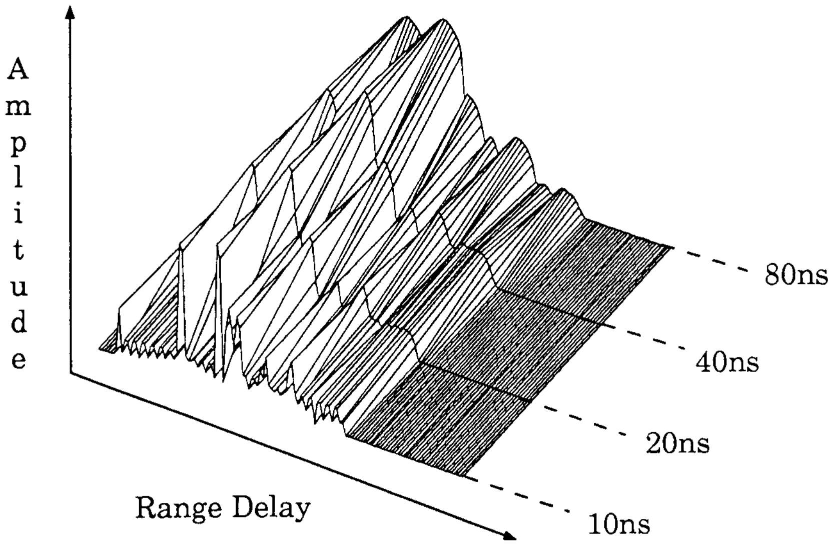

Figure 2: Illustrating typical variation of ship radar return as a function of pulsewidth.

In Figure 2 we show a typical result of a ship return with transmit pulse widths of 10 ns, 20 ns, 40 ns, 80 ns. In the three–dimensional representation shown, we see clearly the coalescence of the ship scatterers as we move from fine to coarse pulsewidths. This is also demonstrated in Figure 4, showing the pulse returns corresponding to the four different pulsewidths. In Figure 4 we also have traced, by connecting the major peaks of each return, the coalescence of the ship scatterers as we move from fine to coarse pulsewidths. We have performed many simulation experiments with the same typical and generic results, as those represented in Figures 2 and 4.

In Figure 3 we have extracted the location of the maxima of the ship radar return as a function of pulsewidth, or equivalently resolution. We see that by varying the pulsewidth we do not quite get uniform localization of the major scattering centers with respect to resolution. Uniform localization corresponds to vertical lines. We call diagrams such as the one depicted in Figure 3 *scale space diagrams*, extending a notion originally introduced by Witkin in computer vision [23]. Our experiments to date regarding uniform localization of radar features are not complete and this issue will be revisited elsewhere,

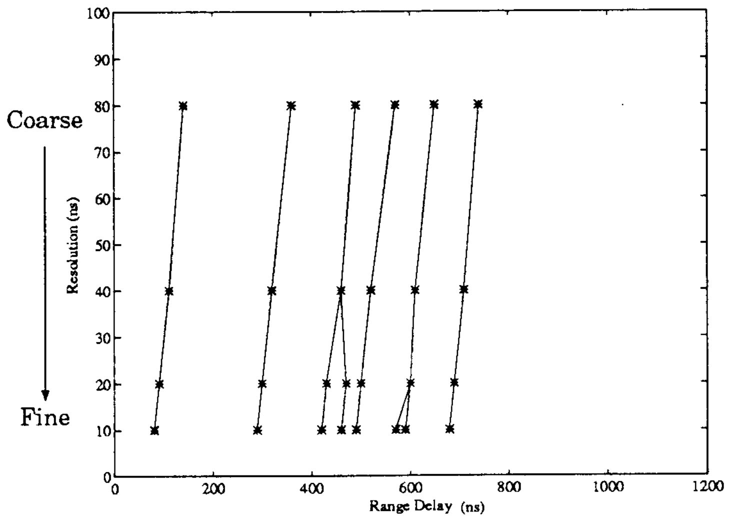

Figure 3: Illustrating a typical scale space diagram for radar pulse returns from a ship

utilizing a substantially larger and statistical experimental database. On the other hand, varying the pulsewidth of the transmitted radar pulse, changes the way individual scatterers combine coherently. Since we are recording the amplitude of the return, the whole operation is in addition quite nonlinear. The overall combination and aggregation of individual scatterers' returns is a very complex operation and we do not have any strong physical basis to expect uniform localization of scattering centers with respect to resolution variations induced by different pulsewidths. However, it is clear that such scale space diagrams provide a "fingerprint" of the ship, since they indicate how scatterers combine; a property of the ship geometry when all other scattering parameters are kept constant.

As mentioned earlier one of our objectives is to show that by combining wavelet representations with clustering algorithms we obtain similar representations of the ship radar return; recapturing in essence the effects of variable pulsewidth data as described here.

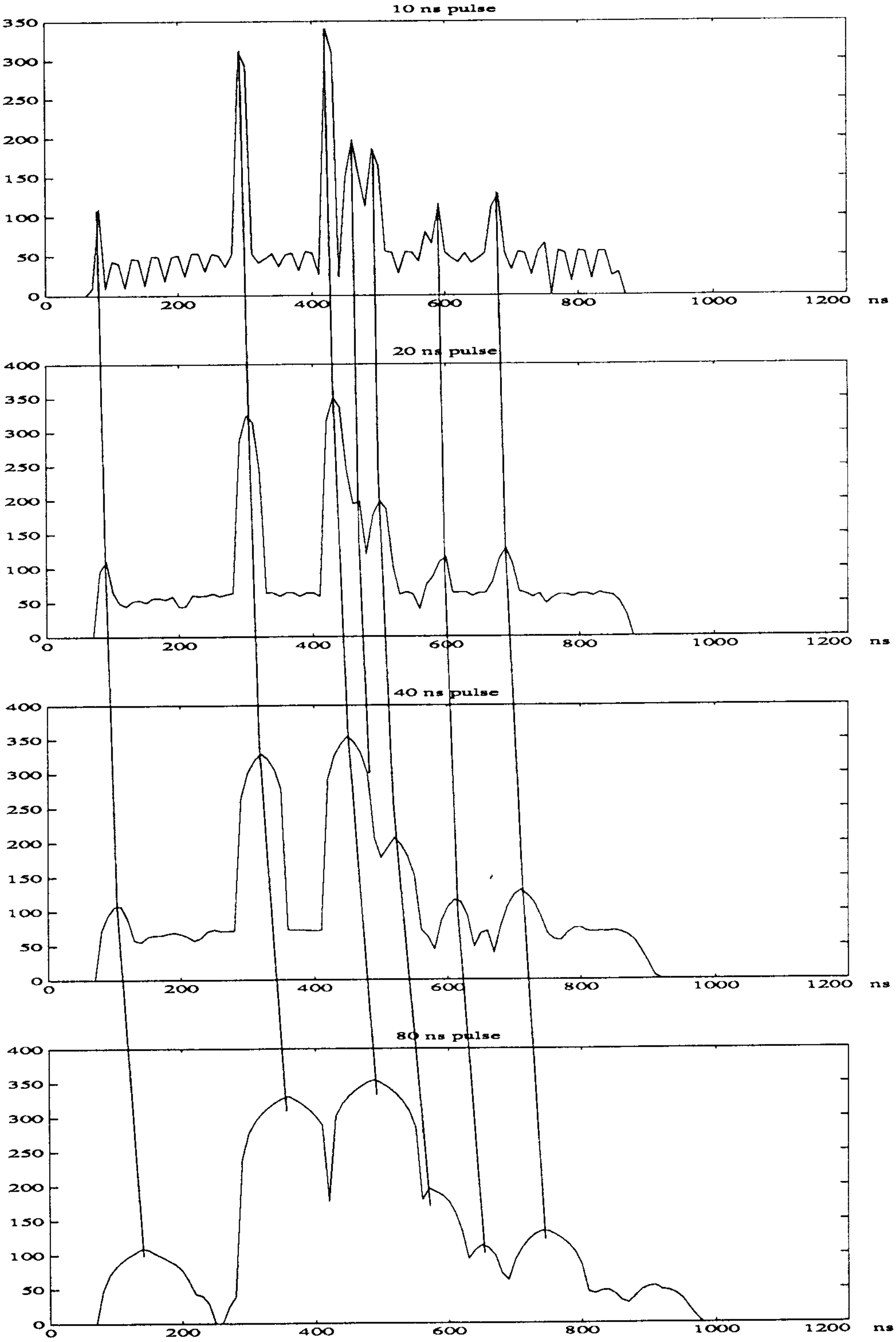

Figure 4: Individual ship pulse returns from the same ship model for different pulsewidths.

4 Wavelet Representation of Radar Returns

Wavelet and wavelet packets and the associated representations have recently attracted great interest as compact and efficient representations of signals in many scales [1]-[9]. Wavelets are extensions of the more well-known Gabor transforms and the resulting Gabor pyramid representations of images [4]. Like the Fourier transform, wavelets decompose functions into coefficients assigned to fundamental building blocks, from which the original function can be reconstructed. But whereas Fourier analysis builds everything out of sines and cosines, wavelet theory relies on translations and dilations of a suitably chosen "mother wavelet". Most often the mother wavelet is a well-localized "blip". The rest of the building blocks are formed by translating the mother wavelet by unit steps and by contracting or dilating by factors of 2. This gives a two-parameter transform: one for the distance translated and one for the power of 2. This approach means that wavelet theory can "zoom in" on details of a function in a way that Fourier analysis cannot. Because the sine and cosine functions extend all along the real line, Fourier coefficients contain only "global," not local, information about a function. In particular, transient behavior is difficult to see in the Fourier transform. But wavelets are custom-made for analysis of this kind; a blip in the function shows up as a blip in the wavelet transform. Thus wavelets are more akin to local representations, albeit simultaneously at different scales. Consequently they are more robust to noise because minor disruptions (noise) remain localized, which makes them easier to edit (filter) out. These properties justify the underlying hypothesis of this study: wavelet representations are particularly suited for radar returns. We shall give some additional reasons later in this section.

Since we are primarily interested in one-dimensional signals (i.e. amplitude representations of time pulses) we describe the basic facts for one-dimensional signals here.

4.1 Wavelet representations

We follow [9] in this brief summary on wavelet fundamentals. Wavelets are functions generated from one single function ψ by dilations and translations,

$$\psi^{a,b}(t) = |a|^{-1/2}\psi(\frac{t-b}{a}) \tag{4.1}$$

The *mother wavelet* ψ has to satisfy (roughly) $\int dx\psi(x) = 0$, which implies at least some oscillations. Thus high frequency wavelets correspond to $a < 1$ or narrow width, while low frequency wavelets have $a > 1$ or wider width. Wavelet transforms [1, 2] represent any arbitrary function f as a superposition of elementary blocks. Any such superposition decomposes f into different scale levels, where each level is then further decomposed with a resolution adapted to the level. One way to achieve such a decomposition is to write f as an integral over a and b of $\psi^{a,b}$ with appropriate weighting coefficients [1, 9]. For digital computations we have to write f as a discrete superposition by introducing the discretization $a = a_0^m$, $b = nb_0a_0^m$, with $m, n \in Z$, and $a_0 > 1$, $b_0 > 0$ fixed. The wavelet decomposition is then

$$f = \sum c_{m,n}(f)\psi_{m,n} \tag{4.2}$$

with $\psi_{m,n}(t) = \psi^{a_0^m, nb_0a_0^m}(t) = a_0^{-m/2}\psi(a_0^{-m}t - nb_0)$. For $a_0 = 2, b_0 = 1$, which is the most widely used selection, there exist very special choices of ψ such that the $\psi_{m,n}$ constitute an orthonormal basis, so that

$$c_{m,n} =< \psi_{m,n}, f >= \int \psi_{m,n}(x)f(x)dx \tag{4.3}$$

in this case. Such bases were constructed by Daubechies [1] and Meyer [3]. These concepts were synthesized to a multiresolution analysis tool for signals by Mallat [2], which gives rise to fast computation algorithms.

In such a multiresolution analysis [2] one has *two functions: the mother wavelet* ψ *and a scaling function* ϕ. One also introduces dilated and translated versions of the scaling function, $\phi_{m,n}(x) = 2^{-m/2}\phi(2^{-m}x - n)$. For fixed m, the $\phi_{m,n}$ are orthonormal. Usually one denotes by V_m the space spanned by the $\phi_{m,n}$. The spaces V_m describe successive approximation spaces, $\ldots V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \ldots$, each with resolution 2^m. This sequence of successive approximation spaces V_m constitutes a *multiresolution analysis* [2, 9]. The multiresolution aspect is a consequence of the requirement: $f \in V_m \iff f(2^m\cdot) \in V_0$. For each m, the $\psi_{m,n}$ span a space W_m which is exactly the orthogonal complement in V_{m-1} of V_m; therefore the coefficients $< \psi_{m,n}, f >$ describe the information lost when going from an approximation of f with resolution 2^{m-1} to the coarser approximation with resolution 2^m. The spaces W_m inherit the scaling property from the V_m: $f \in W_m \iff f(2^m\cdot) \in W_0$. The scaling function ϕ, which is in V_0 can be expanded in V_{-1} resulting in the dilation equation [26]

$$\phi = \sum_n h_n\phi_{-1,n} \tag{4.4}$$

while the mother wavelet is given from the scaling function via

$$\psi = \sum_n g_n\phi_{-1,n} \tag{4.5}$$

where

$$\begin{aligned} h_n &=< \phi_{-1,n}, \phi >= 2^{1/2}\int \phi(2x-n)\phi(x)\,dx \\ g_n &=< \phi_{-1,n}, \psi >= (-1)^n h_{-n+1} \end{aligned} \tag{4.6}$$

The sequences $\{h_n,\ n\ integer\}$ and $\{g_n,\ n\ integer\}$ can be thought of as the impulse responses of two filters, known as *conjugate mirror filters*. Based on these results a useful and constructive starting point for multiresolution analysis is the choice of a scaling function ϕ using (4.4) [9, pp. 140–145].

These concepts result into the following fast algorithm for the computation of the $c_{m,n}(f)$ [2]:

$$\begin{aligned} c_{m,n}(f) &= \textstyle\sum_k g_{k-2n} a_{m-1,k}(f) \\ a_{m,n}(f) &= \textstyle\sum_k h_{k-2n} a_{m-1,k}(f) \end{aligned} \tag{4.7}$$

where $a_{m-1,n}(f) =< \phi_{m-1,n}, f >$ are coefficients characterizing the projection of f onto V_m. The first of (4.7), which is a consequence of (4.5, shows that we can compute the $c_{m,n}$ by means of the same operation (convolution with g, decimation by factor of 2) from the $a_{m,n}$, if the latter were known. On the other hand, the second of (4.7), which is a consequence of the dilation equation (4.4), shows that the $a_{m,n}$ can indeed be computed by means of the same operation (convolution with h, decimation by factor of 2) from the $c_{m,n}$. The procedure is now clear: starting from $< \phi_{0,n}, f >$, we compute the $< \psi_{1,n}, f >$ by the first of (4.7), and the $< \phi_{1,n}$ from the second of (4.7. We then apply (4.7) again to compute the $< \psi_{2,n}, f >$, $< \phi_{2,n}, f >$ from the $< \phi_{1,n}, f >$, etc.. This fast recursive algorithm computes at each level m not only the wavelet coefficients $c_{m,n}$, but also the $a_{m,n}$, which are useful for the computation of the next level wavelet coefficients. The whole process can also be viewed as the computation of successively coarser approximations of f, together with the "difference in information" between every two successive levels. Thus we start with a fine scale approximation of f, $S^0 f$, which is the projection of f onto V_0. Then since $V_0 = V_1 \oplus W_1$, we write $S^0 f = S^1 f + W^1 f$, where $S^1 f$ as the projection of $S^0 f$ (and therefore of f) onto V_1 is the next coarser approximation of f, while $W^1 f$ as the projection of $S^0 f$ on W_1 represents the "lost information" in the transition $S^0 f \rightarrow S^1 f$. This picture repeats for each scale m. In each of these spaces V_m, W_m we have the orthonormal bases $(\phi_{m,n})$, $(\psi_{m,n})$, *n integer* respectively, and therefore

$$S^0 f = \sum_n a_{0,n}\phi_{0,n}, \quad S^1 f = \sum_n a_{1,n}\phi_{1,n}, \quad W^1 f = \sum_n c_{m,n}\psi_{1,n}. \tag{4.8}$$

Formulae (4.7) give the evolution of the coefficients of the orthogonal bases transformations from V_m to V_{m+1}. If we let L_h and L_g represent the linear transformations corresponding to the first and second of (4.7), these computations are represented schematically in the diagram of Figure 5, where $\mathbf{c}^{\mathbf{m}}$ is the vector of coefficients $\{c_{m,n},\ n\ integer\}$ and $\mathbf{a}^{\mathbf{m}}$ is the vector of coefficients $\{a_{m,n},\ n\ integer\}$.

In practice, we stop after a finite number of levels $1, 2, \cdots, J$. At this stage we have replaced the information at the fine scale $\mathbf{a}^{\mathbf{0}}$ with the information in $\mathbf{c}^{\mathbf{1}}, \mathbf{c}^{\mathbf{2}}, \cdots, \mathbf{c}^{\mathbf{J}}$ and a final coarse scale coefficient vector $\mathbf{a}^{\mathbf{J}}$. Equivalently we have replaced $S^0 f$ with $W^1 f, W^2 f, \cdots, W^J f$ and $S^J f$. Because (4.7) are associated with orthonormal wavelet bases, the inverse transformation is given by the adjoint matrices. Indeed

$$S^{m-1} f = S^m f + W^m f = \sum_n a_{m,n}\phi_{m,n} \; + \; \sum_n c_{m,n}\psi_{m,n} \tag{4.9}$$

which results in view of (4.4, 4.5) to

$$\begin{tikzcd}
\mathbf{a}^0 \arrow[r, "L_h"] \arrow[dr, "L_g"] & \mathbf{a}^1 \arrow[r, "L_h"] \arrow[dr, "L_g"] & \mathbf{a}^2 \arrow[r, "L_h"] \arrow[dr, "L_g"] & \mathbf{a}^3 \arrow[r, dashed] & {} \\
 & \mathbf{c}^1 & \mathbf{c}^2 & \mathbf{c}^3 &
\end{tikzcd}$$

Figure 5: Schematic representation of the multiresolution algorithm, analysis part.

$$a_{m-1,n}(f) = \sum_k [h_{n-2k} a_{m,k}(f) + g_{n-2k} c_{m,k}(f)]. \tag{4.10}$$

If the function f is given in sampled form, then one can take these samples for the fine resolution approximation coefficients $a_{0,n}$, and (4.7, 4.10) describe the analysis and synthesis steps of a *subband filtering scheme* with exact reconstruction, on these sampled values, with low-pass filter h and high-pass filter g. In such a scheme, in the analysis step, the incoming sequence $\mathbf{a^0}$ is convolved with two different filters, one low-pass and one high-pass. The two resulting sequences are then subsampled, i.e. only the even (or only the odd) entries are retained, since the other samples are redundant. In the synthesis step, we first interleave with zeros the sequences resulting from the analysis part, and then convolve the resulting "upsampled" sequences with the filters h and g respectively and adding the results, which gives (4.10). Within a time reversal the same filters are used for analysis and synthesis, thus the commonly used term *quadrature mirror filters*.

Most of the orthonormal wavelet bases have infinitely supported ψ, corresponding to filters h, g with infinitely many taps. The construction of Daubechies in [1] gives ψ with finite support, which corresponds to FIR filters h, g. The latter are preferred for applications and implementations. It then follows that such orthonormal bases [1] correspond to a subband filtering scheme with exact reconstruction property, using the same FIR filters for reconstruction as for decomposition. Subband filtering schemes with exact reconstruction have been studied extensively [17]- [20].

The easiest way to ensure compact support for the wavelet ψ is to chose the scaling function ϕ with compact support. It then follows from (4.6 that h will be FIR, and so will be g. Then ψ has compact support since from 4.5 it is the finite linear combination of compactly supported functions. For compactly supported ϕ, the scaled filter transfer function ($i = \sqrt{-1}$)

$$H(\xi) = 2^{-1/2} \sum_n h_n e^{-in\xi} \tag{4.11}$$

which is 2π-periodic, becomes a trigonometric polynomial. Then orthonormality of the $\phi_{0,n}$ implies

$$|H(\xi)|^2 + |H(\xi + \pi)|^2 = 1. \tag{4.12}$$

One is interested in constructing H so that ϕ and ψ are reasonably regular (smooth). The latter requirement implies that H must be of the form

$$H(\xi) = \left(\frac{1+e^{i\xi}}{2}\right)^N \mathcal{L}(\xi), \tag{4.13}$$

where $N \geq 1$ and $\mathcal{L}$ is a trigonometric polynomial.There exist many ways to construct such H [9, chapters 6,7]. The only possible candidate for a scaling function is provided from

$$\hat{\phi}(\xi) = (2\pi)^{-1/2} \prod_{k=1}^{\infty} H(2^{-k}\xi). \tag{4.14}$$

which follows from $\hat{\phi}(\xi) = H(\xi/2)\hat{\phi}(\xi/2)$ which in turn is (4.4) in the Fourier domain. In addition to (4.12), (4.13), H must satisfy $H(0) = 1$. Necessary and sufficient conditions for H to satisfy the above conditions and generate orthonormal wavelet bases are given in [9, Theorem 6.3.6].

One is also interested in the construction of compactly supported wavelets which have L vanishing moments; i.e. $\int x^l\psi(x)\, dx,\ l = 0, 2, \cdots, L-1$. These are important because an orthonormal basis of wavelets can consist of C^{L-1} wavelets only if the mother wavelet ψ has N vanishing moments. They allow efficient approximation by a small number of wavelet coefficients in (4.2). The wavelets ${}_N\psi$ constructed in [9, section 6.4] (where N is as in equation (4.13)), have corresponding ${}_NH$ with $2N-1$ non–vanishing coefficients, have support width $2N-1$, have the highest number of vanishing moments compatible with their support width, and belong (as do the corresponding scaling functions ${}_N\phi$ to $C^{\mu N}$, where $\mu \approx .2$. If we base the construction of wavelets with vanishing moments to the dilation equation (4.4), the condition of L vanishing moments is equivalent with H, as given by (4.11), having a zero of order L at $\xi = \pi$. This is the same as

$$\sum_n (-1)^n n^m h_n = 0,\ m = 0, 1, \cdots, L-1. \tag{4.15}$$

We are interested in investigating such exact reconstruction schemes for high resolution ship radar returns. In particular we are interested in discovering mother wavelets and scaling functions which are physically motivated; e.g. lead to representations where discontinuities (corresponding to identifiable scatterers) are uniformly localized across resolution (pulsewidth).

Using biorthogonal bases (i.e. by relaxing the orthonormality requirement) it is possible to construct examples where the mother wavelet has arbitrarily high regularity. In addition biorthogonal bases permit the construction of symmetric wavelets, which hold promise for the uniform localization of the local maxima of one dimensional pulses across different resolutions [19]. In such schemes, one decomposes f as before (i.e. using (4.7) , but reconstruction becomes (that is (4.10) is replaced by)

$$a_{m-1,n}(f) = \sum_k [\tilde{h}_{n-2k} a_{m,k}(f) + \tilde{g}_{n-2k} c_{m,k}(f)] \tag{4.16}$$

where the filters $\tilde{h}$, $\tilde{g}$ are different from h, g. In order to have exact reconstruction we impose the conditions:

$$\begin{aligned} \tilde{g}_n &= (-1)^n h_{-n+1} \\ g_n &= (-1)^n \tilde{h}_{-n+1} \\ \textstyle\sum_n h_n \tilde{h}_{n+2k} &= \delta_{k,0} \end{aligned} \tag{4.17}$$

Here we have two pairs of filters and therefore two pairs of scaling functions and mother wavelets, ϕ, ψ and $\tilde{\phi}, \tilde{\psi}$. The scaled filter transfer functions are given again by

$$\tilde{H}(\xi) = 2^{-1/2} \sum_n \tilde{h}_n e^{-in\xi}, \qquad H(\xi) = 2^{-1/2} \sum_n h_n e^{-in\xi}. \tag{4.18}$$

and must satisfy $H(0) = 1 = \tilde{H}(0)$. This amounts to normalizing the coefficients so that $\sum_n h_n = \sqrt{2} = \sum_n \tilde{h}_n$. The Fourier transforms of ϕ and $\tilde{\phi}$ are given by

$$\begin{aligned} \hat{\phi}(\xi) &= (2\pi)^{-1/2} \textstyle\prod_{k=1}^{\infty} H(2^{-k}\xi) \\ \hat{\tilde{\phi}}(\xi) &= (2\pi)^{-1/2} \textstyle\prod_{k=1}^{\infty} \tilde{H}(2^{-k}\xi). \end{aligned} \tag{4.19}$$

which are well defined.

Then the following interpretation is valid for (4.16). The functions ϕ and $\tilde{\phi}$ satisfy the dilation equations

$$\phi(x) = \sum_n h_n \phi(2x-n) \ \ and \ \ \tilde{\phi}(x) = \sum_n \tilde{h}_n \tilde{\phi}(2x-n) \tag{4.20}$$

It is easy to see that ϕ and $\tilde{\phi}$ are well-defined square integrable functions, compactly supported if the filters h and $\tilde{h}$ are FIR. Define also

$$\psi(x) = \sum_n g_n \phi(2x-n) \ \ and \ \ \tilde{\psi}(x) = \sum_n \tilde{g}_n \tilde{\phi}(2x-n) \tag{4.21}$$

Then the above subband filtering scheme, with different filter pairs in the analysis and the synthesis steps, with exact reconstruction, corresponds to *two* dual wavelet bases, associated with two different multiresolution ladders

$$\begin{aligned} &\ldots V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \ldots \\ &\ldots \tilde{V}_2 \subset \tilde{V}_1 \subset \tilde{V}_0 \subset \tilde{V}_{-1} \subset \tilde{V}_{-2} \ldots \end{aligned} \tag{4.22}$$

The wavelets $\{\psi_{m,n},\ n\ integer\}$ and $\{\tilde{\psi}_{m,n},\ n\ integer\}$ span subspaces W_m, $\tilde{W}_m$ respectively, which are complements to V_m, $\tilde{V}_m$ but not orthogonal complements. The resulting filter bank structure with the associated filters is depicted in Figure 6.

Again one can construct compactly supported mother wavelets ψ and $\tilde{\psi}$ with varying degree of regularity or vanishing moments [9, chapter 8], [19]. Biorthogonal bases of wavelets with regularity have been recently constructed in [9, 19, 20]. It was shown in [19] that arbitrary high regularity can be achieved for both ψ and $\tilde{\psi}$, provided one chooses

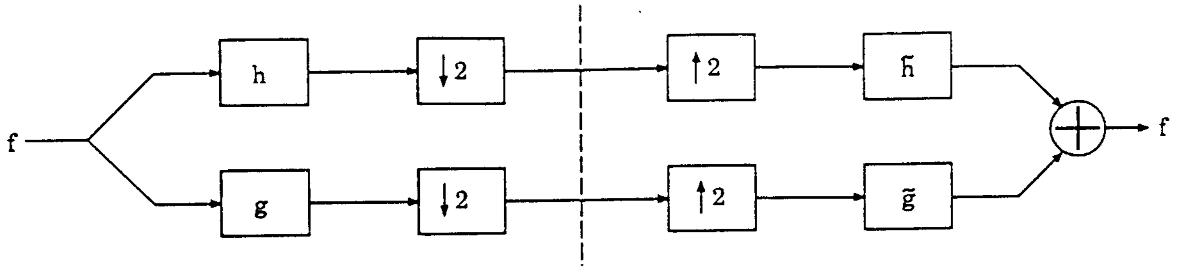

Figure 6: Subband filter bank structure and the associated filter pairs

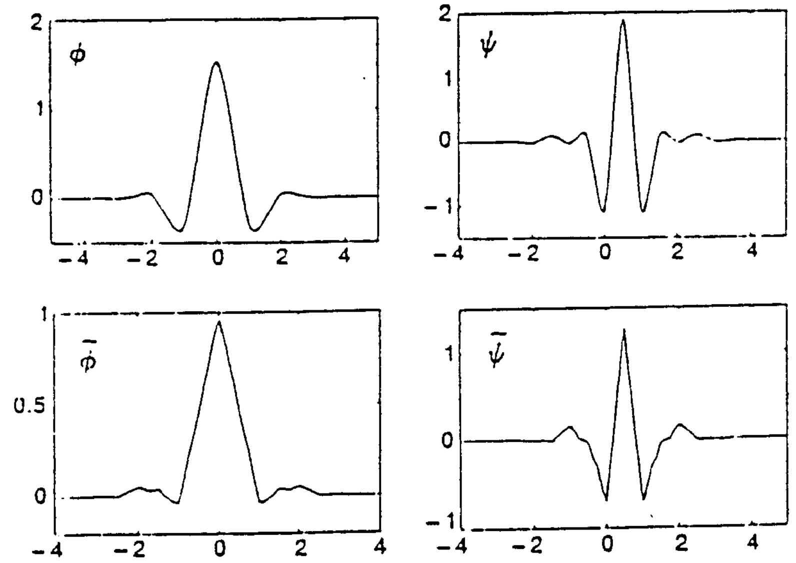

Figure 7: An example of functions $\phi, \tilde{\phi}, \psi, \tilde{\psi}$.

sufficiently long filters. Many examples of biorthogonal wavelet bases with reasonably regular ψ and $\tilde{\psi}$ can be constructed [9]. The regularity of $\tilde{\psi}_{m,n}$, which is linked to the number of zero moments of ψ, is more important than the regularity of $\psi_{m,n}$. Many examples are possible [9, 19], including spline filters, and filters close to orthonormal wavelet filters. An example of scaling functions ϕ, $\tilde{\phi}$ and mother wavelets ψ, $\tilde{\psi}$ from [9, p.280] is shown in Figure 7.

4.2 Application to synthetic ship radar data

We generated a radar return database utilizing the NRL Code 5750 ship radar return simulator. One of our objectives is to organize such databases efficiently so as to minimize the memory required for storage, and to minimize the search time for classification. Both have significant applications in surveillance, tracking, Automatic Target Recognition and battle management. It is important to analyze methods with respect to scale-up in the size of the database. In generating the synthetic data we kept the radar fixed and turned the ship. The parameters in the synthetic data were as follows. Radar frequency: 16.25 GHz, Elevation angle: .023°, Sea state: 3, Pulsewidth: 10 ns Pulse Repetition Interval: 80 ms to 1 s.

In creating the database, we varied the aspect angle from 0° to 360° in increments of 0.05°. We shall refer to this data set as $\mathcal{S}_\infty$. This allowed for large variation in the number and appearance of dominant scatterers. The database contained 7,200 pulses at fine resolution, which corresponded for the sampled data to vectors of length 2^7. We did not change the range gate and therefore each pulse in the database has the same time duration. We denote by $[0, T]$ this common time length of each pulse. In our experiments we took $T = 1280$ ns. We also created an additional database with 1,000 pulses by restricting the data to a finite set of aspect angles 0° to 360° in increments of 45°. We shall refer to this data set as $\mathcal{S}_\epsilon$. We worked only with discrete time data, and each pulse was represented initially as a vector consisting of $2^7 = 128$ samples, reflecting a sampling interval of 10 ns. We shall denote by f the generic sampled pulse in the database. We shall use subindices, such as f_j to denote a specific pulse. The sample values will be denoted by $f(n)$, where $n \in \{1, 2, \cdots, 2^7\}$. Since we will consider subsampled versions of the pulses as well as of various derived signals in the multiresolution representations, we will use index sets for the samples which are subsets of $\{1, 2, \cdots, 2^7\}$ as well.

An important question that arose during this study was the formalism for wavelet representations for signals on an interval. As already indicated all signals in the database are confined on the interval $[0,\ T]$. Among the options available for wavelet bases on an interval are [9]: (a) Use standard wavelets by extending each function to be zero outside the interval of interest; (b) Use a periodic extension of each signal; (c) Use wavelets with compact support included in the interval of interest and supplement them with specially adapted functions at the edges [27, 28]; (d) Use an extension of each signal by reflection and utilize biorthogonal wavelet bases with symmetric scaling functions ϕ and $\tilde{\phi}$ around the middle of the interval of interest. The performance of each method and the associated computational complexity are different. After preliminary experimentation with our data we have used primarily method (a). Methods (b) and (d) work very well with the type of signals we had since the ship radar pulses are zero towards both ends of the interval of interest (see Figures 1 and 4). This makes the results based on periodic extensions more accurate and actually very similar to results based on extension by reflection. Due to this characteristic of our data we found method (a) the easiest to implement and the results were as accurate as any other method. This implementation was further facilitated by the fact that we used FIR filters, typically with small number of taps. As a consequence of these observations we have primarily used extension by zeros, and this is implicitly assumed in the sequel of this paper.

From the given collection of sampled pulses we construct several new signals and spaces following the results on wavelets summarized in section 4.1. We used compactly supported wavelets generated with a scaling function ϕ with compact support satisfying a dilation equation like (4.4), with only finitely nonzero coefficients

$$\phi(t) = \sum_{k=0}^{K} c_k \phi(2t - k). \tag{4.23}$$

Here K and $\{c_k, \ k = 0, 2, \cdots, K\}$ are parameters to be chosen. Note that the c_k are related to the h_k of (4.4) via $c_k = \sqrt{2} h_k$. The corresponding compactly supported wavelet is given by the analog of (4.5)

$$\psi(t) = \sum_{k=0}^{K} (-1)^k c_{-k+1} \phi(2t - k). \tag{4.24}$$

Since $\sum_k h_k = \sqrt{2}$, the c_k satisfy the normalization constraint

$$\sum_{k=0}^{K} c_k = 2. \tag{4.25}$$

Finally to ensure that the translates of $\phi(t)$ are orthogonal we require that

$$\sum_{k=0}^{K} c_k c_{k+2l} = 2\delta_{0,l} \tag{4.26}$$

where $\delta_{0,l}$ is the Kronecker delta function. The condition (4.26) is equivalent to the "power complimentary condition" (4.12), where $H(\xi)$ is the low-pass filter transfer function given in (4.11), and is also equal to

$$H(\xi) = \frac{1}{2} \sum_{k=0}^{K} c_k e^{ik\xi}, \tag{4.27}$$

which is also known as the *symbol* of the dilation equation (4.23).

Let $\mathcal{S}$ denote the set of discretized radar pulses. The fine resolution data will be denoted by $S^0 f(n), \ n \in I^0$, where $I^0 \subseteq \{1, 2, \cdots, 2^7\}$, is the index set of the fine resolution data. We could subsample the given data to economize computations but in our experiments we took I^0 to be the full set $\{1, 2, \cdots, 2^7\}$. We shall let $N = 2^J$ denote the number of samples in the fine resolution data, where J is the maximum possible number of scales that we can consider. In practice one considers scales up to J^* where $J^* < J$. We denote by $\Delta\tau^0$ the sampling interval for the fine resolution; it can be also thought as the resolution of the fine resolution data. Respectively for each resolution m we denote by I^m the subset of I^0 where sampled values of the m^th resolution pulse representation $S^m f$ are computed. I^m is obtained from I^{m-1} by decimation; resulting in our case in $|I^m| = 2^{7-m}$. To use the notation and constructs of section (4.1) we identify $\mathbf{a^0}$ with the vector of sampled data $S^0 f$. Then we use the *pyramid* scheme (4.7) to recursively compute the successive approximations $S^m f$ to the pulse f at various scales m and the residual pulses $W^m f$, following the notation of section (4.1). All signals are digitized and we identify the vector $\mathbf{a^m}$ with $S^m f$ and the vector $\mathbf{c^m}$ with $W^m f$. As we proceed with this analysis step from scale m to the coarser scale $m + 1$, the space of signals becomes smaller (see (4.22), and the length of vectors is halved (see the decimating blocks in Figure 6). Thus at scale 1 we have $N/2 = 2^{J-1}$ samples and resolution $\Delta\tau^1 = 2\Delta\tau^0$. At scale m we have $N/(2^m) = 2^{J-m}$ samples and resolution $\Delta\tau^m = 2^m\Delta\tau^0$. Thus the algorithm recursively splits the initial vector $\mathbf{a^0}$ representing the sampled pulse $S^0 f$ to its components $\mathbf{c^m}$ at different scales indexed by m representing the wavelet residuals $W^m f$. Thus the tree multiresolution scheme of Figure 5 replaces the information in each pulse $f = S^0 f$ with the set $\{W^m f,\ m = 1, 2, \cdots, J^*,\ S^{J^*} f\}$. One can also think of $W^m f$ as representing the details of the pulse f in scale m.

The recursions of this decomposition, as described by (4.7) can be given in terms of matrices L_h, L_g with coefficients completely determined by the coefficients c_k of the dilation equation (4.23). Conversely the reconstruction of f from the set $\{W^m f,\ m = 1, 2, \cdots, J^*,\ S^{J^*} f\}$ proceeds as in (4.10) which utilizes the adjoint matrices L_h^*, L_g^*. In this setting the Discrete Wavelet Transform is uniquely characterized by an orthogonal matrix $T_W \colon \mathbf{R}^N \longrightarrow \mathbf{R}^N$, where $N = 2^J$ is the number of samples available from the signal. The orthogonal matrix T_W is uniquely determined by the coefficients c_k involved in the recursion of the scaling function as described in equation (4.23). It has been shown recently that all "valid" sequences $\{c_k,\ k = 0, 1, \cdots, K\}$ are parametrized by $(K+1)/2 - 1$ "angle" parameters [29] in the range $[0,\ 2\pi)$. This parametrization is useful in obtaining optimal or best wavelets for particular signals. The design of "best" mother wavelets and scaling functions (orthogonal or biorthogonal) for ship radar returns is an important problem in our program. It will be addressed elsewhere.

In Figures 8, 9 and 10, we show typical examples obtained from our experiments. As indicated above $N = 128$, and we used $J^* = 3$. This gives us four scales (including the given fine scale) $m = 0, 1, 2, 3$, with vector lengths $128, 64, 32, 16$ and resolutions 10 ns, 20 ns, 40 ns, 80 ns, respectively. Figure 8 illustrates the multiresolution representation of a typical ship radar pulse using the Haar wavelet, which corresponds to low-pass filter coefficients $\{h_0 = 0,\ h_1 = 0.5,\ h_2 = 0.5,\ h_k = 0\ for\ k \geq 3\}$.

Figure 9 illustrates the multiresolution representation of a typical ship radar pulse using an orthonormal wavelet, with highest number of vanishing moments from [9, chapter 6], constructed from ${}_N H$ with $N = 6$, which corresponds to the filter coefficients shown in Table 1.

Figure 10 illustrates the multiresolution representation of a typical ship radar pulse using the biorthogonal symmetric wavelet, based on splines with less dissimilar lengths $l = 4 = k$, $\tilde{k} = 4$ from [11] which corresponds to the analysis filter coefficients h_k and the and the synthesis filter coefficients $\tilde{h}_k$ shown in Table 2.

From these Figures we see that the multiresolution representations are different depending on the wavelet used. The differences are more pronounced at the coarser resolutions. "Best" representations of the pulses at coarse resolutions are important for our scheme, since the resulting hierarchical data representation utilizes increasingly finer resolution data for classification. This scheme will work very well provided the initial coarse

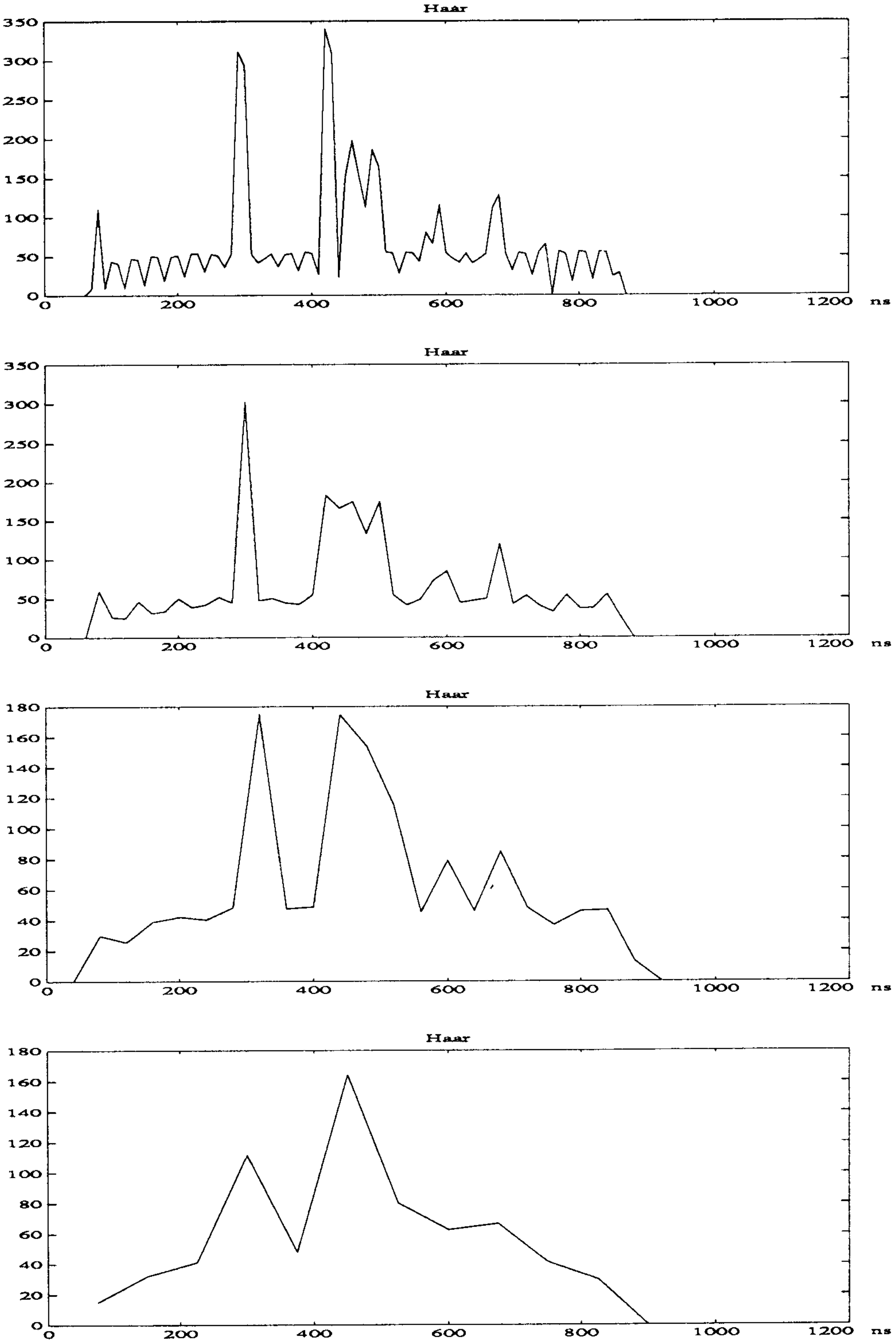

Figure 8: Multiresolution representation of typical ship return pulse using Haar wavelets.

$h_0 = 0.111540743350$	$h_6 = 0.097501605587$
$h_1 = 0.494623890398$	$h_7 = 0.027522865530$
$h_2 = 0.751133908021$	$h_8 = -0.031582039318$
$h_3 = 0.315250351709$	$h_9 = 0.000553842201$
$h_4 = -0.226264693965$	$h_{10} = 0.004777257511$
$h_5 = -0.129766867567$	$h_{11} = -0.001077301085$
$h_k = 0 \ for \ k \geq 12$	

Table 1: Filter Coefficients of Daubechies[8] with $N = 6$.

$h_0 = 0.788484$	$\tilde{h}_0 = 0.852698$
$h_1 = h_{-1} = 0.418092$	$\tilde{h}_1 = \tilde{h}_{-1} = 0.377402$
$h_2 = h_{-2} = -0.040689$	$\tilde{h}_2 = \tilde{h}_{-2} = -0.110623$
$h_3 = h_{-3} = -0.064539$	$\tilde{h}_3 = \tilde{h}_{-3} = -0.023849$
$h_k = 0 \ for \ \|k\| \geq 4$	$\tilde{h}_4 = \tilde{h}_{-4} = 0.037828$
	$\tilde{h}_k = 0 \ for \ \|k\| \geq 5$

Table 2: Filter Coefficients of Biorthogonal Symmetric Wavelet

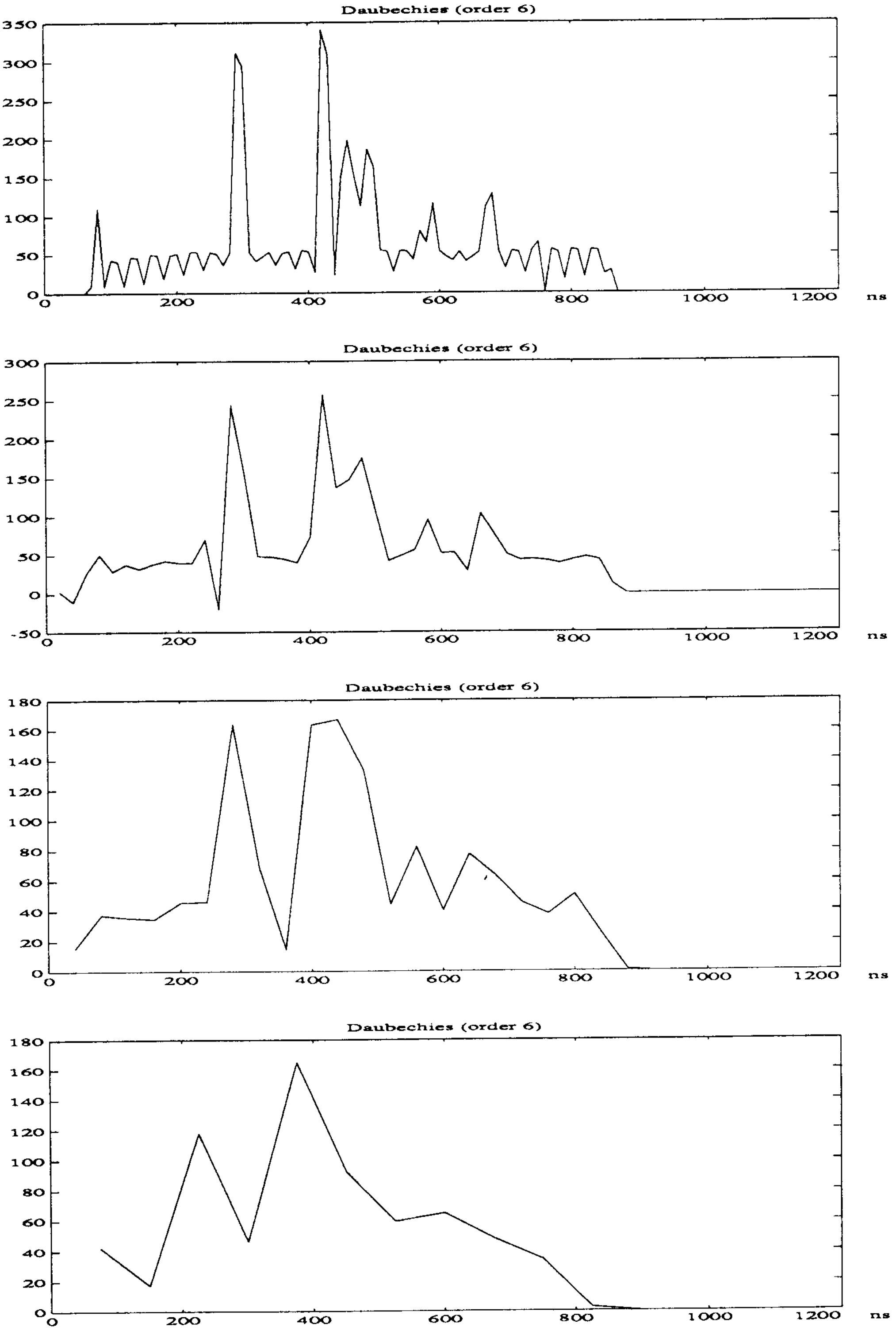

Figure 9: Multiresolution representation of typical ship return pulse using Daubechies wavelets.

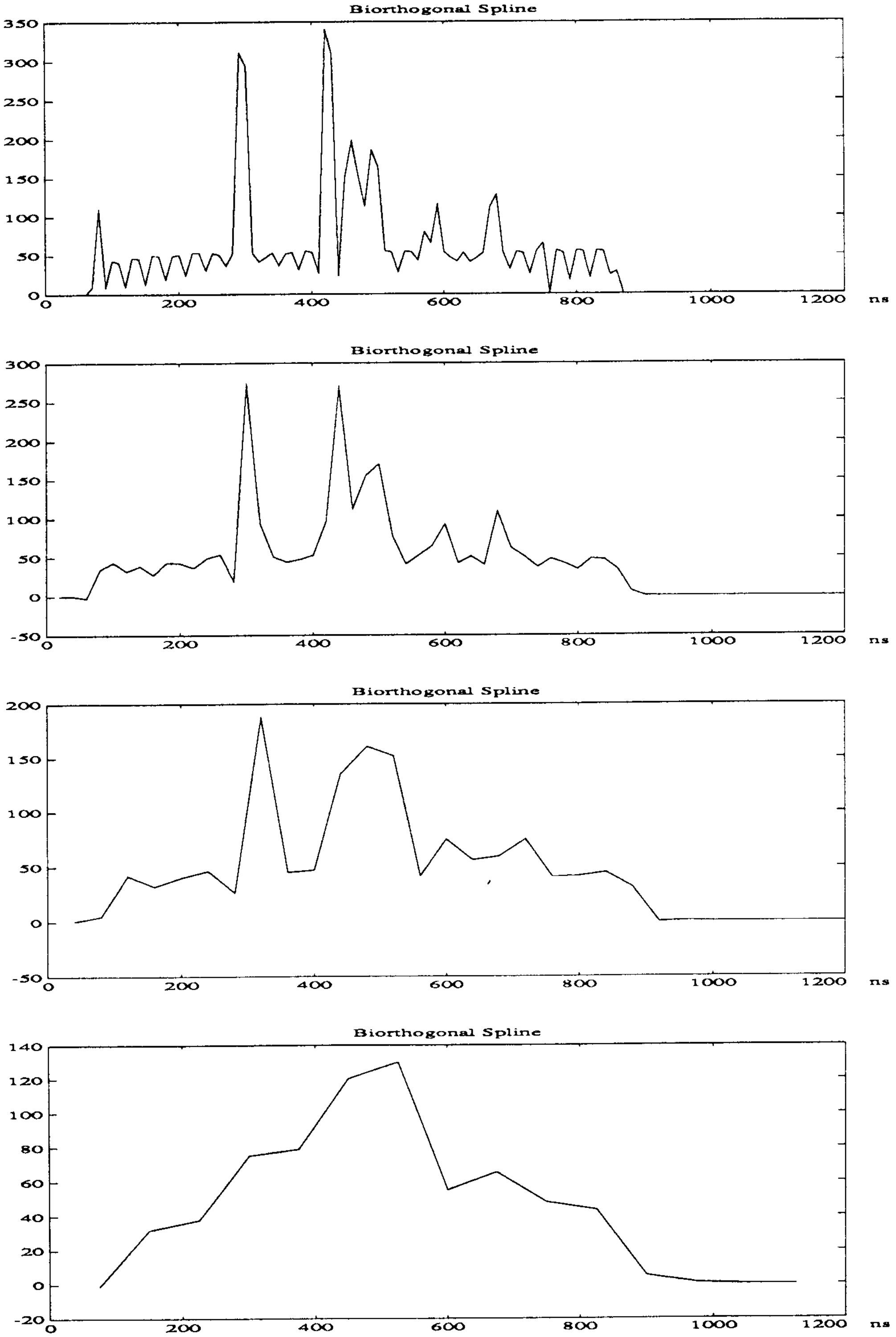

Figure 10: Multiresolution representation of typical ship return pulse using biorthogonal spline symmetric wavelets.

resolution clustering is not completely wrong (see section 6 for further details on this issue). These considerations motivate optimizing the wavelet as discussed earlier. From these figures we observe clearly that the peaks of the radar pulse coalesce as we vary the resolution from fine to coarse, in a manner similar to the one observed when we used variable pulsewidths of comparable resolution; c.f. Figure 4). This is not unexpected and justifies earlier statements regarding the suitability of wavelet analysis for radar signals. The following argument is clearer in continuous time. From (3.1) the received pulse r is represented as a convolution. Let us apply to r the wavelet decomposition described in section 4.1. Then

$$r = \sum_{m<J,n} c_{m,n}(r)\psi_{m,n} + \sum_{n} a_{J,n}(r)\phi_{J,n} \tag{4.28}$$

where the second sum provides an approximation to r at the coarser resolution J. By applying (3.1) to the last equation we obtain that this coarser approximation is given by

$$\begin{aligned} S^J r &= \textstyle\sum_n < r, \phi_{J,n} > \phi_{J,n} \\ &= \textstyle\sum_n < H_{IF} \star h_s \star p, \ \phi_{J,n} > \phi_{J,n} \\ &= H_{IF} \star h_s \star \textstyle\sum_n < p, \phi_{J,n} > \phi_{J,n} \\ &= H_{IF} \star h_s \star S^J p. \end{aligned} \tag{4.29}$$

This means, that within the approximations involved, the "smoothed" return at resolution J is equivalent to the return due to a "smoothed" transmit pulse $S^J p$ at the same resolution. Therefore if we locate the maxima of the amplitude $|S^J r|$, they should coincide with the maxima of the amplitude of a return to a wider pulse $S^J p$. This is the basic reason for the expected performance of wavelet decompositions and representations in accurately describing multiresolution radar returns, as if we varied the pulsewidth. this computation is very accurate if we perform the wavelet decomposition (4.28) on the complex radar return. This will be presented elsewhere. For the purposes of this study we worked directly with the amplitude of the pulses, in which case the argument just given is only an approximation. However, it does provide the physical justification for our methodology.

On a more fundamental basis, the fact that wavelets permit a time-frequency analysis is important for radar signals, since the method described here can be applied equally well to other types of radar signals, including ISAR, SAR, frequency agile radar, etc.. Such applications will be pursued elsewhere.

5 Hierarchical Representation via Tree Structured Vector Quantization

We now turn to the principal problem addressed in our study: how to efficiently compress radar pulse data so as to minimize memory requirements and minimize search recovery time. We aimed at the development of a hierarchical, tree–structured representation, which could utilize the multiresolution representations provided by wavelets. In developing the method described here we were inspired by the recent discoveries of certain deep relations between Vector Quantization (VQ) [13] and decision trees [24].

5.1 Vector quantization as a clustering algorithm

The goal of vector quantization is multidimensional data compression with respect to a fidelity criterion. A fundamental result of Shannon's rate-distortion theory is that a better performance trade-off between the amount of compression and the amount of distortion of a given set of source vectors can always be achieved by directly coding the vectors as opposed to simply coding the scalar components. In its usual form, vector quantization is understood to be generally concerned with the reduction of the memory requirements for data storage or the reduction of bandwidth of transmitted signals under a noise immunity constraint. Such is the case in speech processing applications and image compression. In this form, the fidelity criterion, or *distortion measure*, is a similarity measure between the original set of vector samples and the decoding of their compressed versions. Another important measure is the *rate of the code* which gives an idea of the amount of compression achieved. However, by properly defining a rate-distortion measure between the respective sample distributions one can reinterpret the process of vector quantization in the context of optimal decision theory. In fact, this flexibility of the definition and interpretation of rate and distortion in Shannon's theory has recently lead to very beneficial cross-fertilization between these two areas, in particular between tree-structured vector quantizers [13] and classification (decision) trees [24]. This insightful viewpoint interprets the distortion-rate pair as a valuable quantifier of overlearning.

The crucial element in efficient vector quantizer design is a distortion measure which has the right properties. The most frequently used distortion measure in signal coding is the L^2 distortion measure whether it is computed directly on signal subblocks or transformations of subblocks. However in problems such as the one addressed here the choice of the distorsion measure is critical, since it affects classification performance. Selection of distortion measures in vector quantization is a difficult problem which has not been investigated appropriately.

An $N-$dimensional (memoryless) vector quantizer, which we will denote VQ for convenience, consists of an encoder γ mapping an $N-$dimensional vector space $\mathcal{X}$ to a set of (channel) code symbols $\mathcal{F}$ and a decoder δ mapping these code symbols to a reproduction alphabet $\mathcal{A}$. Sample vectors in $\mathcal{X}$ are usually considered to arise from a random source. For a given code symbol $F \in \mathcal{F}$ if we let $\ell(F)$ denote its length (in bits) then we can define the average rate R in bits per vector of a given code γ by $R = E[\ell(\gamma(X))]$ where the expectation arises from our chosen probabilistic model for the random vector X. The distortion between any input vector $x \in \mathcal{X}$ and its reproduction $\delta(\gamma(x))$ is modeled according to a cost assignment $d(x, \delta(\gamma(x)))$, with which one defines the average distortion of a given VQ: $E[d(X, \delta(\gamma(X)))]$. In general, the design of a VQ involves a trade-off between the conflicting desires of low average code rate (good compression) and low distortion (good fidelity). In all but the simplest cases VQ design entails the use of a training set $\mathcal{L}$ of vector samples from $\mathcal{X}$ since one usually has no other access to the underlying probability distributions. The fact that the decoder can be optimized with respect to a fixed encoder and *vice versa* is a key tool in VQ design customarily referred to as the generalized Lloyd algorithm [13]. Thus, there are three main ingredients necessary to efficiently design a vector quantizer: A good distortion measure, a tractable formula for the generalized centroid induced by this measure, and a code vector initialization procedure. A significant proportion of the computation time of the generalized Lloyd VQ design algorithm is spent searching through $\mathcal{F}$ for the code symbol $F^* = \delta(X)$ which minimizes the distortion $d(X, F)$ for the training sample X. Implementations of the basic algorithm in which the search is exhaustive are called *full-search*. In many VQ systems the time overhead associated with this search is too costly. One very successful approach to avoiding the costs of full search VQ at the expense of an increase in average rate has been the tree-structured vector quantizer (TSVQ); we will confine this discussion to the binary-tree case.

Most recently, variable depth TSVQs have been studied in which initially a large tree is grown according to a "greedy" algorithm and then pruned back in an optimal fashion to reduce the average rate or depth of the tree. However, this tactic can be augmented with a pruning phase which searches for a subtree of the large original tree having an acceptable distortion-rate trade-off. Such a VQ is called a pruned TSVQ (PTSVQ). Here, by pruned subtree of a given tree we refer to one that shares the same root node. Because any tree T has a finite number of such pruned subtrees, one can perform an exhaustive search for the subtree which achieves a best distortion-rate performance. However, generalizing results from the optimal design of classification trees [24], Gray et al. [30] have shown how to efficiently design PTSVQ's.

An important recent development, observed 'by Chou, Lookabough and Gray [30] is that classification trees (CT) can be understood in a vector quantization framework by considering the communication model in which one views the unknown class C as a probabilistic source generating the "input" vector X which in turn is corrupted by channel noise. A classification tree, τ, then takes this observation, "encodes" it and then "decodes" the class decision C^*,

$$C^* = \delta(F^*) = \delta(\gamma(X)) = \delta(\gamma(\zeta(C))), \tag{5.1}$$

as shown in the modified VQ/CT diagram below.

$$\underset{\overbrace{\textit{input classes}}}{\mathcal{C}} \quad \overset{\underbrace{\textit{random channel}}}{\xrightarrow{\zeta}} \quad \underset{\overbrace{\textit{observed vectors}}}{\mathcal{X}} \quad \overset{\underbrace{\textit{distortion encoder}}}{\xrightarrow{\gamma}} \quad \underset{\overbrace{\textit{vector codes}}}{\mathcal{F}} \quad \overset{\underbrace{\textit{class decoder}}}{\xrightarrow{\delta}} \quad \underset{\overbrace{\textit{class decisions}}}{\mathcal{A}} \tag{5.2}$$

Here of course, the reproduction alphabet of class decisions, $\mathcal{A}$, is precisely the set of class labels $\mathcal{C}$. In this diagram the $\tau(\cdot) = \delta(\gamma(\cdot))$ represents the tree classifier. Since we wish to minimize the probability of misclassification,

$$\begin{aligned} \Pr\{\, C \neq \tau(X)\,\} &= E[\,1\{C \neq \tau(X)\}\,] \\ &= E[\,E[\,1\{C \neq \tau(X)\} \,|\, X\,]\,], \end{aligned} \tag{5.3}$$

we may define the so-called *modified distortion* measure $d_M(x,c) = \Pr\{\, C \neq c \,|\, X = x\,\}$ on $\mathcal{X} \times \mathcal{C}$ and therefore study tree classifiers in the context of information theory, and in particular distortion-rate bounds and vector quantization. Conversely, new techniques from the theory of decision trees can be used to design vector quantizers. In fact, an optimal pruning algorithm recently developed for classification trees, the generalized Breiman, Friedman, Olshen and Stone (BFOS) algorithm [30], which removes nodes in an optimal manner, represents the first fundamentally new VQ design algorithm since the generalized Lloyd algorithm [13] and Kohonen's Learning Vector Quantization [5].

VQ in addition is a clustering algorithm. Indeed the codewords, represented by the centroids, can be thought of as representatives of the equivalence class represented by each cell of the VQ (each Voronoi cell). It is in this sense that we use VQ in our approach to the problem of hierarchical representations for ship radar returns.

Of particular interest is tree structured vector quantization (TSVQ), which provides logarithmic (in the number of data) search time *vs* linear (in the number of data) search time provided by full search VQ. TSVQ is a special case of hierarchical VQ [13]. TSVQ is one of the most effective and widely used techniques for reducing the search complexity in VQ. In TSVQ the search is performed in stages. In each stage a substantial subset of candidate Voronoi cells is eliminated from consideration by a relatively small number of operations. In an $m - ary$ tree search with a balanced tree, the input data vector is compared with m predesigned Voronoi vectors at each stage or node of the tree. The nearest (minimum distortion) Voronoi vector determines which of m paths through the tree to select in order to reach the next stage of testing. At each stage the number of candidate Voronoi vectors is reduced to $1/m$ of the previous set of candidates. Such a tree structured search is a special case of a classification tree or decision tree [24]. When $m = 2$ we have a binary tree.

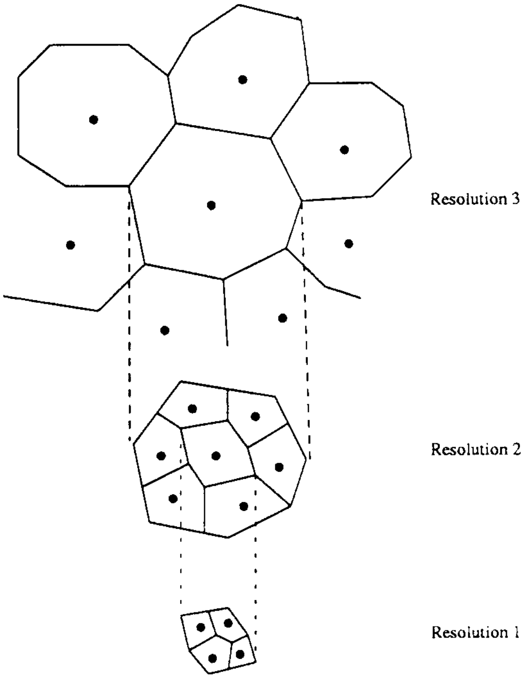

Figure 11: Illustrating a multiresolution TSVQ by splitting Voronoi cells based on different resolution data.

A useful method for designing the tree structure is based on the application the Linde–Buzo–Gray (LBG) algorithm [13] to successive stages using a training set. We have used a variant of this method which is of the "greedy" [31] variation. More precisely our algorithm splits the cell which contributes the largest portion of the current overall distortion.

5.2 A TSVQ conforming with a wavelet multiresolution representation

The algorithm which implements our overall approach is described in this subsection. We first perform a multiresolution wavelet representation of the radar pulses, based on the selection of a mother wavelet, as described in section 4.2. This allows us to consider each pulse reconstructed at different resolutions $S^0 f$, $S^1 f, \cdots$, $S^{J^*} f$. We then proceed by splitting the signal space at various resolutions in cells as indicated pictorially in Figure 11.

The details of the approach and the resulting algorithm follow. First, a multiresolution representation of the data vectors (sampled radar pulses) is made, resulting in each data vector being represented in a number of resolutions. The data vector space (signal space) is then partitioned into cells, or collections of data vectors which are determined by the repeated application of the Linde-Buzo-Gray (LBG) algorithm. LBG is first applied to the coarsest resolution representation of the data vectors $\{S^{J^*}f,\ f \in \mathcal{S}\}$. Since it is the coarsest representation, the corresponding length of the data vectors is the shortest; in our experiments that length was 16 (c.f. section 4.2). As a result this clustering is faster than a clustering performed on the much longer fine resolution representations of the data vectors. The resultant distortion is determined based on a mean squared distance metric, and is computed using the finest resolution representation of the data vectors. The cell (equivalence class of coarse resolution representations) which is the greatest contributor to the total average distortion for the entire partition is the cell which is split in the next application of LBG. A new Voronoi vector is found near the Voronoi vector for the cell to be split and is added to the Voronoi vectors previously used for LBG. LBG is then applied to the entire population of data vectors, again using the coarsest representation of each vector. These steps are repeated until the percentage reduction in distortion for the entire population falls below a predetermined threshold. The partition in the coarsest resolution is then fixed, and further partitioning continues by splitting the cells already obtained based on finer resolution representations of the data vectors in the cell. The algorithm then iterates through the following steps until the allotted number of cells have been allocated, or until total average distortion has been reduced to a requisite level:

(i) The next cell chosen for splitting is the one, among the current terminal cells (nodes), which is the greatest contributor to the total average distortion for the entire population of data vectors. Distortion is computed always using the finest resolution representation of the data vectors.

(ii) If that cell (chosen for further splitting in step (i)) belongs to a subpartition which has been fixed, it is split using the finer resolution representations of the data vectors in the cell. If it belongs to a subpartition which is not fixed already, it is split by the addition of a Voronoi vector and the application of LBG to the entire subpartition using the same resolution representation of the data vectors in the subpartition.

(iii) If the percentage improvement in distortion for the subpartition to which the cell belongs falls below a threshold, the subpartition is fixed and further subdivision of the subpartition is allowed to continue only using finer resolution representations of the data vectors.

Each new layer in the tree corresponds exactly to partitions based on the next finer resolution data representation than the previous layer. In the context of describing the hierarchy of the partitioning of the given signal set $\mathcal{S}$ (the ship radar pulses) we will often use resolution indices to determine the layer in the tree. Thus we may refer to the signals that are contained in a particular element of a partition in the r^{th} layer of the tree as the signals of a given cell at resolution m, where $m = J^* - r$. Thus layer 0 in the tree corresponds to the coarser resolution J^*; in our experiments $J^* = 3$.

Note that the algorithm splits the population of a cell differently depending on whether the cell belongs to a subpartition that has been fixed or not. If its subpartition has not been fixed, then upon splitting, a new Voronoi vector is chosen, close to the Voronoi vector (centroid) for that cell, and LBG is applied to the population of that cell, and all of the other cells in its subpartition. If its subpartition has been fixed, LBG is applied only to the population of that cell, but finer resolution representations of the data vectors are used. In order to maintain this distinction, in the description of the algorithm given below, we must often refer to partitions and subpartitions. We simply note here that once a subpartition is fixed, it contains the same population as some cell at the next finer resolution. For example, once fixed, subpartition s of cell c at resolution m (layer $J^* - m$), would contain the same data vectors as cell s at resolution $m - 1$.

Given a set of data vectors $\mathcal{S}$ at fine resolution 0, a multiresolution representation of those observations $\{S^0 f,\ S^1 f,\ \cdots,\ S^{J^*} f,\ f \in \mathcal{S}\}$, and a tree, $\mathcal{T}$, describing the hierarchy in the partitioning of the observation space, we define:

r tree layer (0 is the top layer, J^* the deepest layer)
m resolution (J^* is coarsest, m is coarser than $m - 1$)
$S^m f_i$ data vector number i at resolution m
o_{rc} a set of ordered pairs of indices describing the contents and the subpartitioning of cell c at tree layer r. There are as many ordered pairs as there are data vectors in cell c at resolution $J^* - r$. The first index of each pair identifies the data vector. The second index identifies the subpartition of cell c where this vector belongs.
o_{rcs} the set of data vectors in cell c, subcell s, at resolution $J^* - r$
v_{rc} the set of Voronoi vectors in the partition described by o_{rc}
v_{rcs} the Voronoi vector for subcell s of cell c, at resolution $J^* - r$
l the set of all leaf nodes in $\mathcal{T}$ that may be split
k current number of leaf nodes in the tree
K maximum number of leaf nodes allowed
0 finest resolution scale
M total number of observations (data vectors)

The algorithm proceeds as follows:

```
main
$m \leftarrow J^*; r \leftarrow 0; c \leftarrow 0; s \leftarrow 0$          Coarsest data, first cell, first subpartition
$o_{rc} \leftarrow \{(1,0),(2,0),...,(M,0)\}$      All data vectors are put in one subpartition.
$v_{rc} \leftarrow$ centroid$(o_{rc}, J^* - r)$         Initialize Voronoi vector to centroid of data.
$\mathcal{T} \leftarrow t_{rc}$                           Initialize tree to single node.
$l \leftarrow \{t_{rc}\}$                         Initialize leaf list to single node.
lbg$(o_{rc}, v_{rc}, J^* - r)$                 Apply LBG to the subpartition.
$z \leftarrow y \leftarrow D(o_{rc}, 0)$                  Store distortion for the subpartition.
splitCell$(r, c, s)$                       Split the subpartition into 2 subcells.
```

```
$k \leftarrow 2$                                                Store the number of leaf nodes.
if$(y - D(o_{rc}, 0) < \epsilon) f_{rc} \leftarrow 1$           If distortion is not much improved,
                                                                fix subpartition.
else $f_{rc} \leftarrow 0$                                      Otherwise, allow further splitting
                                                                of subpartitions.
while $(\{k < K\} \cup \{z > \delta\})$                         Until maximum number of leaves
                                                                or distortion lower bound is reached
    $(r, c, s) \leftarrow \arg\max_{(r,c,s):n(o_{rcs}) \in l} d(o_{rcs}, 0)$    Find the leaf (cell) which contributes
                                                                    the most to total distortion
                                                                    (measured at the finest resolution).
    $y \leftarrow D(o_{rc}, 0)$                                     Store the distortion for that cell's
                                                                    parent cell, utilizing the parent cell's
                                                                    (fixed) subpartitions.
    if$(\{f_{rc} = 1\} \cap \{r < J^*\})$ expandCell$(r, c, s)$     If subpartition is fixed and finer
                                                                    resolutions are available, then
                                                                    expand subpartition at next finer
                                                                    resolution.
    if$(\{f_{rc} = 1\} \cap \{r = M\})$ remLeaf$(r, c, s)$          If subpartition is fixed at the finest
                                                                    resolution, then remove this cell
                                                                    from the list of leaf nodes.
    if$(f_{rc} = 0)$                                                If subpartition is not fixed,
        splitCell$(r, c, s)$                                        Split the subcell via LBG.
        if$(y - D(o_{rc}, 0) < \epsilon)$ $f_{rc} \leftarrow 1$     If distortion is not much improved,
                                                                    fix subpartition.
        else $f_{rc} \leftarrow 0$                                  Otherwise, allow further splitting
                                                                    of subpartitions.
    end if
    $z = totD()$                                                Compute the distortion for the
                                                                    entire tree
end while
exit
```

```
splitCell$(r, c, s)$:
$m \leftarrow J^* - r; i \leftarrow \arg\min_{i:(i,s) \in o_{rc}} ||S^m f_i - v_{rcs}||$    Find the data vector
                                                                closest to the Voronoi vector.
addCell$(r, c, \bar{s})$                                        Make a new subpartition.
addLeaf$(r, c, \bar{s})$
$k \leftarrow k + 1$
$v_{rc\bar{s}} \leftarrow S^m f_i$                              Make the found data vector
$v_{rc} \leftarrow v_{rc} \cup v_{rc\bar{s}}$                   a Voronoi vector.
lbg$(o_{rc}, v_{rc}, J^* - r)$                                  Apply LBG to the subpartitions.
return
```

```
expandCell($r, c, s$):
$\bar{r} \leftarrow r + 1$
addCell($\bar{r}, s, \bar{s}$)                      Create a new cell on the next tree layer
                                                   having the same data vectors as
                                                   those in the subpartition $s$.

addLeaf($\bar{r}, s, \bar{s}$)
remLeaf($r, c, s$)
$o_{\bar{r}s} \leftarrow \{(i, \bar{s}) : (i, s) \in o_{rc}\}$
$v_{\bar{r}s\bar{s}} \leftarrow$ centroid $(o_{\bar{r}s}, \bar{r})$          Initialize the Voronoi vectors for this new cell.
$v_{\bar{r}s} \leftarrow v_{\bar{r}s\bar{s}}$
lbg($o_{\bar{r}s}, v_{\bar{r}s}, \bar{r}$)
return
```

addCell(r, c, s): Creates a new subpartition, s, at resolution $J^* - r$, cell c, and adds a node to the tree to correspond to it.

addLeaf(r, c, s): Adds this node to the list of leaf nodes ready for splitting or expansion.

remLeaf(r, c, s): Removes this node from the list of leaf nodes ready for splitting or expansion.

centroid(o_{rc}, r): Returns the centroid of the data vector in o_{rc}, computed at resolution $m = J^* - r$.

lbg(o_{rc}, v_{rc}, m): Applies the Linde-Buzo-Gray algorithm to data in o_{rc}, using Voronoi Vectors v_{rc}, at resolution $m = J^* - r$.

n(o_{rcs}): Defines the tree node that corresponds to o_{rcs}.

d($o_{rcs}, 0$): Computes the distortion of all data vectors in subpartition s,
cell c and resolution $J^* - r$, weighted by the relative frequency p_{rcs}
All computations are done at the finest resolution.
Specifically, letting
n_{rcs} = the number of data vectors in subcell s, cell c, resolution $J^* - r$,

$p_{rcs} = n_{rcs}/M$
$v_{rcs,0}$ is the Voronoi vector for all data vectors in subpartition s, cell c, resolution $J^* - r$, but recomputed with finest resolution data
Then we have
$d(o_{rcs}, 0) = p_{rcs}(1/n_{rcs}) \sum_{i \in o_{rcs}} ||v_{rcs,0} - S^0 f_i||^2$

D(o_{rc}, 0): Computes the distortion of all data vectors in cell c, resolution $J^* - r$, weighted by the relative frequency of the cell. using finest resolution data.
$D(o_{rc}, 0) = \sum_{s:(i,s) \in o_{rc}} d(o_{rcs}, 0)$

totD(): Computes the distortion for the entire tree, using finest resolution data
$totD() = \sum_{r,c,s:n(r,c,s) \in \{terminal\ \ nodes\}} d(o_{rcs}, 0)$

We have constructed a hierarchical organization of the data as a tree. The tree is conformant with the multiresolution data representations. It is clear that the search of the hierarchically organized database will be much faster than the search of the overall database. The question is: how much performance did we sacrifice? To answer this question we compared the results of the wavelet-TSVQ algorithm with those of straight VQ applied to the finest resolution data, by means of the total (operational) distortion *vs* number of cells performance curve, as well as *vs* the entropy of the final partition. The number of cells is the number of terminal nodes (cells). The entropy of the partition is $E = -\sum_{r,c,s:n(r,c,s) \in \{terminal\ \ nodes\}} p_{rcs} \log p_{rcs}$. Since entropy is a measure of randomness of the population of a partition, the larger E is the better the clustering performed; i.e. the cells of the partition contain well matched pulses. As the typical results depicted in Figures 12, 13, 14, 15 illustrate, the performance achieved by our wavelet-TSVQ algorithm is indeed excellent. Figures 12, 13, describe the results on the data set $\mathcal{S}_1$, where the aspect angle was varied every 0.05°. Figures 14, 15, describe the results on the data set $\mathcal{S}_2$, where the aspect angle was varied every 45°. The mother wavelet used in these results was the Haar wavelet.

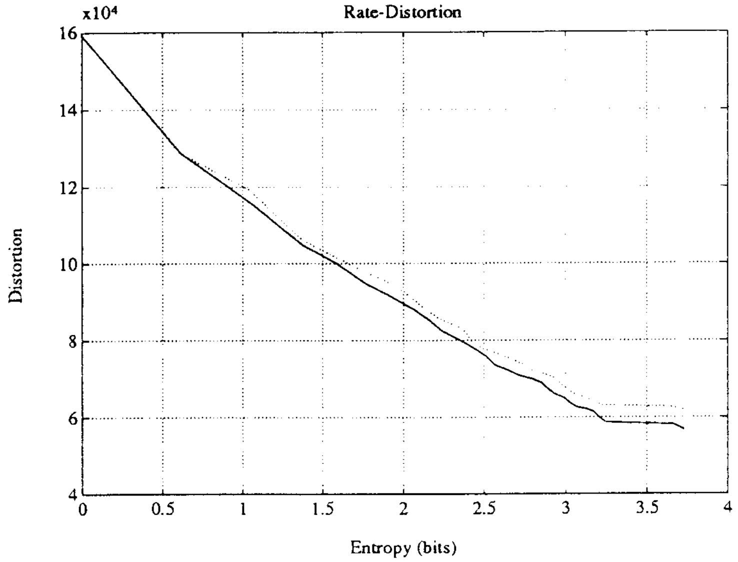

Figure 12: Performance comparison (entropy) between VQ on fine scale pulses vs wavelet-TSVQ on multiresolution pulses; $\mathcal{S}_1$ data set

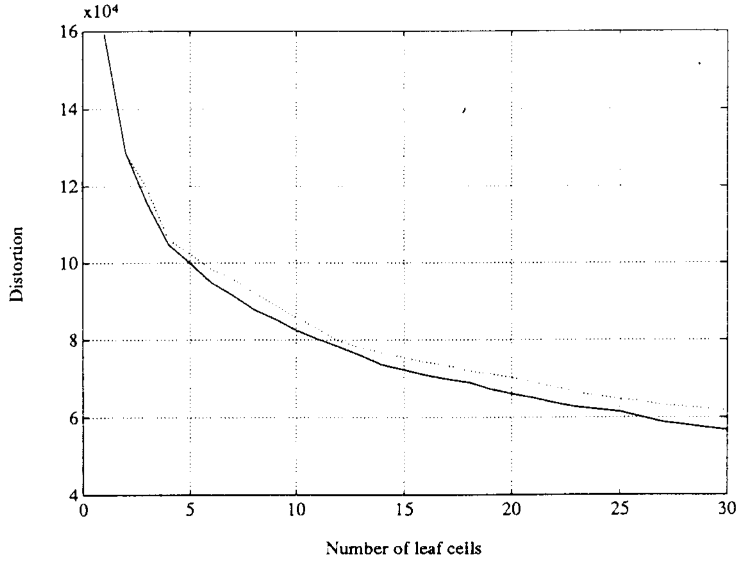

Figure 13: Performance comparison (number of leaf nodes) between VQ on fine scale pulses vs wavelet-TSVQ on multiresolution pulses; $\mathcal{S}_1$ data set

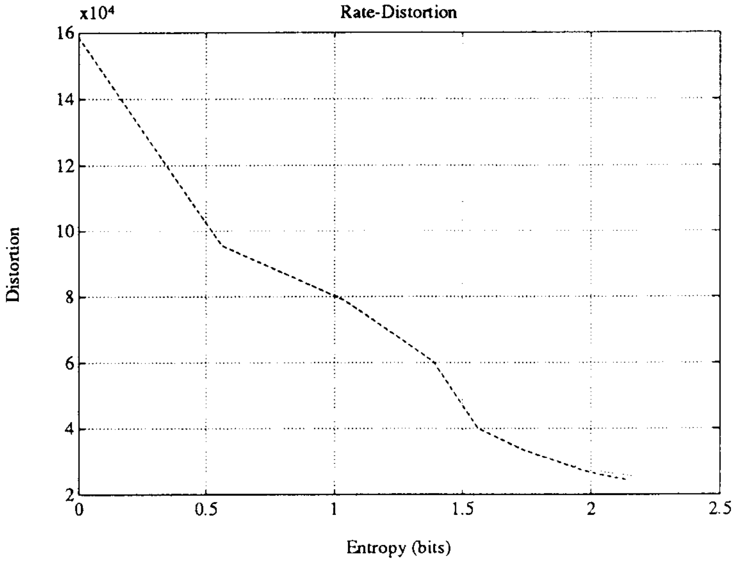

Figure 14: Performance comparison (entropy) between VQ on fine scale pulses vs wavelet-TSVQ on multiresolution pulses; $\mathcal{S}_2$ data set

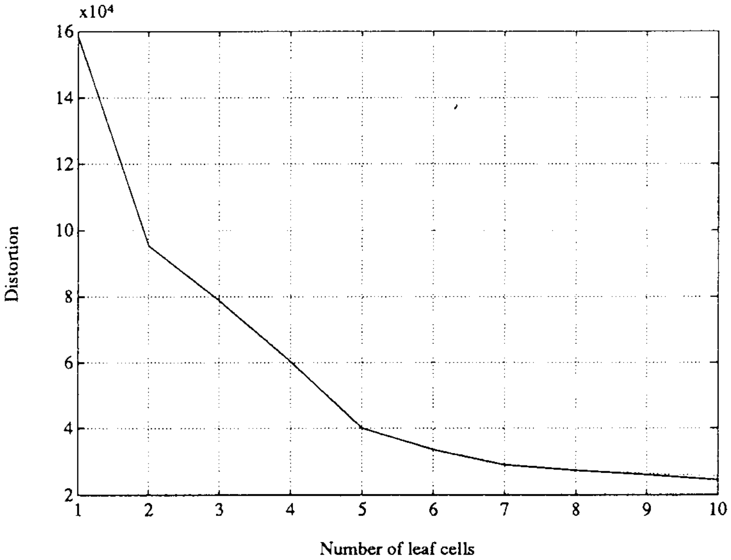

Figure 15: Performance comparison (number of leaf nodes) between VQ on fine scale pulses vs wavelet-TSVQ on multiresolution pulses; $\mathcal{S}_2$ data set

6 Experiments, Interpretations and Conclusions

In this section we examine carefully the trees constructed by our method and show indeed that our wavelet-TSVQ indeed achieves an indexing scheme for ship targets reminiscent of that obtained by varying the aspect angle and the pulsewidth. This is an indexing scheme because it provides a hierarchical organization of the multi–viewpoint (aspect and elevation) multi–pulsewidth radar data from a ship using significant clusterings in the δ, α, ϵ parameter space. We also examine very carefully the resulting cells to discover on what characteristics of the pulse the clustering was based in an essential way.

First in Figure 16 we show the tree constructed by our method for the data set $\mathcal{S}_1$. We show the cells that the algorithm created at each resolution (tree layer) as well as the percentages of pulses that were clustered in each cell. This tree has 30 final cells, and achieved average distortion of 61748, and partition entropy of 3.37. Its performance was excellent in comparison to full VQ search. The notation we have used to designate the cells is as follows: *cell m,k* denotes the cell number k at resolution m.

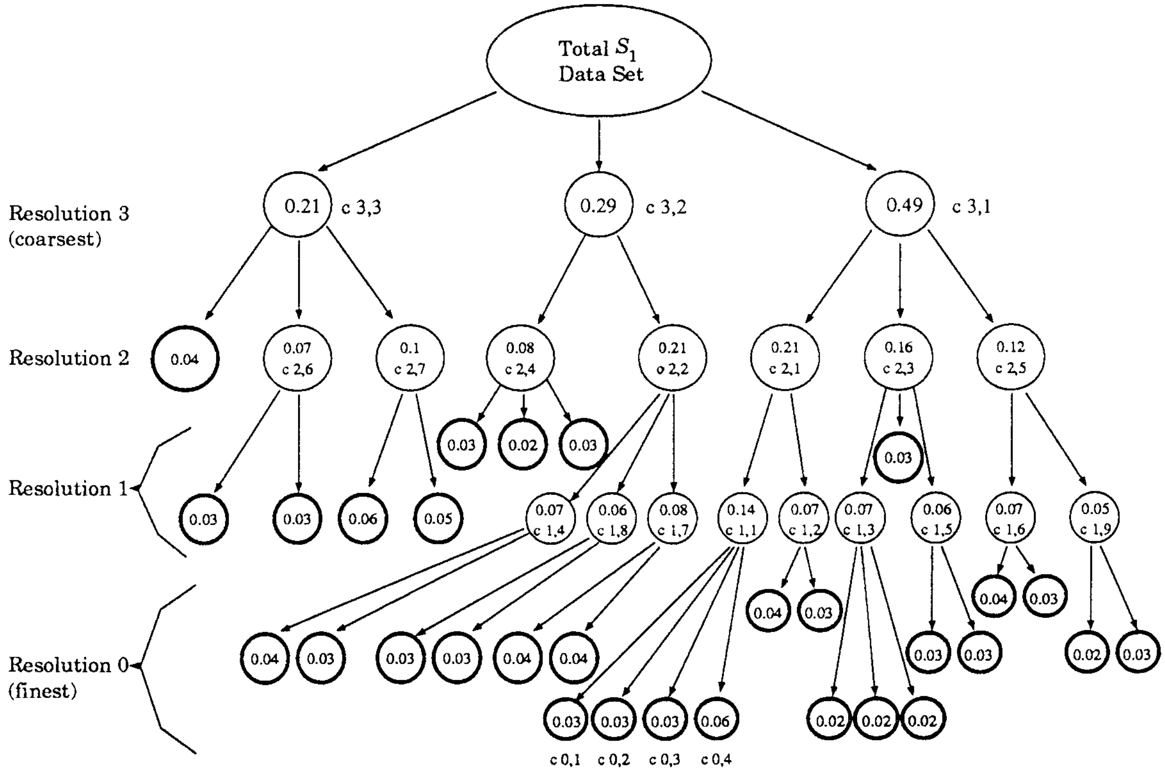

Figure 16: Tree data organization constructed by the wavelet-TSVQ algorithm on the $\mathcal{S}_1$ data set

In Figures 17, 18, 19, 20, we show the pulses in cells of this tree. We see that at the coarse level the wavelet–TSVQ algorithm clusters the radar pulses according to aspect.

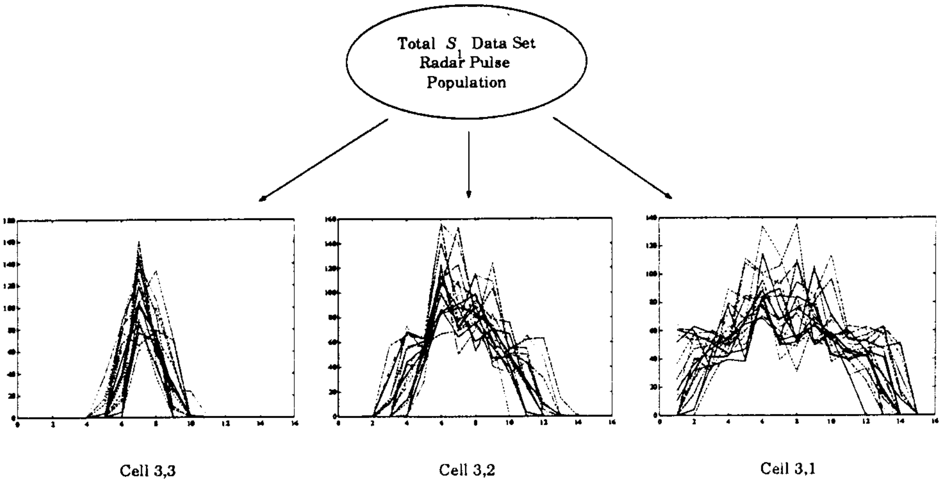

Figure 17: Layer 0, resolution 3 cells in the tree of Figure 16

As we move to finer resolutions the pulses cluster according to the location of their maxima. These maxima correspond to significant scatterers. This is a very nice and natural indexing of the radar returns from a ship. What is an important contribution is that we have developed a systematic, automatic method for constructing this indexing, and we have provided its quantitative parameters and performance measures. To better realize the efficiency of this method it suffices to compare it with conventional methods of indexing radar pulses based on small aspect-elevation cells. The method also reflects the accuracy limitations of the sensor, in the sense that it does not attempt to separate the pulses more than the sensor noise will permit.

In Figure 21 we show the tree constructed by our method for the data set $\mathcal{S}_2$. We show the cells that the algorithm created at each resolution (tree layer) as well as the percentages of pulses that were clustered in each cell. This tree has 11 final cells, and achieved average distortion of 24463, and partition entropy of 2.24. Its performance was excellent in comparison to full VQ search. The notation we have used to designate the cells is as follows: *cell m,k* denotes the cell number k at resolution m.

In Figures 22, 23, 24, 25, we show the pulses in cells of this tree. We see that at the coarse level the wavelet–TSVQ algorithm clusters the radar pulses according to aspect. As we move to finer resolutions the pulses cluster according to the location of their maxima. This is a very nice and natural indexing of a the radar returns from a ship. We also observe that this clustering is sharper in this case, in comparison with the data set $\mathcal{S}_1$, because we have less variability in aspect.

Putting all this together we have discovered in addition an extremely efficient indexing scheme for high range–resolution radar data, which is akin to the *aspect graph* widely used in computer vision based object recognition [22]. Indeed as we show in Figure 26 a multiresolution aspect graph for radar ship data results naturally from our methodology.

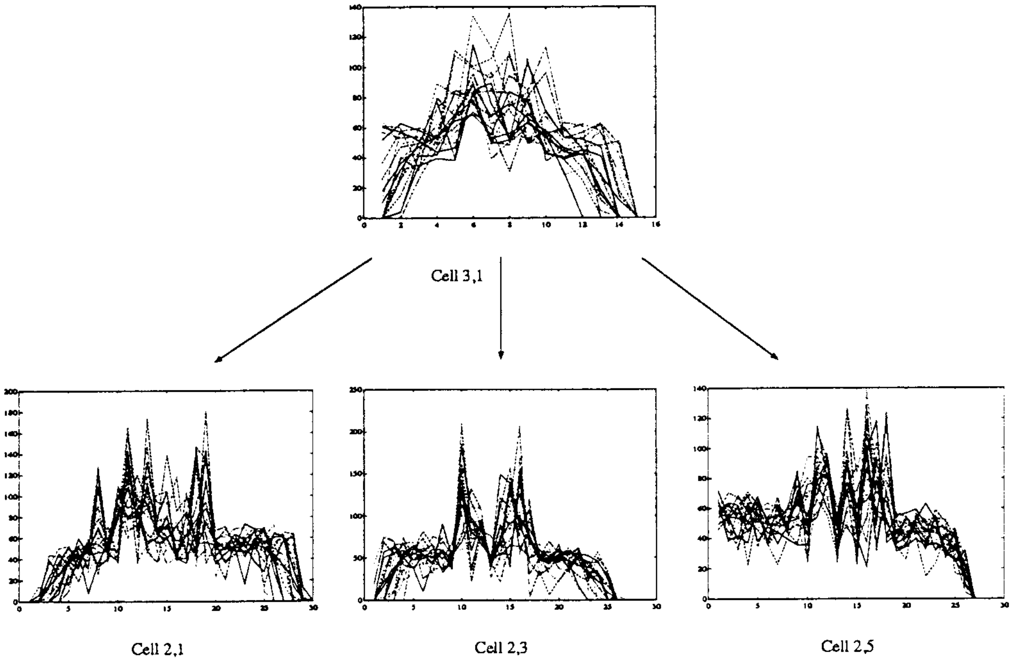

Figure 18: Layer 1, resolution 2 splitting of cells in the tree of Figure 16

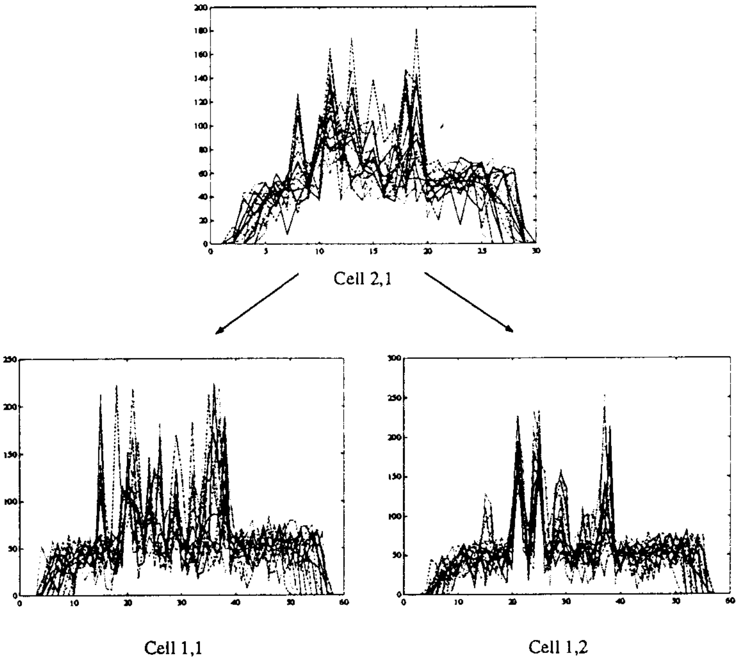

Figure 19: Layer 2, resolution 1 splitting of cells in the tree of Figure 16

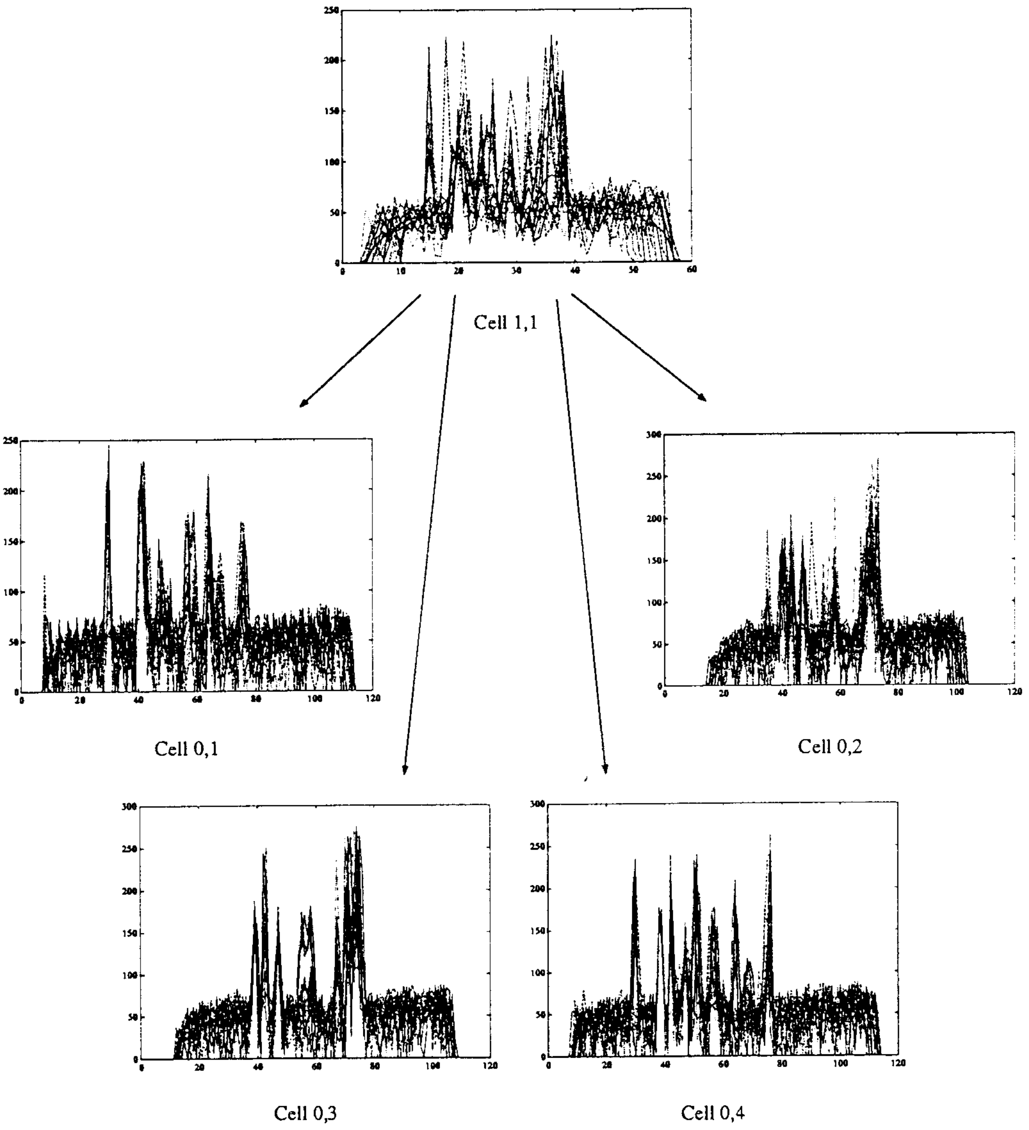

Figure 20: Layer 3, resolution 0 splitting of cells in the tree of Figure 16

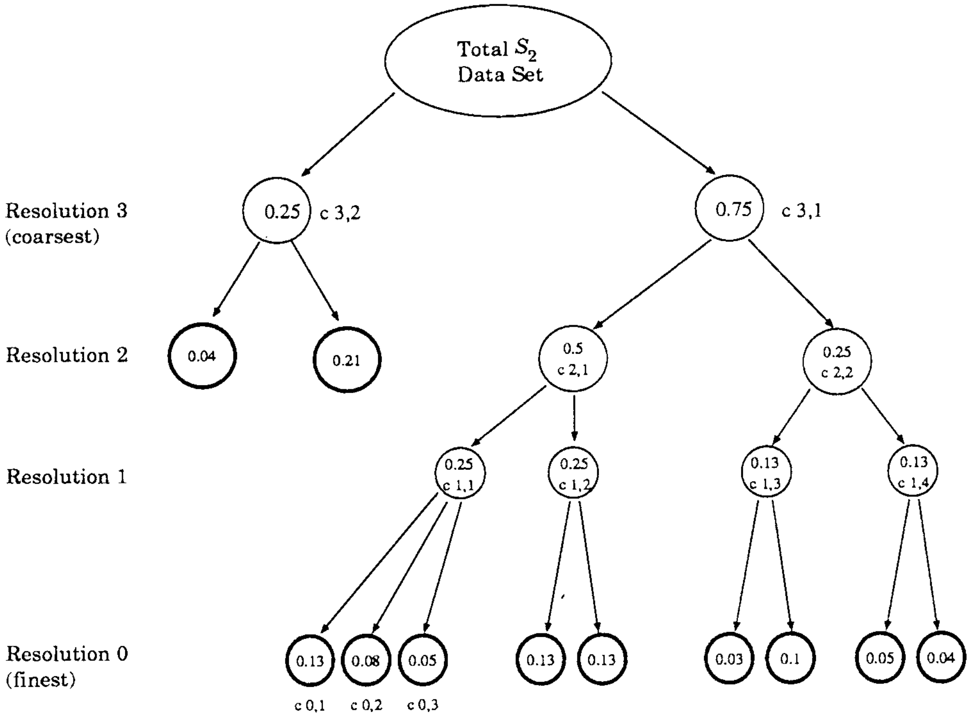

Figure 21: Tree data organization constructed by the wavelet-TSVQ algorithm on the $\mathcal{S}_2$ data set

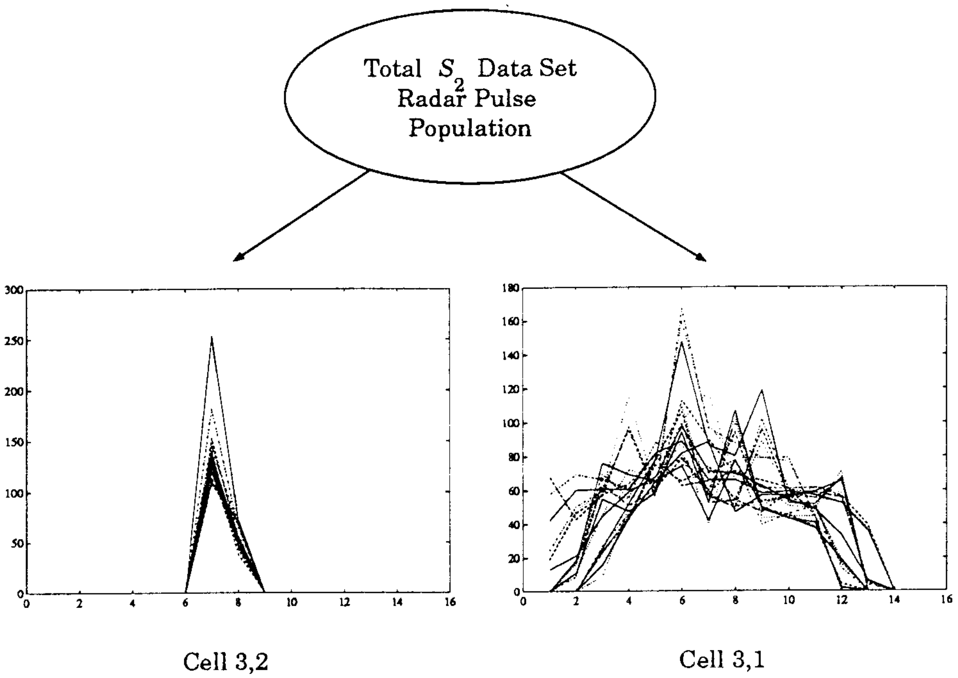

Figure 22: Layer 0, resolution 3 cells in the tree of Figure 21

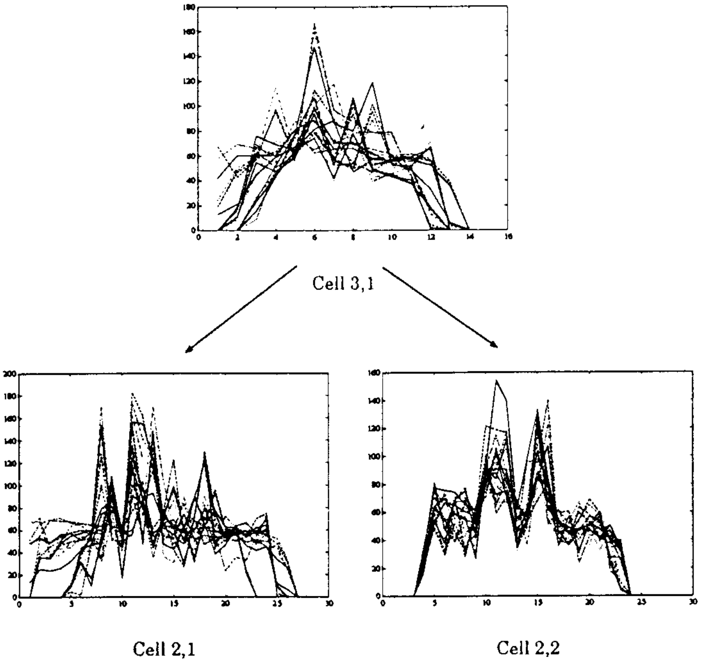

Figure 23: Layer 1, resolution 2 splitting of cells in the tree of Figure 21

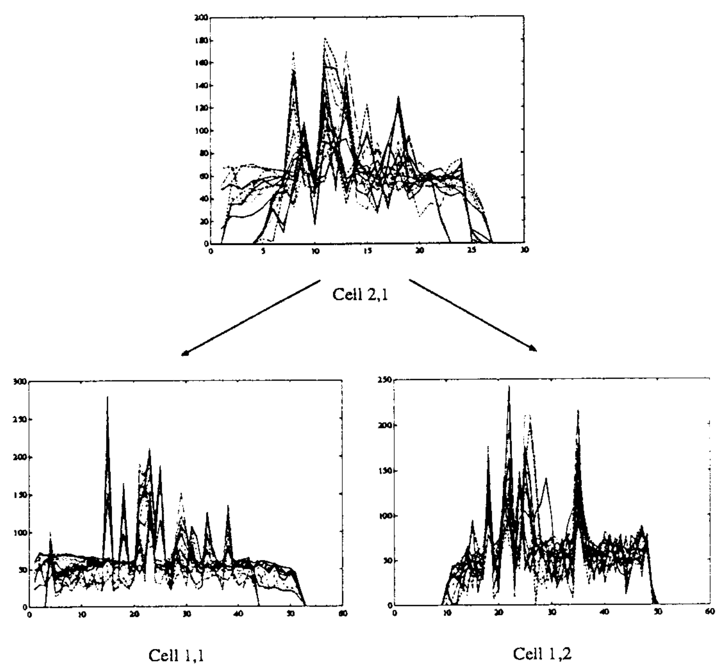

Figure 24: Layer 2, resolution 1 splitting of cells in the tree of Figure 21

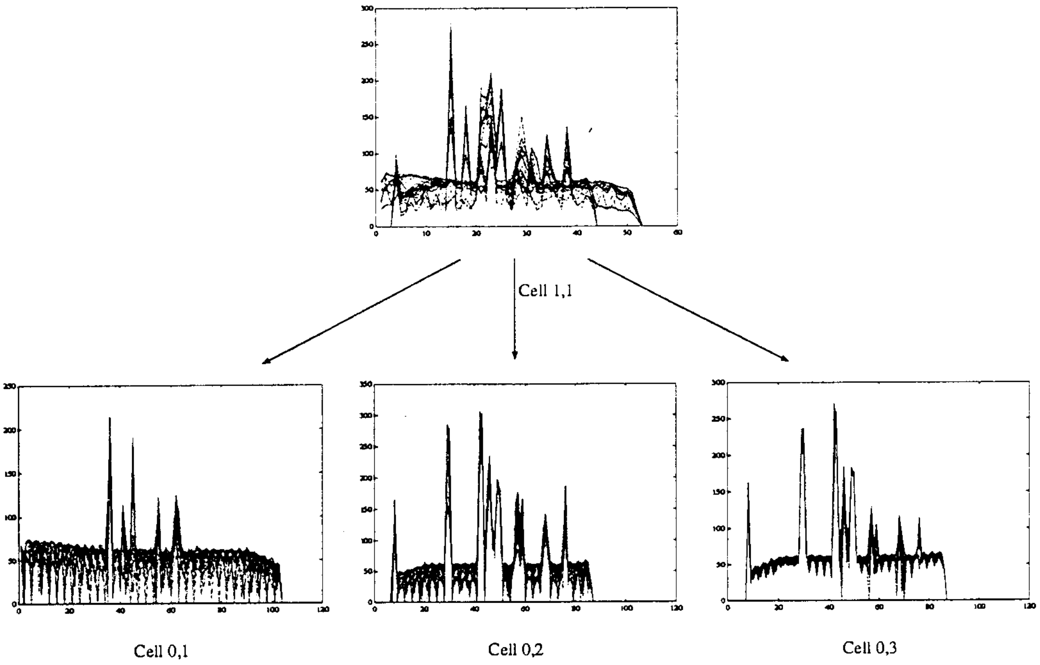

Figure 25: Layer 3, resolution 0 splitting of cells in the tree of Figure 21

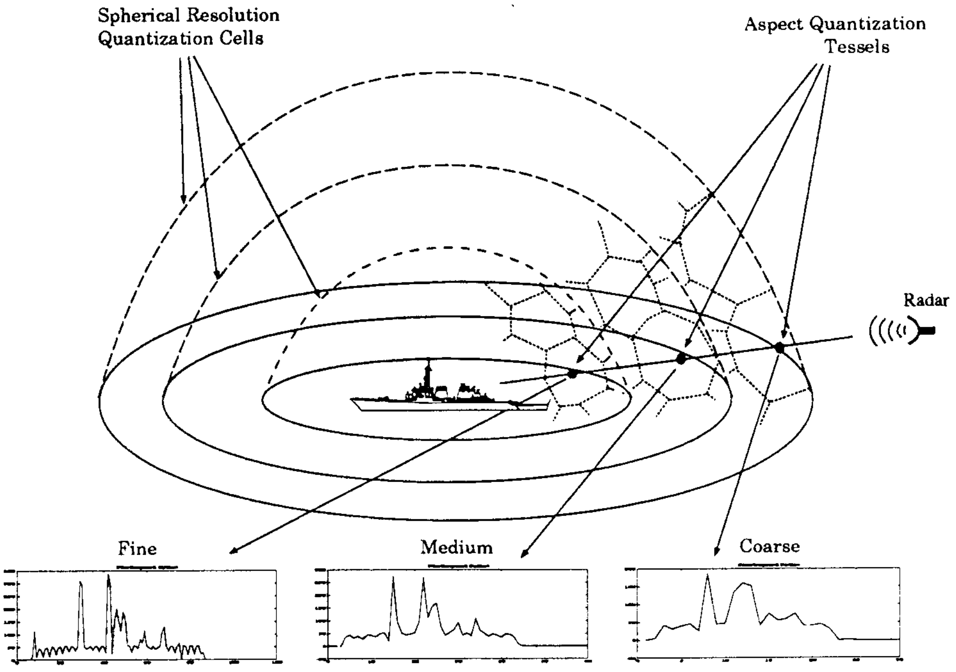

Figure 26: Illustrating a multiresolution aspect graph for radar ship data.

Here the concentric spheres designate different resolutions. The cells on these spheres illustrate aspect equivalence classes for the radar signals (returned pulses). These equivalence classes mean that the pulses in these clusters are difficult to discriminate due to their similarity. As we move inwards in this graph, the outside cells split as we can now get further characteristics of the target based on finer resolution information on the pulse. These characteristics are related to dominant scattering centers. To construct the aspect graph we select as the representative from each equivalence class, the radar pulse corresponding to the centroid of the corresponding Voronoi cell. The resulting graph has geometrically the appearance of the trees depicted in Figures 16 and 21. The nodes (cells in these figures) correspond to aspect–elevation neighborhoods for which the corresponding returned pulses are too similar to be separated. To each node there is associated a "canonical" pulse which corresponds to the centroid. The nodes are given for various resolutions. Transitions from one node to the other indicate either a change in aspect-elevation, or in resolution, of adequate magnitude to cause changes in the pulse that can be discriminated by the sensor. In the present case these changes are due to grouping (or ungrouping) of scatterers, or scatterer visibility (or non visibility) from the particular aspect–elevation cell. It is clear from this discussion that the aspect graph is a reduced but accurate model of the target and can be used to guide the ATR process in model–based ATR. In such an application the received pulse is compared with the "canonical" pulse at each node sequentially as the ATR process evolves. The aspect graph directs the search in an efficient and speedy manner; it is well known that tree based search is logarithmic in the number of terminal nodes, which is a substantial reduction from conventional methods.

These constructs are extremely useful in ATR, ship classification and in the retrieval of data from large radar pulse databases. The techniques are generic however and can be applied to a great variety of signal classification and hierarchical organization problems. We shall pursue several of these developments elsewhere.

Acknowledgments

We are indebted to Jim Preston of AIMS, Inc., for his expert help with the computations and the graphical representations reported in this paper.

References

[1] I. Daubechies, "Orthonormal Bases of Compactly Supported Wavelets", *Comm. Pure Appl. Math. 41*, pp. 909—996, 1988.

[2] S.G. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation", *IEEE Trans. on Pattern Analysis and Mach. Intel.*, Vol. 11, No. 7, pp. 674—693, 1989.

[3] Y. Meyer, *Ondelettes et Operateurs I, II*, Hermann, Paris, 1990.

[4] J.G. Daugman, "Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression", *IEEE Trans. on Ac. Sp. and Sign. Process.*, Vol 36, No. 7, pp. 1169—1179, July 1988.

[5] Teuvo Kohonen, *Self-organization and Associative Memory*, Springer-Verlag, New York, 1989.

[6] S.G. Mallat and S. Zhong, " Characterization of Signals From Multiscale Edges", *NYU, Computer Science Technical Report, Nov. 1991, IEEE Trans. Pattern Anal. Machine Intell.*

[7] S. Zhong and S.G. Mallat, "Compact Image Representation From Multiscale Edges", *Proc. of 3rd International Conf. on Comp. Vision*, Dec. 1990.

[8] O. Rioul and M. Vetterli, "Wavelets and Signal Processing", *IEEE Signal Processing Magazine*, Vol. 8, No.4, pp.14—38, October 1991.

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

[10] M.V. Wickerhauser, "Acoustic Signal Compression with Wave Packets", preprint Yale University, 1989.

[11] M. Antonini, M. Barlaud, P. Mathieu and I. Daubechies, "Image Coding Using Vector Quantization in the Wavelet Transform Domain", *Proc. 1990 IEEE ICASSP*, Albuquerque USA, pp. 2297—2300, April 1990.

[12] R.R. Coifman, Y. Meyer and V.M. Wickerhauser, "Wavelet Analysis and Signal Processing", preprint Yale University, 1990.

[13] A. Gersho and R.M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Press, 1991.

"Reduced Scatterer Ship RCS Models", *NRL Report*, to be published.

[14] G.A. Killen, "Advanced Technology MMW Seeker Testbed: A Multi-Technology Demonstration Sensor", *Proc. 1989 IEEE Radar National Conference*, pp. 35—41, 1989.

[15] *IEEE Proceedings, Special Issue on: radar cross sections of complex objects*, May 1989.

[16] N.C. Currie and C.E. Brown (Edts), *Principles and Applications of Millimeter-Wave Radar*, Artech House, 1987.

[17] M.J. Smith and D.P. Barnwell, "Exact Reconstruction for Tree-Structured Subband Coders", *IEEE Trans. ASSP 34*, pp. 434—441, 1986.

[18] M. Vetterli, "Splitting a Signal into Subsampled Channels Allowing Perfect Reconstruction", *Proc. IASTED Conf. Appl. Signal Processing Digital Filtering*, Paris France, June 1985.

[19] A. Cohen, I. Daubechies and J.C. Feauveau. "Biorthogonal Bases of Compactly Supported Wavelets", *AT&T Bell Labs Techn. Report*, No. TM 11217-900529-07.

[20] M. Vetterli and C. Herley, "Wavelets and Filter Banks: Relationships and New Results", *Proc. 1990 IEEE ICASSP*, Albuquerque USA, April 1990.

[21] D. Marr, *Vision*, Freeman and Co., 1982.

[22] K. Bowyer, D. Eggert, J. Stewman, and L. Stark, "Developing the Aspect Graph Representation for Use in Image Understanding", *Proc. DARPA Image Understanding Workshop 1989*, pp.831—849.

[23] A. Witkin, "Scale Space Filtering", *Proc. Int. Joint Conf. Artificial Intell.*", 1983.

[24] L. Breiman, J.H. Friedman, R.A. Olshen and C.J. Stone, *Classification and Regression Trees*, Wadsworth and Brooks, 1984.

[25] M. Skolnik (Edt), *Radar Handbook*, McGraw-Hill, Inc. 1982.

[26] G. Strang, "Wavelets and Dilation Equations: a Brief Introduction", *SIAM Review*, Vol. 31, No. 4, pp. 614-627, pp. 614–627, Dec. 1989.

[27] Y. Meyer, "Ondelettes sur l'Intervalle", *Rev. Math. Iberoamer.*, 1992.

[28] A. Cohen, I. Daubechies and P. Vial, " Wavelets and Fast Wavelet Transforms on the Interval", AT&T Bell laboratories, preprint.

[29] D. Pollen, "$SU_I(2,\ F[z,1/z])$ for F a subfield of C ", *J. Amer. Math. Soc.*, Vol.3, No. 3, pp. 611–624, July 1990.

[30] P.A. Chou, T. Lookabaugh, and R.M. Gray, "Optimal Pruning with Applications to Tree–Structured Source Coding and Modeling", *IEEE Trans. Inform. Theory*, pp. 299-315, March 1989.

[31] E. Riskin, "A Greedy Tree Growing Algorithm for the Design of Variable rate Vector Quantizers", *IEEE Trans. on Signal Process.*, Vol. 39, No. 11, pp. 2500–2507, Nov. 1991.