IMAGE FEATURE EXTRACTION BY MULTIRESOLUTION ANALYSIS WITH WAVELET TRANSFORM

A Thesis Submitted to the College of

Graduate Studies and Research

in Partial Fulfillment of the Requirements

for the Degree of Master of Science

in the Department of Electrical Engineering

University of Saskatchewan

Saskatoon

By

Zhiqiang Li

Spring 2002

Permission To Use

In presenting this thesis in partial fulfilment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Electrical Engineering

University of Saskatchewan

Saskatoon, Saskatchewan S7N 5A9

Abstract

IMAGE FEATURE EXTRACTION BY MULTIRESOLUTION ANALYSIS WITH WAVELET TRANSFORM

by Zhiqiang Li

This thesis is a study of the 2D-image signal feature extraction by Multi-Resolution Analysis (MRA) with Wavelet Transforms (WT).

The Wavelet Transform is the most significant signal analysis tool developed in last decade and it is extensively used in several fields such as signal processing, image processing, and communication. This thesis is devoted to the development of a novel feature extraction scheme for image analysis with WT. The goal is to utilize the merits of WT in feature extraction for stereo image matching, given that the image features involved are not well pre-defined.

The concept of Multi-Resolution Analysis, which is inherited naturally in WT, is employed in the decomposition of an image. The image properties at each resolution level are represented by the scalars of a wavelet and its translations at that level; in turn the feature space is formed by the wavelet at each decomposition level, precisely, by the wavelet and its dilations.

A novel WT algorithm, the Anchored Wavelet Transform (AWT), for MRA is introduced, and the usefulness of the new algorithm for feature extraction is demonstrated with examples. With the feature extraction scheme proposed, a stereo image matching algorithm is coded which could be a general pre-process step in 3D-object parameter estimation. Other usage could be found easily in image or texture pattern classification, surface segmentation and object tracking in an image sequence.

Acknowledgments

I am honored to express my gratitude to Dr. Wood, my supervisor, for his invaluable guidance, support, encouragement and financial assistance through all the phases of graduate study and the preparation of this thesis.

Finally, I also would like to thank my family and numerous friends who have supported me technically and spiritually. Without their help and encouragement, I would not have been able to work out the difficulties I encountered and finish this thesis successfully.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

2D Two Dimensional

3D Three Dimensional

ASIC Application-Specific Integrated Circuit

AWT Anchored Wavelet Transform

CCD Charge-Coupled Device

DFT Discrete Fourier Transform

DWT Discrete Wavelet Transform

FT Fourier Transform

GLCM Gray-Level Co-occurrence Matrix

HH Highpass and Highpass

HL Highpass and Lowpass

IR Infrared

JTC Joint Transform Correlator

LH Lowpass and Highpass

LL Lowpass and Lowpass

MRA Multiresolution Analysis

OCR Optical Character Recognition

SVD Singular Value Decomposition

UAV Unmanned Aerial Vehicle

VLSI Very Large Scale Integration

WFT Windowed Fourier Transform

WP Wavelet Packet

WT Wavelet Transform

Chapter 1 Introduction

Image feature extraction is an important step in many applications of machine/computer vision. In this chapter, the applications and compositions of machine/computer vision systems are briefly introduced first, to illustrate the basic functionalities of image feature extraction in the system; then an overview of feature extraction in image signal processing is made, and the feature extraction method with Wavelet transform developed in this thesis work is introduced.

1.1 Feature Extraction in Machine/Computer Vision

During the last three decades, machine/computer vision has been an area of active research and development. Machines with vision and computing capabilities offer viable solutions to a wide range of important practical tasks, such as controlling, inspecting, measuring, grading, sorting, guiding, counting and identifying.

The reason for this wide-spread contribution of machine/computer vision to a variety of industries is that optical sensing and digital data processing provides a way to equip machines to substitute for the human in many situations where human beings cannot work quickly and safely, cannot gain physical access, cannot concentrate for long periods, or are simply too unreliable or expensive to employ on a regular basis. For example, mobile robots have been recognized as a good substitution for human beings in a wide variety of constrained environments such as walking under the deep sea, working underground, and repairing the space station. The vision systems for mobile robots are used to help recognize

and track goal objects or locations, plan paths to the goal, avoid obstacles along the chosen path, compute motion parameters, etc. Another example of machine/computer vision systems adopted by the industry is to automate highway toll collection. The important technology that enables this process is the automatic recognizing and reading of the license plate number of a car passing through a toll gate with camera(s). This automatic process eliminates the manual intervention and traffic congestion at the gate and saves people from tedious tasks. Beside the two examples mentioned above, there are still countless machine/computer vision applications that improve our life and work; they allow us to convert the paper text to digital text without typing with Optical Character Recognition (OCR) program installed in many computers, they allow automated industrial inspection by camera for machined parts in factories to screen out defective units, and they allow motion perception in video streams from surveillance system which bring more chance of fast response from guards, etc.

In many of these applications of machine/computer vision, image feature extraction is an essential part of the vision process. A feature is the quantified description of the primitives in the image, and feature extraction is used to generate the description from the image signal. In Virginio Cantoni (1994) [1] edited proceedings, with the theme of "Human and Machine Vision: Analogies and Divergences", vision functionalities are categorized into seven groups when comparing human and machine vision:

1) Retina,
2) Visual Primitives,
3) Camera and Eye Control System,
4) Motion Perception and Interpretation,

5) Visual Representation,

6) Visual Reasoning,

7) Imagination and Learning.

In the following paragraphs, considering the seven sections briefly, it is clear that the quantified description for objects in a vision target is explicitly or implicitly used in almost every section of the vision process.

The retina consists of 120 million photoreceptors with a decreasing distribution density going from the fovea (close to the visual axis) towards the periphery, and has two functions: collect light signals and control the focus of the collection on the determined location. The image sensors (CCD device, sonic sensor, variety of films, etc.) in Machine/Computer Vision system act as the retina in collecting the image signal (when the signal is analog signal, it has to be converted to digital when the signal processing units are digital processors). To direct the image sensor to switch "attention" to relevant area of the scene, more intelligent parts must be involved to define and justify the properties associated with the area in the scene that will validate the area as relevant.

In the Visual Primitives section, the primitives are detected from the visual signals. The visual cortex examines the information from the retina for various parameters of the visual signal, such as orientation, color and velocity. The interacting cells of visual cortex detect fields of various sizes, selectively sensitive to the orientation of primitives such as lines or edges or even surfaces with texture. Correspondingly, much work has been done in machine/computer vision to detect those primitives by comparing their characteristics or features including photometric, geometric, statistic, etc., measures, against predefined criteria or ones generated by higher level modules. The characteristics and features must all

be quantified descriptions for the target primitives. The criteria must be quantified rules for detection, and the machine or computer must be able to understand them.

The eye control system is a sophisticated one composed of different control agents that will stabilize the image of the environment on the retina, visually scan a wide field of view, set and reset gaze orientation quickly, and pursue the selected targets. This control system has an hierarchical structure where there are different activity subsystems interacting with each other to fulfill the given tasks. For example, the target tracking control is accomplished through the combination of movements of body, head and eyes. Similarly, the camera control system contains control loops to do the diffused scanning, attention focusing, target tracking, trajectory detection and collision prediction. This system forms the visual front-end, guided by a high level reasoning system which defines and selects windowed fields or objects to observe.

Motion is one of the dominant primitives in that the visual perception in humans can recover the spatial 3D structure from a dynamic view of the environment which is superior to that obtained from the static primitives from static frames of view. Motion Perception and Interpretation can be achieved by Machine/Computer Vision system in two ways. One way is that a set of recognized objects are tracked along the frame sequence to get the correspondence scheme, and then the motion parameters are estimated by the object displacements in different frames. In another way, the 2D field of instantaneous pixel velocities, called the optical flow, is calculated without aligning on any corresponding objects.

Visual Representation, Visual Reasoning, and Imagination and Learning are at high levels of the vision system. Visual Representation is about how to remember what has been

seen and about the formalisms of representation of visual information for reasoning and learning; Visual Reasoning is about how to understand what has been seen based on represented visual information, taught knowledge and learned experience; Imagination and Learning is about how to use the understood visual information to construct a new format for visual information representation or new logic for reasoning. Correspondingly, in these levels, a machine or computer has to organize the descriptions of the primitives to form an internal view of the world, based on which a spatial layout will be constructed for relational reasoning. After reasoning, the results can be used by machine control systems to command the action of related parts of the machine, including the camera control sub-system. The artificial reasoning systems are used to find inferences and relations between objects (constructed by representation, could be refined by reasoning) according to predefined logic or one built from learning.

From the above outline of vision systems architecture, it is clear that the detection of primitives is the fundamental operation in machine/computer vision. It is the only place that the raw image signal is processed, and the higher-level process will rely on the result of the detection of primitives. Feature extraction is used to generate the quantified description to identify the primitives, therefore feature extraction is the essential part of primitive detection and machine/computer vision systems. Two factors have to be considered in selecting or constructing feature extraction methods for vision systems. One is the performance of feature extraction. Because feature extraction happens in the very beginning of the vision process pipeline and it deals with high-density raw image data, it has an important impact on the performance of the whole vision system. Another one is the flexibility of feature extraction. It should be flexible enough to accommodate to different

image resolutions resulting from the camera control of scanning and gazing. For example, a mobile robot on the way to a work site at high speed may get low quality images input to the vision system, and a rough description to the objects in the images may be enough for it to avoid collision. On the other hand, when it arrives at the work site to pick up a target object, more detailed description to the objects in the image may be needed.

1.2 Introduction to Image Feature Extraction

In this thesis work, the term 'feature' refers to quantified information describing and identifying the visual primitive in a vision systems framework described above; 'feature extraction' is concerned with extracting those quantified pieces of information directly from image signals to feed higher level function modules to make intelligent decisions. In Machine/Computer Vision, visual primitives defined by so called features are the smallest image subsets with specific geometrical and/or structure properties, so the feature is the basic level of description, in a hierarchical description of patterns in the image with increasing complexity.

In terms of pattern recognition, a pattern (primitive is a pattern, the basic pattern in image) is characterized by a set of features and a feature vector is assumed to represent the pattern to be recognized. The elements of a feature vector should be able to characterize a pattern so that it can be attributed to a specific class.

Because feature extraction is used to build the feature vector representing an image primitive, many extraction methods are devised to calculate the parameters of points, line segments, and contours, and treat these parameters as the feature of primitives. Sonka et al. [34] summarized the edge and contour feature extraction and optical flow analysis methods

developed in the past. The steps normally involved in these methods are image pre-processing and primitive description. The pre-processing is done to detect image primitives, and the primitive description is done to find characteristics of primitives to be used in higher-level processes in the vision system.

Edge is a very important primitive used in many vision applications. Edges represent small areas of high local contrast in correspondence with discontinuities in intensity. Therefore the pre-processing for edge feature extraction is to detect these discontinuities, which is called edge detection. The calculus is used to describe the change of continuous intensity functions using derivatives; an image function depends on two variables of the coordinates in image plane surface, and so edge-detecting operators describing edges are expressed by partial derivatives. The operators, such as the Roberts operator, the Laplace operator, and the Prewitt operator, approximate derivatives of image functions using differences computed from one or more convolution masks. For example, the Roberts operator consists of two masks: $\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ and $\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$. When this operator convolves with intensities of four pixels in a 2x2 window in an image, the edge could be found with large absolute results of the convolution. The reason is the operator calculates the intensity differences between adjacent pixels. The actual operators can be designed to be very complex with noise reduction functions and using second derivatives. However, no matter how complex the detector is, the detected edge is far from an ideal edge because of noise and is sometimes incomplete and degraded. Therefore symbolic data combined with detection results are used to represent the edge. The symbol may contain the degree of contrast, position and orientation of the edge segments from the detector. All these pieces

of information including symbols and detection result may assist the next stage process to determine what should be the complete and accurate edge.

Image segmentation is another primitive detection step leading to determining the contour of a region in the image; its main goal is to divide an image into parts that have a strong correlation with objects or areas of the real world contained in the image. The methods include edge-based, thresholding-based, region-based and matching-based segmentation. Edge based segmentation relies on edges found in an image by edge detecting operators and on strategies leading to link edge pieces together to form a reasonable contour construction. The pixel intensity thresholding method is used to find thresholds through intensity histogram analysis on the target image and to compare each pixel intensity to thresholds to determine to which region the pixel belongs, given that many image regions are characterized by constant intensity and thresholds can be established as criteria to separate these regions. Thresholding is computationally inexpensive and fast; although it is the oldest segmentation method, it is still widely used in simple applications. The region-based segmentation techniques are generally better in noisy images, where borders are extremely difficult to detect. Homogeneity is an important property of regions and is used as the main criterion in segmentation, and the basic idea is to divide an image into zones of maximum homogeneity. The simplest homogeneity criterion uses an average intensity of the region. The segmentation algorithm is used to grow the region of interest from an individual pixel to relatively larger regions by gradually merging and splitting the region repeatedly based on homogeneity calculated after each such operation. Matching is another basic approach to segmentation that is used to locate known objects in an image, to search for specific patterns. Matching is often based on

directly comparing intensity properties of image sub-regions, though some other methods such as derived features of sub-primitives (smaller primitives) and higher-level image presentations are also used.

After primitives are detected, the identifications or the characteristics, called features, of the found primitives will be extracted to complete the image feature extraction. The direct geometric properties, such as location in the image and length of the line segments, can be taken as features. Anyhow, more efforts are normally necessary to extract more sophisticated primitive characteristics as features, such as curvature measurement parameters, signatures (contour distance function coefficients), chord length and angle, etc. Mathematical transforms, such as Fourier Transforms, Hough transforms, B-spline representation, etc., can also be used to process primitives, and the coefficients of the transform result are the extracted features. For example, if $z(t)$ is the approximate function to a line segment (such as an edge) in an image, the Fourier representation of $z(t)$ is $z(t)=\sum_n T_n e^{int}$, then T_n series is an eligible representation of the original line segment.

Optical flow reflects the image changes due to motion during a time interval dt, and the optical flow field is the velocity field that represents the three-dimensional motion of points across a series of two-dimensional images. Pre-processing is the optical flow computation based on two assumptions: the intensity of any point in the series is constant over time and nearby points move in a similar manner. After the optical flow computation, the velocity field is established, from which the points of interest can be found, such as those most active ones or those in a group with similar activities. Then the mutual velocity

between observer and interested point, the trend of point and depth information can be extracted to describe the point.

All the feature extraction methods described above are fairly mature technologies for which a researcher in this area can easily find algorithms or even program libraries to implement them for the machine/computer vision task at hand. However, these methods are not flexible because they can only extract primitive features that stick to the default resolution of the image to be processed. In other word, these methods cannot be used for multi-resolution feature extraction, since they are not flexible and sometimes are not reliable. For example, when the Roberts operator is used to detect an edge by correlating the 2x2 adjacent pixel intensities, if the image is with high resolution, the operator may find two edges from a real edge of a box, because that edge may be one pixel wide in low resolution and three pixels wide in high resolution. So, in the past few years, some researchers began using Wavelet Transforms, which is a multi-resolution analysis method, to extract features in image signals. In the next section, the usage of Wavelet Transform in feature extraction will be introduced. In Chapter 2, some efforts in this particular area will be reviewed. A new feature extraction method implementing a novel Anchored Wavelet Transform, which is fast because of its parallel computation and flexible because of its multi-resolution solution, will be introduced in the following chapters.

1.3 Wavelet Transform can be used in Image Feature Extraction

As mentioned before, image feature extraction is an important step in many research and applications of machine/computer vision. These applications use feature extraction for object classification/recognition, image registration and presentation, and for image

compression and communication. Further, feature extraction is a critical step in processing image signals because it is in the very beginning of the image processing hierarchy and the higher-level process will rely on the result from it. However, it is difficult to find reliable and robust features. According to William S. Meisel (1972) [32], one has to consider the following four general requirements in allocating the feature space:

- Low Dimensionality

The features defining the pattern space must be relatively few in number. This is necessary to make the algorithms both meaningful and computationally feasible.

- Sufficient Information

The features used must retain sufficient information to allow the desired operation in subsequent processing. This is application dependent.

- Geometrical Consistency

The features must make different patterns distinguishable. On the other hand, differences that are irrelevant to the quality to be recognized need not be reflected in the feature space.

- Consistency of Features Throughout the Samples

A feature is intended to allow comparison of one unique aspect of different patterns, and must be suited to this purpose.

From the following paragraphs it is shown that the basis wavelet functions for Multi-Resolution Analysis satisfy all the requirements to form the feature space in image analysis.

The main concept in WT analysis is the representation of a function as a superposition of wavelet basis functions. From a single finite duration mother wavelet, $\psi(x)$, a family of orthonormal basis functions can be obtained through dilations and translations.

$$\psi_{m,n}(x) = 2^{-m/2}\psi\left(2^{-m}x - n\right). \qquad 1.1$$

Where m and n are integers. By variation of the integer n, the translation process transforms the mother function into a separate vector space with distinctive spatial orientation. Similarly, by changing m, the dilation process amounts to a transformation of the original mother function into another vector space possessing a different resolution. Naming the vector spaces by $\mathbf{V}_m$'s with m as resolution-level; we have the multi-resolution representations of the wavelet decomposition:

$$\ldots \mathbf{V}_m \subset \mathbf{V}_{m-1} \ldots . \qquad 1.2$$

and

$$f(x) \in \mathbf{V}_0 \Leftrightarrow f\left(2^m x\right) \in \mathbf{V}_m . \qquad 1.3$$

A function then can be decomposed as:

$$f(x) = \ldots + \sum_n c_{m,n}\psi_{m,n}(x) + \sum_n c_{m+1,n}\psi_{m+1,n}(x) + \ldots . \qquad 1.4$$

where $\ldots, \{c_{m,n}\}, \{c_{m+1,n}\} \ldots$ are the wavelet coefficient sets at each resolution level.

The requirements for feature extraction are satisfied when we use these coefficients as the features of the image.

Since the basis functions are orthonormal, any set of coefficients selected as a feature would help to discriminate the image patterns, so 'Geometrical Consistency' is sustained.

The basis functions are dilated and/or translated from the single mother wavelet function, so it is very safe to say 'Consistency of Features throughout the Samples' is satisfied for selected features.

Wavelet decomposition reduces the resolution as well as the size of the image, so the dimensionality is reduced. Furthermore, the wavelet transform is localized, so the features could be localized. This means that when we analyze an image property at a particular position, we can use only a few WT coefficients to represent the property concerned. 'Low Dimensionality' can be the case if we use WT properly as a feature extraction tool.

As for 'Sufficient Information', because we can decompose or reconstruct an image to any resolution level that does not exceed the original level, any image analysis application can find the sufficient level of WT that will give sufficient information.

For image signals with sharp edges and other discontinuities, one problem with Fourier transforms and Fourier synthesis is that in order to accommodate a discontinuity, high frequency terms appear and are not localized, but are added everywhere. WT offers the advantage of viewing the images with global as well as with local perspectives in different scales. Global effects such as illumination and the coarse shapes of the objects can be viewed more clearly in lower scales, while local effects such as edges appear only at higher resolutions. So wavelets and wavelet transforms are a natural advance from classic Fourier techniques in image processing, filtering and reconstruction, and they hold promise in shape and light interpretation.

1.4 Object of This Thesis

As described in Section 1.3, wavelet decomposition is a feasible tool to extract features for image analysis. This thesis seeks to develop a generalized scheme to incorporate the WT into the image property interpretation at each pixel location. The goal of the scheme is to develop a novel WT algorithm for this subject.

There has already been a great amount of effort devoted into this area by researchers with different interests. While other WT applications reduce the resolution of the original signal when undertaking the wavelet decomposition, this thesis searches for an algorithm, the Anchored Wavelet Transform (AWT), for wavelet decomposition that will keep the decomposition resolution unchanged for a particular region of interest. Each pixel in the image would have wavelet representations in any desired decomposition level in this wavelet algorithm. Other wavelet algorithms will inevitably shift the focus position by $\left(2^m-1,2^m-1\right)$ after m levels of decomposition, which makes it very difficult to introduce contextual information into the analysis, while AWT should be a non-shift wavelet decomposition algorithm which will keep focus on the point of interest. The pixel contextual information would be readily available right in the WT coefficient set.

An image feature extraction method is to be designed based on this new algorithm, which will be implemented to track dominant features in stereo images or in image sequences.

1.5 Organization of This Thesis

Research reports and applications on feature extraction by wavelet transform methods in image analysis are discussed in Chapter Two. Chapter Three briefly describes the theory

of Multi-Resolution Analysis and Wavelet Transforms. Chapter Four introduces the concept of Anchored Wavelet Transform based on the theory introduced in Chapter Three. Chapter Five presents examples of AWT in feature extraction. A points-tracking approach is also presented in this chapter. Chapter Six gives the conclusion of this research and suggestions for future work in this area.

Chapter 2 Literature Review

There are two main directions in modern wavelet research: one is to find a set of basis functions and transforms that will give an informative, efficient, and useful description of a function or signal and provide an efficient features localization in both spatial and frequency or scale domains. Another one is multiresolution analysis where the decomposition of a signal is in terms of the resolution of details. This chapter gives an overview of wavelet applications in feature extraction for image processing, which covers the developments in both directions. Section 2.1 introduces the wavelet analysis method involved in a variety of image processing areas, including pattern classification/recognition, image indexing, image registration and segmentation. Section 2.2 focuses on the existing research achievements of WT Multiresolution Analysis in image feature extraction. WT MRA is also the emphasis of this thesis work in the following chapters. Section 2.3 briefly introduces the developments of WT in hardware implementation. In Section 2.4, the core concept of this thesis work is presented.

2.1 Wavelet Analysis in Image Processing

2.1.1 Feature Extraction by WT

Compared to other classical signal processing method, such as Fourier representation, Windowed Fourier Transform (WFT), etc., WT is superior in image feature extraction. Different feature extraction methods were studied by Huynh et al. (1998) [22] for the classification of underwater mammals. Support for the usefulness of the wavelet

representation is provided by comparison with the performance of a Windowed Fourier Transform (WFT) and Fourier representation. The result shows that, although WFT representation is much more appropriate for the signal used by the authors, attributed as a cocktail sound, including biological sounds, mixed with environmental sound and man-made sounds, than a Fourier representation, the wavelet representation has a most profound effect on the classification. Tian et al. (1999) [3] studied and benchmarked the performance of Singular Value Decomposition (SVD), Wavelet Packet (WP) and Gray-Level Co-occurrence Matrix (GLCM) based features for a cloud classification system. The SVD and WP achieved similar performance, while GLCM fell slightly behind.

Lehr and Lii (1996) [18] tried to unify the process of noise reduction, segmentation, feature extraction and classification to develop a general pattern recognition technique using only WT. The research result shows that the WT technique has the advantage of being computationally efficient even though it might not be optimal.

Beside the most widespread application of the wavelet transform in data compression, wavelet analysis techniques in feature extraction is another important area that has been extensively researched for signal and image processing because of its possible advantage over other feature extraction methods.

Solka (1998) [16] detailed the use of low-level features in a particular pattern detection problem, to identify the man-made regions in Unmanned Aerial Vehicle (UAV) imagery. WT was used to extract region boundaries. Deschenes et al. (1998) [30] presented an approach for recognizing 3D vehicles from the 2D IR images at different angles. For the classification, edge detection was used to eliminate the misleading information related to the temperature variations due to vehicle operation. Then edge tracking produced the 1

pixel thick closed contour for each image. The invariant Fourier descriptor vectors were used to describe the contour, and were the inputs to a neural network classifier. The emphasis of the paper is put mainly on the wavelet edge detection which is based on the extraction of local maxima of the wavelet decompositions. Fernandez and Huntsberger (1998) [10] produced a multiresolution edge representation also based on local maxima of the wavelet transform to detect polygons.

Espinal et al. (1998) [9] introduced a novel texture metric that can be used for texture classification, image segmentation or target recognition based on the fractal dimension and the 2D WT. The texture metric is called the fractal signature of the texture, which is the coarseness change in measured area. The authors used either 16x16 or 32x32 windows to screen through the image, and get the first 3 levels of WT decomposition to calculate 9 signatures for the section under the window. The algorithm gives better segmentation and computational efficiency compared to those of Gabor filters.

Mammography is the most reliable method for the early detection for breast cancer. The problem is that the number of mammograms produced and then examined by the radiologists is increasing dramatically. Automatic computer pre-screening programs are a most attractive solution to the problem. Chen and Lee (1997) [6] and Yu and Guan (1999) [29] both introduced WT feature extraction methods to detect the clustered microcalcifications in digitized mammograms. Strickland and Zoucha (1998) [27] developed a similar detection method by subband decomposition.

There is growing interest in applying wavelet features for image indexing as well. Kuo and Liang (1997) [17] introduced a wavelet-based image indexing system for image storage and retrieval over the network. Wavelet domain features used include subband

significance, decomposition structure, luminance and chrominance histograms, and the significance map of the lowest frequency channel.

2.1.2 Image Registration

In many image analysis applications, 'similar' images are compared with one another to detect certain properties of the involved objects. To accomplish this, a proper alignment of the images is required, that is, the transformation to relate points of one image to corresponding points of another image must be found based on some similarity measurement.

Li and Zhou (1996) [14] presented a wavelet-based point feature extraction algorithm to select distinct and consistent landmark features across images to register visual and IR imagery. Because visual and IR data have very different grayscale characteristics, the edge-based point feature extraction was then used to boost landmark-detection performance. The algorithm employs a wavelet-based edge detector to locate contours and select point features from the contour based on local statistical information of intensity.

Olive (1995) [15] and Pan (1996) [13] also presented algorithms for image matching using wavelet analysis.

2.1.3 Segmentation

Segmentation is an important low-level task for image processing, which is to group pixels in an image according to some arbitrary criterion. Rao and Prasad (1995) [24] proposed an image segmentation algorithm based on pixel intensity histograms in different resolution through wavelet analysis. Chen and Lee (1997) [6] employed the multiresolution wavelet and Gaussian Markov Random Field feature extraction scheme as pre-processing before fuzzy-c-means clustering operation.

Most techniques mentioned above tend to focus on a particular level of wavelet decomposition, and to use partial functionality of the WT, either for noise reduction, edge detection, or automatic correlation.

A greater advantage of WT lies in MRA, which addresses the relation between decomposition levels. The full function of WT could be applied in extracting image properties through adequately designed WT MRA. In next section some MRA presentations will be reviewed again in more detail.

2.2 Features from Wavelet MRA

The last three or four years have seen an explosion of MRA techniques in image processing research. Strickland and Zoucha (1998) [27], Hajj et al. (1996) [12] and Zhang et al. (1999) [33] employed filter banks in subband decomposition MRA. WT MRA inherits all filter bank MRA benefits and offers an efficient framework for extracting object or primitive features from images at various scales, which naturally incorporates most advantages of wavelet analysis.

2.2.1 Coarse-to-Fine Strategy

Olive (1995) [15] developed a framework that automatically delivers such a transformation for each pair of 'similar' images. This method uses the multiresolution structure given by the 2D wavelet transform. The method described applies the techniques derived from a block-matching algorithm, which is to seek minimum distance among nearby blocks in the image pair. The distance is measured by the difference of WT details between the image pair. The accomplishment of this presentation is to measure displacement over a large range in low resolution with low accuracy, whilst to measure

displacements over a short range in high resolution with higher accuracy. This coarse-to-fine strategy effectively avoids the risk of getting trapped into a local minimum in matching the 'similar' blocks between images. Pan (1996) [13] presented a complete algorithm for top-down image matching using a complex wavelet pyramid to solve the general stereo image matching problem. The algorithm iterates the propagation from the match result at coarser level to finer level by interpolation. Both image matching works by Olive and Pan use the multiresolution approach to guide the similarity search from coarse to fine. To reflect the coarser level result into the finer level, interpolation has to be used in both cases because of the decimation in the WT decomposition process. The interpolation can be a great computational burden in the match iteration.

Rao and Prasad (1995) [24] proposed an image segmentation scheme based on multiresolution, and successive approximations of the image histogram. The algorithm involves the selection of a threshold from the image histogram of the statistical distribution of intensity levels in the image. The paper deals with a multiresolution technique for analyzing histogram data rather than image data itself. By WT MRA, the algorithm begins with a coarse, initial segmentation of the image, which is obtained by selecting thresholds from a coarse sampling of a low-pass filtered image histogram. This segmentation is refined by selecting thresholds from increasingly better approximations to the histogram, with finer resolution.

2.2.2 Coarse-back-to-Fine: By Reconstruction

Yu et al. (1998) [29] presented an image processing procedure for the automatic detection of clustered microcalcifications in digitized mammograms. The proposed method uses wavelet features and gray level statistical features to identify potential

microcalcification pixels before a formal analysis procedure is carried out based on structure features describing individual microcalcifications. The wavelet features are generated by a four-level wavelet decomposition and reconstruction operation. The reconstruction is done for each level by setting the transform values of other levels to zero. After the operation, there are 4 same size images representing different frequency band distributions of the original image. The gray level statistical features, median contrast and normalized gray level value at each pixel, are also used. The four filter-banded image values at each pixel and two statistical features of each pixel form the feature space. A feed forward neural network is used to produce the likelihood map of calcification corresponding to the original mammogram.

The wavelet decomposition and reconstruction operation utilizing the WT result is a new and effective pixel-feature-extraction scheme proposed by the paper.

2.3 Special Topics on Feature Extraction by WT

Even though the two topics introduced in this section are not closely related with the development of this thesis, they are two important branches in which the WT feature extraction method will develop in the future.

2.3.1 Globalize the Local Features from WT

With the wavelet analysis method, the features can be locally detected. On the other hand, to use these features globally, some special techniques will be needed.

Boles and Boashash (1998) [31] presented a new polar coordinate representation for wavelet analysis to globalize the features for recognizing humans from images of the iris of the eyes. Sakalli and Yan (1998) [20] presented a method for feature-based coding of

human facial images. The proposed face-coding scheme is cascaded in three stages: face content objects location, wavelet decomposition and vector quantization. The initial stage focuses on analyzing the image content in terms of objects, including eyes, nose, mouth, and their spatial relations. This paper gives an exceptional point of view on utilizing wavelet decomposition in image processing. The object localization in the first stage of the proposed scheme compensates for the inevitable weakness of wavelet analysis on the image signal. The wavelet analysis can only capture local features of the image by itself.

2.3.2 Hardware implementation of WT for feature extraction

There are an increasing number of implementations of wavelet feature filters using optical systems. Both Tripathi and Singh (1998) [25] and Soon et al. (1998) [4] demonstrated the use of the Joint Transform Correlator (JTC) by wavelet filters to recognize pre-defined targets. In the VLSI design of WT, several architectures have been proposed for the 2D discrete wavelet transform. The presentation from Yu and Chen (1998) [7] introduced a VLSI implementation for separable two-dimensional Discrete Wavelet Transform (DWT) decomposition. The performance of the proposed architecture can reach N x N clock cycles to compute an N x N 2D DWT. With filter length of L, the resource requirement is only of 2*N*L–2*N, 3L multipliers and 3(L-1) adders.

Obviously, more hardware designs for WT will be expected in the future because it will facilitate the real time application of wavelet analysis.

From the literature survey we can see that wavelet transforms have been widely used to extract features for image analysis. Some research has been done for MRA by wavelet transform. However, only those methods with indirect reference to multilayer information were found in the literature, such as using interpolation functions and wavelet

reconstruction methods or employing coarse-to-fine or coarse-back-to-fine strategies. The more effective feature representation by direct combination of number of levels of WT information has not been studied. There is a need to continue and extend the current research on WT MRA in order to extract and present multi-layer features efficiently for image processing.

2.4 Algorithm Developed in This Thesis

In this thesis the method of feature extraction by wavelet multiresolution analysis is studied. The goal is to develop a new algorithm to form the feature space by combining the wavelet decompositions in different levels or resolutions. The new algorithm will be implemented for stereo image matching, given that the image features involved are not well pre-defined.

Wavelet multiresolution analysis is the subband decomposition that offers the advantage of viewing the image with global as well as with local perspectives in different scales. Global effects such as illumination and the coarse shapes of the objects can be viewed more clearly in lower scales, while local effects appear at higher resolutions.

By wavelet transform, the algorithm developed here will extract features for each point in the 2D image. Each feature is the correlation result between a surrounding window of the point and a dilation of a base wavelet. In lower scales, the windows and the dilation for base wavelet are bigger; in higher scales, the windows and the dilation for base wavelet are smaller. The combination of the wavelet correlation results from different scales, representing the coarse-to-fine looking of the interested point, forms the feature space. The new algorithm, AWT, will generate wavelet analysis results from multiple resolution layer

in only one pass of decomposition, where interpolation or wavelet reconstruction are not necessary any more. In turn, the feature extraction method using AWT will inherit all the advantages of the coarse-to-fine and coarse-back-to-fine MRA methods, but with the form of direct merging of different levels of information. The feature space formed will include dimensions or measures from different resolutions of the original image, which will be readily used by any pattern recognition/classification, image segmentation or registration application.

Chapter 3 Multi-Resolution Analysis and Wavelet Transform

This chapter presents the background theory of Multiresolution Analysis and Wavelet Transform, on which the feature extraction algorithm developed in this thesis is based.

Because both the mathematics and the practical introduction to wavelets are best served by using the concept of resolution that is employed to define the effects of changing scale in signal details, the presentation here goes from MRA to WT.

This chapter thus starts with an intuitive interpretation of the MRA by sequential approximation of a vector, which gives a clear view of the process of MRA. Then the conditions and formulations for approximation of functions in terms of the MRA used in the thesis are deduced and formalized in detail. The scaling function, wavelet and general algorithm in WT are finally introduced.

3.1 Vector Approximation – Intuitive Introduction to MRA

Consider the N-dimensional, real-valued vector space in which the vector can be expressed in the form of $\mathbf{x} = [x_1, x_2, ..., x_N]$. This vector space is N-dimensional linear so that there exist N linearly independent (not necessarily orthogonal) basis vectors $\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_N$, and any vector $\mathbf{x}$ in the space can be represented by a linear combination of these basis vectors. In other words, there is a unique set of scalars $\alpha_1, \alpha_2, ..., \alpha_N$ so that $\mathbf{x} = \alpha_1 \mathbf{a}_1 + \alpha_2 \mathbf{a}_2 + ... + \alpha_N \mathbf{a}_N$. Let us name this N-dimentional vector space $\mathbf{V}_N$.

To approximate vectors in this N-dimensional space by vectors in a subspace with one lower dimension, only N-1 of the N basis vectors, say $\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_{N-1}$, are used to generate all linear combinations. The set of all such linear combinations is also a vector space with the dimension of N-1 and every vector in it also belongs to $\mathbf{V}_N$. This vector space of dimension N-1 is thus one of the *vector subspaces* of $\mathbf{V}_N$, and can be written as $\mathbf{V}_{N-1}$. By continuing to drop one basis vector at each step, sub spaces $\mathbf{V}_{N-2}, ..., \mathbf{V}_1$ with dimension N-2,...,1 can be similarly constructed respectively. The subspace $\mathbf{V}_1$ only has the single vector $\mathbf{a}_1$ as its basis. These vector spaces, $\mathbf{V}_N, \mathbf{V}_{N-1}, ..., \mathbf{V}_1$, form a nested sequence of vector space and subspaces,

$$\mathbf{V}_1 \subset \mathbf{V}_2 \subset ... \subset \mathbf{V}_N \ . \tag{3.1}$$

In case a vector $\mathbf{x}$ in $\mathbf{V}_N$ is going to be approximated by a vector in $\mathbf{V}_{N-1}$, the best approximation in the sense of minimum mean-squared error is made by choosing the vector (say $\mathbf{x}_{N-1}$) in $\mathbf{V}_{N-1}$ for which the length of the error vector

$$\mathbf{e}_{N-1} \equiv \mathbf{x} - \mathbf{x}_{N-1} \tag{3.2}$$

is minimized. The approximation vector $\mathbf{x}_{N-1}$ can thus be obtained by solving equations for $\mathbf{e}_{N-1}$ in

$$\langle \mathbf{e}_{N-1}, \mathbf{a}_k \rangle = 0 \ , \tag{3.3}$$

Where k = 1,2,...,N-1. In a geometric analogy, $\mathbf{x}_{N-1}$ is the vector obtained by dropping a perpendicular base vector from $\mathbf{x}$ to $\mathbf{V}_{N-1}$ as illustrated in Figure 3.1.

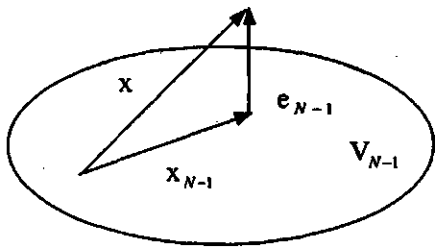


Figure 3.1 A geometric analogy for finding the best approximating vector to x in the vector space .

The vector $\mathbf{x}_{N-1}$ is thus called the orthogonal projection of $\mathbf{x}$ on $\mathbf{V}_{N-1}$. The error vector $\mathbf{e}_{N-1}$ is orthogonal to every vector in $\mathbf{V}_{N-1}$, including the basis vector of $\mathbf{V}_{N-1}$. By continuing the process of projecting $\mathbf{x}_{N-1}$ on $\mathbf{V}_{N-2}$ to yield $\mathbf{x}_{N-2}$, and so on, throughout the entire sequence of subspaces, a sequence of othogonal projections of $\mathbf{x}$ in the subspaces $\mathbf{V}_{N-1}$, $\mathbf{V}_{N-2},\ldots,\mathbf{V}_1$ is obtained. The projections are $\mathbf{x}_{N-1}$, $\mathbf{x}_{N-2},\ldots,\mathbf{x}_1$ repectively. Note that an approximating vector $\mathbf{x}_k$ is the orthogonal projection of not only $\mathbf{x}_{k+1}$ but also of the vector $\mathbf{x}_{k+2},\ldots,\mathbf{x}_{N-1}$ and $\mathbf{x}$ on $\mathbf{V}_k$. With $\mathbf{x}_N = \mathbf{x}$ let

$$\mathbf{e}_k = \mathbf{x}_{k+1} - \mathbf{x}_k \qquad 3.4$$

for k=1,...,N-1, denote the error between the projections on successive subspaces. The sub-sequential projections from $\mathbf{x}$ are approximations to the original vector and the quality of these approximations either stay the same or get worse going from approximation vector $\mathbf{x}_{N-1}$ to $\mathbf{x}_1$. Another way of looking at this sequential approximation is that the subspace $\mathbf{V}_{N-1}$ contains the finest approximations to vectors in $\mathbf{V}_N$ whereas $\mathbf{V}_1$ contains the coarsest approximations.

The error vector $\mathbf{e}_{N-1}$ can be interpreted as the amount of "detail" of x that is lost in projecting to its approximation $\mathbf{x}_{N-1}$ because of

$$\mathbf{x} - \mathbf{e}_{N-1} = \mathbf{x}_{N-1} \qquad 3.5$$

The detail vector $\mathbf{e}_{N-1}$ belongs to a vector subspace of $\mathbf{V}_N$, and all vectors in this subspace are orthogonal to $\mathbf{V}_{N-1}$. Let us call this vector subspace $\mathbf{W}_{N-1}$. The dimension of $\mathbf{W}_{N-1}$ is one in the current situation, and $\mathbf{V}_{N-1}$ and $\mathbf{W}_{N-1}$ are orthogonal to each other. Furthermore, since any vector in $\mathbf{V}_N$ can be expressed as the combination of a vector in $\mathbf{V}_{N-1}$ plus a vector in $\mathbf{W}_{N-1}$, subspaces of $\mathbf{V}_{N-1}$ and $\mathbf{W}_{N-1}$ are the orthogonal complements of each other in $\mathbf{V}_N$. Similar relations can be deduced between $\mathbf{V}_{N-2}$ and $\mathbf{W}_{N-2}$, and so on. In general, the error vector between any successive levels can be seen as the detail that is lost in approximating a vector in a finer level with a vector in a coarser level. On the other hand, the error vector can be seen as the detail that can be added to the vector approximation in a coarser level to make up the vector being approximated in the finer level as well.

The vectors $\mathbf{e}_{N-1}, \mathbf{e}_{N-2}, \ldots, \mathbf{e}_1, \mathbf{x}_1$ form an orthogonal vector set, that is, the inner product of any pair of these vectors is zero. They come from the one-dimensional subspaces $\mathbf{W}_{N-1}, \mathbf{W}_{N-2}, \ldots, \mathbf{W}_1$ and $\mathbf{V}_1$ respectively. Moreover,

$$\mathbf{x} = \mathbf{e}_{N-1} + \mathbf{e}_{N-2} + \ldots + \mathbf{e}_1 + \mathbf{x}_1 . \qquad 3.6$$

Thus the original vector can be reconstructed from the coarsest approximation vector $\mathbf{x}_1$ from $\mathbf{V}_1$ and the vectors from the various detail levels. Each of these detail vectors is an

orthogonal projection of $\mathbf{x}$ on the corresponding one-dimensional subspace $\mathbf{W}_i$ where i = 1,...,N-1. Then the vector space $\mathbf{V}_N$ is said to be the *direct sum* of these subspaces $\mathbf{W}_{N-1}, \mathbf{W}_{N-2}, ..., \mathbf{W}_1$ and $\mathbf{V}_1$, which is written as:

$$\mathbf{V}_N = \mathbf{W}_{N-1} \oplus \mathbf{W}_{N-2} \oplus ..., \mathbf{W}_1 \oplus \mathbf{V}_1, \tag{3.7}$$

or in another form as:

$$\mathbf{V}_N = \mathbf{W}_{N-1} \oplus \mathbf{V}_{N-1}, \mathbf{V}_{N-1} = \mathbf{W}_{N-2} \oplus \mathbf{V}_{N-2}, ..., \mathbf{V}_2 = \mathbf{W}_1 \oplus \mathbf{V}_1 . \tag{3.8}$$

From the development shown above, it is also clear that any vector in $\mathbf{W}_k$, including basis vectors of $\mathbf{W}_k$, and any vector in $\mathbf{V}_k$, including basis vectors of $\mathbf{V}_k$ can be expressed as a linear combination of basis vectors of $\mathbf{V}_{k+1}$:

$$\mathbf{e}_k = \alpha_1 \mathbf{a}_1 + \alpha_2 \mathbf{a}_2 + ,..., + \alpha_{k+1} \mathbf{a}_{k+1}, \quad \mathbf{e}_k \in \mathbf{W}_k , \tag{3.9}$$

$$\mathbf{x}_k = \beta_1 \mathbf{a}_1 + \beta_2 \mathbf{a}_2 + ,..., + \beta_{k+1} \mathbf{a}_{k+1}, \quad \mathbf{x}_k \in \mathbf{V}_k , \tag{3.10}$$

where k=N-1,...,1, $\alpha_1, \alpha_2, ..., \alpha_{k+1}$ and $\beta_1, \beta_2, ..., \beta_{k+1}$ are unique scalars sets.

The nesting of vector spaces represented by Formula 3.1, the act of breaking down a vector into an hierarchy of approximations by Equations 3.8 and 3.9, and the relations between details and approximations expressed be Equation 3.9 and 3.10, form the core of the MRA used in this thesis.

It is attractive to claim the details filtered out from MRA as features of the vector. In other words, $\mathbf{W}_{N-1} \oplus \mathbf{W}_{N-2} \oplus ... \mathbf{W}_k$ forms the feature space to classify the vector in $\mathbf{V}_N$ as long as k is small enough so that enough features of the sample vectors are taken in order to separate the vector clusters. The following sections will demonstrate that the wavelet

transform generates a natural setting for similar MRA in approximating functions with finite-energy, such as image signals. The function features in various levels will be screened out as MRA going to the coarse level.

3.2 MRA, Scaling Function and Wavelet

In Section 3.1, an N-dimensional, real-valued vector space is used to intuitively illustrate the vector orthogonal approximation and reconstruction, which are the central ideas of MRA. In this section, an MRA that involves approximation of functions in a sequence of nested linear vector spaces will be formally defined and constructed. The goal is to construct the nested linear vector space to which any function with finite energy can be projected; the error vectors, or the details during the projection, form the features of the function.

3.2.1 Formal definition of an MRA

Not every sequence of nested vector spaces yields an MRA arena. In addition to the nesting property, the vector spaces need to satisfy other conditions and relations stated in Section 3.1. The formal definition of the MRA used in this thesis is the following (Burrus et al. 1998 [8]): An MRA consists of the nested linear vector spaces $...\mathbf{V}_{-1} \subset \mathbf{V}_0 \subset \mathbf{V}_{+1}...$ such that

1. The union of these subspaces is *dense* in the space of square integrable functions, $L^2(R)$
2. The intersection of these subspaces is a singleton set containing a zero function or zero vector

3. If $f(t) \in \mathbf{V}_k$ then $f(2t) \in \mathbf{V}_{k-1}$, and vice versa

4. There exists a function (called a *scaling function*) $\phi(t)$ such that $\{\phi(t-k): k$ integer$\}$ is a basis for $\mathbf{V}_0$.

The first requirement mentions one set of space being dense in another space. A simple explanation to the concept of denseness is to consider the relationship of two sets of real number A and B, where $A \subset B$. The set A is said to be dense in B if for every element $b \in$ B an element in A can be found that is as close to b as needed. One straightforward example for the explanation is to take the set of rational numbers as A and the set of real numbers as B. The set A is dense in B since for every real number x in set B, it is possible to find a rational number in set A that is as close to x as desired. Stated another way with the sense of multi-resolution, it is possible to find a sequence of rational numbers q_1, $q_2, \ldots, q_n$, such that as $n \to \infty$ we have $q_n \to x$ and q_n is a finer approximation to x than q_{n-1}. A classic instance of this example is provided in the definition of e, the base of the natural logarithm, as $e = \lim_{n \to \infty} \left(1 + \frac{1}{n}\right)^n$, where the limit is taken as n steps through positive integer values. The fact that A is dense in B lets us make approximations to irrational numbers with rational numbers in practical situations, such as numeric computation. Thus the first requirement of the MRA definition implies that the union of the nested linear vector spaces yields a space, $\mathbf{V}_{-\infty}$, that is not necessarily the same as the space of signals with finite energy, $\mathbf{L}^2(\mathbf{R})$, but is dense in it. In other words, for every signal $f(t) \in \mathbf{L}^2(\mathbf{R})$, there is a signal in $\mathbf{V}_{-\infty}$ that is as close to $f(t)$ as needed.

The second requirement states that the only signal common to all the vector spaces is the all-zero signal or the zero vector. This requirement combined with that of being dense in $\mathbf{L}^2(\mathbf{R})$, gives an extended formula of the nested linear vector spaces:

$$\{0\} \leftarrow,\ldots,\mathbf{V}_{-1} \subset \mathbf{V}_0 \subset \mathbf{V}_{+1},\ldots,\rightarrow \mathbf{L}^2(\mathbf{R}) . \qquad 3.11$$

In feature extraction, the formula 3.11 guarantees that there are enough projections or approximations that can be made so that the desired classification is ensured. In other words, according to the development in Section 3.1, the formula 3.11 promises that a proper k can be found for $\mathbf{W}_{N-1} \oplus \mathbf{W}_{N-2} \oplus \ldots \mathbf{W}_k$ to be a valid feature space. Anyway, in image processing, $\mathbf{V}_0$ is the start point of the sequential projection, where the resolution is decided by the image sampling resolution. The value of k is also confined by the size of the image.

The third requirement brings in dilation and states that a factor-of-two dilation of a vector belonging to a subspace at a certain level yields a vector in the next coarser subspace. Conversely, dilating a factor of one half yields a function in the next finer subspace.

The final requirement requires that there be a scaling function $\phi(t)$ in the basis functions such that the set $\{\phi(t-n): n \text{ integer}\}$ is linearly independent, and any function $f_0(t) \in \mathbf{V}_0$ is expressible as:

$$f_0(t) = \sum_{n=-\infty}^{+\infty} a(0,n)\phi(t-n) , \qquad 3.12$$

for a sequence of scalars $a(0,n)$ where $n = 0,\pm1,\pm2,\ldots$.

In general, this section defines the MRA as

- Consisting of nested linear vector spaces to which $f(t) \in \mathbf{L}^2(\mathbf{R})$ can be projected sequentially
- Function going to next coarser level by a factor-of-two dilation
- There is one basis function in $\mathbf{V}_0$.

3.2.2 Construction of the Orthogonal MRA from Scaling Function

Let $\phi(t)$ be a function that satisfies the following:

1. It integrates to one:

$$\int_{-\infty}^{\infty} \phi(t)dt = 1 \,. \tag{3.13}$$

2. It has unit energy:

$$\|\phi(t)\|^2 = \int_{-\infty}^{\infty} |\phi(t)|^2 dt = 1 \,. \tag{3.14}$$

3. The set consisting of $\phi(t)$ and its integer translates is orthogonal:

$$\langle \phi(t), \phi(t-n) \rangle = \delta(n) \,. \tag{3.15}$$

4. There is a sequence of scalars $h(n)$ where $n = 0, \pm 1, \pm 2, \ldots$, such that

$$\phi(t) = \sum_{n=-\infty}^{\infty} h(n)\sqrt{2}\phi(2t-n) \,. \tag{3.16}$$

The function $\phi(t)$ defined here is called the Scaling Function. The $\sqrt{2}$ term in Equation 3.16 maintains the unity norm of the scaling function.

The set $\{\phi(t-k)$:k integer$\}$ consists of linearly independent vectors because the scaling function and its translations is orthogonal. Therefore, it serves as a basis for a linear vector space, $\mathbf{V}_0$. Substituting $2^k t$ for t in Equation 3.15, we then have

$$\langle \phi(2^k t), \phi(2^k t - n) \rangle = 2^{-k} \delta(n) . \qquad 3.17$$

Equation 3.17 indicates that the set $\{\phi(2^k t - l)$:l integer$\}$ is also an orthogonal set for a given integer k, which in turn is the basis for the linear vector space, $\mathbf{V}_k$. That is, the subspaces $...\mathbf{V}_1, \mathbf{V}_0, \mathbf{V}_{-1}, ...$ and so on, satisfy $f(t) \in \mathbf{V}_k \Leftrightarrow f(2t) \in \mathbf{V}_{k-1}$. Because the basis vector set for $\mathbf{V}_{k-1}$, $\{\phi(2^{k-1} t - l)$:l integer$\}$ can be reconstructed by linear combinations from the basis vector set for $\mathbf{V}_k$, $\{\phi(2^k t - l)$:l integer$\}$, $\mathbf{V}_{k-1} \subset \mathbf{V}_k$ is obtained, which is equivalent to a nesting of all the subspaces involved. $k = 0, \pm 1, \pm 2, ...$, and so on, so the union of the subspaces is dense in $\mathbf{L}^2(\mathbf{R})$. The notation for the basis set of $\mathbf{V}_k$, $\{\phi(2^k t - l)$:l integer$\}$, can be simplified as $\phi_{k,l}(t)$, where $k = 0, \pm 1, \pm 2, ...$, represents the dilation and $l = 0, \pm 1, \pm 2, ...$, represents the translation.

Now the MRA is complete because all four properties listed above are satisfied, given the definition of the scaling function. That is, the initial scaling function $\phi(t)$ and its dilation equation are sufficient to form an MRA. To finish the MRA process similar to that described in Section 3.1, the detail (denoted by **e** in Section 3.1) lost when $f(t) \in \mathbf{V}_k$ is projected into a coarser level in $\mathbf{V}_{k-1}$ needs to be found, because **e** is the feature of the vector sought according to Section 3.1. The wavelet serves to measure **e**.

3.2.3 Wavelet

Wavelets are the set of functions, $\{\varphi(2^k t - l): k, l \text{ integer}\}$ or $\varphi_{k,l}(t)$, dilated and translated from the mother wavelet $\varphi(t)$ or noted as $\varphi_{0,0}(t)$. This set of wavelets is expected to be dense to the space of differences between spaces based on various scales of the scaling function. In turn, Since the set resides in the space spanned by the next narrower scaling function, $\mathbf{W}_0 \in \mathbf{V}_1$, they can be represented by linear combination of the scaling function set $\phi(2t)$ (Burrus et al. 1998 [8]):

$$\varphi(t) = \sum_n h_1(n)\sqrt{2}\phi(2t - n), n \in \mathbf{Z}. \qquad 3.18$$

The 'span' used in the last statement means that if a set of vectors is the basis of a space, then the set spans the space. Therefore, Theorem 17 in (Burrus et al. 1998 [8]) is quoted to declare the necessary and sufficient conditions for the existence of such wavelets:

If the scaling coefficients $h(n)$ in 3.16 satisfy the conditions for existence and orthogonality of the scaling function and the wavelet is defined by equation 3.18, then the integer translates of this wavelet span $\mathbf{W}_0$, the orthogonal complement of $\mathbf{V}_0$, both being in $\mathbf{V}_1$, i.e., the wavelet is orthogonal to the scaling function at the same scale,

$$\int \phi(t - n)\varphi(t - m)dt = 0, \qquad 3.19$$

if and only if the coefficients $h_1(n)$ are given by

$$h_1(n) = \pm(-1)^n h(N - n), \qquad 3.20$$

where N is an arbitrary odd integer chosen to conveniently position $h_1(n)$.

Because $\mathbf{V}_1$ can be any space in the nested space list as long as it is the initial space from which the projection is started, in other words, any space can be deemed as $\mathbf{V}_1$, the properties of function projection stated in the above theorem can be generalized as:

$$\mathbf{V}_k = \mathbf{V}_{k-1} \oplus \mathbf{W}_{k-1}, \tag{3.21}$$

$$\mathbf{V}_{k-1} \perp \mathbf{W}_{k-1}, \tag{3.22}$$

given the relation between scaling function coefficients and wavelet coefficients as in Equation 3.23, and the scaling function defined in Section 3.2.2. In practice, these two assumptions are the starting points to design the wavelet transform (which is not included in this thesis), and the orthogonal wavelets used in this thesis are all designed satisfying these assumptions. Another important property of orthogonal wavelets is:

$$\langle \varphi(t), \varphi(t-n) \rangle = \delta(n) . \tag{3.23}$$

Now, a function projection or approximation scheme that is similar to that described in Section 3.1 is obtained:

$$e_k(t) = \sum_n \alpha_{k,n} \varphi_{k,n}(t), \quad e_k(t) \in \mathbf{W}_k , \tag{3.24}$$

$$f_k(t) = \sum_n \beta_{k,n} \phi_{k,n}(t), \quad f_k(t) \in \mathbf{V}_k , \tag{3.25}$$

$$\mathbf{V}_0 = \mathbf{V}_{-1} \oplus \mathbf{W}_{-1}, \mathbf{V}_{-1} = \mathbf{V}_{-2} \oplus \mathbf{W}_{-2}, \ldots, \mathbf{V}_{-M+1} = \mathbf{V}_{-M} \oplus \mathbf{W}_{-M} \tag{3.26}$$

$$\begin{aligned} &\mathbf{V}_0 = \mathbf{W}_{-1} \oplus \mathbf{W}_{-2} \oplus, \ldots, \oplus \mathbf{W}_{-M} \oplus \mathbf{V}_{-M} \\ &\Rightarrow f_0(t) = e_{-1}(t) + e_{-2}(t) + \ldots + e_{-M}(t) + f_{-M-1}(t) , \end{aligned} \tag{3.27}$$

where k and n are integer, and $-M \le k \le -1$.

If the scaling function is well-behaved, then at a high scale, the scaling is similar to a Dirac delta function and the initial projection is just to sample the function. In other words, at high enough resolution, samples of the signal are very close to the scaling coefficients, $\beta_{0,n}$. In practice one is only given the samples of a function $f_0(t)$, so $\mathbf{V}_0$ in the above formulas represents the highest resolution space.

If $\mathbf{W}_{-1}, \mathbf{W}_{-2}, \ldots, \mathbf{W}_{-M}, \mathbf{V}_{-M}$ form the complete function space that contains $f_0(t)$, then the basis of that subspace, $\phi_{-M,l}(t)$ and $\varphi_{k,l}(t)$: $-M \le k \le -1$, form the perfect feature space to describe $f_0(t)$. The scalar sets α, β are extracted feature measurements along with the $f_0(t)$ approximated by the coarser and coarser $f_k(t)$. For example, the $\beta_{k,l}$ feature measurement for $f_0(t)$ on the $\varphi_{k,l}(t)$ base, can be said to be in the direction of the vector $\varphi_{k,l}(t)$. These statements are the theoretical basis for the feature extraction algorithm developed in this thesis.

3.3 General WT Algorithm

In the function projection process, the scaling functions and wavelets are not dealt with directly, only the coefficients α, β need to be considered, because they are the function features wanted. Indeed, most WT applications concentrate only on α, β.

In order to work directly with the WT coefficients, the relationships between the function approximation coefficients at a lower scale level in terms of those at higher scale will be derived.

Substituting $2^j t-k$ for t in Equations 3.16 and 3.21, $\phi_{j,k}(t)$ and $\varphi_{j,k}(t)$ can be obtained by:

$$\begin{aligned}\phi_{j,k}(t)=\phi(2^j t-k)&=\sum_{n=-\infty}^{\infty}h(n)\sqrt{2}\phi(2(2^j t-k)-n)\\&=\sum_{n=-\infty}^{\infty}h(n)\sqrt{2}\phi(2^{j+1}t-2k-n)\end{aligned}, \qquad 3.28$$

$$\begin{aligned}\varphi_{j,k}(t)=\varphi(2^j t-k)&=\sum_{n=-\infty}^{\infty}h_1(n)\sqrt{2}\phi(2(2^j t-k)-n)\\&=\sum_{n=-\infty}^{\infty}h_1(n)\sqrt{2}\phi(2^{j+1}t-2k-n)\end{aligned}. \qquad 3.29$$

Changing variables $2k+n$ to m, Equations 3.28 and 3.29 become:

$$\phi_{j,k}(t)=\phi(2^j t-k)=\sum_{m=-\infty}^{\infty}h(m-2k)\sqrt{2}\phi(2^{j+1}t-m), \qquad 3.30$$

$$\varphi_{j,k}(t)=\varphi(2^j t-k)=\sum_{m=-\infty}^{\infty}h_1(m-2k)\sqrt{2}\phi(2^{j+1}t-m). \qquad 3.31$$

According to Equations 3.24, 3.25 and 3.27, function projection on the j+1 scale level can be expressed as:

$$\begin{aligned}f_{j+1}(t)&=\sum_{k=-\infty}^{\infty}\beta_{j+1,k}\phi(2^{j+1}t-k)\\&=\sum_{k=-\infty}^{\infty}\beta_{j,k}\phi(2^j t-k)+\sum_{k=-\infty}^{\infty}\alpha_{j,k}\varphi(2^j t-k)\end{aligned}. \qquad 3.32$$

Because $\phi_{j,k}(t)$ are orthogonal to each other according to Equation 3.17, and $\phi_{j,k}(t)$ are orthogonal to $\varphi_{j,k}(t)$ as well according to Equation 3.22, $\beta_{j,k}$ and $\alpha_{j,k}$ can be found by taking the inner product:

$$\beta_{j,k} = \langle f_{j+1}(t), \phi_{j,k}(t) \rangle = \int f_{j+1}(t)\phi(2^j t - k)dt \ , \qquad 3.33$$

$$\alpha_{j,k} = \langle f_{j+1}(t), \varphi_{j,k}(t) \rangle = \int f_{j+1}(t)\varphi(2^j t - k)dt \ . \qquad 3.34$$

By using Equation 3.30 to Equations 3.33, and 3.31 to 3.34, and interchanging the sum and integral, 3.33 and 3.34 become:

$$\beta_{j,k} = \sum_{m=-\infty}^{\infty} h(m-2k)\sqrt{2}\int f_{j+1}(t)\phi(2^{j+1}t - m)dt = \sum_{m=-\infty}^{\infty} h(m-2k)\sqrt{2}\beta_{j+1,k} \ , \qquad 3.35$$

$$\alpha_{j,k} = \sum_{m=-\infty}^{\infty} h_1(m-2k)\sqrt{2}\int f_{j+1}(t)\phi(2^{j+1}t - m)dt = \sum_{m=-\infty}^{\infty} h_1(m-2k)\sqrt{2}\beta_{j+1,k} \ . \qquad 3.36$$

Equations 3.35 and 3.36 are the general wavelet transform algorithm used in most applications including those for feature extraction introduced in the literature review. By Equations 3.35 and 3.36, the original sampled function is decomposed into serial detail functions and a coarsest approximation function.

This chapter briefly described the MRA and WT by which a finite-energy function can be projected into the space with a different scale level or resolution. The wavelet coefficients are the features extracted for the function at the corresponding level. The general WT algorithm implementing the projection is used to calculate scale coefficients $\beta_{j,k}$ and wavelet coefficients $\alpha_{j,k}$ in the coarser level from the next finer level scale coefficients $\beta_{j+1,k}$. The content of this chapter forms the theoretical basis of this thesis.

Chapter 4 Feature Extraction by Anchored Wavelet Transform

In Chapter 3, the feature extraction system based on Multiresolution Analysis and Wavelet Transform is described. The scalar sets α, β are extracted feature measurements while the function $f(t)$ is projected to the coarser and coarser level. The widely used WT algorithm is used to calculate scale coefficients $\beta_{j,k}$ and wavelet coefficients $\alpha_{j,k}$ in coarser scaling levels from the previous finer level scale coefficients $\beta_{j+1,k}$, and the projection process starts from the sampled version of $f(t)$, which is the starting point in all applications of digital signal processing.

This chapter focuses on developing a novel algorithm within the framework described in Chapter 3, to extract features at, or more precisely, at and around, each specific sample point, $f_0(p)$, of the function, where p is the index to the initial samples that the algorithm starts with. Section 4.1 will show that the general WT algorithm is not sufficient to extract features at and around each sample point, while Section 4.2 will develop an Anchored WT algorithm which is capable to do so. Section 4.3 will interpret the features extracted by the AWT physically.

4.1 A Close Look at the Common WT Algorithm

According to Equations 3.35 and 3.36 in Chapter 3, the common WT algorithm is used to calculate the scaling coefficients and wavelet coefficients from scaling coefficients at a previous finer scaling level:

$$\beta_{j-1,k} = \sum_{m=-\infty}^{\infty} h(m-2k)\beta_{j,m}, \tag{4.1}$$

$$\alpha_{j-1,k} = \sum_{m=-\infty}^{\infty} h_1(m-2k)\beta_{j,m}. \tag{4.2}$$

The implementation of Equation 4.1 and 4.2 is illustrated in Figure 4.1 where the down-pointing arrows denote a decimation or down-sampling by two and the other boxes denote a convolution by $h(-n)$ or $h_1(-n)$. In Figure 4.1, the scaling coefficients $\beta_{j-1,k}$ and wavelet coefficients $\alpha_{j-1,k}$ at the coarser level are obtained from finer level scaling coefficients $\beta_{j,m}$, and the even coarser level $\beta_{j-2,k}$ and $\alpha_{j-2,k}$ are obtained from $\beta_{j-1,m}$. Repeating this decomposition process, the coarsest level can be reached, when all the function feature measurements α are extracted.

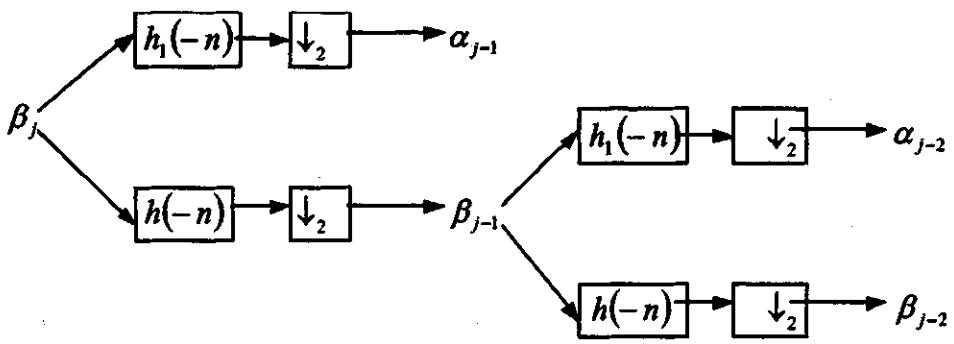


Figure 4.1 Wavelet Decomposition.

The decimation or down-sampling means that the sample number of $\beta_{j-1,k}$ in the coarser level is half of the number of $\beta_{j,m}$, and the same rate holds for $\alpha_{j-1,k}$. This is because the convolution between $h(-n)$ (or $h_1(-n)$) and $\beta_{j,m}$ starts from every other sample position in the $\beta_{j,m}$ series with a two-sample interval. For example, assuming the initial function sampling starts from index one, $h(n)\{_{=0;n<-1 \text{ or } n>1}^{\neq 0;-1\le n\le 1}$ and $h_1(n)\{_{=0;n<-1 \text{ or } n>1}^{\neq 0;-1\le n\le 1}$, the function projection on $\varphi_{j-1,1}(t)$ at level j-1 is done by the convolution starting from $\beta_{j,1}$ at level j, which yields $\alpha_{j-1,1}$, then function projection on $\varphi_{j-1,2}(t)$ is done by the convolution starting from $\beta_{j,3}$ to yield $\alpha_{j-1,2}$. This example is illustrated in Figure 4.2 for better understanding.

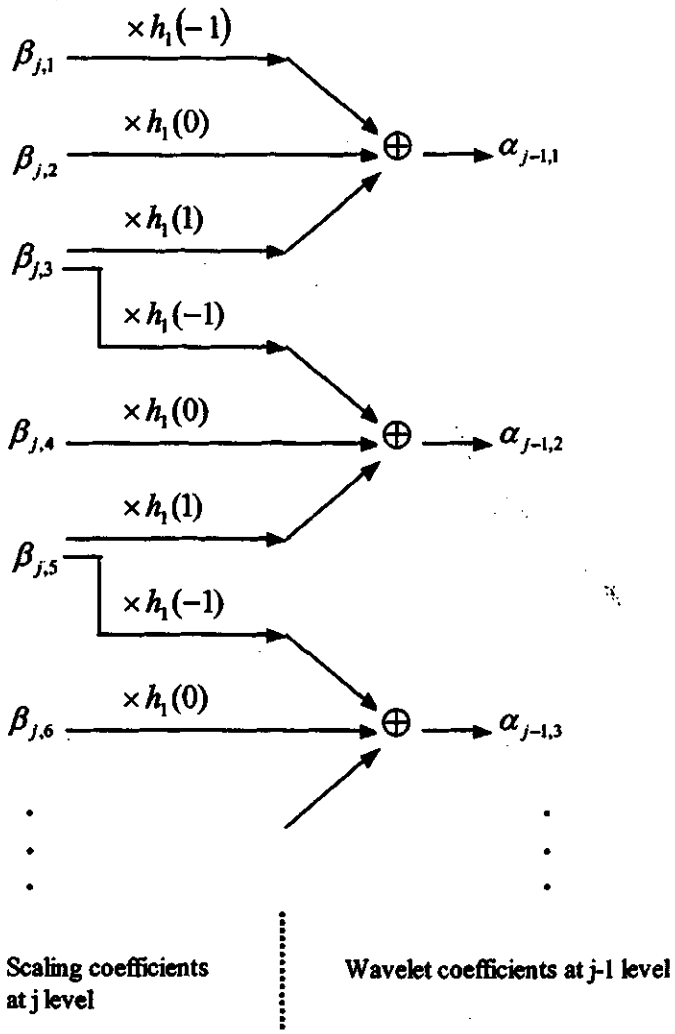


Figure 4.2 Illustration for down-sampling.

Clearly, the common WT algorithm is not sufficient to extract features at and around each sample point, because the down-sampling reduces the feature numbers at the coarser level by half. After two decompositions of a function, the function approximation can be expressed as

$$\begin{aligned} f_0(t) &= \sum_{n=1}^{L} \beta_{0,n}\phi(2^0 t - n) \\ &= \sum_{n=1}^{L/2} \alpha_{-1,n}\varphi(2^{-1}t - n) + \sum_{n=1}^{L/4} \alpha_{-2,n}\varphi(2^{-2}t - n) + \sum_{n=1}^{L/4} \beta_{-2,n}\phi(2^{-2}t - n) \end{aligned} \quad 4.3$$

where $f_0(t)$ is the approximation to the function $f(t)$ that is sampled by a delta function, and L is assumed to be the length of the initial sample space, which is an integer multiple of four. If the coefficients are indexed by initial indexing, we have

$$f_0(t)=\sum_{n=1}^{L}\beta_{0,n}\phi(2^0 t-n)$$
$$=\sum_{n=1}^{L/2}\alpha_{-1,n*2}\varphi(2^{-1}(t-n*2))+\sum_{n=1}^{L/4}\alpha_{-2,n*4}\varphi(2^{-2}(t-n*4))+\sum_{n=1}^{L/4}\beta_{-2,n*4}\phi(2^{-2}(t-n*4)) \quad . 4.4$$

The coefficient sets obtained are

Level 0:	$\beta_{0,1}$,	$\beta_{0,2}$,	$\beta_{0,3}$,	$\beta_{0,4}$,	$\beta_{0,5}$,	$\beta_{0,6}$,	$\beta_{0,7}$,	$\beta_{0,8}$, ...
Level 1:		$\alpha_{-1,2}$		$\alpha_{-1,4}$		$\alpha_{-1,6}$		$\alpha_{-1,8}$ ...
Level 2:				$\alpha_{-2,4}$				$\alpha_{-2,8}$ ...
				$\beta_{-2,4}$:				$\beta_{-2,8}$: ...

In the above illustration, even though the function features can be found at initial sample location $k=4,8,...$ as $\{\alpha_{-1,4},\alpha_{-2,4},\beta_{-2,4}\}$, $\{\alpha_{-1,8},\alpha_{-2,8},\beta_{-2,8}\}$,..., the features at location $k=1,2,3,5,6,7...$ are not available because of the decimation.

There are two general solutions that can be found in current research efforts of others who intend to extract point features of the function by WT. One is to interpolate the error function at the coarser level; the other one is to reconstruct the initial sample from each level of wavelet coefficients.

According to Equation 3.2.3.7, the error function at scaling level *j* can be expressed as

$$e_j(t)=\sum_n \alpha_{j,n}2^{j/2}\varphi_{j,n}(t),\ \ e_j(t)\in \mathbf{W}_j. \qquad 4.5$$

The approximation error of the original function (not the initial sampling to the function) at the initial sampling time spot T for the *jth* projection is

$$e_{j,T} = e_j(T) = \sum_n \alpha_{j,n} 2^{j/2} \varphi(2^j T - n). \qquad 4.6$$

This is the interpolation to the function approximation error at the j level from the wavelet coefficients. The interpolation to the coarsest projection can be done in the same way. Then after interpolating to each projection error and the coarsest projection, the following series can be obtained: $e_{-1,0}, e_{-1,1}, \ldots, e_{-1,L}$; $e_{-2,0}, e_{-2,1}, \ldots, e_{-2,L}$; ...; $f_{-M,0}, f_{-M,1}, \ldots, f_{-M,L}$, where $f_{-M,k}$ are the interpolation results from the coarsest approximation by Equation 3.2.3.8. The features extracted for location T are $\{e_{-1,T}, e_{-2,T}, \ldots, e_{-M,T}, f_{-M,T}\}$. Because the closed form expression for $\phi(t)$ and $\varphi(t)$ may not be available in that the wavelet design focus is normally on $h(-n)$ and $h_1(-n)$, and even though $\phi(t)$ and $\varphi(t)$ are available, the complexity of the interpolation is overwhelming, so the interpolation solution is not a good choice.

Another general solution is to reconstruct initial samples from wavelet coefficients at an individual coarser level and from the scaling coefficients at the coarsest level. Reconstruction is the reverse process to decomposition, which is to recover the scaling coefficients at a finer level from scaling coefficients and wavelet coefficients at the next coarser level. The reconstruction can be expressed by:

$$\beta_{j+1,k} = \sum_{m=-\infty}^{\infty} h^*(k-2m)\beta_{j,m} + \sum_{m=-\infty}^{\infty} h^*_1(k-2m)\alpha_{j,m}. \qquad 4.7$$

$h^*(n)$ is $h(n)$ used in decomposition and $h_1^*(n)$ is $h_1(n)$, in the case of the orthogonal wavelet. The reason for this relation can be found in any wavelet text, which is out of the scope of this thesis.

Figure 4.3 illustrates the reconstruction process that recovers the initial samples from which the decomposition is started.

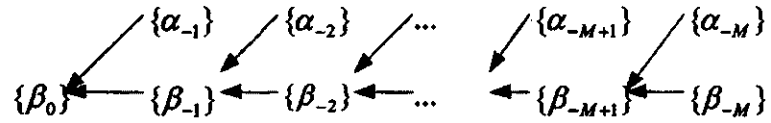


Figure 4.3 Reconstruction of initial samples

In order to extract point features of the function, one can repeat the reconstruction that only includes coefficients at one level by setting all others to zero, as implemented in Figure 4.4.

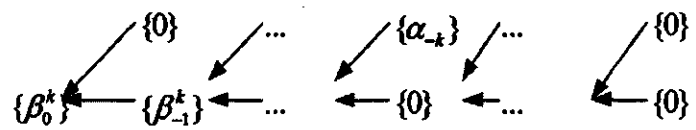


Figure 4.4 Reconstruction from individual level

$\{\beta_0^k\}$ reflects the effects of k^{th} level wavelet coefficients on the original samples., where the subscript '0' stands for the reconstruction targeted at the original sample level, and the superscript 'k' indicates that the reconstruction is from level k wavelet coefficients only. After the repetition of this reconstruction, the following series can be obtained: $\beta^{-1}{}_{0,0}, \beta^{-1}{}_{0,1}, ..., \beta^{-1}{}_{0,L}$; $\beta^{-2}{}_{0,0}, \beta^{-2}{}_{0,1}, ..., \beta^{-2}{}_{0,L}$; ...; $\beta^{-M}{}_{0,0}, \beta^{-M}{}_{0,1}, ..., \beta^{-M}{}_{0,L}$. From these series, the point features of the function are extracted: $\{\beta^{-1}{}_{0,T}, \beta^{-2}{}_{0,T}, ..., \beta^{-M}{}_{0,T}\}$, where T is the point location.

Besides these two general solutions for extracting point features of the function by WT, which use indirect methods, interpolation and reconstruction, to calculate function

approximation at each original sampling time spot in each scaling level, a novel direct point feature extraction algorithm, AWT, will be deduced in the next section.

4.2 AWT—Extension to WT Algorithm

In order to make up the decimation of the scaling and wavelet coefficients, the new relationships between the function approximation coefficients at a lower scale level in terms of those at a higher scale have to be derived. Before the derivation is started, the description of 'index' needs to be clarified. The index is called 'level j index' if it is dilated by 2^j along the time axis. For example, let $\phi_{j,k} = \phi(2^j t - k) = \phi(2^j(t - 2^{-j}k))$, then k is called the j^{th} level index because k is the dilation result of 2^j from the initial index $2^{-j}k$ along the time axis, where j is the scaling level. On the other hand, let $\phi_{j,l} = \phi(2^j(t-l))$, then l is called the initial sample level index, because l is the location measurement along the time axis without any dilation.

In the following sections, all the indices are the initial sample level index without dilation, except as declared separately.

According to Equation 3.16, and Equation 3.18, if we assume the starting level is the initial sampling level, V_0, then the scaling function and wavelet function at the coarser level next to it are:

$$\phi_{-1,k}(t) = \phi_{-1,2l}(t) = \phi(2^{-1}(t-2l)) = \sum_{n=-\infty}^{\infty} h(n)\sqrt{2}\phi(2^0(t-2l-2^0 n)), \quad 4.8$$

$$\varphi_{-1,k}(t) = \varphi_{-1,2l}(t) = \varphi(2^{-1}(t-2l)) = \sum_{n=-\infty}^{\infty} h_1(n)\sqrt{2}\phi(2^0(t-2l-2^0 n)), \quad 4.9$$

where k is level –1 index (even though l is level –1 index as well, but $2l$ is level zero index). If the initial samples can be deemed as the scaling coefficients for the function projection to the function space of the finest level, $\mathbf{V}_0$, then the function approximations at this level and the next coarser level can be expressed as:

$$\begin{aligned} f_0(t) &= \sum_{m=-\infty}^{\infty} \beta_{0,m} 2^{0/2} \phi\left(2^0 t - m\right) \\ &= \sum_{l=-\infty}^{\infty} \beta_{-1,2l} 2^{-1/2} \phi\left(2^{-1}(t-2l)\right) + \sum_{l=-\infty}^{\infty} \alpha_{-1,2l} 2^{-1/2} \varphi\left(2^{-1}(t-2l)\right) \end{aligned} \quad . \quad 4.10$$

Because $\phi_{-1,2l}(t)$ are orthogonal to each other according to Equation 3.17, $\phi_{-1,2l}(t)$ are orthogonal to $\varphi_{-1,2l}(t)$ as well according to Equation 3.19, and $\varphi_{-1,2l}(t)$ are orthogonal to each other according to Equation 3.23, $\beta_{-1,2l}$ and $\alpha_{-1,2l}$ can be found by taking the inner product:

$$\beta_{-1,2l} = \left\langle f_0(t), \phi_{-1,2l}(t) \right\rangle = \int f_0(t) 2^{-1/2} \phi\left(2^{-1}(t-2l)\right) dt, \quad 4.13$$

$$\alpha_{-1,2l} = \left\langle f_0(t), \varphi_{-1,2l}(t) \right\rangle = \int f_0(t) 2^{-1/2} \varphi\left(2^{-1}(t-2l)\right) dt. \quad 4.14$$

By using Equation 4.8 to Equation 4.13, and 4.9 to 4.14, and interchanging the sum and integral, 4.13 and 4.14 become:

$$\beta_{-1,2l} = \sum_{n=-\infty}^{\infty} h(n) \int f_0(t) \phi\left(2^0(t-2l-2^0 n)\right) dt = \sum_{m=-\infty}^{\infty} h(n) \beta_{0,2l+2^0 n}, \quad 4.15$$

$$\alpha_{-1,2l} = \sum_{n=-\infty}^{\infty} h_1(n) \int f_0(t) \phi\left(2^0(t-2l-2^0 n)\right) dt = \sum_{m=-\infty}^{\infty} h_1(n) \beta_{0,2l+2^0 n}. \quad 4.16$$

Now by shifting the time axis by one unit of the initial sampling interval, the scaling function and wavelet function at the next coarser level can be expressed as the following:

$$\phi_{-1,2l+1}(t) = \phi\left(2^{-1}((t-1)-2l)\right) = \sum_{n=-\infty}^{\infty} h(n) \sqrt{2} \phi\left(2^0\left(((t-1)-2l)-2^0 n\right)\right), \quad 4.17$$

$$\varphi_{-1,2l+1}(t)=\varphi\left(2^{-1}((t-1)-2l)\right)=\sum_{n=-\infty}^{\infty}h_1(n)\sqrt{2}\phi\left(2^0\left(((t-1)-2l)-2^0n\right)\right). \qquad 4.18$$

The function approximations at level zero and function decomposition at the next coarser level are:

$$\begin{aligned} f_0(t)&=\sum_{m=-\infty}^{\infty}\beta_{0,m}\phi\left(2^0t-m\right)\\ &=\sum_{l=-\infty}^{\infty}\beta_{-1,2l+1}2^{-1/2}\phi\left(2^{-1}((t-1)-2l)\right)+\sum_{l=-\infty}^{\infty}\alpha_{-1,2l+1}2^{-1/2}\varphi\left(2^{-1}((t-1)-2l)\right). \end{aligned} \qquad 4.19$$

Then, by taking inner product, and substituting $\phi_{-1,2l+1}(t)$ and $\varphi_{-1,2l+1}(t)$ according to Equations 4.17 and 4.18 :

$$\begin{aligned} \beta_{-1,2l+1}&=\left\langle f_0(t),\phi_{-1,2l+1}(t)\right\rangle=\int f_0(t)2^{-1/2}\phi\left(2^{-1}((t-1)-2l)\right)dt\\ &=\sum_{n=-\infty}^{\infty}h(n)\beta_{0,(2l+1)+2^0n}\ , \end{aligned} \qquad 4.20$$

$$\begin{aligned} \alpha_{-1,2l+1}&=\left\langle f_0(t),\varphi_{-1,2l+1}(t)\right\rangle=\int f_0(t)2^{-1/2}\varphi\left(2^{-1}((t-1)-2l)\right)dt\\ &=\sum_{n=-\infty}^{\infty}h_1(n)\beta_{0,(2l+1)+2^0n}\ . \end{aligned} \qquad 4.21$$

Adding Equation 4.10 and Equation 4.19, we obtain:

$$\begin{aligned} &f_0(t)+f_0(t)\\ &=\sum_{l=-\infty}^{\infty}\beta_{-1,2l}2^{-1/2}\phi\left(2^{-1}(t-2l)\right)+\sum_{l=-\infty}^{\infty}\alpha_{-1,2l}2^{-1/2}\varphi\left(2^{-1}(t-2l)\right)\\ &+\sum_{l=-\infty}^{\infty}\beta_{-1,2l+1}2^{-1/2}\phi\left(2^{-1}((t-1)-2l)\right)+\sum_{l=-\infty}^{\infty}\alpha_{-1,2l+1}2^{-1/2}\varphi\left(2^{-1}((t-1)-2l)\right), \end{aligned} \qquad 4.22$$

$$\begin{aligned} f_0(t)&=\sum_{l=-\infty}^{\infty}\frac{1}{2}\beta_{-1,2l}2^{-1/2}\phi\left(2^{-1}(t-2l)\right)+\sum_{k=-\infty}^{\infty}\frac{1}{2}\alpha_{-1,2l}2^{-1/2}\varphi\left(2^{-1}(t-2l)\right)\\ &+\sum_{l=-\infty}^{\infty}\frac{1}{2}\beta_{-1,2l+1}2^{-1/2}\phi\left(2^{-1}((t-1)-2l)\right)+\sum_{l=-\infty}^{\infty}\frac{1}{2}\alpha_{-1,2l+1}2^{-1/2}\varphi\left(2^{-1}((t-1)-2l)\right). \end{aligned} \qquad 4.23$$

The sets $\phi_{-1,2l}(t)$ and $\phi_{-1,2l+1}(t)$, are the basis of $\mathbf{V}_{-1}$, onto which the f_0 is projected. The sets $\varphi_{-1,2l}(t)$ and $\varphi_{-1,2l+1}(t)$ are then the basis of the space $\mathbf{W}_{-1}$ which covers the error of the projection.

By combining Equations 4.15 and 4.20, 4.16 and 4.21, we can see, the number of $\{\beta_{-1}\}$ and the number of $\{\alpha_{-1}\}$ are same as the number of $\{\beta_0\}$:

$$\beta_{-1,m} = \sum_{n=-\infty}^{\infty} h(n)\beta_{0,m+2^0 n} \qquad 4.24$$

$$\alpha_{-1,m} = \sum_{n=-\infty}^{\infty} h_1(n)\beta_{0,m+2^0 n} \qquad 4.25$$

In the next level projection, four sets of scaling function and wavelet function for $\mathbf{V}_{-2}$ and $\mathbf{W}_{-2}$ can be obtained by shifting the time axis using $i = 0,1,2,3$:

$$\phi_{-1,2^2 l+i}(t) = \phi\left(2^{-2}((t-i)-2^2 l)\right) = \sum_{n=-\infty}^{\infty} h(n)\sqrt{2}\phi\left(2^{-1}(((t-i)-2^2 l)-2^{+1}n)\right), \qquad 4.26$$

$$\varphi_{-1,2^2 l+i}(t) = \varphi\left(2^{-2}((t-i)-2^2 l)\right) = \sum_{n=-\infty}^{\infty} h_1(n)\sqrt{2}\phi\left(2^{-1}(((t-i)-2^2 l)-2^{+1}n)\right). \qquad 4.27$$

The function approximations at $\mathbf{V}_{-1}$ (and decomposition at $\mathbf{V}_{-2}$) are:

$$\begin{aligned} f_{-1}(t) &= \sum_{m=-\infty}^{\infty} \beta_{-1,2m} 2^{-1/2} \phi\left(2^{-1}(t-2m)\right) \\ &= \sum_{l=-\infty}^{\infty} \beta_{-1,2^2 l+0} 2^{-2/2} \phi\left(2^{-2}((t-0)-2^2 l)\right) + \sum_{l=-\infty}^{\infty} \alpha_{-1,2^2 l+0} 2^{-2/2} \varphi\left(2^{-2}((t-0)-2^2 l)\right) \\ &= \sum_{l=-\infty}^{\infty} \beta_{-1,2^2 l+2} 2^{-2/2} \phi\left(2^{-2}((t-2)-2^2 l)\right) + \sum_{l=-\infty}^{\infty} \alpha_{-1,2^2 l+2} 2^{-2/2} \varphi\left(2^{-2}((t-2)-2^2 l)\right), \end{aligned} \qquad 4.28$$

$$f_{-1}(t)=\sum_{m=-\infty}^{\infty}\beta_{-1,2m}2^{-1/2}\phi\left(2^{-1}((t-1)-2m)\right)$$

$$=\sum_{l=-\infty}^{\infty}\beta_{-1,2^2l+1}2^{-2/2}\phi\left(2^{-2}((t-1)-2^2l)\right)+\sum_{l=-\infty}^{\infty}\alpha_{-1,2^2l+1}2^{-2/2}\varphi\left(2^{-2}((t-1)-2^2l)\right) \quad 4.29$$

$$=\sum_{l=-\infty}^{\infty}\beta_{-1,2^2l+3}2^{-2/2}\phi\left(2^{-2}((t-3)-2^2l)\right)+\sum_{l=-\infty}^{\infty}\alpha_{-1,2^2l+3}2^{-2/2}\varphi\left(2^{-2}((t-3)-2^2l)\right).$$

Then the following relation can be obtained by applying the same transformation to Equations 4.26 -- 4.29:

$$\beta_{-2,2^2l+i}=\sum_{n=-\infty}^{\infty}h(n)\beta_{-1,(2^2l+i)+2^1n}\Rightarrow\beta_{-2,m}=\sum_{n=-\infty}^{\infty}h(n)\beta_{-1,m+2^1n}, \quad 4.30$$

$$\alpha_{-2,2^2l+i}=\sum_{n=-\infty}^{\infty}h_1(n)\alpha_{-1,(2^2l+i)+2^1n}\Rightarrow\alpha_{-2,m}=\sum_{n=-\infty}^{\infty}h_1(n)\beta_{-1,m+2^1n}. \quad 4.31$$

where $i=0,1,2,3$.

The function decomposition on level 2 is

$$f_{-1}(t)=\sum_{i=0}^{3}\Big(\sum_{l=-\infty}^{\infty}\frac{1}{4}\beta_{-2,4l+i}2^{-2/2}\phi\left(2^{-2}((t-i)-4l)\right) + \sum_{l=-\infty}^{\infty}\frac{1}{4}\alpha_{-2,4l+i}2^{-2/2}\varphi\left(2^{-2}((t-i)-4l)\right)\Big). \quad 4.32$$

Now, let us generalize the new algorithm to calculate the scaling and wavelet coefficients and function decomposition from Equations 4.24, 4.25, 4.30, 4.31, 4.23 and 4.32:

$$\beta_{j,m}=\sum_{n=-\infty}^{\infty}h(n)\beta_{j+1,m+2^{j-1}n}, \quad 4.33$$

$$\alpha_{j,m}=\sum_{n=-\infty}^{\infty}h_1(n)\beta_{j+1,m+2^{j-1}n}, \quad 4.34$$

$$f_{j+1}(t)=\sum_{i=0}^{2^j-1}\Big(\sum_{l=-\infty}^{\infty}2^j\beta_{j,2^jl+i}2^{j/2}\phi\big(2^j\big((t-i)-2^{-j}l\big)\big)$$
$$+\sum_{l=-\infty}^{\infty}2^j\alpha_{j,2^jl+i}2^{j/2}\varphi\big(2^j\big((t-i)-2^{-j}l\big)\big)\Big) \qquad 4.35$$
$$=\sum_{m=-\infty}^{\infty}2^j\beta_{j,m}2^{j/2}\phi\big(2^j(t-m)\big)+\sum_{m=-\infty}^{\infty}2^j\alpha_{j,m}2^{j/2}\varphi\big(2^j(t-m)\big).$$

where $j \le -1$ is integer.

This algorithm is called Anchored Wavelet Transform (AWT), in that the algorithm uses only the initial sample level index during the transformation, and coefficients at each level are available for any location. In Figure 4.3, AWT is applied to the example declared in Section 4.1 for interpretation of the word 'anchored' and the difference between the AWT algorithm and the common WT.

From Equations 4.33 and 4.34, we can see that the decimation effect disappears so that the function projections on each $\{\phi_j\}$ and $\{\varphi_j\}$ can be obtained, which form the feature space for the function.

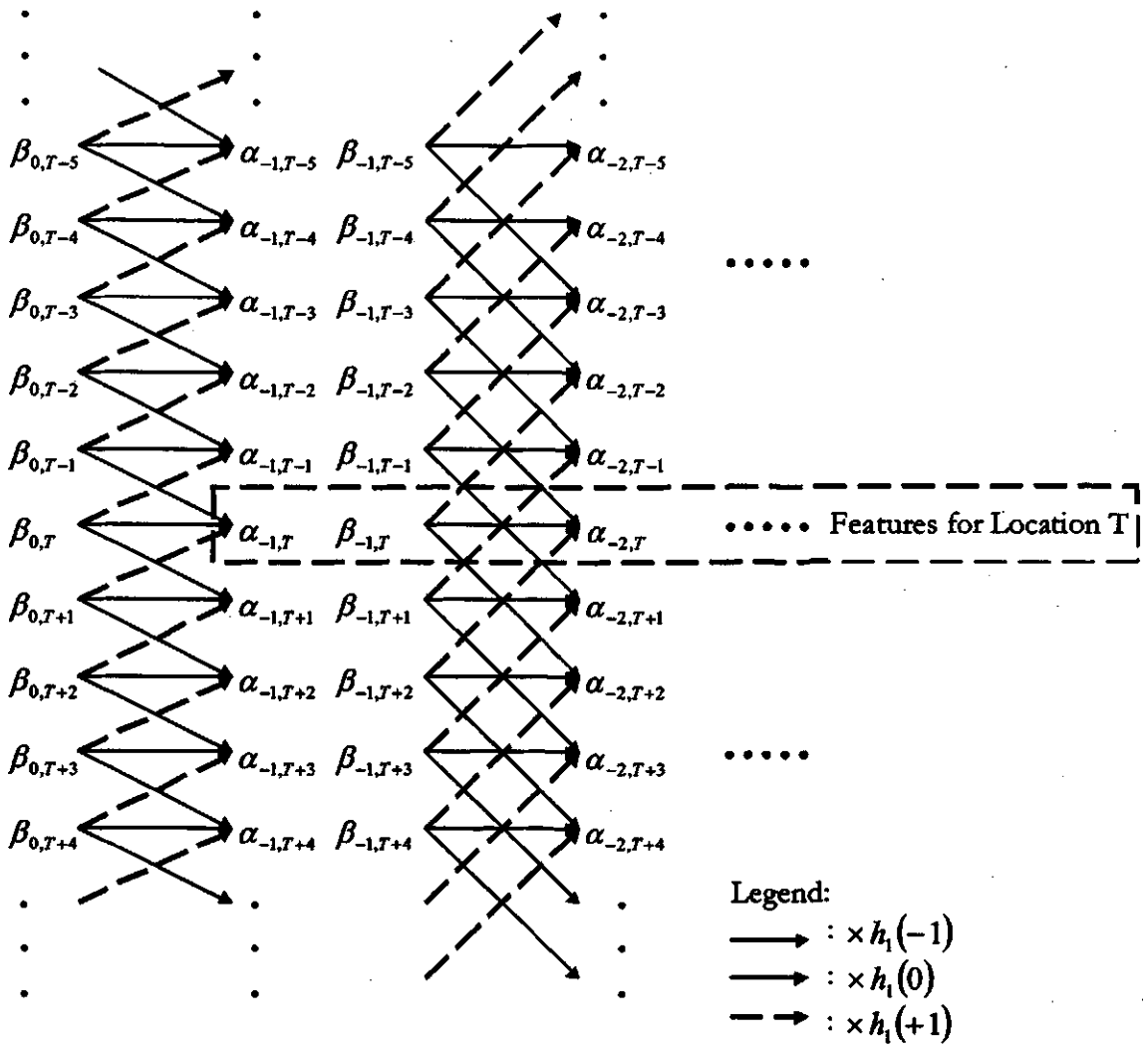


Figure 4.5 AWT algorithm illustration

The feature extraction method developed is to repeatedly apply Equations 4.33 and 4.34 on series $\{\beta_j\}$, where the initial value of j is -1, standing for the first decomposition from initial samples. The feature measurements extracted for any location T are the wavelet coefficients at each scaling level and the scaling coefficient at the coarsest level:

$$\{\alpha_{-1,T}, \alpha_{-2,T}, ..., \alpha_{-M,T}, \beta_{-M,T}\}, \qquad 4.36$$

where M is the number of repetitions of AWT.

4.3 Interpretation of the Features Extracted by AWT

There are two interpretations to the features extracted by AWT: the features measure the frequency component amplitudes in different frequency bands of the function spectrum, and measure the degree of correlation between the shape of the function and the shape of the wavelet.

In the discipline of digital signal processing, the 'filtering' of a sequence of numbers is achieved by convolving the sequence with a sequence of numbers called the filter impulse response. Equations 4.33 and 4.34 show that the scaling and wavelet coefficients at different levels of scale can be obtained by convolving the previous finer level scaling coefficients by the time-reversed recursion sequence $h(-n)$ and $h_1(-n)$ with a dilated time interval. In other words, the coefficients at one level are obtained by filtering scaling coefficients at the previous finer scaling level. The filter implemented by $h(-n)$ is a lowpass filter, the one implemented by $h_1(-n)$ is a highpass filter and the two filters are mirror filters with the same stop frequency (Burrus et. al. 1998 [8]), under the assumption that orthogonal wavelet and scaling functions are used in the transform.

Assume $h(-n)$ starts with the $-P$ element, ends with the M element, and has stop frequency of ω_1:

$$|H(\omega_1)| = \left| \sum_{n=-P}^{M} h(-n) e^{i\omega_1 n} \right|. \qquad 4.37$$

Let us look at how the filters are applied at each scaling level from Equations 4.33 and 4.34:

- Projection to scaling level –1: the sequence applied in convolution is $h^{-1}(-n)=\{h(-P),h(-P+1),...,h(M)\}$, its stop frequency is ω_1;
- Projection to scaling level –2: the sequence applied in convolution is $h^{-2}(-n)=\{h(-P),0,h(-P+1),...,h(M-1),0,h(M)\}$.

  Its stop frequency is $\omega_1/2$, because

$$H^{-2}(\omega)=\sum_{n=-2P}^{2M}h^{-2}(-n)e^{i\omega n}=\sum_{m=-P}^{M}h(-m)e^{i2\omega m}; \qquad 4.38$$

- In general, in projecting to scaling level $j\le -1$, the sequence applied in convolution is $h^{j}(-n)=\{h(-P),0,..,0,h(-P+1),...,h(M-1),0,...,0,h(M)\}$, the number of zeros between adjacent $h(-m)$ members is $2^{-j-1}-1$, and the stop frequency is $\omega_1/2^{-j-1}$, because of

$$H^{j}(\omega)=\sum_{n=-2^{-j-1}P}^{2^{-j-1}M}h^{j}(-n)e^{i\omega n}=\sum_{m=-P}^{M}h(-m)e^{i2^{-j-1}\omega m}; \qquad 4.39$$

- The sequence $h_1^{-j}(-n)$ has the same stop frequency as that of $h^{-j}(-n)$.

From the above analysis we can see that, the spectrum of initial samples $\beta_{0,m}$ is divided into a lowpass and highpass band in the first projection, resulting in the scaling and wavelet coefficients at lower level, $\beta_{-1,m}$ and $\alpha_{-1,m}$. As the projections proceed, the lowpass bands at each lower level are further equally divided. Figure 4.6 shows this frequency band dividing process. W_{-k} is the function approximation error space at level k, which is described in Section 3.2.

From Figure 4.6, we can see that the features $\alpha_{j,m}$ extracted by AWT are the frequency component measurements for the initial samples, in different frequency bands.

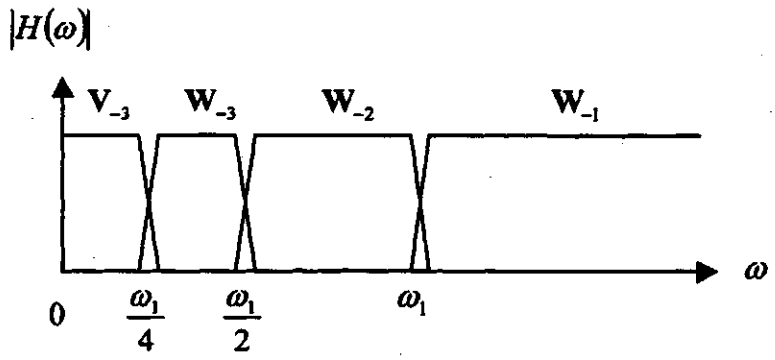


Figure 4.6 Frequency Bands for AWT analysis

Another interpretation to the features extracted is that they measure the degree of correlation between the shape of the function and the shape of the wavelet. This can be explained by looking at Equation 4.35. When approximating function from scaling level $j+1$ to level j, if the function shape change at location T is very similar to the shape of $\varphi_{j,T}$, then the feature measurement $\alpha_{j,T}$ will be relatively large. This is similar to FT. The FT result shows the level of resemblance between a particular periodical waveform and the target signal, while wavelet coefficient shows the resemblance between a wavelet and the target signal change. For AWT, taking into account the scaling effect, we can also obtain the shape information of the function in a different range from the features extracted.

In this chapter, the widely used WT algorithm is shown to be not sufficient to directly extract features at any location of a function because of the decimation, while the AWT algorithm is proposed to do so. The feature extraction method developed is to repeatedly apply Equations 4.33 and 4.34 on series $\{\beta_j\}$. The feature measurements extracted for any

location T are the wavelet coefficients at each scaling level and scaling coefficients at the coarsest level: $\{\alpha_{-1,T}, \alpha_{-2,T}, ..., \alpha_{-M,T}, \beta_{-M,T}\}$, where M is the number of repetitions of AWT. There are two interpretations to the features extracted: the features measure the frequency component amplitudes in different frequency bands of the function spectrum, and measure the degree of correlation between the shape of the function and the shape of the wavelet.

Chapter 5 Experiments of AWT Feature Extraction for Point Tracking in an Image Sequence

In Chapter 4, the AWT algorithm was developed to extract one dimensional function features at specified points. The feature measurements extracted for any location T are the wavelet coefficients at each scaling level and scaling coefficients at the coarsest level: $\{\alpha_{-1,T},\alpha_{-2,T},...,\alpha_{-M,T},\beta_{-M,T}\}$, where M is the number of repetitions of AWT. There are two interpretations to the features extracted: the features measure the frequency component amplitudes in different frequency bands of the function spectrum, and measure the degree of correlation between the shape of the function and the shape of the wavelet.

In order to demonstrate the effectiveness of the wavelet features extraction in image analysis, the AWT algorithm will be used to extract image point features, and track points in an image sequence.

5.1 2D AWT and Used Wavelets

To process a 2D digital signal, such as an image, Equations 4.33 and 4.34 are first applied on each row of the image, and then applied to each column. Equations 4.33 and 4.34 are written as:

$$\beta_{j,m}=\sum_{n=-\infty}^{\infty}h(n)\beta_{j+1,m+2^{-j-1}n}\ , \qquad 5.1$$

$$\alpha_{j,m} = \sum_{n=-\infty}^{\infty} h_1(n)\beta_{j+1,m+2^{-j-1}n} \,. \qquad 5.2$$

As introduced in Chapter 4, $h(n)$ is a low-pass filter and $h_1(n)$ is a mirror high-pass filter. In each decomposition level, Equations 5.1 and 5.2 are applied to each row of the low-pass image from the last level, and two images are generated: low-passed image $\beta^{L}{}_{j,(m,i)}(n)$ and high-passed image $\alpha^{H}{}_{j,(m,i)}(n)$, where j is the decomposition level and (m,i) is the 2D coordinate. These two images both have the same size as the original image. Then after Equations 5.1 and 5.2 are applied to each column of $\alpha^{H}{}_{j,(m,i)}(n)$ and $\beta^{L}{}_{j,(m,i)}(n)$, four same size images are generated: $\alpha^{HH}{}_{j,(m,i)}(n)$, $\alpha^{HL}{}_{j,(m,i)}(n)$, $\alpha^{LH}{}_{j,(m,i)}(n)$ and $\beta^{LL}{}_{j,(m,i)}(n)$. Figure 5.1 illustrates two level decomposition of an image signal. The features extracted for point (X,Y) are

$$\begin{aligned}
&\{\alpha^{HH}{}_{-1,(X,Y)}, \alpha^{HL}{}_{-1,(X,Y)}, \alpha^{LH}{}_{-1,(X,Y)}, \\
&\alpha^{HH}{}_{-2,(X,Y)}, \alpha^{HL}{}_{-2,(X,Y)}, \alpha^{LH}{}_{-2,(X,Y)}, \ldots \\
&\alpha^{HH}{}_{-M+1,(X,Y)}, \alpha^{HL}{}_{-M+1,(X,Y)}, \alpha^{LH}{}_{-M+1,(X,Y)}, \beta^{LL}{}_{-M,(X,Y)}\},
\end{aligned} \qquad 5.3$$

where M is the number of the decomposition level.

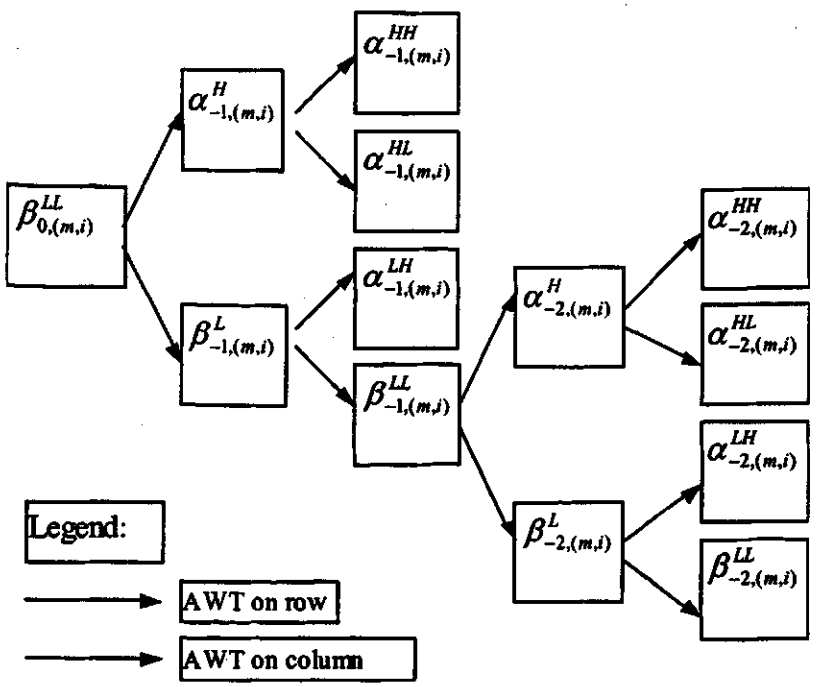


Figure 5.1 Two levels of AWT on 2D image.

The orthogonal wavelets used in this thesis are Haar, Daubechies-4 and Daubechies-6, which are presented in Burrus et al. 1998 [8]. The scaling function coefficients of these wavelet systems, $h(n)$, are listed in Table 5.1. Because Haar, Daubechies-4 and Daubechies-6 wavelets are orthogonal wavelets, the wavelet function coefficients, $h_1(n)$, can be obtained from $h(n)$ by Equation 3.20.

Table 5.1 Scaling function coefficients

n	Haar *h(n)*	Daubechies-4	Daubechies-6
0	0.5	0.3415063	0.2352336
1	0.5	0.5915063	0.5705585
2		0.1584937	0.3251825
3		-0.0915063	-0.0954672
4			-0.0604161
5			0.0249088

5.2 Examples of Applying WT and AWT on an Image

Figures 5.2, 5.3, 5.4, 5.5 and 5.6 show the result of applying the WT and the AWT algorithm to a sample image. The wavelet used is Daubechies-4. The right side of Figure 5.2 shows the result of applying the general WT algorithm. The size of the transformed images shrinks to one fourth of the original one, so the values in these images cannot be used as point features directly because of the decimation. Details of the image and the experiment are given in Appendix A.

Original image Wavelet Transform level one

Figure 5.2 Original image and one level WT result

In Figure 5.3 – 5.6, the symbol LL means that the low pass filter $h(n)$ has been applied to both horizontal and vertical directions. HH means that the high pass filter $h_1(n)$ has been applied to both directions, HL means the high pass has been applied to the horizontal and low pass to the vertical, and LH vice versa.

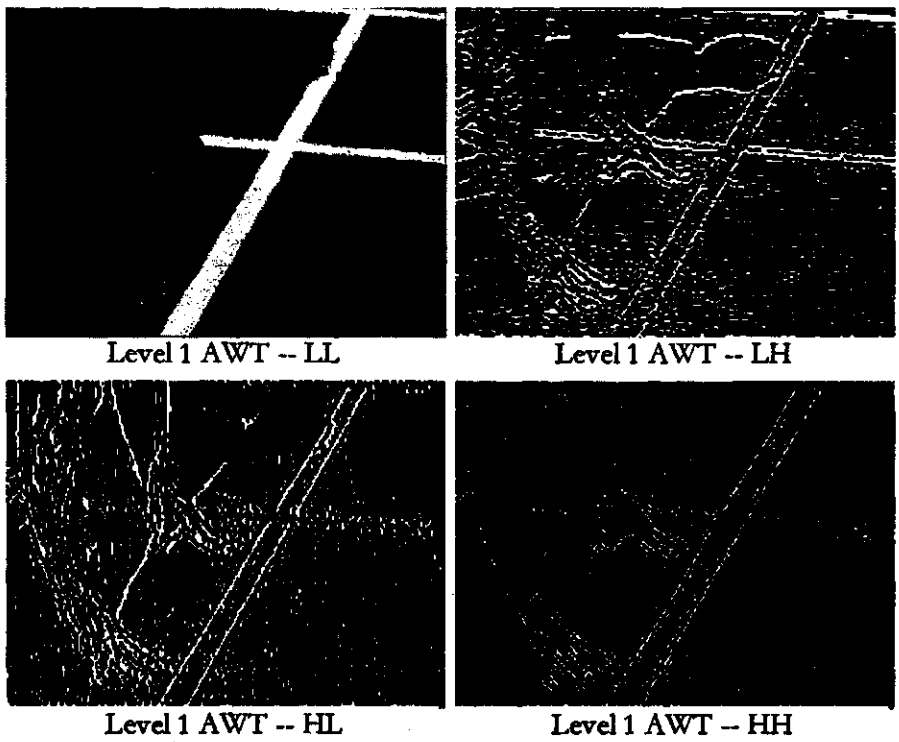


Figure 5.3 Level 1 AWT result

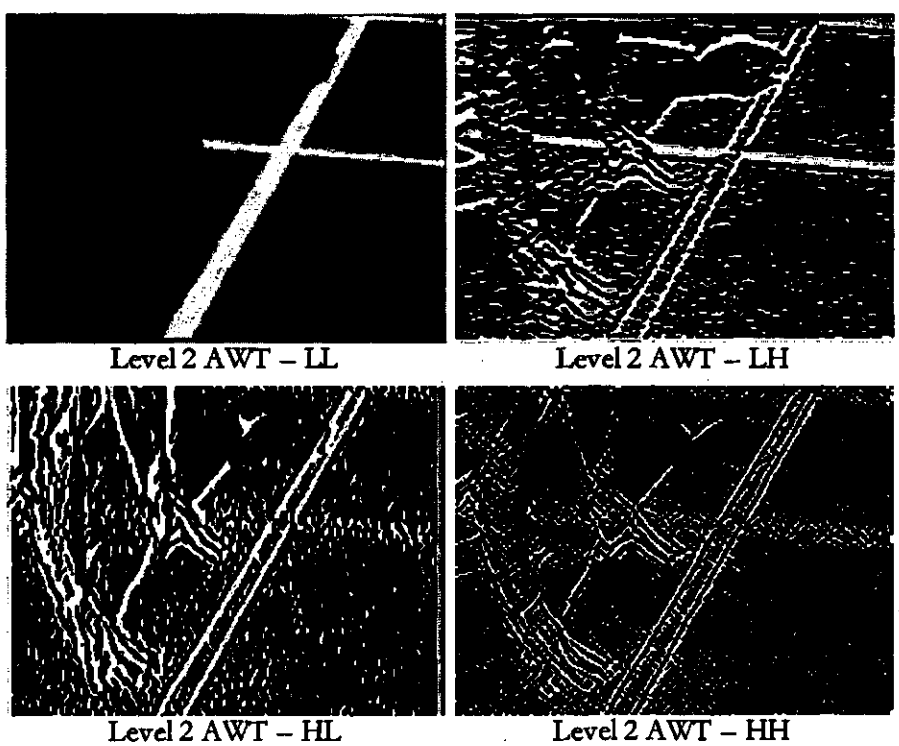


Figure 5.4 Level 2 AWT result

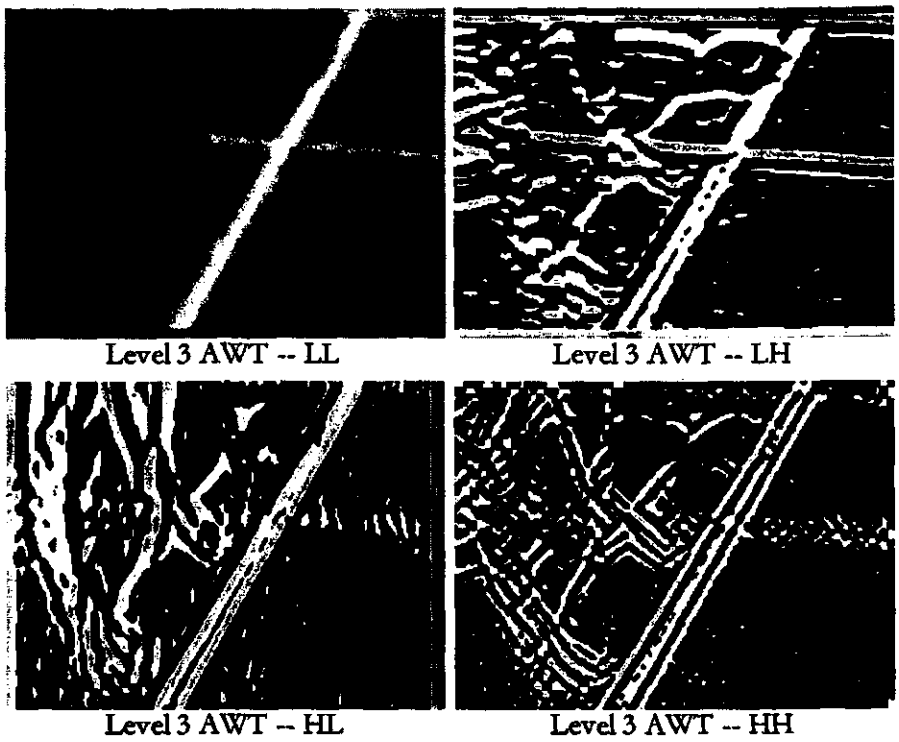


Figure 5.5 Level 3 AWT result

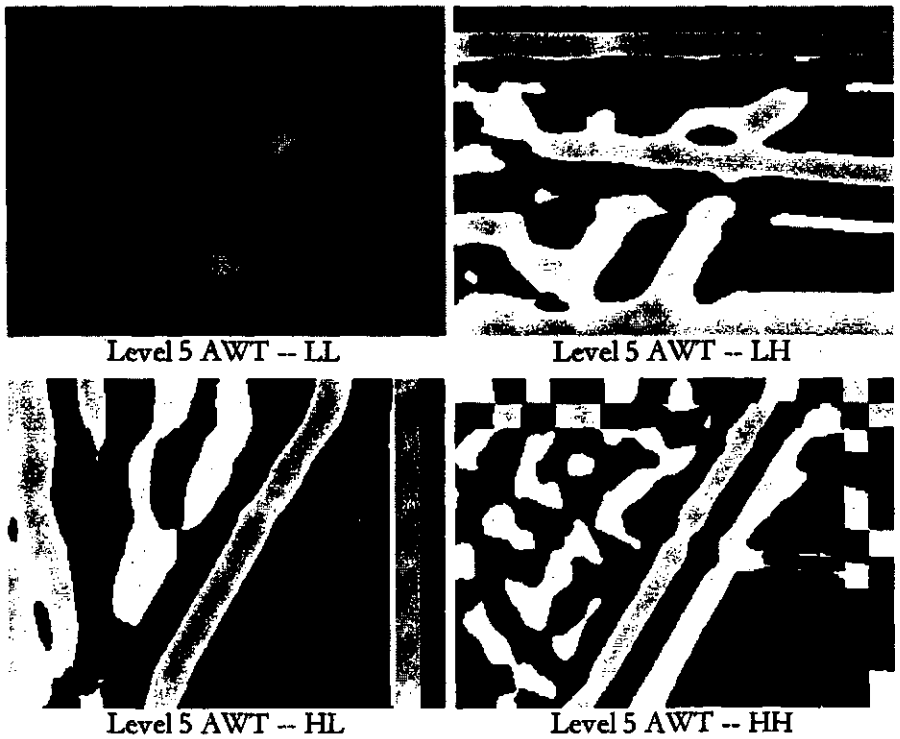


Figure 5.6 Level 5 AWT result

On the other hand, the AWT algorithm transforms an image into same size images containing different image information, which are suitable to be features.

From Figures 5.3, 5.4, 5.5, and 5.6 we can see that:

- LL parts tend to be blurrier when the transformation goes deeper. In level 5 of the AWT transform, the LL image is almost segmented into several unique-intensity regions, the unique-intensity represents the 'average' pixel intensity in the corresponding region;
- LH parts mainly contain the horizontal lines, with different width in different levels;
- HL parts mainly contain the vertical lines, with different width in different levels;
- HH parts contain the intersections of the lines.

5.3 Algorithms for Selecting the Feature-Rich Points and Tracking Points

Based on the observations in Section 5.2 and the interpretations to AWT presented in Chapter 4, the rule for selecting feature-rich points is obtained:

$$\underset{x,y}{Max}\left(\sum_{j=1}^{M}\left(\left(\frac{\alpha_{j,(x,y)}^{HH}}{\beta_{M,(x,y)}^{LL}}\right)^2+\left(\frac{\alpha_{j,(x,y)}^{HL}}{\beta_{M,(x,y)}^{LL}}\right)^2+\left(\frac{\alpha_{j,(x,y)}^{LH}}{\beta_{M,(x,y)}^{LL}}\right)^2\right)\right). \qquad 5.4$$

where M is the AWT level, and *(x, y)* is the point coordinate in the image. The Formula 5.4 states that, if the image has maximum overall relative energy (biggest shape variances) in all scales at point *(x, y)*, then point *(x, y)* is selected as a feature-rich point. This point will then be tracked as described in the next section. When multiple points are

desired, Formula 5.4 can be repeated to find them, but excluding the previously selected points in each search. Another way of selecting multiple points is to separate the whole image into sections and seek a maximum in each section. In this thesis, the image was divided into 25 evenly distributed sections with the same size, to get 25 local maxima. Then nine points with the largest energy were finally selected.

Tracking a point requires us to find the matching points in an image sequence. After selecting the feature-rich points in the 1st frame in the image sequence, the points tracking algorithm will select the most similar point in the 2nd frame for each selected point in the 1st frame, select most similar point in the 3rd frame for each selected point in 2nd frame,..., and so on, until the end frame in the image sequence is reached.

The similarity is measured by the Euclidean distance D between two points in feature space which is $\mathbf{W}_{-1} \cup \mathbf{W}_{-2},\ldots,\cup\mathbf{W}_{-M}$, where M is the total number of AWT levels. D can be expressed as:

$$\underset{(x1,y1),(x2,y2)}{D} = \sum_{j=1}^{M}\left(\left(\alpha^{HH}_{j,(x1,y1)} - \alpha^{HH}_{j,(x2,y2)}\right)^2 + \left(\alpha^{HL}_{j,(x1,y1)} - \alpha^{HL}_{j,(x2,y2)}\right)^2 + \left(\alpha^{LH}_{j,(x1,y1)} - \alpha^{LH}_{j,(x2,y2)}\right)^2\right), \quad 5.5$$

where *(x1, y1)* is the point coordinate in one image, *(x2, y2)* is the point in another image. The points match decision rule is: Point (X2, Y2) in image B is the similar point to point (X1, Y1) in image A, if the Euclidean distance between (X1, Y1) and (X2, Y2) in feature space is the minimum, among distances between (X1, Y1) and any point in image B:

$$Min\left(\underset{(X1,Y1),(x2,y2)\in B}{D}\right) = \underset{(X1,Y1),(X2,Y2)}{D}. \quad 5.6$$

5.4 Experiments Result and Discussion

This section gives the results of experiments tested to demonstrate the effectiveness of the AWT algorithms to identify specific feature rich points in an image, and to track those points from image to image in a video sequence. The detailed description of the experimental implementation is given in Appendix A.

Figure 5.8 shows the points tracking result, where the points tracking algorithms developed in Section 5.3 are used in tracking 9 points within an image sequence. The wavelet used is Daubechies-4, and the number of the AWT decomposition levels M is 6. Figure 5.7 lists the original images zoomed to 57% of actual size.

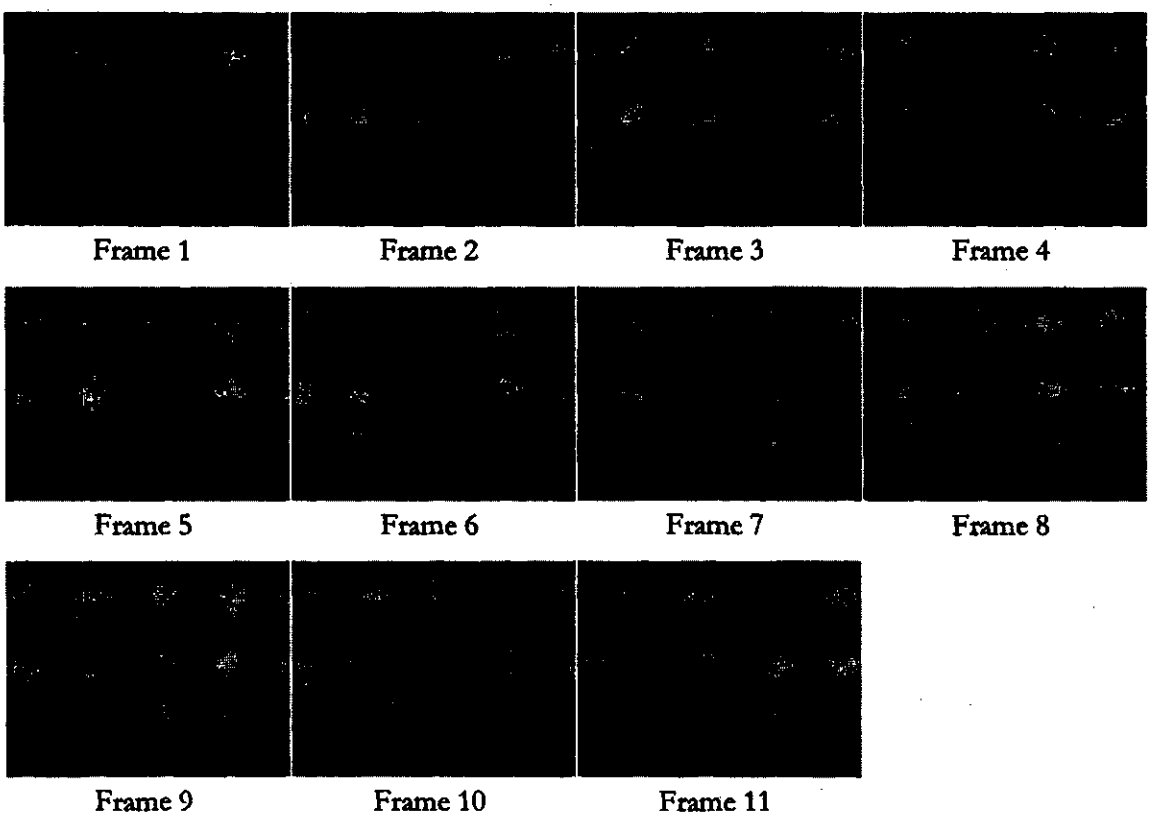
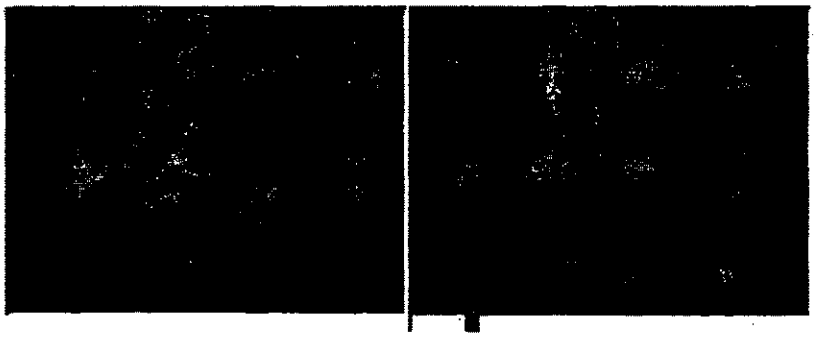


Figure 5.7 Original image sequence.

The points, which are marked in the image of 'Frame 1 Points Selected' in Figure 5.8, are selected by the feature-rich points selecting algorithm expressed by Formula 5.4. The marked points in the other images in Figure 5.8 are the match results to the points in the previous image with the same mark index. The actual matching process is going forward, from frame 1 to frame 11, and then going backward, from frame 11 to frame 1:

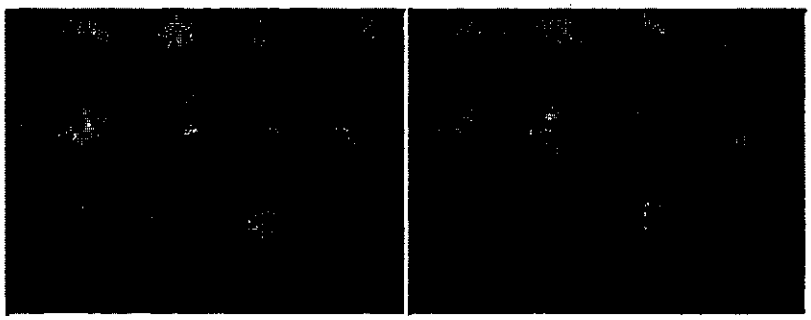
$1\rightarrow2\rightarrow3\rightarrow4\rightarrow5\rightarrow6\rightarrow7\rightarrow8\rightarrow9\rightarrow10\rightarrow11\rightarrow10\rightarrow9\rightarrow8\rightarrow7\rightarrow6\rightarrow5\rightarrow4\rightarrow3\rightarrow2\rightarrow1$,

where the numbers are the image frame numbers, and the arrow means to find similar points in current image corresponding to the marked points in the previous image. Therefore there are two column images in Figure 5.8; the left side column images are the forward matching images while the right column images are the result of backward matching. The exact location of the point marked is at the upper-left corner of the background rectangle of the index marker.



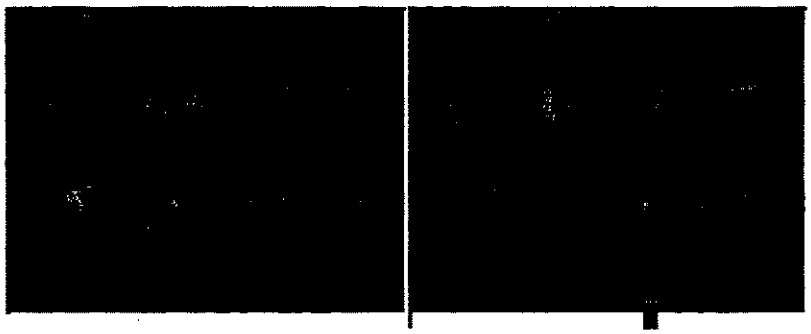
Frame 1 Points Selected

Frame 1 Backward Match



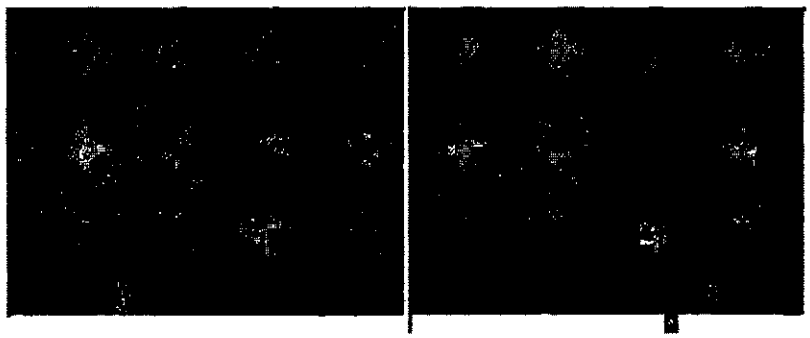
Frame 2 Forward Match

Frame 2 Backward Match



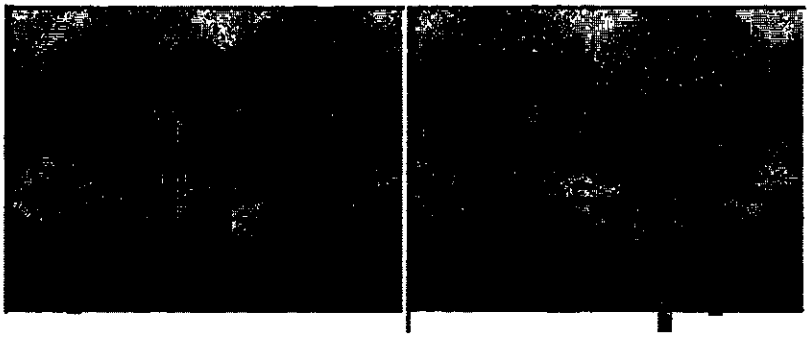
Frame 3 Forward Match

Frame 3 Backward Match

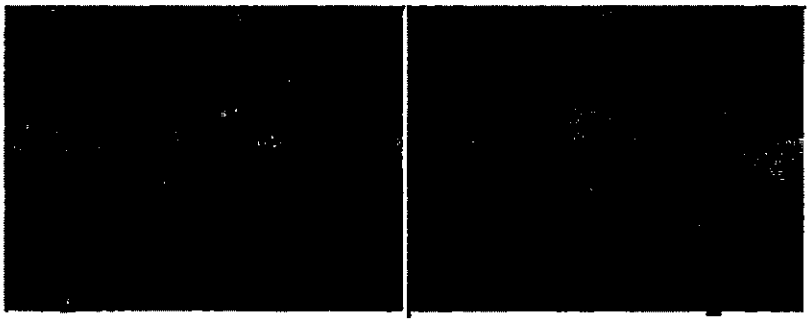


Frame 4 Forward Match

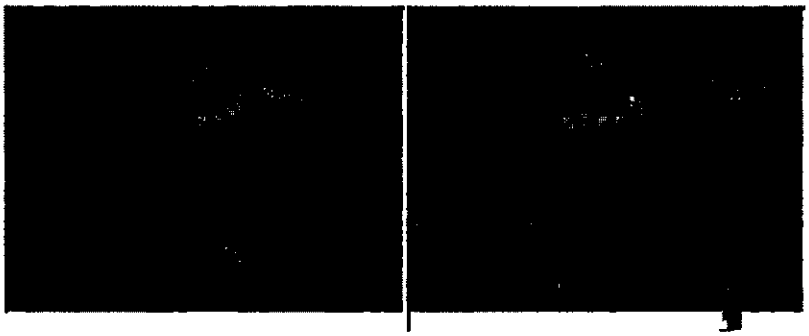
Frame 4 Backward Match



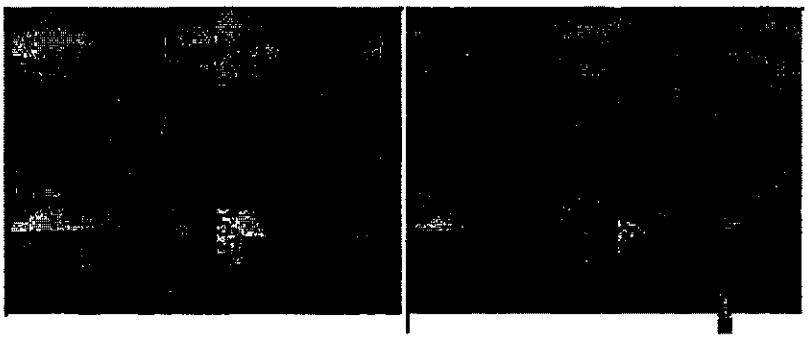
Frame 5 Forward Match Frame 5 Backward Match



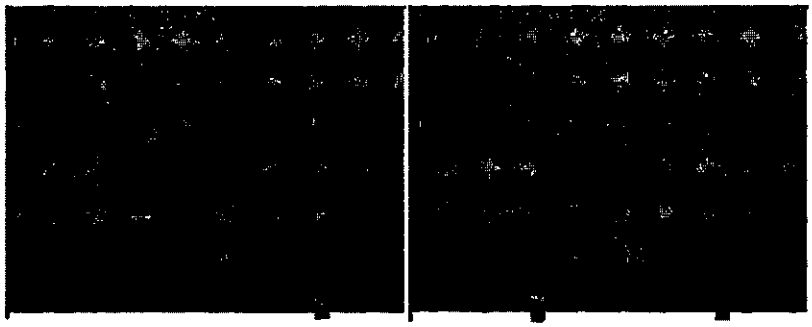
Frame 6 Forward Match Frame 6 Backward Match



Frame 7 Forward Match Frame 7 Backward Match

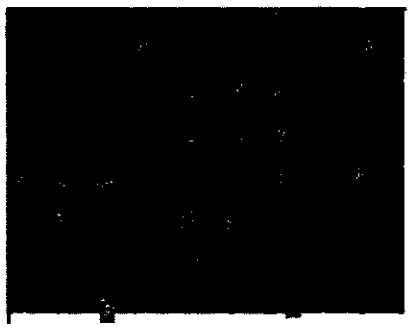


Frame 8 Forward Match Frame 8 Backward Match



Frame 9 Forward Match Frame 9 Backward Match

Frame 10 Forward Match Frame 10 Backward Match



Frame 11 Forward Match

Figure 5.8 Feature-rich points tracking result using Daubechies-4 wavelet in the AWT feature extraction.

From the marked positions in the image of 'Frame 1 Points Selected', we can see that the points selected by the feature-rich points selecting algorithm are all located in a region with significant underlying shape variance. As expected, these points are either near the lines that separate regions with large intensity contrast, or near the intersection of such lines. There are 9 points selected with index zero to eight. What should be noticed is that markers '2' and/or '5' occlude the marker of point '0'. This happens in all images in Figure 5.8.

The matching result, determined by visually comparing similar points with those chosen by the algorithm, is summarized in Table 5.2. From Table 5.2 we can see that the mismatch rate is only 8%. This demonstrates the high effectiveness of the matching algorithm, considering that the quality of the image sequence is not high. Furthermore, because the matching algorithm is fairly simple, its effectiveness comes mainly from the success of the AWT feature extraction algorithm.

Tables 5.3 and 5.4 summarize the match result of using Haar and Daubechies-6 wavelet in the AWT feature extraction algorithm. The initial feature-rich points tracked are manually set at the same locations as in the previous experiment, and the image sequence is the same as well. The results show that Daubechies-6 has similar performance to that of Daubechies-4; they all have a better performance than Haar wavelets. All these three wavelets can track more than half of the points from frame 1 to frame 11.

Table 5.2 Summary of the matching result using Daubechies-4 wavelet in the AWT

Forward Matching	Total Points	Number of Mismatched	Index for Mismatched Points
Frame1 to Frame2	9	1	No.8
Frame2 to Frame3	9	0	N/A
Frame3 to Frame4	9	2	No.8, No.7
Frame4 to Frame5	9	1	No.8
Frame5 to Frame6	9	0	N/A
Frame6 to Frame7	9	1	No.8
Frame7 to Frame8	9	1	No.8
Frame8 to Frame9	9	1	No.8
Frame9 to Frame10	9	1	No.5
Frame10 to Frame11	9	0	N/A
Backward Matching	Total Points	Number of Mismatched	Index for Mismatched Points
Frame11 to Frame10	9	1	N/A
Frame10 to Frame9	9	0	N/A
Frame9 to Frame8	9	0	No.5
Frame8 to Frame7	9	0	N/A
Frame7 to Frame6	9	2	No.5
Frame6 to Frame5	9	0	No.5
Frame5 to Frame4	9	0	N/A
Frame4 to Frame3	9	1	No.5
Frame3 to Frame2	9	1	No.5
Frame2 to Frame1	9	0	No.5, No.4
Total:	180	15	
Mismatch Rate:		8%	

Table 5.3 Summary of the matching result using Haar wavelet in the AWT

Using Haar wavelet in the AWT feature extraction			
Forward Matching	Total Points	Number of Mismatched	Index for Mismatched Points
Frame1 to Frame2	9	2	No.8,No.5
Frame2 to Frame3	9	0	N/A
Frame3 to Frame4	9	2	No.8, No.7
Frame4 to Frame5	9	3	No.8, No.7, No.5
Frame5 to Frame6	9	1	No.8
Frame6 to Frame7	9	2	No.8, No.5
Frame7 to Frame8	9	3	No.7, No.6, No.5
Frame8 to Frame9	9	3	No.7, No.6, No.5
Frame9 to Frame10	9	2	No.7, No.6
Frame10 to Frame11	9	1	No.8
Total:	90	19	
Mismatch Rate:		21%	

Table 5.4 Summary of the matching result using Daubechies-6 wavelet in the AWT

Using Daubechies-6 wavelet in the AWT feature extraction			
Forward Matching	Total Points	Number of Mismatched	Index for Mismatched Points
Frame1 to Frame2	9	2	No.8, No.7
Frame2 to Frame3	9	1	No.8
Frame3 to Frame4	9	0	N/A
Frame4 to Frame5	9	0	N/A
Frame5 to Frame6	9	1	No.7
Frame6 to Frame7	9	1	No.4
Frame7 to Frame8	9	0	N/A
Frame8 to Frame9	9	1	No.4
Frame9 to Frame10	9	1	No.7
Frame10 to Frame11	9	0	N/A
Total:	90	7	
Mismatch Rate:		8%	

However, when images 'Frame 1 Point Selected' and 'Frame 11 Forward Match' (see Figure 5.8) are compared, points No.4, No.5, No.7 and No.8 are all mismatched. It seems

the propagation of matching error can degrade the reliability of the points tracking in the sequence. Now let us scrutinize the cause for mismatch. The image sequence captures the scene of the water rushing out under the bridge, where there is a pillar and rock at the riverbank. As introduced in Section 5.3, initial feature-rich points are selected according to the degree of underlying shape change. From image 'Frame 1 Point Selected' we can see that

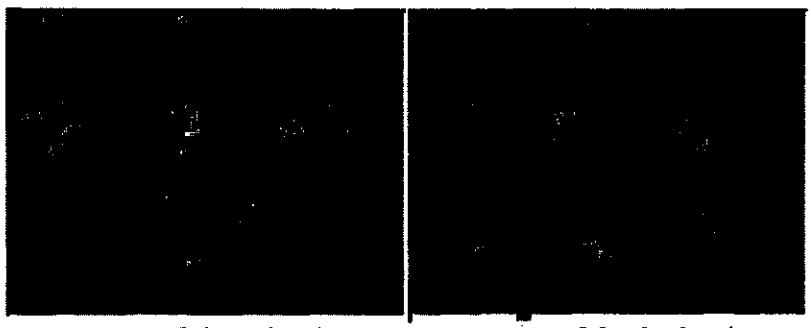
- the point No.4 is selected at the intersection of a gray color region and a white region,
- the point No.5 is selected at the intersection of bright water and a dark rock; the selection depends on the shape of the rock as well as the water,
- the point No.7 is selected at the intersection of bright water, gray riverbank and a half bright and half dark round shape pillar,
- the point No.8 is selected on the water with strap pattern, near the vertical gray shade on the pillar.

The first mismatching for the point No.8 is in $1 \rightarrow 2$ (forward match, from frame 1 to frame 2), where the water shape changed around the original No. 8 position, whereas the nearest pattern match goes to a location on the bridge. There is also a strap pattern under the new matched location, near a stick shaped object. We can see from the images that the causes of the first mismatching for point No.7 and No.5 are similar to No.8: the water shape changed. The only mismatch for the point No.4 happens when similar texture appeared under the point. After the first mismatch, points No.5 and No.8 are totally lost because they never go to any feature-rich location again, while point No.7 sticks on a new location. It is clear that the cause of mismatch is the change of the shape underlying the

tracked points, because the features extracted for point tracking are the shape measurements in different scales (in different frequency band).

It is a natural solution to discard the points that are not located on relatively static image objects, to get robust points for the tracking method. The method showed above is to track the points forward and backward, and discard mismatched points in the same image. Such as in image 'Frame 1 Points Selected' and 'Frame 1 Backward Match', it is quite reasonable to discard point No.4, No.5, No.7 and No.8 to get reliable tracking results.

Even though Figure 5.8 shows the points tracking frame by frame in the sequence to describe the tracking process, it is not necessary to do so in practice. The algorithms developed are based on the MRA concept, so that they can tolerate relatively larger viewport movement to the scene. Figure 5.9 shows matching points in frame 1 to frame 11 directly, using Daubechies-4 wavelet in the AWT. It is obvious that the result is acceptable compared to those in Figure 5.8: except point No.8, which is unfortunately selected in the fast changing water, other points are well matched.



Frame 1: Selected points Frame 11: Matched points

Figure 5.9 Matching points in Frame 1 to Frame 11.

In this chapter, the simple feature-rich points selecting and matching algorithms are developed using the AWT feature extraction algorithm, and experiments of tracking points in a real world image sequence with poor quality are done. To get more reliable point tracking results, the forward and backward matching method is suggested. The experimental results show that the AWT algorithm is effective in extracting point features that can be applied in 2D digital signal analysis.

Chapter 6 Conclusions

6.1 Summary

In this research, feature extraction in image analysis was studied. It was shown in Chapter one that wavelet decomposition is a feasible tool to extract features for image analysis. A literature survey of feature extraction using wavelet transforms in image analysis applications was given in Chapter two, where wavelet MRA is recognized as an effective tool for image feature extraction. In Chapter three, the concept of feature extraction by Multiresolution Analysis and its implementation in the form of wavelet decomposition was introduced. The Anchored Wavelet Transform was designed in Chapter four to improve the WT algorithm introduced in Chapter three for applications in image analysis such as feature identification and tracking. Through image feature-rich point allocating and tracking examples in Chapter five, we can see that the resulting AWT algorithm exhibits great effectiveness in directly extracting features at any location in the target image. In addition, results in this chapter also indicate that features extraction by AWT algorithm provides a tool not only for points tracking in image sequence, but also possibly for other image analysis application, such as object classification/recognition, surface segmentation and image registration.

6.2 Conclusions

With the AWT feature extraction algorithm, many benefits can be expected in image analysis applications. First of all, the features extracted represent the frequency component amplitudes in different frequency bands of the function spectrum, so that the noise in the image is isolated and can be depressed. Secondly, the features extracted measure the degree of correlation between the shape of the target function and the shape of the wavelets automatically in different scales. This is of profound benefit to image analysis because the relative information for a group of points in different sized windows of the image is quantified for any location in the image. At the finer scaling level, smaller windows are applied to the image to quantify the relative information among the points in the window under the location of interest, while at the coarser level, larger windows are applied to quantify the relation amongst more points under the location of interest. Finally the features extracted are obtained directly in a one-pass process with the AWT algorithm. Compared with a conventional WT feature extraction algorithm where reconstruction or interpolation is necessary in a decomposition-construction process, the AWT algorithm simplifies the feature extraction.

Although the AWT algorithm is only applied to image feature-rich point allocating and points tracking in this thesis, the algorithm was developed not only for this purpose. It is suitable for broader areas of signal processing. We do not foresee any difficulties in doing so.

The AWT algorithm does not employ sample decimation in the decomposition process and a set of features will be extracted for each image location. These algorithms impose relatively large demand on the power of a processing unit, and the size of the memory in a

conventional computer based computing platform. This constrains some on-line image processing application implementing the AWT algorithm.

In addition, there is a great need for an efficient algorithm to select the proper wavelet for each application, which is not addressed in this thesis. Therefore, although our research has shown promise for practical use, more effort is needed to bring the AWT approach from theoretical study to practical industrial applications.

6.3 Research Contributions

In this research the application of MRA and WT in image feature extraction is studied. A novel AWT is introduced into the process of feature extraction. As the core of the AWT, a non-shift wavelet decomposition model is developed with the objective to properly represent image properties with WT coefficients. AWT is an extension to ordinary WT methods; it inherits the flexibilities of WT and is a useful tool for feature extraction with its multiresolution analysis capability.

6.4 Suggestions for Future Work

Even though proved effective and promising by examples, the AWT feature extraction algorithm developed in this research is the first attempt in applying wavelet MRA directly in signal analysis, and implemented in a PC. The following research could improve the performance and practicability of the algorithm.

- An imperative task is to study the way of selecting the proper wavelets among the existing wavelets system, or designing custom wavelets for a particular application. The reason of doing so lies in the physical interpretation of the features extracted by

the AWT, which measure the shape of the target function according to the shape of the wavelet itself. When there is a particularly shaped object in the signal needing to be emphasized, a similar shaped wavelet could be used to optimally extract information of interest. In this thesis, different wavelets used in the AWT to extract features for points tracking give different results. This result also indicates that it is necessary to study the wavelet selection algorithm further.

- Concerning the implementation of the AWT in image processing, the computational load will grow with the increase of the size of the image signal. The feature extraction program running on a PC can only process relatively small-sized images due to the limited memory size and single CPU speed. A fully parallel computing model with large memory is necessary to merge the AWT into on-line image processing or into a portable facility. The ideal candidate for this is the implementation of the AWT into ASIC. Theoretically, the computation speed of parallel hardware is much faster than that of serial executed software. Hardware implementation of the wavelet decomposition will make the real time AWT feature extraction possible. Therefore, research work should continue in AWT microchip design.

References

1. Virginio Cantoni, Human and Machine Vision, Analogies and Divergencies, Plenum Press, A Division of Plenum Publishing Corporation, 1994.
2. A.K.Louis, P. Maab and A. Rieder, Wavelets, Theory and Applications, John Wiley & Sons Ltd., 1997.
3. Bin Tian, Mukhtiar A. Shaikh, Mahmood R. Azimi_sadjadi, Thomas H. Vonder Haar and Donald L. Reinke, A Study of Cloud Classification with Neural Networks Using Spectral and Textural Features, IEEE Transactions on Neural Networks, Vol. 10, No. 1, 138-151, 1999.
4. Boon Yi Soon, Mohammad S. Alam and Mohammad A. Karim, Improved Feature Extraction by Use of a Joint Wavelet Transform Correlator, Applied Optics, Vol. 37, No. 5, 821-827, 1998.
5. Charles K. Chui, An Introduction to Wavelets, Academic Press, Inc., 1992.
6. C. H. Chen and G. G. Lee, On Digital Mammogram Segmentation and Microcalcification Detection Using Multiresolution Wavelet Analysis, Graphical Models and Image Processing, Vol. 59, No. 5, 349-364, 1997.
7. Chu Yu and Sao-jie Chen, Design of An Efficient VLSI Architecture for 2-D Discrete Wavelet Transforms, IEEE Transactions on Consumer Electronics, Vol. 45, No. 1, 135-140, 1998.
8. C. Sidney Burrus, Ramesh A. Gopinath, and Haitao Guo, Introduction to Wavelets and Wavelet Transform, Prentice Hall, 1998.

9. Fausto Espunal, Terrance Huntsberger, Bjorn D. Jawerth and Toshiro Kubota, Wavelet-based Fractal Signature Analysis for Automatic Target Recognition, Optical Engineering, Vol. 37, No. 1, 177-174, 1998.

10. Gabriel Fernandez and Terrance Huntsberger, Wavelet-based System for Recognition and Labeling of Polyhedral Junctions, Optical Engineering, Vol. 37, No. 1, 158-165, 1998.

11. George S. Kapogiannopoulos and Manos Papadakis, Character Recognition Using a Biorthogonal Discrete Wavelet Transform, Proceedings of SPIE, Vol. 2825, Wavelet Applications in Signal and Image Processing IV, 384-393, 1996.

12. Hazem M, Hajj, Truong Q. Nguyen and Roland T. Chin, Proceedings of SPIE, Vol. 2825, Wavelet Applications in Signal and Image Processing IV, 330-341, 1996.

13. He-Ping Pan, General Stereo Image Matching Using Symmetric Complex Wavelets, Proceedings of SPIE, Vol. 2825, Wavelet Applications in Signal and Image Processing IV, 697-720, 1996.

14. Hui Henry Li and Yi-Tong Zhou, A Wavelet-Based Point Feature Extractor For Multi-Sensor Image Registration, Proceeding of SPIE, Vol. 2762, Wavelet Applications III, 524-534, 1996.

15. Jean-Christophe Olive, Joachim Deubler and Christian boulin,Automatic Registration of Images by a Wavelet-Based Multiresolution Approach, Proceedings of SPIE, Vol. 2569, Wavelet Applications in Signal and Image Processing III, 234-242, 1995.

16. J. L. Solka, D. J. Marchette, B. C. Wallet, V. L. Inwin and G. W. Rogers, Identification of Man-made Regions in Unmanned Aerial Vehicle Imagery and

Videos, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 8, 852-857, 1998.

17. Kai-Chieh Liang and C.-C. Jay Kuo, Progressive Indexing, Retrieval and Transmission of Wavelet Compressed Image Database, Proceedings of SPIE, Vol. 3169, Wavelet Applications in Signal and Image Processing V, 190-199, 1997.
18. Mark E. Lehr and Keh-Shin Lii, Template Basis Techniques to Pattern Recognition, Proceedings of SPIE, Vol. 2825, Wavelet Applications in Signal and Image Processing IV, 972-981, 1996.
19. Michael Unser, Ten Good Reasons for Using Spline Wavelets, Proceedings of SPIE, Vol. 3169, Wavelet Applications in Signal and Image Processing V, 422-431, 1997.
20. Mustafa Sakalli and Hong Yan, Feature-Based Compression Of Human Face Images, Optical Engineering, Vol. 37, No. 5, 1520-1529, 1998.
21. Mysore R. Raghuveer, Object Detection Through Matched Wavelet Transform, Proceeding of SPIE, Vol. 2762, Wavelet Applications III, 45-50, 1996.
22. Quyen Q. Huynh, Leon N Cooper, Nathan Intrator and Harel Shouval, Classification of Underwater Mammals Using Feature Extration Based on Time-Frequency Analysis and BCM Theory, IEEE Transactions on Signal Processing, Vol. 46, No. 5, 1202-1207, 1998.
23. Raghuveer M. Rao and Ajit S. Bopardikar, Wavelet Transforms, Introduction to Theory and Applications, Addison-Wesley Longman, Inc., 1998.
24. Ramana L. Rao and Lakshman Prasad, Image Segmentation by Multiresolution Histogram Decomposition, Proceedings of SPIE, Vol. 2569, Wavelet Applications in Signal and Image Processing III, 766-774, 1995.

25. Renu Tripathi and kehar Singh, Pattern Discrimination Using a Bank of Wavelet Filters in a Joint Transform Correlator, Optical Engineering, Vol. 37, No. 2, 532-538, 1998.

26. Robert E. Karlsen, Thomas Meiztler, Grant R. Gerhart and David Gorsich, Comparative Study of Wavelet Methods for Ground Vehicle Signature Analysis, Proceeding of SPIE, Vol. 2762, Wavelet Applications III, 314-321, 1996.

27. Robin N. Strickland and Douglas M. Zoucha, Object Detection Using Subband Decomposition, Optical Engineering, Vol. 37, No. 1, 320-330, 1998.

28. Sheng Zhong, Harold Szu, Francis Chin and Qing-Yun Shi, Constraints in The Wavelet Transform Domain for Stereo Vision Correspondence Matching, Proceeding of SPIE, Vol. 2762, Wavelet Applications III, 575-586, 1996.

29. Songyang Yu, Ling Guan and Stephen Brown, Automatic detection of clustered Microcalcifications in Digitized Mammogram Films, Journal of Electronic Imaging, Vol. 8, No. 1, 76-82, 1999.

30. Sylvain Deschenes, Yunlong Sheng and paul C. Chevrette, Three-dimensional Object Recognition from Two-dimensional Images Using Wavelet Transforms and Neural Networks, Optical Engineering, Vol. 37, No. 3, 763-770.

31. W. W. Boles and B. Boashash, A Human Identification Technique Using Images of the Iris and Wavelet Transform, IEEE Transactions on Signal Processing, Vol. 46, No. 4, 1195-1188, 1998.

32. William S. Meisel, Computer-Oriented Approaches to Pattern Recognition, Academic Press, Inc., 1972.

33. Xiao-Ping Zhang, Mita D. Desai and Ying-Ning Peng, Othorgonal Complex Filter Banks and Wavelet: Some Properties and Design, IEEE Transactions on Signal Processing, Vol. 47, No. 4, 1039-1048, 1999.

34. Milan Sonka, Vaclav Hlavac and Roger Boyle, Image Processing, Analysis, and Machine Vision, Second Edition, PWS Publishing, 1999.

Appendix A: Implementation of Experiments

In Chapter 5, two experiments were described to visually show the difference between applying WT and AWT on images, and to track points within image frames grabbed from a video clip.

In this Appendix, the implementations of these experiments will be described.

A.1 Data description

The data used for Section 5.2 is an image file DibSample.dib saved in the attached CD. DibSample.dib is a Device Independent Bitmap (DIB) with the following properties:

- 24 Bit Per Pixel (Bpp) RGB (Red Green Blue) bitmap. R, G and B component are all unsigned 8 bit integers,
- Width of the bitmap is 283 pixels,
- Height of the bitmap is 212 pixels,
- No compression applied on this image file.

The data used for Section 5.3 is an image sequence grabbed from an uncompressed video clip Nino83b.avi, which is also saved in the attached CD. The utility used to grab images from the video clip is AviEdit.exe. This executable is built from Microsoft Visual C/C++ 5.0 code, specifically, the sample project AviEdit. The AviEdit project and source code are also saved in the attached CD under the \AviEdit directory. The images grabbed from Nino83b.avi have the following properties:

- 24 Bpp RGB bitmap. R, G and B component are all unsigned 8 bit integers;

- Width of the bitmap is 240 pixels;
- Height of the bitmap is 180 pixels;
- No compression applied on this image file.

DIB is in Windows image format with the following data structure:

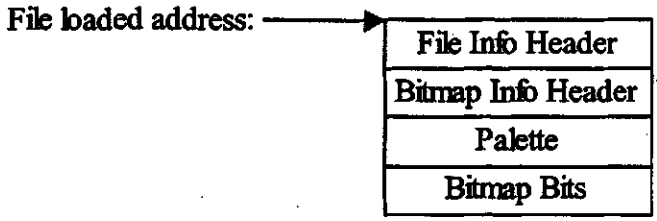


Figure A.1 Image file format.

Because the images used in the experiments are all 24Bpp, there is no palette used (palette is only used for 1/4/8Bpp images.). After the headers are stripped out, the address for the image bits is available.

A.2 Algorithm implementations

The algorithm implementations are in Microsoft Visual C/C++ project AWTa that were created during this project and saved in the attached CD with source code.

The implementations are composed of three entry functions: WT, AWT and PointTracking. As described in Section A.1, the input data element is an 8 bit unsigned integer. The transformed output is also 24Bpp images with 8 bit unsigned integer components. The interim data during calculation is of float data type, implicitly or explicitly.

A.2.1 User interface

To make the application easy to use, a very simple dialog based user interface was programmed through which the algorithm functions can be executed. The following procedures are used to execute the image processing.

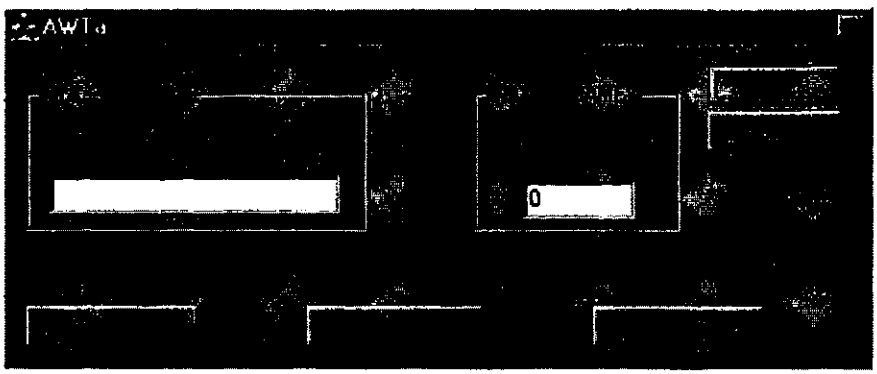


Figure A.2: The dialog window.

- FILE PATH/NAME: Input the full path and name of an image file to be processed into this edit box before pressing the WT/AWT button. Input full path, where a sequence of image files reside, into this edit box before pressing the Point Tracking button.
- TOTAL FRAMES: Input the number of frames to which the points tracking algorithm will be applied.
- WT: Pressing this button will transform the image using the standard WT algorithm. A new image file will be generated with 'WT' suffixed to its file name. For example, if the file name in FILE PATH/NAME is 'c:\thesis\DibSample.dib', then the new file generated will be named 'c:\thesis\DibSampleWT.dib' and the file will be saved in the same directory as the input image file.

- AWT: Pressing this button will transform the image using the AWT algorithm. Four new image files will be generated with 'LL', 'HL', 'LH' and 'HH' suffixed to their file names. The newly generated files will be in the same directory as the input image file.
- Point Tracking: Pressing this button will do the points tracking experiment described in Chapter 5. The input files must have a name using a digit starting from '0' and with the extension of '.dib'. For example, if FILE PATH/NAME is 'c:\thesis' and TOTAL FRAMES is 4, then the four files 0.dib, 1.dib, 2.dib and 3.dib will be used by this application. The same number of files will be generated in the same directory as the input files, with 'r' suffixed to each file name, such as 0r.dib, 1r.dib, 2r.dib and 3r.dib. The output files are the same image files as input, but with point number tags drawn in the image to show where the interesting points are, and their id numbers. Figure 5.8 gives an example of a point tag.

A.2.2 WT

WT(...) is the function implementing the standard WT algorithm to transform the image data. Figure A.3 is the flow chart of the WT(...) function.

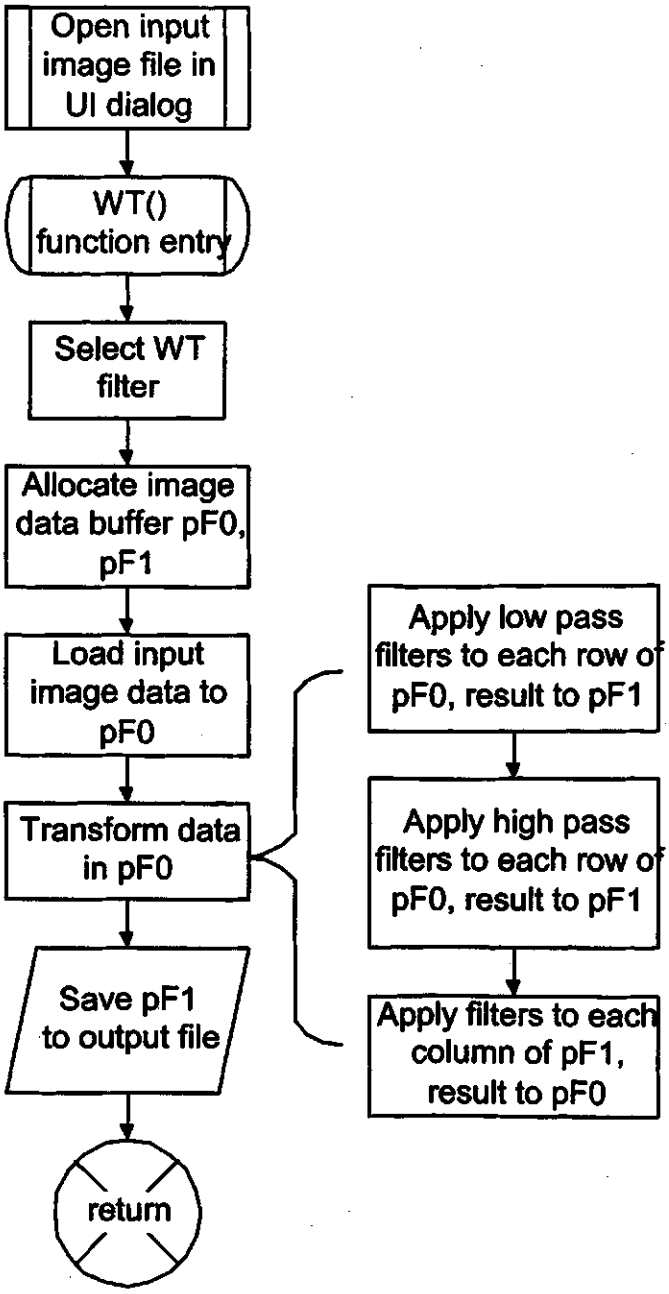


Figure A.3 Flow chart for WT(...) function.

A.2.3 AWT

AWT(...) is the function implementing AWT algorithm to transform the image data. Figure A1.4 is the flow chart of the AWT(...) and AWTConv(...) functions .

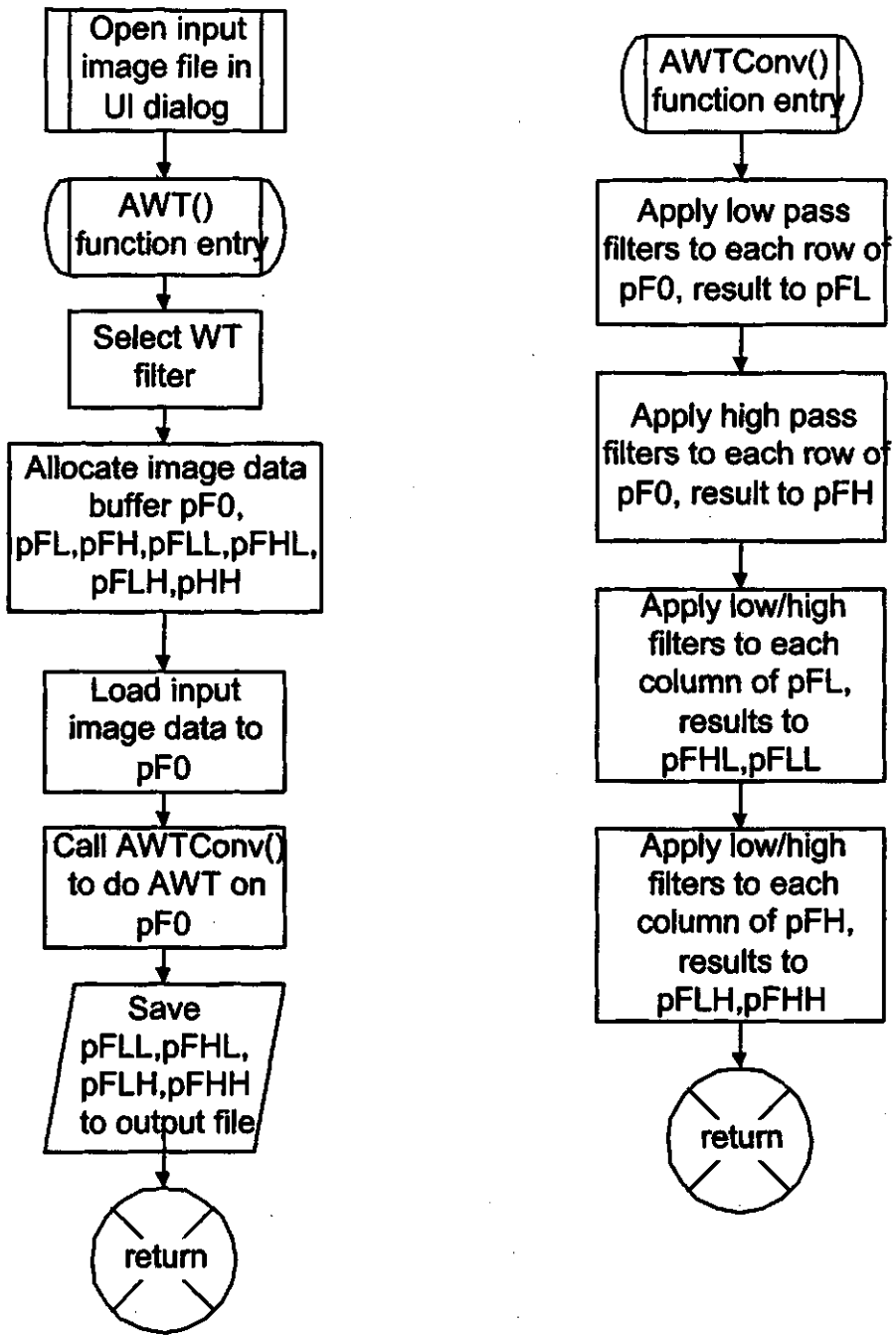


Figure A.4 Flow chart for function AWT(...) and AWTConv(...).

A.2.4 PointTracking

PointTracking(...) is the function implementing the points tracking experiment. Figure A.5 is the flow chart of the PointTracking(...) function. AWTConv(...) is used in this function to calculate the scale and wavelet coefficients which are used as point features.

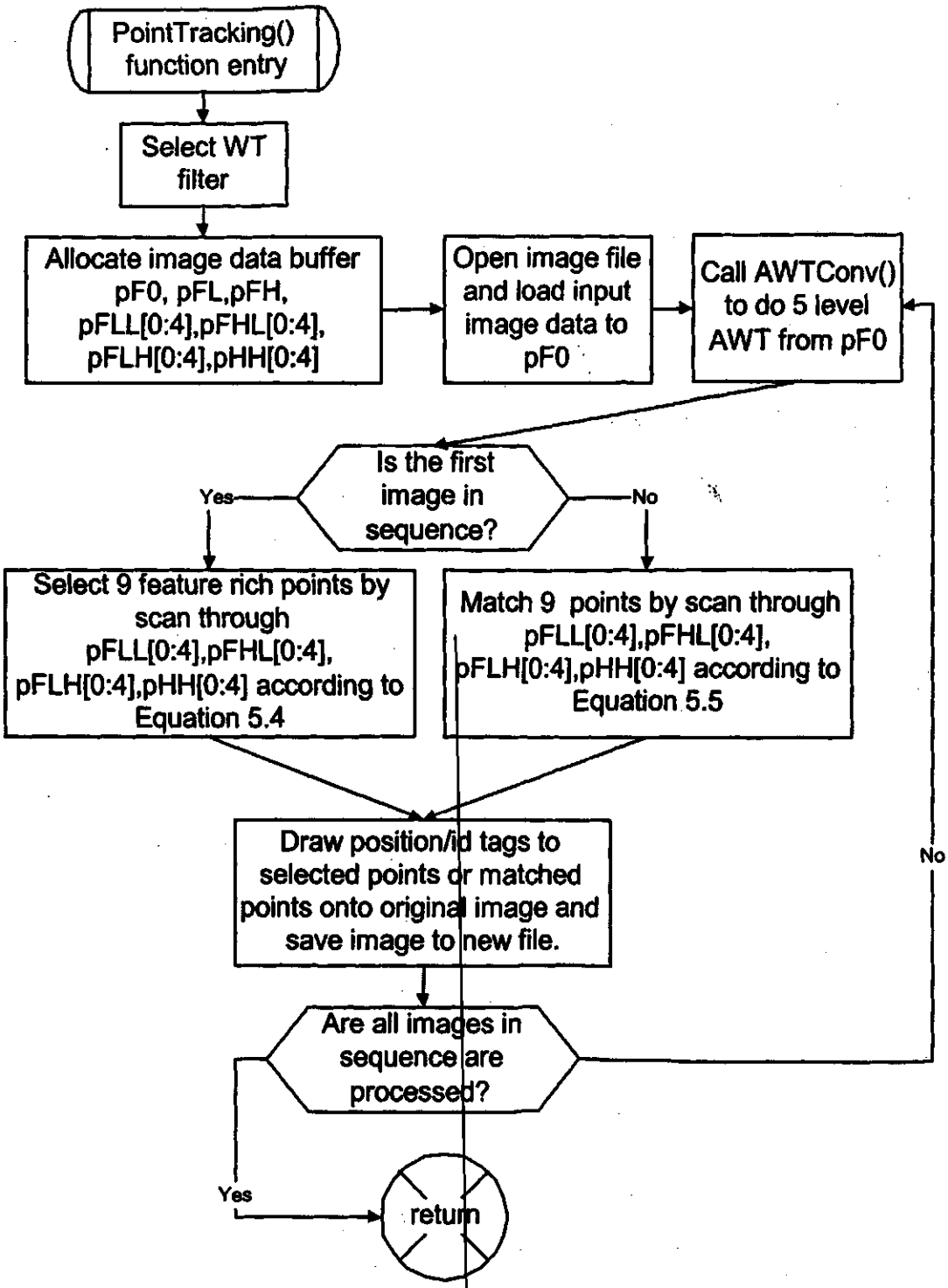


Figure A.5 Flow chart for PointTracking(...) function.

As shown in Figure A.5, after the feature-rich points are selected in the first image or matched points are found in subsequent images, a small rectangle id tag is drawn in the original image at the location associated with the tag. The marked images are saved to disk and can be opened by any image editing software application for viewing. For example, point (20,20) is selected in the first image and is marked with tag '0', then its matched point (x, y) in next image is also marked with tag '0'. The viewer can then make a subjective judgment whether two points with the same tag in two consecutive images are true matched points.

A.3 Algorithm implementation summary

The experiments conducted for this thesis are implemented in the form of a software application programmed in Microsoft Visual C/C++. To run the application, double click on AWTa.exe.

The code presented here is for the experiments described in Chapter 5. Other applications could utilize these algorithms' source code without much change. Extra coding and debugging efforts may be needed to apply those algorithms to other kind of images.

Following is the list of source files:
Algorithm.cpp,
Algorithm.h,
AWTa.cpp,
AWTa.h,
AWTa.rc,
AWTaDlg.cpp,
AWTaDlg.h
Dib.cpp,
Resource.h,

ShowDib.h,

StdAfx.cpp,

StdAfx.h.