Three-Dimensional Object Registration Using Wavelet Features

by

Julie S. Chalfant

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

PhD in Mechanical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2008

Author ....
Department of Mechanical Engineering
December 11, 2007

Certified by...
Nicholas M. Patrikalakis
Kawasaki Professor of Engineering
Thesis Supervisor

Accepted by ....
Lallit Anand
Chairman, Department Committee on Graduate Students
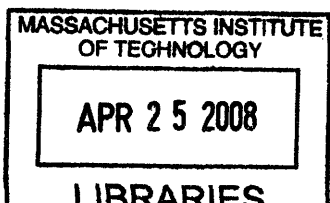
Three-Dimensional Object Registration Using Wavelet Features

by

Julie S. Chalfant

Submitted to the Department of Mechanical Engineering
on December 11, 2007, in partial fulfillment of the
requirements for the degree of
PhD in Mechanical Engineering

Abstract

Recent developments in shape-based modeling and data acquisition have brought three-dimensional models to the forefront of computer graphics and visualization research. New data acquisition methods are producing large numbers of models in a variety of fields. Three-dimensional shape-based matching and registration (alignment) are key to the useful application of such models in areas from automated surface inspection to cancer detection and surgery. The three-dimensional models in these applications are typically huge. State-of-the-art simulations in computational fluid dynamics produce upward of four terabytes of data per second of flow. Research-level magnetic resonance imaging (MRI) resolutions can reach 1 cubic micro-meter. As a result, object registration and matching algorithms must handle very large amounts of data.

The algorithms developed in this thesis accomplish automatic registration and matching of three-dimensional voxelized models. We employ features in a wavelet transform domain to accomplish registration. The features are extracted in a multi-resolutional format, thus delineating features at various scales for robust and rapid matching. Registration is achieved through seeking peaks in sets of rotation quaternions using a voting scheme, then separately identifying translation. The method is robust to occlusion, clutter and noise. The efficacy of the algorithm is demonstrated through examples from solid modeling and medical imaging applications.

Thesis Supervisor: Nicholas M. Patrikalakis
Title: Kawasaki Professor of Engineering

Acknowledgments

My deepest appreciation to Professor Nick Patrikalakis for his guidance, insight and patience as my thesis advisor, and to the ever helpful, encouraging, and discerning Professors Vivek Goyal and Dave Gossard and Doctor Wonjoon Cho as my thesis committee. Much thanks also to the postgraduates and students of the MIT Ocean Engineering Design Laboratory, especially Dr. Martin Reuter.

Special thanks and appreciation to Jack and Paul, without whom this would have taken half as long but been half as fun. And, of course, to Steve.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1   Background and Motivation

Recent developments in shape-based modeling and data acquisition have brought three-dimensional models to the forefront of computational research. New data acquisition methods are producing large numbers of models, leading to the proliferation of three-dimensional models available on the Internet and used in applications from manufacturing to medicine to gaming. Three-dimensional shape-based matching is key in such activities as inspection, computer-enhanced surgery, and computer vision. Data collection and storage techniques have advanced to the point that researchers today generate overwhelmingly large sets of data. Research-level magnetic resonance imaging (MRI) resolutions have reached 80 nanometers [66]. Model data for testing 3D algorithms range up to 69 million triangles [76]. Experimental techniques used for fluid dynamics can collect upwards of 4Gb of data per second of flow. State-of-the-art numerical simulations are producing 3 orders of magnitude more than that [103]. Object matching methods must be designed to handle the plethora of data provided.

A current area of very active research involves shape-based matching of 3D medical images obtained through methods such as MRI or Computed Tomography (CT), which produce voxelized three-dimensional models. Advances in speed and accuracy of such matching, especially given the large size of the data sets, could lead to further advances in areas such as computer-aided surgery, cancer detection, disease diagnosis,

and prosthesis fitting. While specifically developed with the medical imaging application in mind, the methods developed in this work are equally applicable to voxelized models in areas such as manufacturing, in which inspection procedures may include CT scanning. Indeed, any voxelized model or even a polyhedral model which has been voxelized can be addressed using these methods.

This research builds upon work by members of the MIT Ocean Engineering Design Laboratory who developed a procedure for shape-based matching using shape intrinsic fingerprints of 3D models in NURBS representation [39]. However, the algorithms and methods developed in [39] depend upon the NURBS format of the data and cannot be applied to polyhedral or voxelized models, which do not have the mathematical representation required for the classification of umbilics.

For rapid object matching, a reduction in the amount of data is extremely helpful. Transforms of scanned data from a three-dimensional object into an auxiliary space, e.g. via the wavelet transform, have high potential. Features in wavelet space can provide appropriate landmarks for registration, or alignment, and for similarity evaluation.

A major drawback to using the wavelet transform in object registration is its lack of invariance to rotation or translation. A newly developed transform, called the dual-tree complex wavelet transform (DTCWT) [37], is nearly invariant to translation and much less sensitive to rotation than the traditional discrete wavelet transform. We employ the DTCWT to accomplish matching of three-dimensional objects by extracting features in a multi-resolutional format, thus rapidly detecting and matching feature points.

1.2   Research Objectives

The objective of this thesis is to develop a shape-based registration technique for voxelized representations of solids using the wavelet transform. The key idea is to extract features in a multi-resolutional format for rapid matching of 3D models.

1.3 Thesis Organization

Chapter 2 provides an introduction to object matching and reviews research in the area. Chapter 3 introduces basic wavelet theory, describes recent research in wavelets, and describes the dual-tree complex wavelet transform. The object matching algorithms developed by this research are described in Chapter 4. The algorithms are applied to several examples in Chapter 5. Chapter 6 draws conclusions and makes recommendations for future work.

Chapter 2

Review of Previous Work

2.1 Object Registration

Object registration is the process of aligning images or objects to a common coordinate system. The sample objects may have undergone transformation such as translation, rotation or scaling, and the representation may suffer from occlusion, clutter, or noise, or may be only a part of the full model.

- Rigid registration describes the alignment of objects that have undergone translation and rotation only.

- Non-rigid registration includes scaling (uniform and non-uniform), affine motion, or morphing type transformations caused by such changes as growth, surgery or disease. It does not include articulation.

Registration can also be applied to such challenges as identification of objects in a scene and alignment of different views of an object to make a whole. Registration can be accomplished on a few objects for comparison to one another, or it can be used on a large number of objects to establish an atlas, or average, version which is then used to determine normal and abnormal differentiation among a population.

2.1.1 Applications.

Applications for object matching can be found in myriad fields. Examples include:

- Medicine: Registration of data from different modalities (e.g. MRI, CT, Ultrasound), from different patients, or from the same patient at different times, can assist in research, diagnosis and therapy. Specific applications include cancer detection, surgery planning, image guided and enhanced surgery, patient positioning for radiotherapy, research into effects of disease on the brain, and disease monitoring [13, 24, 81].

- Molecular Biology: Finding geometrically-possible combinations of molecule docking sites is the first step in the protein docking problem, which has applications in drug design, brain chemistry interactions, and disease methodology research [22, 7, 86]. In addition, partial matching can aid in detection of specific amino acids within a protein sequence [3].

- Computer vision: Object recognition from a scene including occlusion and clutter assists in robot task planning or pose estimation [82].

- Computer Aided Design: Data fusion of different views of an object into a single entity assists in reverse engineering either from actual parts or from physical models [4].

- Manufacturing: Automated surface inspection requires registration and matching as a key step [4].

- Cultural Heritage: Detection of man-made objects in shallow sub-surface radar or sonar data can aid in mapping archaeological sites on land and under water [83].

- Security: Some applications include reconstructing surfaces from separate views and automatic detection of items in aerial photographs or satellite data [3], and recognition of faces or other personnel features in video data.

- Military: Identification of land mines from shallow sub-surface radar data [108] and identification of underwater mines from sonar data [65] can lead to automatic detection.

- Meteorology: Tracking changing shapes over time is applicable for study of cloud formations in atmospheric data [29].

2.2   Methodologies

Object registration is the subject of a huge body of previous work. In the following, we will concentrate on methodologies for volumetric models and 3D free-form surfaces, represented as clouds of points, voxels, or triangulated models, including "polygon soups" in which the triangles do not necessarily align properly, leaving singularities, overlaps, or holes. Some 2D methods and parametric surface methods that are applicable to the problem statement will be described as well.

Major methodologies used in object registration can be classified as follows:

1. Moment theory involves pose estimation by matching center of mass and moments of inertia. While this is simple and fast, small variations in shape can cause similar items to have quite different orientation of principal axes. In addition, this method requires a complete model and cannot accommodate partial matching or polygon soup models. Moment theory is often used to initially pose two objects for further refinement using another method.

2. Registration using geometric features is the alignment and comparison of objects based on geometric features such as points, curvatures, ridges, segments, etc. Objects may have undergone rigid transformations (rotation and translation), scaling, skew affine motions, or even significant transformations such as lesions or growth. In general, these methods are accurate but time consuming. Some require initial correspondence information to avoid local minima.

   Although the term registration actually describes the alignment of objects to the same coordinate axis system, it is also used to describe the class of object matching methods that include registration as a first step.

3. Registration using voxel similarity is the alignment and comparison of objects based on comparisons of voxel representations including intensity, color, illumi-

nation and reflectance. The primary method used is mutual information, which is a statistical measure of similarity. These methods do not require initial segmentation, but may reach a local minimum. In addition, the global minimum may not be the best alignment. These methods are used often to compare data from different medical imaging modalities.

4. Deformable models are used in the search of a known object in a scene. A model is grossly fit to data then deformed using physically-based rules for closer alignment. This method requires feature detection to initially match the model to the image. It is used extensively in face recognition and medical imaging [59, 61].

The following sections review registration using geometric features and registration using voxel similarity in more detail.

2.2.1 Registration Using Geometric Features

Registration using geometric features methods align and compare objects using features ranging from single points to large segmented sections of the object. The gold standard of these methods is the Iterative Closest Point (ICP) algorithm presented by Besl and McKay in 1992 [5], which compares each point in the sample to the closest point in the model and finds a registration by minimizing the average Euclidean distance between points. Since this method iterates to a solution, it will find a local minimum, and therefore requires some initial correspondence information or a huge number of sample cases (essentially matching all possible rotations and translations) to find the correct solution. Numerous modifications to this method incorporate factors such as local shape [9], outliers and occlusion [107], and partial matching [58].

Hausdorff distance, which measures the greatest Euclidean distance instead of the average Euclidean distance between two sets of points, can be used similarly. Agarwal et al. [2] apply Hausdorff distance to the protein docking problem, but address only translation, not rotation or scaling. Guerra and Pascucci [20] use a modified

Hausdorff distance on line segments to match models to range data of objects that have undergone rigid transformations.

Geometric feature registration methods in general expand on this idea by extracting and matching like features. Many of these methods find a coarse registration which is then refined using a fine registration method such as ICP.

Augmented Points

Augmented point methods still match points in one set to points in another set. However, they reduce the number of points compared and increase the amount of information known about each point.

Chua and Jarvis [10] present an algorithm which determines full and partial matches of surfaces represented by point cloud data. They determine principal curvatures and directions to form a Darboux frame. Seed points are then matched to model points with curvature, distance and direction constraints and the optimal transformation is found. This method avoids local minima, but is slow and depends on a polynomial fit to determine principal curvatures.

Wang et al. [93] developed a non-rigid registration method using a feature point mesh to compare triangulated models of human brains. The algorithm finds points on the study brain corresponding to the atlas brain's selected and labeled points using an objective function measuring Euclidean distance, surface normal, and a curvature measure. This method is useful in the comparison of many similar objects, but requires initial manual processing including labeling points and triangulation, plus initial alignment.

Barequet and Sharir [3] present a method involving "footprints" tailored to the data type and computed at each point in the two surfaces. The footprints represent "amount of material" for volumetric models and surface curvature for surface models. Registration is achieved through a voting scheme that separates translation and rotation. This method is resistant to noise and does not depend upon surface derivatives. However, the method requires a descriptive footprint customized for each class of data, and may settle to a local optimum.

Feature Extraction

Sun et al. [82] present a method that computes a projection of geodesic rings onto the tangent plane at each point, called a "point fingerprint." Candidate points on the sample and model are selected using an irregularity measure, which is the ratio of largest to smallest radius in the tangent plane of a projected geodesic circle. This method uses geodesic distance and can incorporate additional information such as color, but may generate false correspondences.

Ko et al. [38] developed a method of full and partial matching of NURBS surfaces that uses the intersections of iso-curvature lines of Gaussian and mean curvature to establish correspondence between surfaces. This method solves a non-linear polynomial equation system using the Interval Projected Polyhedron (IPP) algorithm, which robustly determines a global minimum. The KH method can accommodate translation, rotation, scaling and partial matching, but requires a NURBS representation of the model.

Krsek et al. [40] developed an algorithm that uses the endpoints of connectors between lines of zero mean curvature as the points of interest to form a first level hypothesis, which is tested by performing ICP on all connector points in the two surfaces, then on all points in the objects. This method provides a good initial guess for ICP so a global minimum can be found, but it cannot register objects without prominent features and it is slow.

Stein and Medioni [77] developed a method that uses small surface patches and lines of discontinuities to compare surfaces. The surface patches are represented with a ring of surface normals called a splash, and the 3D curves are represented with polygonal approximations called super segments. Angles between normals for splashes and angles between sequential segments for super segments are collated in a hash table and candidate hypotheses are retrieved and verified by computing the transformation from model coordinates to sample coordinates and applying a least squares calculation on all matching features.

Ko et al. [39] presented a method that uses umbilics to match full and partial sur-

faces represented in NURBS form. Since umbilics are resistant to small perturbations of a surface, they will persist even with variations in the surface. There are generally few umbilics in a surface, so they are compared one by one to find the optimum transformation. This process requires umbilics, which are not always present, and which can be difficult to find in point cloud or polyhedral representations of surfaces.

Segmentation

Segmentation methods divide an object into segments, then compare the objects using the segments as feature points. Thus, the number of feature "points" is greatly reduced.

Dey et al. [16] developed a method that segments an object through the use of critical points of a height function. Registration is then carried out by matching every translation and every rotation, but the set of points is small (on the order of 10 per model). This method is invariant to translation, rotation, mirroring and scaling and robust against small variations in shape, but cannot accommodate partial matching.

Srinark et al. [75] use progressively simpler meshes as multiresolution surfaces, then divide those surfaces into patches which are used for matching. Rigid motion to match patches is used; some non-rigid motion between patches is allowed, as is occlusion and noise.

Dorai and Jain [18] proposed the segmentation of objects into Constant Shape Maximal Patches (CSMPs), or regions of like shape index, based on principal curvatures, whose average normal is mapped to a Gaussian sphere. A surface connectivity list records connectivity information about the segmented object. A shape spectrum is formed of shape index versus surface area. The shape spectrum can be used to quickly eliminate poor matches, then the connectivity list and Gaussian sphere map can be used for detailed object matching. One drawback is that full recoverability of the original shape is not guaranteed.

Mori et al. [51] propose the use of a cluster of invariant features called a Shapeme, in histogram form, for 2D object recognition. It is useful only for global matching. Shan et al. [69] extend this work to three dimensions and partial matching through

splitting models into sections and comparing all sections. This method is dependent on the selection of sections, and provides no registration information.

Full Object

Full object methods provide a 2D representation of the entire object at each single point. Theoretically a single point could be used to match two surfaces; however, the methods are generally used to produce point pairs which are then used for registration.

Johnson and Hebert [33] describe the position of all points on a surface relative to an oriented point by two parameters: distance from a line through the normal, and distance from the tangent plane. These parameters are collected in a 2D histogram, called a spin image. Spin images are computed in advance for every point in a model and stored in a database. Sample data is then correlated with the model data and point correspondences are established, then confirmed using ICP. This method requires no feature extraction or segmentation, but cannot accommodate partial matching or scaling and has a long registration time. Yamany and Farag [99] modified this to use simplex angle and Euclidean distance as the two parameters.

Spectral Methods

Spectral methods use a spectral transform such as Fourier or Wavelet, and measure distance between objects in spectral space. Several wavelet methods are used in classification, but few achieve registration.

Xue et al. [98] use a wavelet-based attribute vector (WAV) at each voxel in 3D MR images of brains to accomplish registration given an initial correspondence. The discrete wavelet transform (DWT) is used for object registration with initial correspondence information. This method is used to compare like objects and accomplishes object registration with warping, but will settle to a local minimum.

Yeom et al. [102, 31] use Gabor filtering on voxelized objects. The resulting features are matched using a dynamic link association which is composed of rigid motion of a graph followed by elastic deformation. Objects are classified into groups; however, no registration information is provided. In another application, Yeom and

Javidi [101] use principal component analysis and Fisher linear discriminant analysis to classify objects that have undergone Gabor filtering.

Wang and Chua [92] use rotation-invariant spherical 3D Gabor filters to correlate faces from 3D range data to 2D images. The matching is accomplished using the Hausdorff distance for comparison. This method is tailored to faces only.

Nelleri et al. [55] accomplish very preliminary three dimensional object recognition using the Mexican Hat wavelet, a rotationally symmetric 2D continuous wavelet, on complex 2D hologram representations of 3D objects. They are able to recognize squares and triangles. This method achieves rough classification, but no registration.

Laga et al. [41] use the L_2 distance between spherical wavelet coefficients as a similarity measure for database searching. They also use the spherical wavelet energy signatures as a 1D comparison; this is faster but less discriminating. Objects are first aligned using centers of mass and principal axes, and scaled using maximum distance from center of mass. This method does not achieve registration.

Numerous examples of the use of wavelets in 2D image matching exist as well, including database search [30, 91], object recognition [67, 54, 90, 21], registration [56, 11, 105, 104], iris recognition [50, 45, 73], and texture classification [8, 28, 106, 100, 63].

Combination Methods

Lowe [43] developed the Scale Invariant Feature Transform, or SIFT, which matches multiple types of features from one image to another, then draws lines between the correlated features. Although this is only applied in two dimensions, it is extremely fast, robust, and accurate.

Urschler et al. [85] combine corner detection, local (SIFT), and global (shape context) 3D descriptors to achieve non-rigid registration of medical images. They implement a 3D version of the SIFT algorithm described above that is not rotation invariant.

2.2.2 Registration Using Voxel Similarity Measures

Voxel similarity methods compare the overall similarity of voxel intensity between two objects. They often use probability density functions to minimize overall difference in order to achieve registration.

Partitioned intensity uniformity. Woods et al. [96] proposed a voxel similarity method for intermodality comparison of images. The basic assumption is that all pixels with a particular intensity in the first modality, A, represent the same tissue type; therefore, values of corresponding pixels in the second modality, B, should also be similar to each other. Thus, one parcels the voxels of A into bins of common intensity. The mean and standard deviation values of the corresponding voxels in B are then calculated and minimized for registration.

Joint entropy. Joint entropy methods minimize the entropy in a combined image. Since the entropy of two unrelated images will be the sum of the entropies of the two individual images, but the entropy of two aligned images will be less than the sum of the individual entropies, minimizing the joint entropy should achieve registration [24]. The global minimum may not be the best match; for example, alignment of most of the air or blank space in the images may reach a minimum without properly aligning the objects of interest. In addition, interpolation required for rotation of the image will change the PDF values, also possibly leading to incorrect registration. Partially overlapping datasets also create great difficulties.

Mutual information (MI). Much current research in medical imaging uses some form of MI [24]. MI methods overcome the problems of joint entropy by using the information contributed by each individual image along with the combined image information. Mutual information was simultaneously introduced for the assessment of medical images by Collignon [12] and Viola [88]. The method forms a 2D histogram of mutual information, which is maximized when the dispersion of significant correlation is minimized [23]. Usually one image is maintained static while the second image is

rotated and translated, then interpolated to determine values that correspond to nodes in the static image.

Normalized mutual information (NMI) methods overcome the dependence of mutual information on the volume of image overlap by normalizing. If the images have identical fields of view, MI and NMI perform equivalently. If the fields of view differ, normalizing with respect to the joint entropy of the overlap volume is considerably more robust than standard mutual information, according to Studholme et al. [80].

Numerous methods based on mutual information exist. Hastreiter and Ertl [23] worked to improve the speed of optimization by using hardware accelerated imaging operations. Wang et al. [89] used a cumulative distribution instead of a probability density function. Xu and Chen [97] attempt to overcome the lack of spatial information problem by including a gradient-based term in the mutual information analysis. Shekhar et al. [70] accomplish fully automatic elastic registration of PET and CT scans through an initial global rigid registration followed by elastic sub-volume registration.

Other methods. Many other voxel comparison methods exist, including cross-correlation, Fourier domain based cross-correlation, minimization of variance of grey values, histogram clustering, minimization of intensity differences, and maximization of zero crossings, as described in the review by Maintz and Viergever [46]. All share the advantage of using all available information throughout the registration process instead of selecting discrete features, but share the disadvantage of considerable computational costs.

Luan et al. [44] state that MI methods use only the quantity of pixels of any certain intensity, while the quality of those pixels has a great effect on registration as well. They add a qualitative analysis of pixel use to the MI methodology to achieve rigid registration. This method can accommodate large angles of initial misalignment.

Xu and Chen [97] combine wavelet decomposition with voxel similarity measures in that mutual information methods are applied to the approximation coefficients and spatial information methods are applied to the detail coefficients. They register 3D

23

voxelized objects that have been rotated about a single axis.

Malsch et al. [47] perform fully automatic non-rigid registration of voxelized objects by identifying anatomical landmarks in both images and matching them elastically. All points in the object are first classified as to tissue type using rough thresholds. Feature points are then selected as those points which have high variation around them in that many of the surrounding voxels are of a different tissue type. Initial rigid registration to a second image is accomplished using a voxel intensity method such as mutual information or cross correlation. Elastic registration is achieved using thin plate spline interpolation.

2.2.3 Summary Critique of 3D Object Matching Methods

Object registration and matching is an area of active research, with many new methods to improve speed and quality being developed. Tables 2.1 and 2.2 compare the capabilities of the methods described above.

In the matching of voxelized objects, such as required in medical imaging applications, the primary methods employed are voxel intensity methods. Most voxel comparison techniques are iterative, and can therefore settle in a local minimum. To avoid this, some methods use another approach such as moment theory for initial alignment, then use mutual information for fine registration [24].

A second problem is that the global minimum of voxel intensity methods can actually be incorrect. For example, by aligning only the air around an object, mutual information and joint entropy methods will give very high probabilities for all the air and zero probabilities for the remaining pixels, thus giving a very low entropy value - the global minimum, but obviously incorrect. Therefore, it is necessary to establish a comparison window within which a local minimum is sought [24].

In addition, the voxel intensity matching process is extremely slow, ranging from several minutes to hours, depending upon the size of the objects being registered.

Geometric feature methods solve many of these problems; they are somewhat faster and many do not settle into a local minimum; however, most are formulated for point cloud data or meshed objects.

	Surface Type	Registration Type	Partial Match	Scaling	Local Min
Geometric Feature Methods					
ICP [5]	PCD, Mesh	Rigid	No	No	Yes
ICRP [58]	PCD, Mesh	Rigid	Yes	No	Yes
Hausdorff Distance [2, 20]	PCD	Rigid	No	No	Yes
Darboux Frames [10]	PCD	Rigid	Yes	No	No
Feature Point Mesh [93]	PCD	Non-Rigid	Yes	Yes	Yes
Footprints [3]	PCD, Mesh	Rigid	Yes	No	No
Point Fingerprint [82]	Mesh	Rigid	Yes	No	No
KH Method [38]	B-Spline	Rigid	Yes	No	No
Lines of 0 H [40]	PCD	Rigid	Yes	No	No
Splash [77]	Mesh	Rigid	Yes	No	No
Umbilics [39]	B-Spline	Rigid	Yes	Yes	No
Dynamical Systems [16]	PCD	Rigid	No	Yes	No
Hierarchical Patches [75]	Mesh	Semi-Rigid	Yes	No	No
Spin Images [33]	Mesh	Rigid	No	No	No
Surface Signatures [99]	PCD	Rigid	Yes	No	No
WAV [98]	Voxel	Non-Rigid	No	No	Yes
Combination [85]	Voxel	Non-Rigid	No	No	Yes
Voxel Intensity Methods					
Mutual Information [88, 95]	Voxel	Rigid	Yes	No	Yes
Hardware Accelerated [23]	Voxel	Rigid	No	No	Yes
Cumulative Distribution [89]	Voxel	Rigid	No	No	Yes
Gradient Descent [6]	Voxel	Non-Rigid	No	No	Yes
PET/CT Auto Elastic [70]	Voxel	Non-Rigid	No	No	No
Block Match [47]	Voxel	Non-Rigid	Yes	No	Yes

Table 2.1: Capabilities of various object matching methods. The first column is the name of the method and cited paper. The second column is the surface type addressed; PCD indicates point cloud data. The third and fourth columns state whether the method can achieve partial matching and matching across different scales, respectively. The fifth column indicates whether the method will settle into a local minimum.

	Year	Time
Geometric Feature Methods		
ICP [5]	1992	650 sec
Hausdorff Distance [2]	2003	NR
Darboux Frames [10]	1996	70-153 sec
Feature Point Mesh [93]	2003	NR
Footprints [3]	1997	73-340 sec
Point Fingerprint [82]	2003	NR
KH Method [38]	2002	225 sec
Lines of 0 H [40]	2002	NR
Splash [77]	1992	35-1800 sec
Umbilics [39]	2003	NR
Dynamical Systems [16]	2003	NR
Hierarchical Patches [75]	2006	NR
Spin Images [33]	1997	415 sec
Surface Signatures [99]	2002	120 sec
WAV [98]	2003	NR
Combination [85]	2006	1600-2200 sec
Voxel Intensity Methods		
Mutual Information [88, 95]	1995	400 sec
Hardware Accelerated [23]	1998	400 sec
Cumulative Distribution [89]	2003	NR
Gradient Descent [6]	2000	150 sec
PET/CT Auto Elastic [70]	2005	2700-4500 sec
Q-MI [44]	2008	NR
Block Match [47]	2006	1000 sec

Table 2.2: Speed of registration methods. The times listed are those reported in the literature; NR indicates that computation time was not reported. Note that the times cannot be compared one-for-one; they are greatly affected by object size and computer platform. Years of publication are included to assist in comparison; older methods are likely to have been run on slower computers but with smaller models.

A method which accomplishes registration of voxelized objects, without initial correspondence, despite occlusion and clutter, and especially one that reduces the impact of large data sets, is needed.

Wavelet transforms provide a potential method to accomplish this. Wavelet methods to date have used data-intensive forms such as Gabor wavelets, continuous wavelet transforms, or steerable pyramids. A less data-intensive method such as the discrete wavelet transform would be ideal, provided that invariance to rotation and translation could be accomplished.

Chapter 3

Theoretical Background

3.1 Review of the Wavelet Transform

The wavelet transform is a multiresolution method of analyzing data which decomposes an image according to scale while maintaining localized information. Basic wavelet theory is well established and has been implemented in many commercially available products such as the MATLAB Wavelet Toolbox. Current research in wavelet theory is expanding areas such as translation and rotation invariance, irregular grids, stability, non-separable 2D wavelets, and wavelets tailored to specific shapes, such as spherical wavelets. A discussion of current research into invariant forms is included in Section 3.2.

A basic introduction to the wavelet transform is included here; more details can be found in texts such as those by Addison [1], Daubechies [14], Stollnitz, DeRose and Salesin [78] or Strang and Nguyen [79]. Those already familiar with the wavelet transform may want to proceed to Section 3.3 for a description of the Dual-Tree Complex Wavelet Transform.

3.1.1 Continuous Wavelet Transform (CWT)

Mother Wavelet. A mother wavelet, $\psi(t)$, is a function that can be used as a wavelet; i.e., it meets the admissibility conditions of zero mean, finite energy, and

finite admissibility coefficient:

$$\text{Zero mean (no DC component)} \qquad \int_{-\infty}^{\infty} \psi(t)\, dt = 0$$

$$\text{Finite energy (localized)} \qquad \int_{-\infty}^{\infty} |\psi(t)|^2\, dt < \infty$$

$$\begin{array}{c}\text{Finite admissibility coefficient}\\ \text{(inverse transform exists)}\end{array} \qquad 0 < C_g = \int_{0}^{\infty} \frac{|\hat{\psi}(f)|^2}{|f|}\, df < \infty,$$

where $\hat{\psi}(f)$ is the Fourier transform of the wavelet function. The mother wavelet is dilated and translated by various values a and b respectively:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right).$$

Normalizing by $a^{-1/2}$ conserves energy (so that wavelets of different scales have the same energy). Thus, for each scale a,

$$\int_{-\infty}^{\infty} |\hat{\psi}_{a,b}(f)|^2\, df = \int_{-\infty}^{\infty} |\hat{\psi}(t)|^2\, dt.$$

Wavelet transform. This translated and dilated wavelet is then convolved with a signal, $x(t)$, to produce the wavelet transform [1]:

$$T_{a,b} = \int_{-\infty}^{\infty} x(t) \psi_{a,b}^*(t)\, dt \qquad (3.1)$$

where ψ^* is the complex conjugate of ψ, and $T_{a,b}$ are the wavelet coefficients. The inverse wavelet transform is [1]:

$$x(t) = \frac{1}{C_g} \int_{-\infty}^{\infty} \int_{0}^{\infty} T_{a,b} \psi_{a,b}(t) \frac{da\, db}{a^2},$$

where C_g is the finite admissibility coefficient defined above.

This implementation of the wavelet transform is known as the Continuous Wavelet Transform (CWT), since it involves the integral of a continuous function over all scales and locations. Figure 3-1 shows the CWT coefficients, $T_{a,b}$, of a signal, $x(t)$, using the Mexican Hat wavelet, $\psi(t)$.
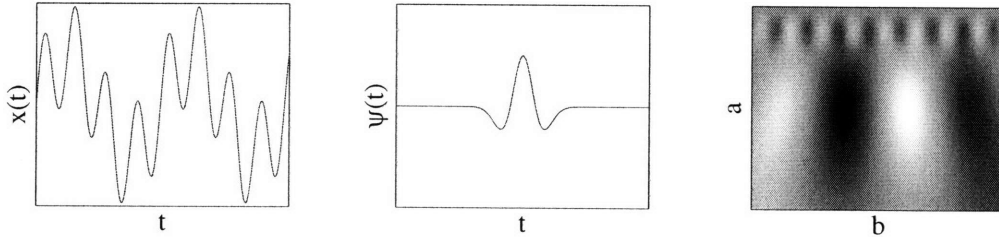
Figure 3-1: The Mexican Hat wavelet $\psi(t)$ (shown center) is applied to the signal $x(t)$ (shown left), resulting in the CWT coefficients $T_{a,b}$ (shown right) at various scales, a, and locations, b.

Multi-Dimensional Wavelets. Two-dimensional wavelets can be constructed as above, using a coordinate vector, $\mathbf{t} = (t_1, t_2)$. The two-dimensional wavelet transform is then given by

$$T_{a,\mathbf{b}} = \frac{1}{a} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(\mathbf{t}) \psi^* \left(\frac{\mathbf{t} - \mathbf{b}}{a} \right) d\mathbf{t},$$

where $\mathbf{b} = (b_1, b_2)$. This can be expanded to D dimensions through the use of D-dimensional coordinate vectors and weighting factors of $a^{-D/2}$.

Practical Application. In theory, the CWT is applied continuously over the entire signal and the entire range of scales. When used in practice, it is applied at discrete locations and discrete scales, which are specifically selected to accentuate the features being analyzed. The CWT can be computed using a Fourier transform of the wavelet and of the signal, applying the convolution theorem. Even at discrete scales and locations, the CWT is extremely data intensive, because, at <u>each</u> scale, it retains a full set of data the same size as the original signal.

3.1.2 Discrete Wavelet Transform (DWT)

The Discrete Wavelet Transform (DWT) is a less data intensive method with a fast implementation algorithm ($O(n)$, where n is the number of data points in the signal) [1].

The most common form of the DWT is discretized on a dyadic scale [1] where the size of the steps between locations, ($n2^m$), is directly linked to the scale, (2^m). Thus, equation (3.1) becomes:

$$T_{m,n} = \frac{1}{\sqrt{2^m}} \int_{-\infty}^{\infty} x(t)\psi^*\left(\frac{t - n2^m}{2^m}\right) dt.$$

When an orthonormal wavelet basis is chosen, the original signal can be perfectly reconstructed without redundancy, using the wavelet coefficients.

Dyadic discrete wavelets have associated scaling functions, ϕ, of the same form as the wavelets:

$$\phi_{m,n}(t) = \frac{1}{\sqrt{2^m}} \phi\left(\frac{t - n2^m}{2^m}\right), \tag{3.2}$$

but which have a mean equal to 1, instead of zero:

$$\int_{-\infty}^{\infty} \phi(t)\, dt = 1. \tag{3.3}$$

Figure 3-2 shows several wavelet functions with the associated scaling functions.

The convolution of the scaling function with the signal produces a smoothed version of the signal:

$$S_{m,n} = \int_{-\infty}^{\infty} x(t)\phi_{m,n}(t)\, dt.$$

The scaling coefficients, $S_{m,n}$, are referred to as approximation coefficients, since they provide a lower resolution approximation of the signal. The wavelet coefficients, $T_{m,n}$, are known as detail coefficients, since they provide the detail required to obtain the original signal from the smoothed signal. The original signal can be reconstructed

Haar   Daubechies 4   Daubechies 8
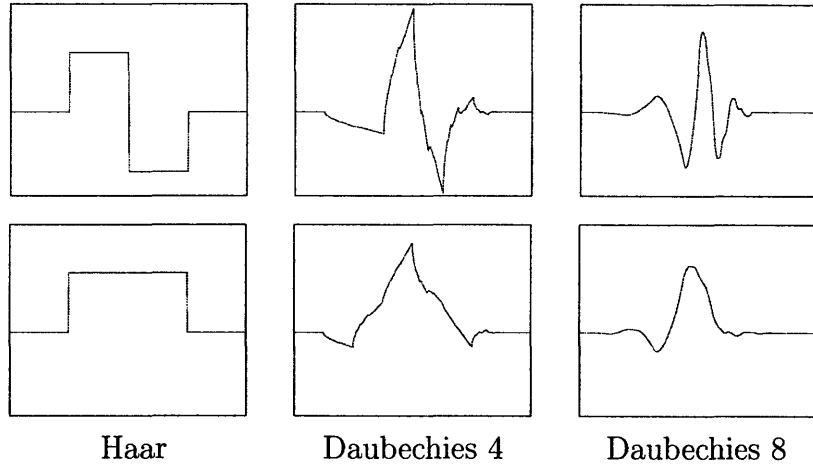
Figure 3-2: Discrete wavelet functions (top row) and their associated scaling functions (bottom row).

from the approximation coefficients at a scale m_0 plus all detail coefficients from m_0 back to the original signal. This is represented as:

$$x(t) = \sum_{n=-\infty}^{\infty} S_{m_0,n} \phi_{m_0,n}(t) + \sum_{m=-\infty}^{m_0} \sum_{n=-\infty}^{\infty} T_{m,n} \psi_{m,n}(t),$$

and is represented graphically in Figure 3-3. Figure 3-4 shows a signal along with its approximation and detail coefficients at several levels of decomposition.
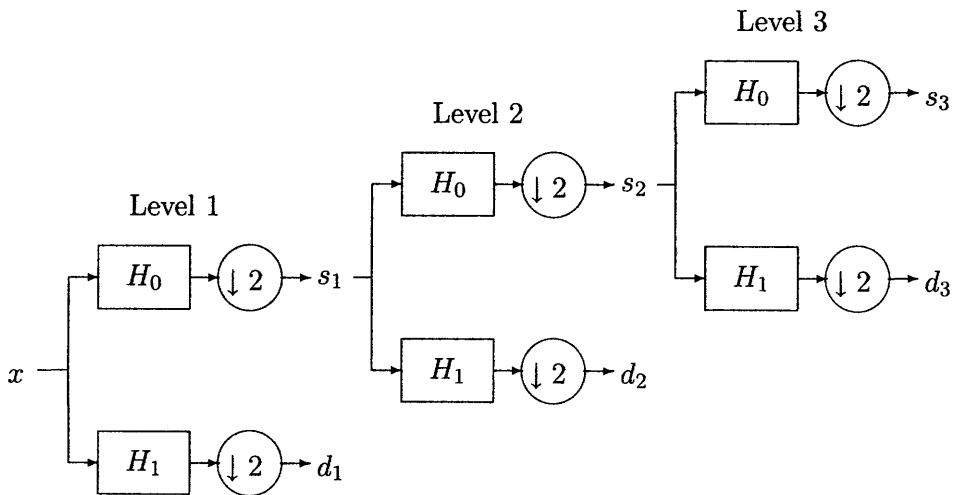
Figure 3-3: The wavelet decomposition of a signal, x. H_1 is the wavelet function, producing detail coefficients d after downsampling by 2. H_0 is the scaling function, producing approximation coefficients s. The signal x can be reconstructed by combining s_3 with d_3, d_2, and d_1, by combining s_2 with d_2 and d_1, or by combining s_1 with d_1. Adapted from [37].

32
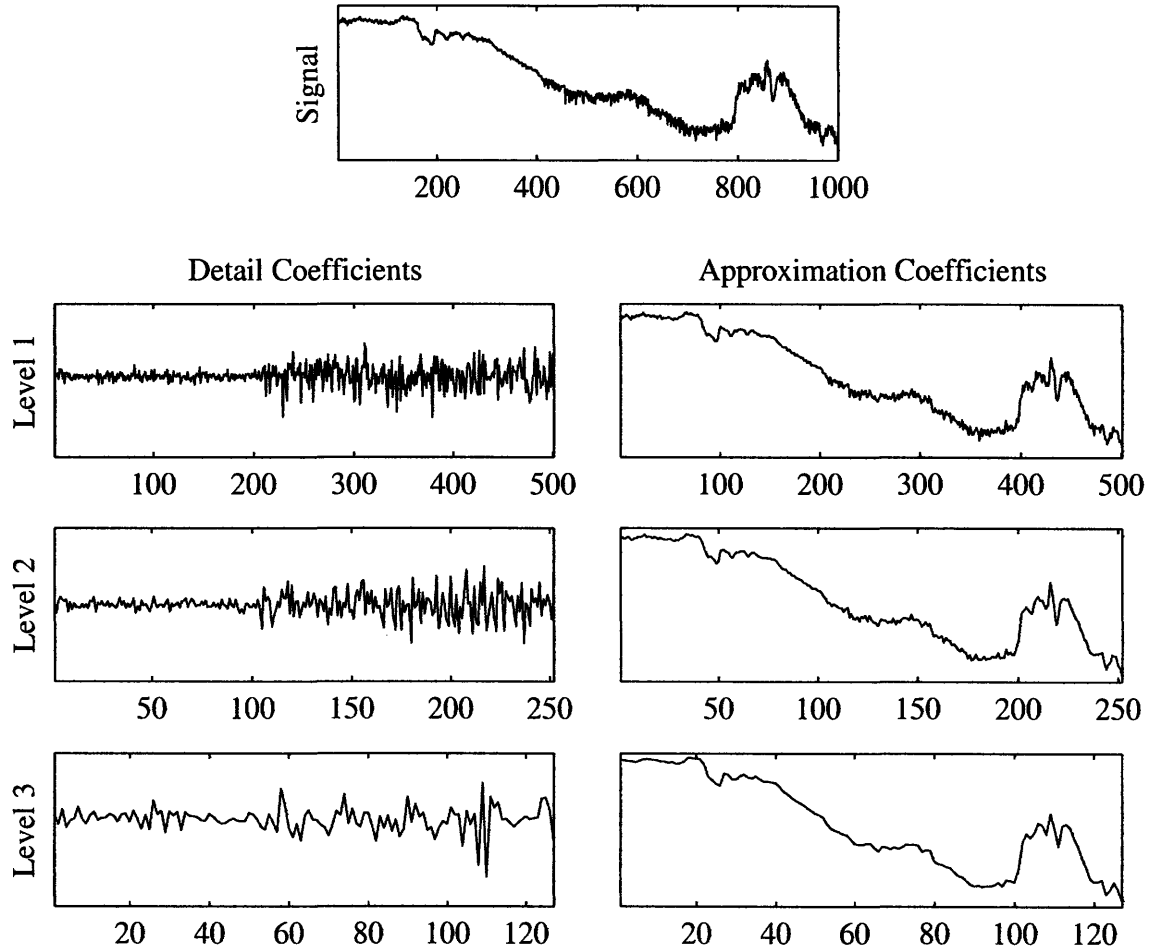
Figure 3-4: A signal is shown in the top row. Each subsequent row is composed of the wavelet decomposition of the signal into detail (left) and approximation (right) coefficients. Note that the approximation at any one level is formed by removing the detail at that level from the approximation at the next higher level.

3.1.3 Two-Dimensional DWT

The most common form of the two-dimensional DWT is composed of tensor products of the one-dimensional DWT, resulting in the following four wavelets:

- 2D scaling function: $\phi(t_1, t_2) = \phi(t_1)\phi(t_2)$

- 2D horizontal wavelet: $\psi^h(t_1, t_2) = \phi(t_1)\psi(t_2)$

- 2D vertical wavelet: $\psi^v(t_1, t_2) = \psi(t_1)\phi(t_2)$

- 2D diagonal wavelet: $\psi^d(t_1, t_2) = \psi(t_1)\psi(t_2)$

These "separable" wavelets are termed horizontal, vertical and diagonal because they accentuate features in the horizontal, vertical and diagonal directions, respectively. An image and its wavelet decomposition using separable wavelets are shown in Figure 3-5.
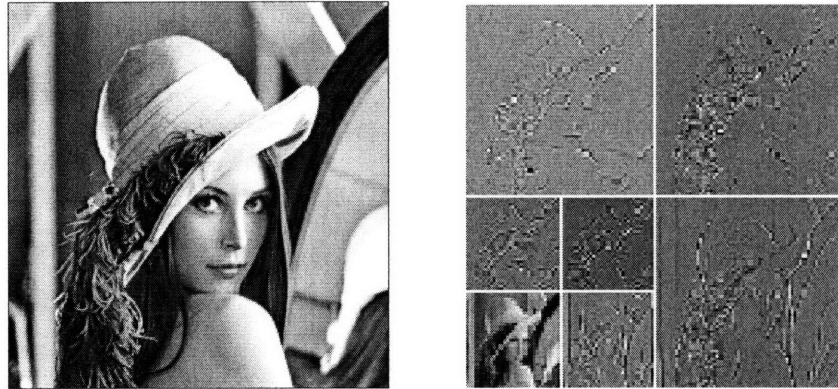
Figure 3-5: Two-level, 2D DWT with separable wavelets. The left block contains the original image [53]. The right block contains, clockwise from the top left: first-level horizontal wavelet coefficients, first-level diagonal wavelet coefficients, first-level vertical wavelet coefficients, and second-level coefficients (clockwise from top left: horizontal, diagonal and vertical wavelet coefficients, then second-level approximation coefficients).

It is also possible to construct non-separable wavelets which are designed in two dimensions and use other grid forms. An example using a quincunx grid, which is structured like the pattern of 5 spots on a die, is shown in Figure 3-6. A wavelet decomposition using non-separable wavelets is shown in Figure 3-7.

```
X  O  X  O  X  O              O     O     O          O     O     O
O  X  O  X  O  X           O     O     O
X  O  X  O  X  O  ⟹           O     O     O  ⟹     O     O     O
O  X  O  X  O  X           O     O     O
X  O  X  O  X  O              O     O     O          O     O     O
O  X  O  X  O  X           O     O     O
```
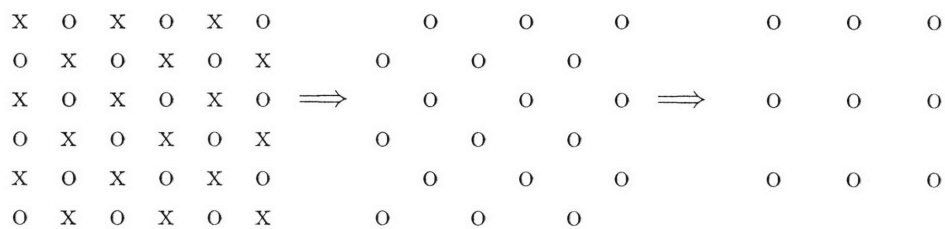
Figure 3-6: A demonstration of downsampling by two, alternating between rectangular and quincunx grids. This sample begins as a rectangular grid. Removing all the x's leaves a quincunx grid of o's. Removing alternate rows of o's returns to a rectangular grid.

Figure 3-7: Two-level, 2D DWT with non-separable wavelets on quincunx grid. The left block contains the original image. The right block contains, clockwise from the top right: first-level wavelet coefficients, second-level wavelet coefficients, second-level approximation coefficients.

3.2 Invariance

One major drawback of the DWT is that it varies with both translation and rotation; a small shift in location of the signal can result in a large change in the corresponding wavelet coefficient values. Since the signal is downsampled at each level of wavelet decomposition, the data points to which the wavelet is applied are different when the signal is shifted by other than the coarsest scale used in the transform. This can produce quite different wavelet coefficients for signals that differ only by translation; see Figure 3-8.

Tensor product wavelets are strongly oriented with the axes, and are therefore not rotation invariant. In 2D, the horizontal wavelets accentuate horizontal features in an image, the vertical wavelets accentuate vertical features in an image, and the diagonal wavelets accentuate diagonal features in an image, for both diagonals. See Figure 3-9. Even non-separable wavelets are not rotation invariant because non-separable discrete wavelets are not isotropic.

The continuous wavelet transform (CWT), since it does not include downsampling, is translation invariant if the signal is shifted by an integer multiple of the grid spacing, and is rotation invariant if an isotropic wavelet is used; however, the CWT is very memory intensive: a full set of coefficients with the same number of data points as the original signal or object is generated at each scale. Since we will be operating on data

Input	Level 1	Level 2	Level 3
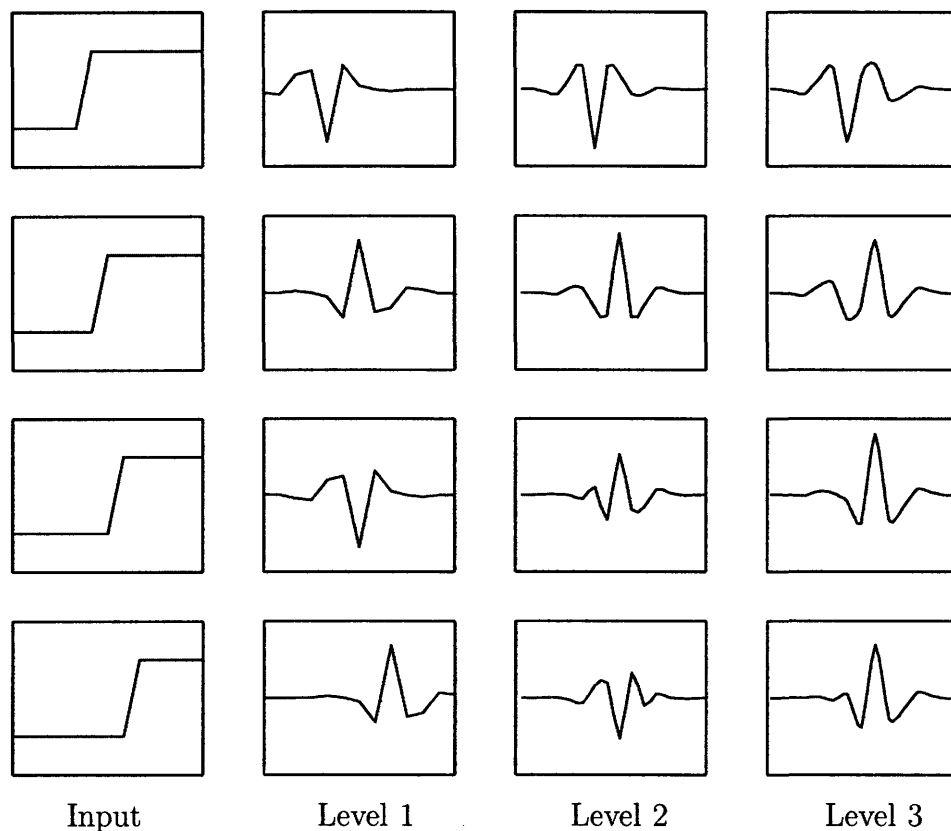
Figure 3-8: Translation variance of the DWT. The left column shows a step function, shifted to the right by one sample in each row. The next three columns show the wavelet coefficients at the first, second and third levels of decomposition. Note that the wavelet coefficients do not show a corresponding shift to the right; instead, the shape of the plots changes radically with a one step shift in the input. Adapted from [37].

sets that can be very large, we desire to use the efficiencies of the discrete wavelet transform. Much current research into wavelets addresses the issues of rotation and translation invariance.

3.2.1 Invariance; Review of Previous Work

The DWT can be made translation or 'shift' invariant to a single grid spacing by omitting downsampling, and can be made nearly shift invariant by averaging several DWTs over a range of signal time origins; however, these methods result in the same data intensity problem found with the CWT. In order to achieve some degree of rotation invariance, some methods use averages of the three bands to achieve fea-

36

Figure 3-9: Rotation variance of the DWT. Note that the top right hand image accentuates the horizontal lines of the mill, the bottom right hand image accentuates the diagonal lines (both +45° and -45°), and the bottom left hand image accentuates the vertical lines. If the mill were rotated, then the coefficient values in each subsection would be quite different.

ture vectors, and other methods attempt to design relatively isotropic non-separable wavelets.

Pun and Lee [42] developed a non-redundant, shift-invariant wavelet transform by averaging four redundant wavelet transforms at each level of resolution. The computational complexity of this method is $O(n \log n)$. However, it is not rotation invariant.

Simoncelli et al. [35, 72] developed steerable pyramids, a redundant frame representation which is rotation and translation invariant, but which is data intensive.

Mrázek and Weickert [52] presented a translation and rotation invariant method that combines wavelet transforms with nonlinear diffusion filters. The iterated filter is not completely rotation invariant, but the asymmetry decreases with a higher number of iterations.

Hill et al. [25] presented a rotation-invariant feature vector based on Kingsbury's DTCWT. They used the six oriented sub-bands of the complex wavelet output and summed energies to produce isotropic, rotationally-invariant feature vectors. This is only applicable to an entire image or large sections of an image.

Do and Vetterli [17] presented contourlets, which are non-separable 2D wavelets designed on a discrete basis. However, they are specifically designed to be anisotropic in order to capture directionality in images.

Van De Ville et al. [87] designed quasi-isotropic polyharmonic B-splines which are non-separable, multi-dimensional basis functions on a quincunx grid. The functions become more isotropic with increased order. They are not shift invariant.

Tymczak et al. [84] designed a set of non-separable multi-wavelets which are nearly isotropic. This format is based on the lifting scheme and thus works well for image processing. They are not shift invariant.

Kingsbury [36, 37] proposed the Dual-Tree Complex Wavelet Transform (DTCWT) which downsamples only after the second level of wavelet decomposition, thus producing two similar trees, offset by one grid spacing. This method produces twice the data of the dyadic DWT in one dimension, but is still much less data intensive than the CWT. In addition, it is much more resistant to variance with rotation than the DWT. This transform meets our requirements of reduced sensitivity to both translation and rotation. It is described in more detail below.

3.3 Dual-Tree Complex Wavelet Transform

Kingsbury [36, 37] proposed the Dual-Tree Complex Wavelet Transform (DTCWT) which downsamples only after the second level of wavelet decomposition, thus producing two parallel trees. Subsequent decompositions evenly interleave data in the two redundant trees. Advantages of the DTCWT include [37]:

- Approximate shift invariance

- Reduced sensitivity to rotation

- Limited redundancy, independent of the number of levels of resolution

- Efficient computation

Approximate shift invariance can be achieved using a real DWT by retaining the samples that would be removed during downsampling at the first level. These samples are placed in a separate, parallel tree. In subsequent levels of decomposition, retained samples in the two trees are evenly interleaved. The samples of the two trees do not interleave evenly if the same wavelet is used on each tree; the samples must be offset by using different wavelets on each tree. See Figure 3-10.

Kingsbury [37] suggests two methods of achieving approximate shift invariance at lower levels: odd/even filters and q-shift filters. The odd/even method uses odd-length wavelets on one tree and even-length wavelets on the other tree, alternating at each level to promote greater symmetry. This method works, but creates a somewhat asymmetric sub-sampling structure, produces different frequency responses in the two trees, and must use biorthogonal wavelets.

The q-shift method fixes all three of these problems. Kingsbury designed a wavelet which is even length but not strictly linear phase, thus allowing perfect reconstruction using orthonormal wavelets where the reconstruction wavelets are the exact reverse of the analysis wavelets. These wavelet filters are designed to have a group delay of one-quarter sample. The time reverse has a group delay of three-quarters sample, thus producing a half-sample difference between the two trees when used as shown in Figure 3-11. Details of the design of the q-shift filter can be found in [37]. An example q-shift wavelet and scaling function are shown in Figure 3-12.

Kingsbury stores the outputs of the two trees as the real and imaginary parts of complex wavelet coefficients; thus, he labels his method the Dual-Tree Complex Wavelet Transform.

An example of the near shift invariance of the DTCWT, contrasted with the shift variance of the DWT, can be seen in Figure 3-13.

The DTCWT is extended to d dimensions by separably applying the wavelets in each direction. This requires an additional redundancy in each dimension; 2^d redun-

Real DWT:

Input samples	x	x	x	x	x	x	x	x	x	x	x	x
Level 1	a	b	a	b	a	b	a	b	a	b	a	b
Level 2	a	·	·	b	a	·	·	b	a	·	·	b
Level 3	a	·	·	b	·	·	·	·	a	·	·	b

Odd/even DTCWT:

Input samples	x	x	x	x	x	x	x	x	x	x	x	x
Level 1	a	b	a	b	a	b	a	b	a	b	a	b
Level 2	·	a	·	b	·	a	·	b	·	a	·	b
Level 3	·	a	·	·	·	b	·	·	·	a	·	·

Figure 3-10: An example of the sampling location for a real DWT with sampling rate doubled and an odd/even DTCWT, creating trees a and b. 'a' and 'b' indicate the location of the sample used in the wavelet decomposition in tree a and b respectively. Note that the samples are evenly spaced at all levels of decomposition in the DTCWT but not in the DWT beyond the first level. Adapted from [36].
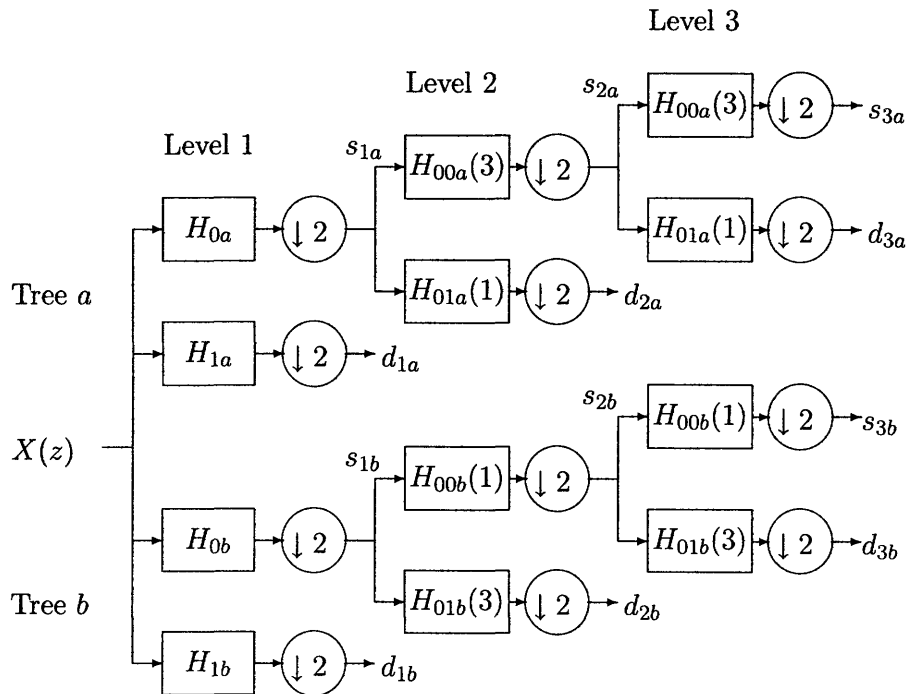
Figure 3-11: DTCWT using q-shift wavelets. H_1 and H_0 are the first-level discrete wavelet and scaling function filters respectively. H_{01} and H_{00} are the q-shift wavelet and scaling function filters; (1) indicates a one-quarter shift and (3) indicates a three-quarter shift. Adapted from [37].

Scaling Function   Wavelet
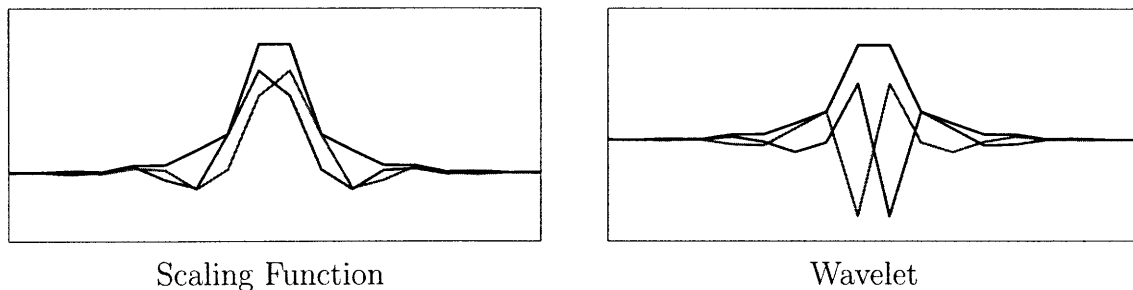
Figure 3-12: Q-shift wavelet and scaling function. Blue lines are tree a, red lines are tree b, and black lines are $|a + jb|$.

dant trees are retained, where d is the dimensionality [36]. Note that the redundancy depends only on the dimensionality, not on the size of the dataset or the number of levels of decomposition.

The shift invariance of the one-dimensional DTCWT continues in higher dimensions; Figure 3-14 demonstrates this in two dimensions. A white circular disk is shifted one pixel to the right in each of three subsequent images, as shown in the left column of Figure 3-14. The level three DWT wavelet coefficients for each of these shifted images are shown in the second column. The third column shows each set of wavelet coefficients subtracted from the one in the previous row; if the DWT were shift invariant, the resulting image would be blank. The fourth and fifth columns show the corresponding DTCWT wavelet coefficients and the subtracted, shifted wavelet coefficients. Note that the DTCWT is not completely shift invariant; at a higher level of resolution, artifacts can be seen.

A quantitative analysis of shift invariance is provided in [37] and is summarized in Appendix A. The quantitative analysis indicates that the DTCWT is approximately 10 to 30 dB more invariant to translation, depending on the q-shift wavelet selected.

Although not rotation invariant, the DTCWT provides good directional selectivity in higher dimensions. In two dimensions, the DTCWT produces twelve sub-images strongly oriented in six directions: ± 15 deg, ± 45 deg, and ± 75 deg [37], as compared to real wavelets which produce three sub-images: two strongly oriented horizontally and vertically, and the third including components from both diagonals. Higher dimensional DTCWTs produce $(4^d - 2^d)/2$ directional sub-bands at each level [37],
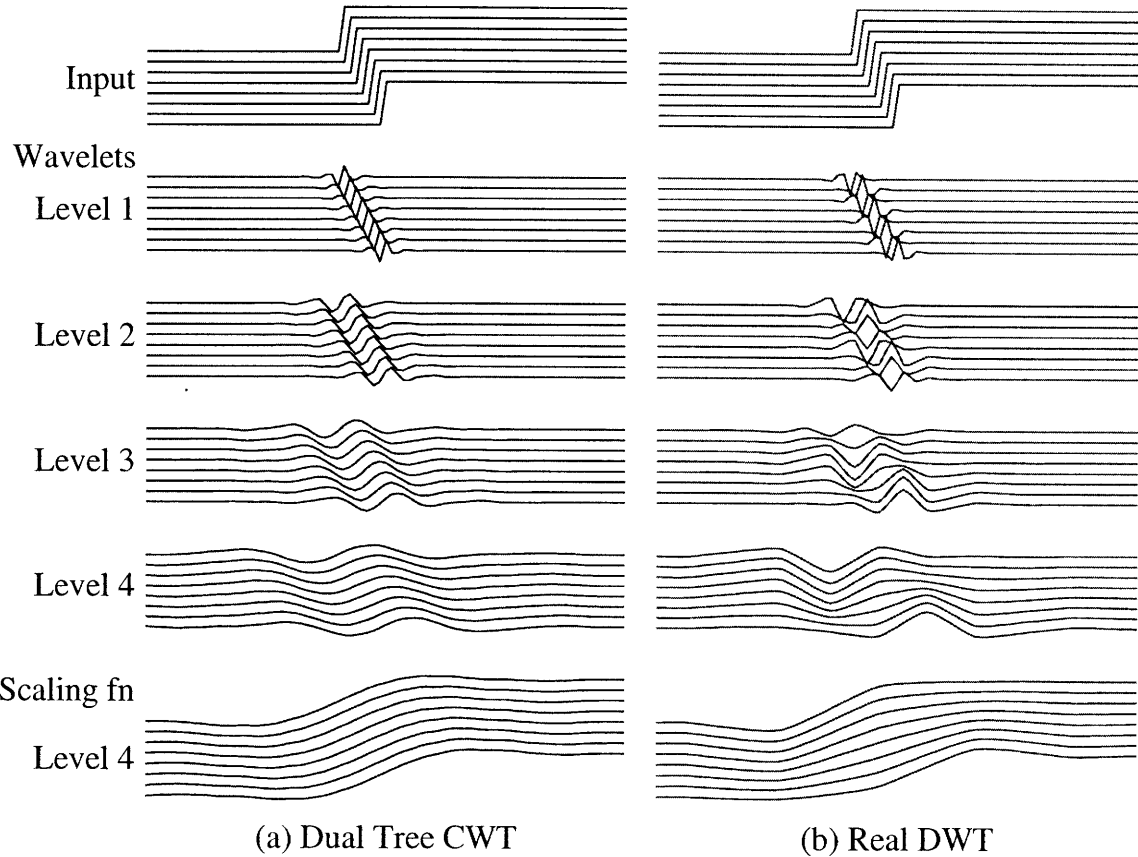
Figure 3-13: The left column is the DTCWT; the right column is the DWT. The top row is the input: eight step functions, each shifted one sample to the right. The next four rows contain the wavelet coefficients at four levels. The bottom row contains the scaling coefficients at the fourth level of decomposition. Note that the DTCWT is nearly shift invariant while the real DWT is not. Adapted from [37].

where d is the dimension. This directional selectivity can be seen in Figure 3-15, where the 2D reconstructed wavelets for the DWT (bottom row) and DTCWT (top row) can be seen. Note that the DWT has significantly greater incidence of artifacts.

The approximate shift invariance and reduced sensitivity to rotation make the Kingsbury DTCWT an excellent candidate for use in object matching.
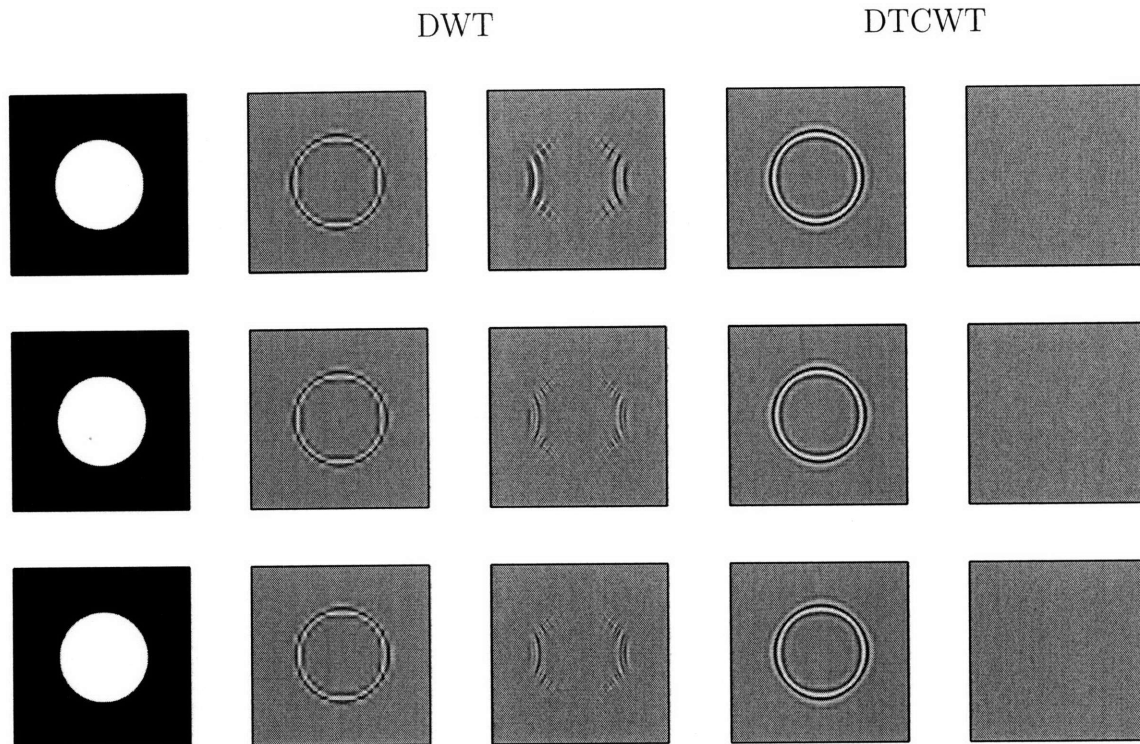
Figure 3-14: Shift variance of the 2D DWT and DTCWT. The left column is the input; three white disks on black backgrounds, each shifted one pixel to the right from the preceding row. The second column is the reconstruction of level three DWT wavelet coefficients for the corresponding input. The third column shows the wavelet coefficients shifted back to the original location and subtracted from the previous row; if the DWT was shift invariant, the resulting image would be blank. The fourth and fifth columns show the wavelet coefficients and the subtracted, shifted wavelet coefficients for the DTCWT.
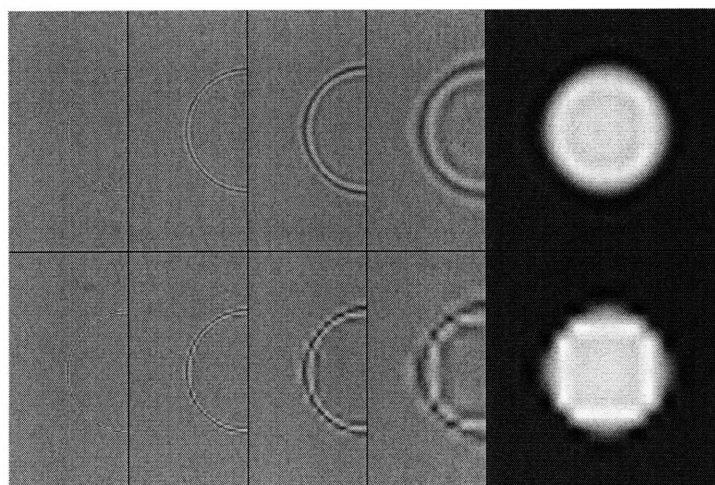
Figure 3-15: Rotation invariance of the DTCWT (top) and DWT (bottom). Adapted from [37].

Chapter 4

Object Registration

In this section we present an algorithm that employs the wavelet transform to accomplish global, rigid, shape-based registration of three-dimensional voxelized objects which have undergone translation and rotation. The method accommodates noise, which can be induced in models through events such as the sensing process, transformation and voxelization. Occlusion and clutter are accommodated as well. We do not include scaling, shearing, warping, or other motions that distort the appearance or functionality of an object.

Given two voxelized models, A and B, we determine the rotation and translation such that:

$$A = \mathbf{R}B + \vec{T} \tag{4.1}$$

where $\mathbf{R}$ is the orthogonal rotation matrix and $\vec{T}$ is the translation vector.

An overview of the algorithm is as follows:

1. Apply a wavelet transform to both objects.

2. Select large magnitude wavelet coefficients in A and B as feature points.

3. Search within the lists of feature points for matching triplets of points.

4. For each matching set of triplets, determine the rotation.

5. Use a voting scheme to select the best rotation.

6. Using the selected rotation, determine the translation.

At the completion of the algorithm, the quality of the resulting match between A and the rotated and translated B can be assessed.

The steps of the algorithm and the design choices developed therein are described in the following sections.

4.1 Wavelet Transform

As discussed in Section 3.2, a transformation that is invariant to rotation and translation (or nearly so) is desired for object registration. If the transformation results vary greatly with rotation and/or translation of the object, then corresponding feature points in each model will have completely different signatures and will be useless for registration purposes.

At the same time, a transformation that is computationally simple with minimal data requirements is desired, especially when working on the very large data sets that are common today. For example, the Mexican Hat Wavelet is a symmetric, 2D wavelet which is thus invariant to rotation; however, it is a continuous wavelet transform with the corresponding huge data sets and slow computation, which greatly reduces its utility.

Kingsbury's Dual-Tree Complex Wavelet Transform (DTCWT) [37], described in detail in Section 3.3, is nearly invariant to translation and is much more resistant to rotation than the discrete wavelet transform (DWT), while maintaining the $O(n)$ computation of the DWT with only a 2^d increase in data size, where d is the dimensionality. The increase in date size can be eliminated by not retaining the first level wavelet coefficients. For these reasons, we use the DTCWT as the wavelet transform in this algorithm.

Within the DTCWT family, there are a number of filters which we can choose from. For the first level discrete wavelet filter, we select the nearly symmetric fil-

ter with the shortest length. For the second- and greater-level filter, we select the shortest-length q-shift filter. The short-length filters were selected so that abrupt changes in the composition of the object would be identified. Longer filters bring in voxels over a greater range, giving weight to areas beyond edges and thus blurring the distinct change. This affects feature point selection. Further effects of filter selection is an area recommended for future research.

4.2 Feature Points

Applying the three-dimensional DTCWT to each model produces, at each level of resolution, an approximation (scaling function coefficients) and detail (wavelet coefficients). There are several possibilities for sources of feature points, listed below and shown in Figure 4-1:

- The original object, with no wavelet transform applied.

- The scaling function coefficients at some specific level of resolution.

- The wavelet coefficients at a single or at multiple levels of resolution.

- The gradient of the wavelet coefficients.

From the two-dimensional example shown in Figure 4-1, it can be seen that the scaling coefficients can contain large swaths with extreme values; note the large contiguous areas of black or white shown. This is especially true for surface models with a constant value for the entire interior. In contrast, the wavelet coefficients highlight changes in intensity and therefore produce more discrete extrema, making them a good choice for feature points.

Heuristic tests upon many three-dimensional models at many different orientations confirm this. The wavelet coefficients produced the highest percentage of feature points that matched in location from one model to another and were thus the most useful for object registration. Therefore, we use the extrema of the wavelet coefficients as feature points.

The three-dimensional DTCWT produces 28 complex sub-bands of wavelet coefficient information; instead of using 28 separate data sets, we reconstruct one level of wavelet coefficients, excluding the average coefficients, in order to produce a single set of real wavelet coefficient data.

Figure 4-2 is a two-dimensional example of feature point selection using the reconstructed wavelet coefficients. Note that obvious points such as the intersections of lines are automatically selected as features.

Another consideration is whether to select features based on magnitude only or whether to include the sign. As the maxima highlight points where the shape of the wavelet corresponds well with the shape of the object, and the minima highlight points where the shape of the wavelet corresponds extremely poorly with the shape of the object, we match only maxima with maxima and minima with minima. Therefore, at each level of decomposition, we retain two lists of feature points: those with large magnitude and positive sign, and those with large magnitude and negative sign. This has the added advantage of providing two separate, shorter lists of points for comparison, thus reducing the computation time of the algorithm.

4.2.1 Invariance

Although the DTCWT is nearly shift-invariant and much less affected by rotation than other methods, it is not completely invariant to either rotation or translation. Therefore, even though many of the DTCWT wavelet coefficient extrema identify the same features in each object, not all features are identified in both objects. This results in a set of feature points with many outliers.

As an example, we separately apply the DTCWT and the the discrete wavelet transform (DWT) to a two-dimensional image that has been translated by 42 pixels in the x and y direction, and to another that has been rotated by 30 degrees counterclockwise. We then compare the number of corresponding maxima of the wavelet coefficients at the third level of decomposition; 73-80% of the DTCWT feature points correspond, while only 40-47% of the DWT feature points correspond. See Figures 4-3 and 4-4 for a visual comparison and Table 4.1 for a numerical comparison.
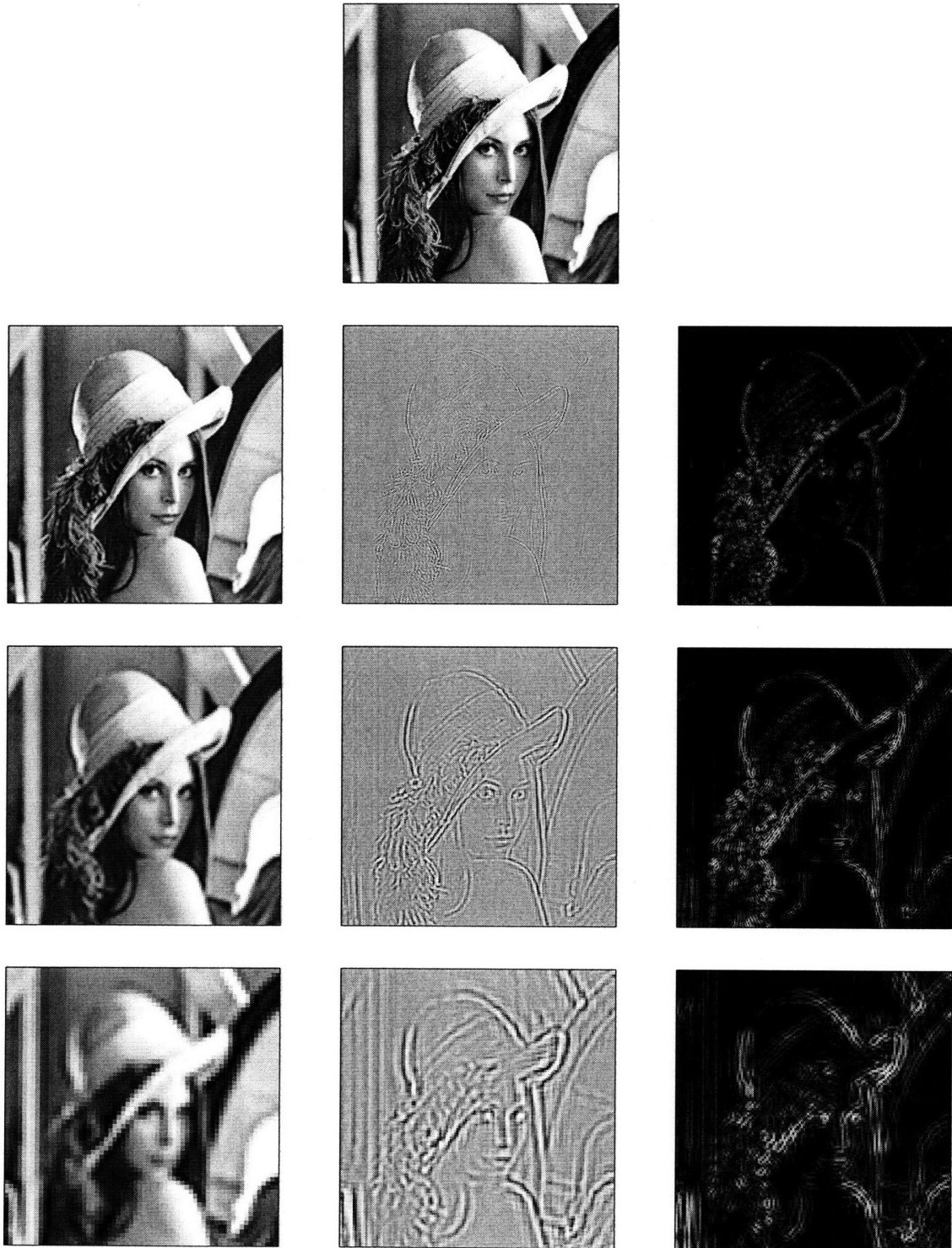
Figure 4-1: A two-dimensional example of the DTCWT. The top figure is the original image. The three subsequent rows are the level one, two and three decompositions in that order, with average (scaling function) coefficients on the left, detail (wavelet) coefficients in the center, and the gradient of the wavelet coefficients on the right.
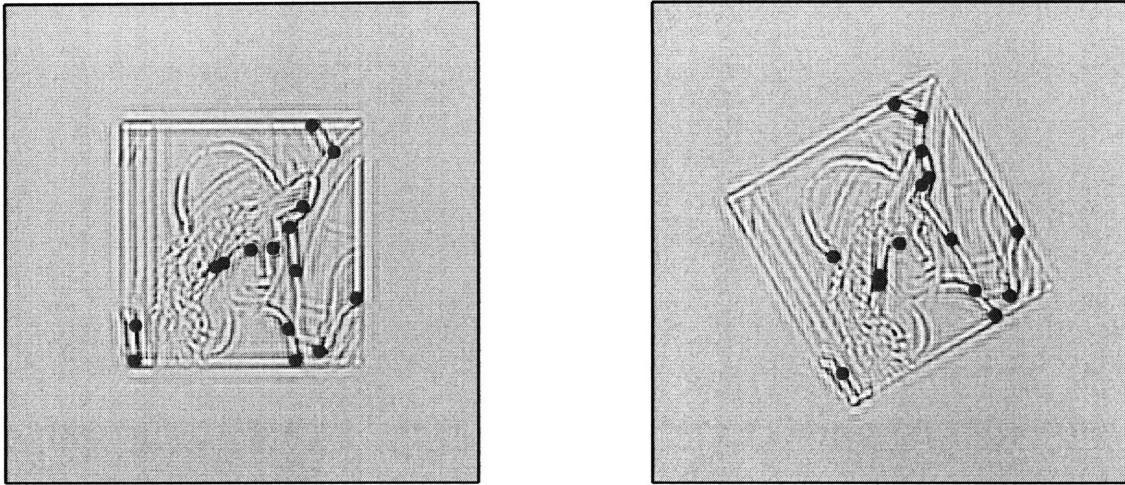
Figure 4-2: A two-dimensional example of extrema in the DTCWT. The figure on the left shows the level three reconstructed wavelet coefficients of the original image. The right-hand figure shows the the level three reconstructed wavelet coefficients of the rotated and translated image. Extrema are denoted by red dots.

It is unlikely that the feature points correspond by chance. Appendix B derives the method for calculating the probability that k of m points correspond in position in an object of size n, given a size p pixel (or voxel) ring. Table 4.1 includes the probability of randomly selected feature points matching by chance given the size of the objects and the number of feature points selected in the examples shown in Figures 4-3 and 4-4.

	DWT	DTCWT
Translation		
number	6	12
percent	40	80
probability	1.52×10^{-4}	1.85×10^{-14}
Rotation		
number	7	11
percent	47	73
probability	9.45×10^{-6}	2.66×10^{-12}

Table 4.1: Comparison of feature point correspondence for DWT and DTCWT in translation and rotation. Number of corresponding feature points, percent of total feature points, and probability of matching that number of points by chance are shown for the third level DWT or DTCWT decomposition of two translated or rotated versions of the Lena image, shown in Figures 4-3 and 4-4. Fifteen total maxima were selected in each image for comparison.
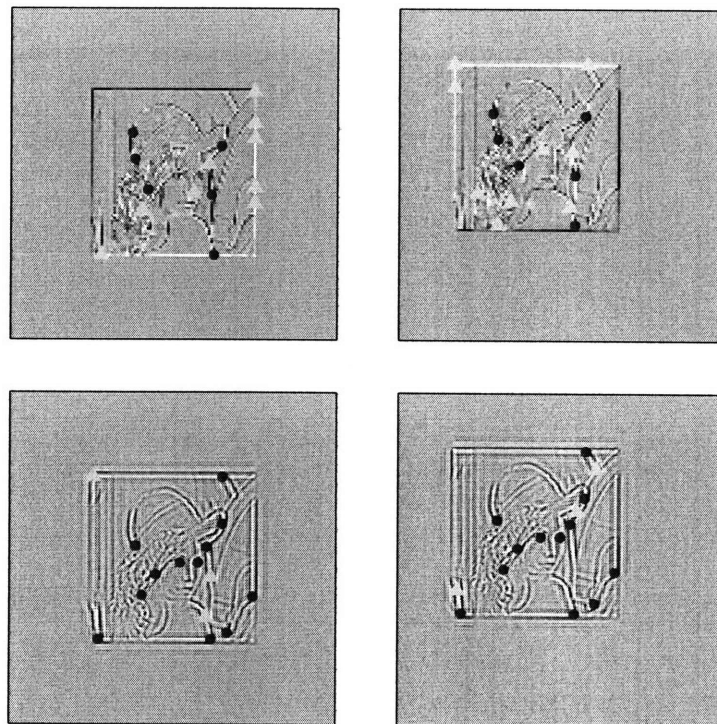
Figure 4-3: Comparison of DWT and DTCWT extrema in translation. The right-hand image was translated -42 pixels in the x and y directions prior to application of the transform. The top row is the DWT; the bottom row is the DTCWT. The 15 wavelet coefficients with the greatest value (positive only) in each image are marked. Corresponding feature points are denoted by red circles; extrema which do not correspond in location are denoted by yellow triangles.
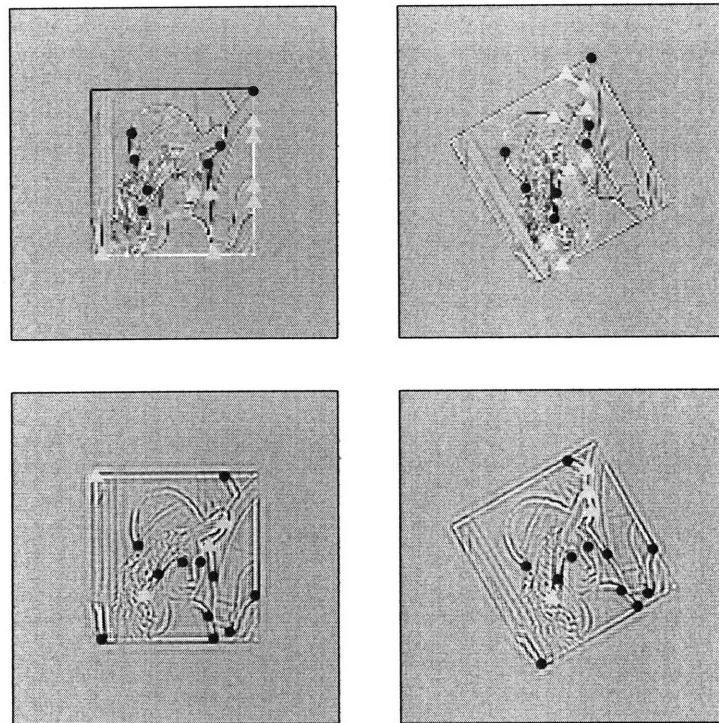
Figure 4-4: Comparison of DWT and DTCWT extrema in rotation. The right-hand image was rotated 30 degrees counterclockwise prior to application of the transform. The top row is the DWT; the bottom row is the DTCWT. The 15 wavelet coefficients with the greatest value (positive only) in each image are marked. Corresponding feature points are denoted by red circles; extrema which do not correspond in location are denoted by yellow triangles.

4.2.2 Wavelet Coefficient Magnitude

A one-to-one correspondence of wavelet coefficients based on coefficient value cannot be used to pinpoint the location of a feature. The magnitude of the wavelet coefficient is different at corresponding points in two objects, and the order of the points differs when listed by magnitude, i.e., the feature with the greatest coefficient magnitude may not coincide with the feature of the greatest coefficient magnitude in the rotated and translated object. However, both of the features denoted with greatest magnitude in each object are likely to be represented in a group of points of large magnitude for each object.

As an example, in Figure 4-5, the six wavelet coefficients with the largest magnitudes are marked with colored dots on the level three reconstructed wavelet coefficients of two versions of the Lena picture, one translated from the other. The location and coefficient value are shown in Table 4.2. Note that although five of the six points match in location, only one matching pair holds the same ranking in each list, and points which seem to be very close in value do not necessarily correspond in position.

Therefore, the values of the wavelet coefficients are used to screen the image for feature points, but the particular value at each point is not used for locating matching points. Instead, a representation is used in which the top maxima are equally weighted, the bottom minima are equally weighted, and no other points are considered.

4.2.3 Quantity of Feature Points

The proper quantity of feature points must also be determined. Too few points will not achieve registration, but too many points slow the algorithm and increase the number of false positives by increasing the likelihood of random correspondence.

The upper limit on the number of feature points is introduced by the potential of saturating the object with so many feature points that randomly matching feature points cause false positive rotations. This is discussed more in the discussion on levels of resolution in Section 4.2.4.
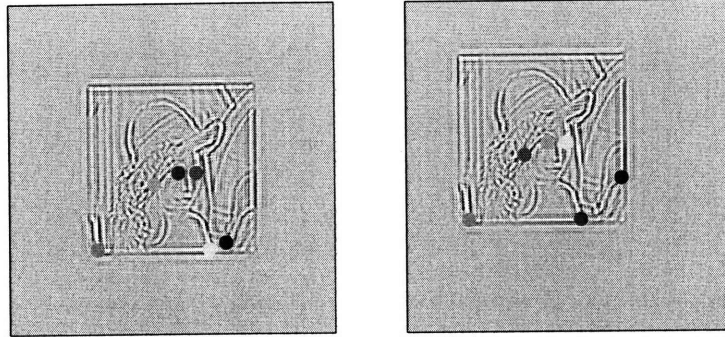
Figure 4-5: The third level DTCWT decomposition of two translated versions of the Lena image. The six points with highest magnitude of coefficients are marked with colored dots. In order of descending magnitude, the colors are: green, red, yellow, blue, cyan and magenta. The coefficient magnitudes and correspondence is listed in Table 4.2.

A			B		
Order	Color	Coefficient Magnitude	Order	Color	Coefficient Magnitude
1	green	224.7	1	green	206.3
2	red	177.3	5	cyan	164.7
3	yellow	170.2	2	red	180.7
4	blue	159.8	n/a		
5	cyan	159.7	6	magenta	159.3
6	magenta	151.1	3	yellow	173.1
n/a			4	blue	171.2

Table 4.2: Coefficient values of corresponding points in the third level DTCWT decomposition of two translated versions of the Lena image shown in Figure 4-5. The points from the left-hand image, denoted A, are listed in descending order of magnitude in the left-hand column. The points that correspond in location in the right-hand image are listed in the right-hand column. Note that the order of magnitude is different, and that for the most part, the magnitudes are quite different.

Another consideration in the upper limit on the number of feature points is the computational complexity of the algorithm, which depends both upon the size of the model and the number of feature points. The computational complexity is the greater of $O(n)$, where n is the number of voxels in the model, and $O(\binom{m}{3}^2)$, where m is the number of feature points. This is described in more detail in Section 4.5.1. Choosing 30 feature points yields $\binom{30}{3}^2 = 16$ million as a worst case. In actuality, a quarter or less of these become viable triangles for testing. Depending upon the number of outliers, the speed of the regitration portion of the algorithm runs from 18 to 56 seconds. Increasing the number of feature points greatly slows the algorithm.

The lower limit on the number of feature points was determined using the following line of reasoning and was confirmed heuristically. The total number of potentially matching triangles is $\binom{m}{3}$, where m is the number of feature points. Assuming three quarters of the feature points are outliers (an assumption on the high end), this becomes $\binom{m/4}{3}$. Of those, some will be eliminated because they are collinear or too small; we assume this is 20% (a number that is not extreme based on heuristics). There are four sets of feature points: positive and negative coefficients at both the second and third levels of resolution. This calculation yields 4 matching triplets for 10 feature points, 32 matching triplets for 20 points, 180 matching triplets for 30 points, and 384 matching triplets for 40 points. The many examples that were run in the process of generating and verifying this algorithm support this line of reasoning. Ten to fifteen feature points often produced a correct rotation but occasionally did not. Twenty-five to thirty-five points reliably produced the correct rotation.

4.2.4 Levels of Resolution

Another variable is the number of levels of resolution that is used in the matching algorithm. The first factor in this is that the number of points from which feature points are selected reduces by 2^d at each level of resolution, where d is the dimensionality. Therefore, at some point there are too few points and the object becomes saturated with feature points. On the examples we are running, we reach saturation at the fourth or fifth level of resolution, so we select our lowest level of resolution to

be three. See Table 4.3.

Model	Level	n	p	f
Brain	1 and 2	2,457,600	3.52e-09	0.89
MRI	3	307,200	0.0020	7.12
	4	38,400	-	56.95
Knee	1 and 2	7,077,888	2.41e-12	0.31
	3	884,736	3.24e-06	2.47
	4	110,592	0.14	19.78
Gooch	1 and 2	11,534,336	8.10e-14	0.19
	3	1,441,792	1.29e-07	1.52
	4	180,224	.0273	12.13
	5	22,528	-	97.08
Shrek	1 and 2	1,741,916	3.63e-08	1.26
	3	217,739	.0116	10.04
	4	27,217	-	80.35

Table 4.3: The saturation level of resolution for various models. Level indicates the level of resolution (note that levels 1 and 2 have the same number of points and are therefore listed together). Column n indicates the number of voxels in each level of resolution. Column p indicates the probability of randomly matching 7 of 20 feature points with a 4-voxel ring. For the rows with no probability listed, the probability is close to 1. Column f indicates the percentage of voxels in the entire image that fall within a 4-voxel ring of 20 feature points, assuming no voxel rings overlap each other or the edges of the model. Note that each model becomes saturated at the fourth to fifth level of resolution.

Through testing on a number of models, it was also determined that very few of the successfully matching triangles were produced by the first level of resolution. By eliminating the first level wavelet coefficients, we save time and memory space in the algorithm with no penalty in the quality of the registration.

4.3   Feature Point Matching

Beginning with all possible combinations of three feature points in models A and B, we reduce the number used by selecting only corresponding triplets with essentially the same distances and angles between the three points.

We first eliminate collinear triplets and very small triangles. At various places in the algorithm we use a tolerance, designated *maxr*, for a range of pixels within which we seek a match; for example, when matching feature points, or when determining final quality of a match. We define a small triangle as one with a perimeter less than six times *maxr*.

We separate the rotation and translation, calculating the rotation first. For potentially matching triplets, we calculate a rotation using the unit quaternion method of Horn [26], who found a closed form solution to the absolute orientation problem. Quaternions are described in more detail in Appendix C. The quaternion representation has an ambiguity in that q and $-q$ describe the same rotation; we eliminate this by restricting one quaternion coefficient, q_1, to be greater than zero.

4.4 Voting Scheme

Potentially corresponding triplets that are correctly paired will generate transformations that are close in value to one another, differing only by noise and error in the calculation of the rotation. The outliers, which generate incorrectly corresponding triplets, will produce widely varying transformations that are unlikely to correspond. Therefore calculating rotations for each set of potentially corresponding triplets generates a set of potential rotations with a cluster at the desired rotation and noise elsewhere. We use a histogram voting scheme to locate the cluster and thus the desired rotation.

4.4.1 Rotation Representation

The voting scheme could be based on any of the possible representations of rotation, such as the orthogonal rotation matrix, Euler angles, unit quaternions, or the axis-angle representation; details are presented in Appendix C. Using the rotation matrix would require searching for a peak in 9-space, which is more cumbersome than other choices. Euler angles suffer from gimbal lock problems and are therefore not a stable representation upon which to base the voting scheme. Two solid choices, both of

56

which would require searching for a peak in 4-space, are quaternions and the axis-angle representation. As shown in Appendix C, the conversion from one format to the other is:

$$
\begin{aligned}
q_0 &= \cos(\alpha/2) \\
q_1 &= \sin(\alpha/2)\cos(\beta_x) \\
q_2 &= \sin(\alpha/2)\cos(\beta_y) \\
q_3 &= \sin(\alpha/2)\cos(\beta_z),
\end{aligned}
$$

where $[q_0\ q_1\ q_2\ q_3]$ are the coefficients of the quaternion, and α, β_x, β_y and β_z are the rotation angle and direction cosines of the axis-angle representation. Note that this conversion is nonlinear; evenly spaced bins in one representation do not correspond to evenly spaced bins in the other. Also note that the degrees of freedom in each representation are reduced by one: $\beta_x^2 + \beta_y^2 + \beta_z^2 = 1$, and $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$.

If we use equally spaced histogram bins for voting based on quaternions, then a bin for a single quaternion coefficient will correspond to a greater arc length in the bins near -1 and 1 than in the bins near zero. However, the bins of the remaining coefficients will correspond to much shorter arc lengths; if one quaternion value is close to one, the remaining values must be close to zero to maintain the unit magnitude requirement. Thus the overall size of the 4-space bin does not vary greatly.

As an alternative, one could base the voting scheme on the axis-angle representation. This has the seeming advantage that equally spaced bins correspond to a linear division of degrees in real space. However, bins of equal numbers of degrees at the equator of a sphere are much larger than those at the poles.

The voting scheme based on quaternions is the most viable of the alternatives and has the added advantage that we calculate our rotation in the quaternion format and therefore require no conversion before voting.

During the voting process, the quaternion coefficient bins are equally spaced from -1 to 1 for each coefficient, except q_1 which runs from 0 to 1 to prevent redundancy. This means that many of the bins will be empty since only unit quaternions are used.

The magnitude of all possible quaternions along with a band corresponding to those bins likely to hold unit quaternions is shown in Figure 4-6.
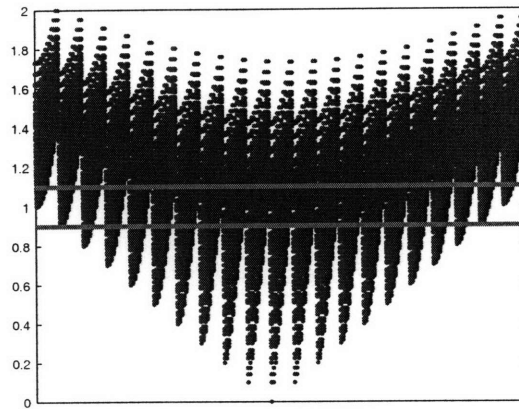
Figure 4-6: Unit quaternion bin magnitude. The magnitude of each possible quaternion bin based on a bin size of 0.1 is graphed. Only those with magnitudes close to 1, between the red lines, have the potential to be filled.

4.4.2 Histogram Bin Size and Origin

A proper bin size is required for determining the peak. If the bins are too large, then there may be sufficient outliers in an incorrect bin to obscure the desired result. In the extreme, a single bin would be of no assistance at all. Conversely, bins that are too small may split the correct transformation peak into separate bins, again obscuring the peak. In this extreme, each bin would have one or zero items in the bin and no peak would be found.

Many treatises on correct bin size for a histogram [32, 68, 71] discuss matching the underlying statistics of the data. In our application, we should have a single large peak and much surrounding noise. We are interested only in finding the peak and not in any analysis of items not in the peak; therefore, statistical analysis of non-peak bins is not required, and the number of samples in non-peak bins is not relevant. See Figure 4-7.

The goal bin size is one that allows us to quickly identify the peak in the data, and to pinpoint it with sufficient accuracy for our desired application. Our algorithm achieves global registration of voxelized objects explicitly enough to provide an initial

58
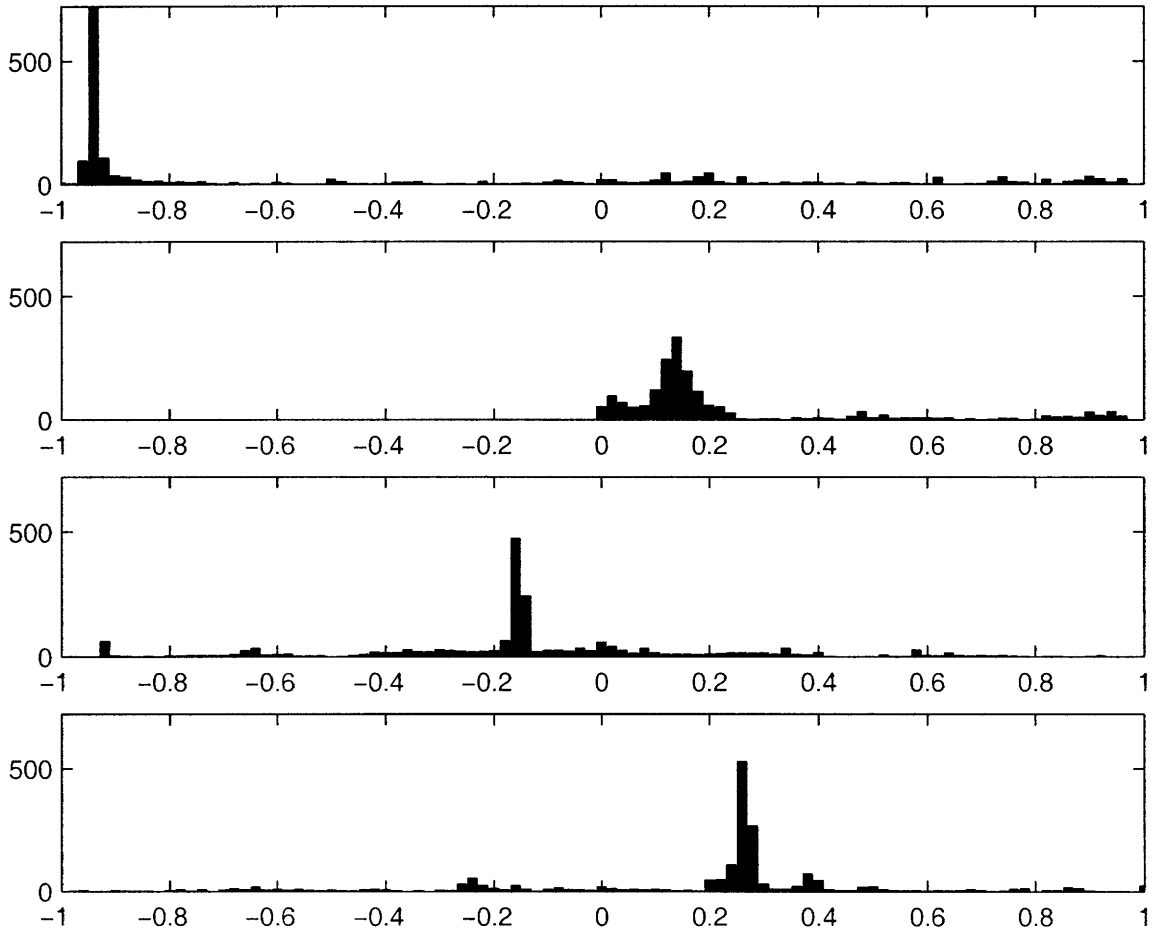
Figure 4-7: A histogram of each coefficient in the rotation quaternion voting scheme. A significant peak occurs at the proper rotation with noise elsewhere. Note that this is a one-dimensional histogram of each coefficient and does not show the interrelation between coefficients; the actual histogram performed is four dimensional.

estimate for fine registration systems such as ICP or voxel intensity measures to be successfully applied without settling into local minima. We define sufficient accuracy as less than two degrees error in each Euler angle. Within this defined goal, a bin size that is too small increases computational time, and a bin size that is too large includes outliers which can mask the proper result.

We use the quaternion coefficients in our histogram, while our definition of sufficient accuracy is based on Euler angles. The nonlinear conversion is as follows (see Appendix C):

$$\phi = \arcsin(-2(q_2 q_4 + q_1 q_3))$$
$$\theta = \arctan\left(\frac{-2(q_3 q_4 - q_1 q_2)}{q_1^2 - q_2^2 - q_3^2 + q_4^2}\right)$$
$$\psi = \arctan\left(\frac{-2(q_2 q_3 - q_1 q_4)}{q_1^2 + q_2^2 - q_3^2 - q_4^2}\right).$$

A bin size of 0.1 for the quaternion coefficients corresponds to an Euler angle bin size ranging from approximately 2 to 10 degrees, with some ranging up to 15 degrees, depending upon which quaternion coefficient and where in the range from -1 to 1 the bin occurs. For example, Table 4.4 shows quaternion bin edge values, corresponding Euler angle bin edges, and Euler angle bin sizes at various quaternion values.

Using a histogram to identify a peak in data leads to the problem of properly placing the origin in order to avoid splitting the peak. If the bin edge occurs exactly upon the peak, the peak will be split with only half the number of samples appearing in two adjacent bins. This may obscure the peak from detection.

Scott [68] developed a method called an Averaged Shifted Histogram (ASH) which eliminates this problem by taking multiple histograms of equal bin widths but shifted bin locations and averaging the results. The method is statistically comparable to kernel estimators with much reduced computational complexity. The method is implemented by selecting a bin size, h, and a number of histograms to be averaged, m. Next, a histogram is performed on the set of data using a small bin size, $\delta = h/m$. Adjacent bins are then averaged. In three dimensions, the histogram value at sample

Quaternion				Euler Angle			Bin Size		
q_0	q_1	q_2	q_3	θ	ϕ	ψ	θ	ϕ	ψ
-0.4	0.3	-0.6	-0.7	-80	-3	123	3	6	8
-0.3	0.3	-0.6	-0.7	-83	3	131	2	8	7
-0.2	0.3	-0.6	-0.7	-85	11	138	2	7	8
-0.1	0.3	-0.6	-0.7	-87	18	146	-	-	-
-0.9	0.0	-0.3	-0.1	-5	-36	14	13	2	4
-0.9	0.1	-0.3	-0.1	-18	-34	18	13	2	4
-0.9	0.2	-0.3	-0.1	-31	-32	22	12	3	2
-0.9	0.3	-0.3	-0.1	-43	-29	24	-	-	-
-0.3	0.9	-0.2	-0.1	-142	4	26	0	3	11
-0.3	0.9	-0.1	-0.1	-142	7	15	1	4	11
-0.3	0.9	0.0	-0.1	-143	11	4	1	4	12
-0.3	0.9	0.1	-0.1	-144	15	-8	-	-	-

Table 4.4: Quaternion values, corresponding Euler angles, and resulting Euler angle bin sizes. In each section, consecutive rows indicate adjacent bins in an equally spaced bin scheme for quaternions. Only one quaternion changes value. The closest corresponding Euler angles based on an xyz rotation are calculated for each bin edge and the resulting bin sizes based on Euler angle are shown. Note that all angles are rounded, occasionally resulting in a bin size of zero. In addition, the bin edges are first normalized to unit quaternions, so the calculated Euler angles do not correspond exactly with the quaternion bin edge. The quaternion bins selected for this example all had magnitudes between 0.95 and 1.05 to minimize the variation in bin edge location.

x is [68]:

$$\hat{f}(\mathbf{x}) = \frac{1}{h_1 h_2 h_3} \sum_{i_1} \sum_{i_2} \sum_{i_3} (m - |i_1|)(m - |i_2|)(m - |i_3|) n_{k_1+i_1, k_2+i_2, k_3+i_3},$$

where i_j runs from $1 - m_j$ to $m_j - 1$, and **x** is in bin I_{k_1,k_2,k_3} which contains n_{k_1,k_2,k_3} values.

Through testing it was verified that the identification of the peak is fairly robust to variations in bin size. Bin width options from 0.001 to 0.2 and numbers of averaged histograms from 1 to 20 were tested; a range of bin widths from 0.001 to 0.05 with 2 to 10 histograms averaged produced acceptable results. We selected a bin width of 0.02 and two averaged histograms.

4.4.3 Locating the Peak Quaternion

By using the Average Shifted Histogram method, we identify a small histogram bin at the center of the peak. We use the values for q from this small bin instead of some combination of the averaged values in the larger shifted bin. This has the effect of pinpointing samples in the center of the peak, thus reducing the variation of the samples used to determine the final rotation quaternion. To further reduce the impact of outliers within our bin, we take the mean of the available values then normalize the resulting quaternion.

4.4.4 Translation

We then take the top few rotations (those with the greatest number of votes), and calculate a translation. We multiply the list of points from model B by the selected rotation, then determine the single translation that aligns the greatest number of points from B with A. Note that the translations determined by lower levels of resolution are half the length of the next higher level.

4.4.5 Matching Feature Point Count

At this point, for the top few rotations (those with the greatest number of votes), we apply the rotation and translation to the list of feature points from model B, and determine proximity to points from model A. This is a confirmation of the quality of our rotation. A proper rotation and translation will produce a larger number of correspondences between the two lists of feature points than an improper rotation and translation.

4.5 Analysis

4.5.1 Computational Complexity

The computational complexity of the algorithm depends upon the size of the objects and the number of feature points. There are three main parts of the algorithm which make significant contributions to computational complexity.

The first is the application of the wavelet transform and reconstruction of the wavelet coefficients. Per Kingsbury [37], the computational complexity of the DTCWT is $O(n)$, where n is the number of voxels in the object.

The second major contributor is the screening of all possible triplets of feature points, which is $O\binom{m}{3}$.

These triplets of feature points are pared down by eliminating collinear points, small triangles, and triplets without matching triplets in the other model. In the worst case, if no triplets were eliminated and the perimeter and angles of all triplets in one model matched those of all triplets in the second model, the complexity of the algorithm would be $O\left(\binom{m}{3}^2\right)$.

4.5.2 Speed

As stated above, the speed of the algorithm depends both upon the size of the object and the number of feature points selected. Table 4.5 shows speed of the algorithm in two sections; the first is the wavelet transform and feature point selection which

depends upon the size of the object. The second is the point matching, voting, and registration which depends upon the number of feature points. The algorithm is run on a Core 2 Duo processor. Times are elapsed time.

The first time, which measures the part of the algorithm that is of complexity $O(n)$, is very consistent. The second time, which measures the part of the algorithm that is of worst case complexity $O\left(\binom{m}{3}^2\right)$, varies greatly depending upon the orientation of the model, as would be expected. As the rotation and translation from one model to the other changes, the number of outliers changes, and thus the number of matching triangles changes.

Model	n	Time 1	m	$\binom{m}{3}^2$	Time 2
Knee MRI	7,077,888	14.4	30	16,483,600	18.2
Shrek Solid Model	3,932,160	12.8	30	16,483,600	33.1
Large Brain MRI	25,186,304	108.9	30	16,483,600	42.1
Brain MRI	7,614,464	7.0	15	207,025	1.9
Brain MRI	7,614,464	7.0	20	1,299,600	4.9
Brain MRI	7,614,464	7.0	25	5,290,000	13.6
Brain MRI	7,614,464	7.0	30	16,483,600	56.5
Brain MRI	7,614,464	7.0	35	42,837,025	374.2

Table 4.5: Speed of the wavelet registration algorithm. n is the number of voxels in the object. m is the number of feature points. The algorithm is divided into two parts. Time 1, with complexity $O(n)$, includes the wavelet transform and feature point selection. Time 2, with complexity $O\left(\binom{m}{3}^2\right)$, includes point matching, voting, and registration. Times are elapsed time.

The algorithm compares favorably with other algorithms of its type. As discussed in section 2.2.3, state of the art geometric feature registration algorithms run in the neighborhood of 70 to 300 seconds. This algorithm runs in the neighborhood of 20 to 150 seconds, depending upon the size of the object.

One intended use of the algorithm is as a pre-processing step for voxel intensity methods which accomplish very fine elastic registration but which require initial coarse registration in order to avoid incorrect local minima. State of the art elastic registration methods run many minutes up to hours; the speed of this algorithm

indicates its suitability for such an application.

Chapter 5

Examples and Applications

In this section we show examples from a variety of applications. Both non-homogeneous models with continuously changing intensities throughout such as MRI scans and homogeneous models with a constant value in the interior such as voxelized surface models are included. We show two examples with artificially added Gaussian noise, one application of temporal matching, and one application with occlusion and clutter.

An important note is that the algorthm is not tuned to specific types of models. All examples were run using the second and third resolution wavelet coefficients with 30 feature points selected. The voting scheme is a four dimensional averaged shifted histogram with bin size of 0.02 for each quaternion coefficient and two bins averaged. A 4-voxel ring is used for comparison of points and objects. Examples were run on a Core 2 Duo machine.

Results format. The results for each registration are presented in tabular format showing the transformations with the eight highest votes in descending order. Angles are the pitch, roll and yaw Euler angles, θ, ϕ and ψ, required for registration as determined by the algorithm. Δ_r is a pseudo-Manhattan distance which is the sum of the magnitude of the distance between each recommended angle and the inverse of the induced rotation. Vote is the number of votes received in the averaged shifted histogram voting scheme. Similarly, the translation is the number of voxels in the i, j and k directions required for registration as determined by the algorithm, and

Δ_t is the Manhattan distance between the recommended translation and the inverse of the induced translation. The level 2 and level 3 columns indicate the number of positive and negative feature points that align within 4 voxels given the recommended rotation and translation. Results are rounded to the nearest degree or voxel.

The top eight vote receiving results are presented. The top result is the recommended answer; the rest are provided for comparison and discussion.

5.1 Non-Homogeneous Objects

These objects are fully three dimensional, varying in intensity throughout the object. Both examples shown here are Magnetic Resonance Imaging (MRI) data. However, this description fits many types of data such as Computed Tomography (CT), Ultrasound (US), and functional MRI (fMRI).

These examples were formed by taking an object, then randomly selecting and applying a three-dimensional Euler angle rotation and three-dimensional translation to produce a second object. These two objects were registered using the wavelet extrema algorithm and the resulting registration is compared to the induced rotation and translation.

Brain MRI. The first example is an MRI scan of a brain [48]; see Figure 5-1. The original model was rotated 28, 37 and -34 degrees in a pitch-roll-yaw Euler angle scheme, then translated -14, 12 and -3 voxels to create the second model. Total elapsed time was 63 seconds. Results are displayed in Table 5.1. All of the top eight results are close in value.

Knee MRI. The second example is an MRI scan of Paul Debevec's knee, which is available online courtesy of a torn ligament [15]; see Figure 5-2. The original object was rotated by pitch-roll-yaw Euler angles of -20, 4, and 41 degrees, then translated 13, 8 and 11 voxels in the i, j and k directions respectively. Registration was performed; the top 8 results are displayed in Table 5.2. The total elapsed time

| Angles | | | Δ_r | Vote | Translation | | | Δ_t | Level 2 | | Level 3 | |
θ	ϕ	ψ			i	j	k		+	-	+	-
-27	-36	33	4	297	14	-13	2	2	10	14	13	20
-25	-37	32	5	259	14	-13	3	1	10	14	13	20
-26	-36	30	7	248	15	-12	3	1	9	14	13	20
-26	-34	31	8	228	15	-13	1	3	10	12	12	20
-28	-35	35	3	227	13	-14	0	5	10	14	13	19
-24	-37	29	9	219	15	-12	3	1	9	12	12	20
-29	-35	33	4	210	13	-13	4	3	10	12	12	19
-26	-36	35	3	208	13	-13	1	4	10	14	13	19

Table 5.1: Registration results for Brain MRI in descending order of vote. The correct rotation in pitch-roll-yaw Euler angles is -28, -37 and 34 degrees, with a translation of 14, -12 and 3 voxels.

Figure 5-1: Brain MRI scan [48]. The left column is a single slice of the original object. The middle and right columns are single slices of the reconstructed wavelet coefficients at the second and third levels of resolution respectively. The top row is model A. The bottom row is model B, which has been rotated and translated.

was 44 seconds. Note that the top 7 results are very close to one another. The eighth, which is incorrect, received significantly fewer votes.

| Angles | | | Δ_r | Vote | Translation | | | Δ_t | Level 2 | | Level 3 | |
θ	ϕ	ψ			i	j	k		+	-	+	-
20	-4	-40	1	5582	-12	-9	-10	2	15	4	22	22
20	-4	-42	1	4513	-12	-9	-10	2	14	4	22	22
21	-3	-40	3	4012	-12	-8	-10	2	16	4	22	22
18	-5	-40	3	3082	-13	-8	-10	1	14	4	23	22
20	-3	-43	2	2859	-12	-9	-10	3	13	4	21	22
18	-4	-42	3	2743	-13	-9	-10	2	14	4	23	22
20	-4	-38	3	2609	-12	-4	-11	5	13	4	22	24
22	5	-31	20	591	-14	3	-2	20	14	2	19	16

Table 5.2: Registration results for Knee MRI in descending order of vote. The correct rotation in pitch-roll-yaw Euler angles is 20, -4, and -41 degrees, with a translation of -13, -8 and -11 voxels.

5.2 Homogeneous Objects

The objects in this section are solid models which do not vary in the interior. If not already in voxelized form, mesh or other surface model types can be voxelized. We used Binvox [49], a binary voxelization tool produced by Patrick Min based on the work of Nooruddin and Turk [57].

Shrek. The 3D mesh model of Shrek, produced by Pocci [62], was voxelized using Binvox [49]. Two tests were run. The first was a registration of the full model and the second a cropped portion of the model.

The first test, using the full Shrek model with an induced rotation in pitch-roll-yaw Euler angles of 12, 15 and -14 degrees and an induced translation of 11, -3 and 11 voxels, had an elapsed time of 262 seconds. Results are shown in Table 5.3. The second test using the cropped model had an elapsed registration time of 67 seconds. See Table 5.4 and Figures 5-3 and 5-4.
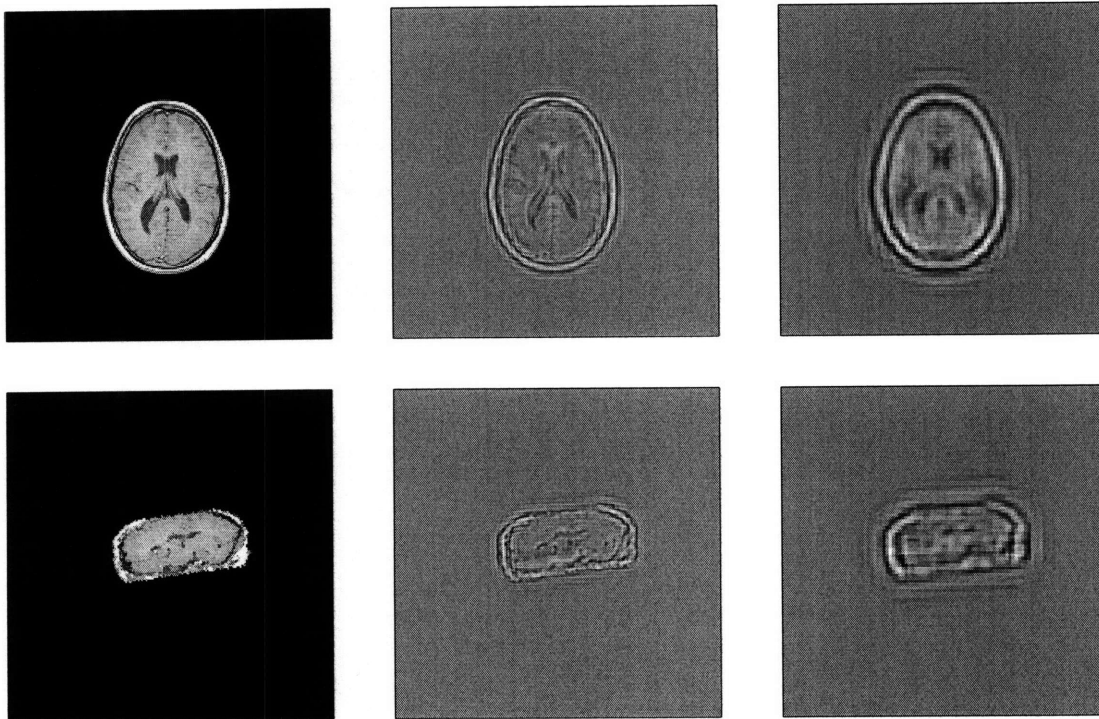
Figure 5-2: Knee MRI scan. The left column is a single slice of the original object. The middle and right columns are single slices of the reconstructed wavelet coefficients at the second and third levels of resolution respectively. The top row is model A. The bottom row is model B, which has been rotated and translated.

| Angles | | | Δ_t | Vote | Translation | | | Δ_t | Level 2 | | Level 3 | |
θ	ϕ	ψ			i	j	k		+	-	+	-
-12	-14	14	1	802	-12	3	-10	2	10	5	22	24
-13	-15	15	2	761	-11	3	-10	1	11	4	23	24
-13	-14	15	2	724	-11	3	-10	2	10	4	23	24
-13	-15	17	4	639	-8	2	-10	5	3	4	22	24
-13	-14	17	4	622	-8	3	-9	5	4	5	23	24
-12	-14	12	3	615	-11	2	-9	3	10	5	22	24
-12	-16	12	2	580	-12	2	-10	2	9	5	23	24
-15	-15	17	6	496	-7	2	-7	8	2	4	22	24

Table 5.3: Registration results for the full Shrek solid model in descending order of vote. The correct rotation in pitch-roll-yaw Euler angles is -12, -15 and 14 degrees, with a translation of -11, 3 and -11 voxels.

| Angles | | | Δ_r | Vote | Translation | | | Δ_t | Level 2 | | Level 3 | |
θ	φ	ψ			i	j	k		+	-	+	-
-10	-21	-30	1	820	13	-7	4	4	11	9	22	26
-10	-19	-30	2	704	12	-8	3	3	11	9	22	26
-11	-18	-28	5	614	15	-7	3	3	10	9	22	26
-9	-15	-28	7	475	15	-8	-3	9	10	5	22	26
-10	-23	-31	4	473	15	-5	6	4	10	10	22	26
-11	-26	-32	8	350	12	-5	10	10	7	7	22	26
-10	-13	-28	9	329	15	-8	-3	9	11	5	22	26
-8	-8	-26	17	255	14	-12	-8	17	4	5	21	26

Table 5.4: Registration results for a cropped version of the Shrek solid model in descending order of vote. The correct rotation in pitch-roll-yaw Euler angles is -10, -20, and -30 degrees, with a translation of -14, 6 and -3 voxels.

Figure 5-3: Shrek model produced by Pocci [62] and voxelized using Binvox [49].

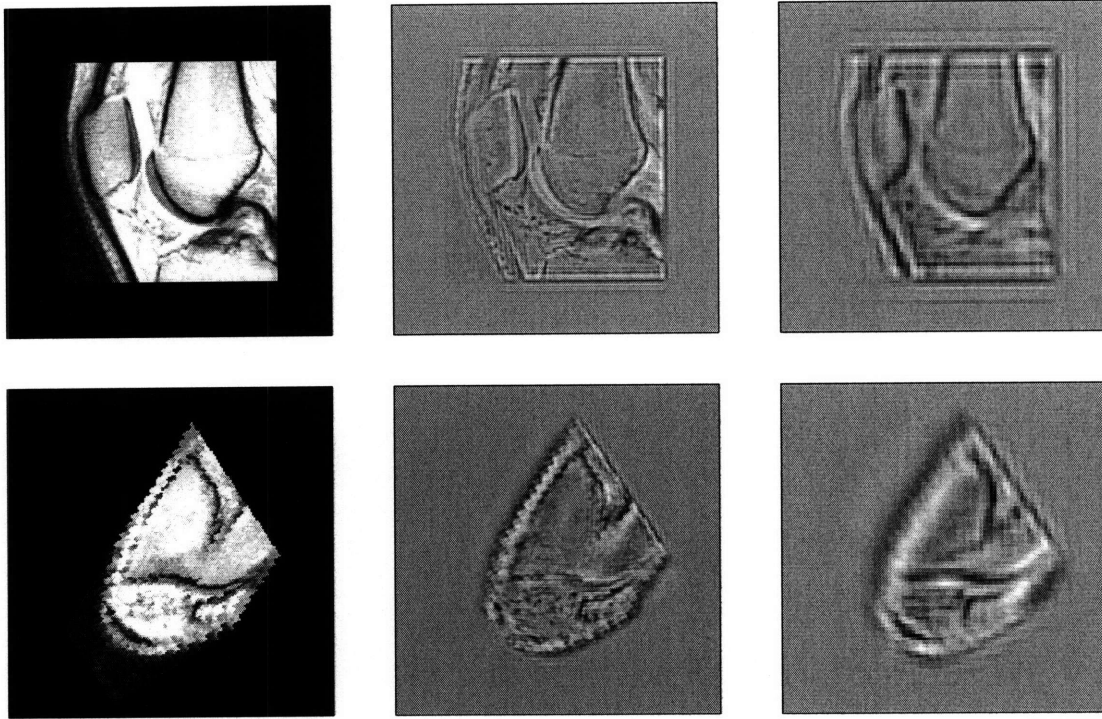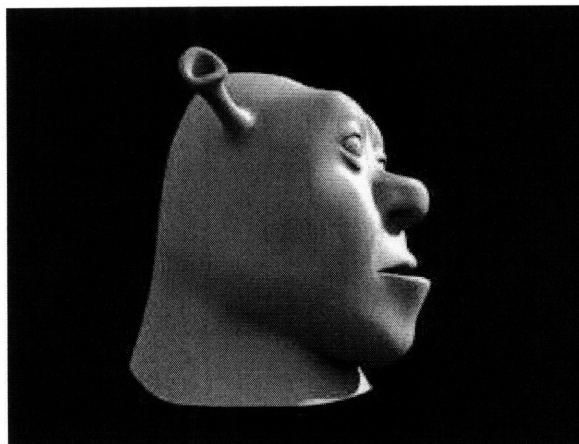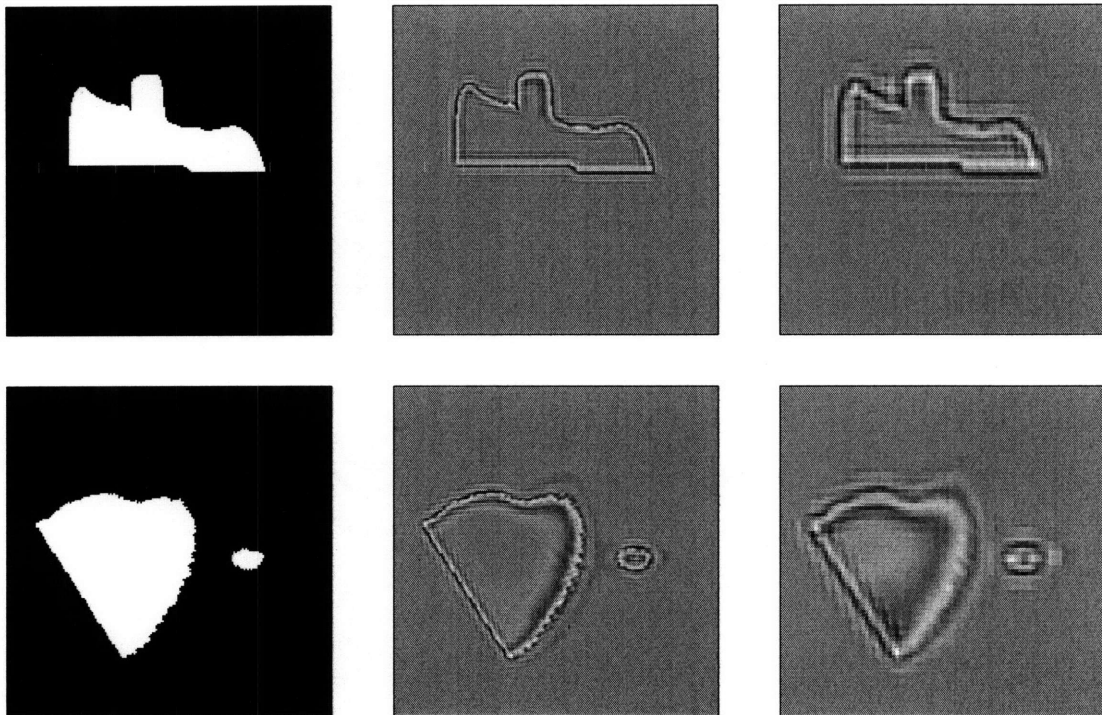Figure 5-4: Cropped Shrek solid model. The left column is a single slice of the original object. The middle and right columns are single slices of the reconstructed wavelet coefficients at the second and third levels of resolution respectively. The top row is model A. The bottom row is model B, which has been rotated and translated.

5.3 Noise

Noise can be induced in 3D models in many ways. In medical imaging, noise is induced by the patient, either through movement or over time through growth and change. All sensing methods induce some type of noise. Manipulating the objects in their electronic form also induces noise. To test our method for noise sensitivity in a controlled manner, we added Gaussian noise to the object to obtain varying signal to noise ratios, then applied the algorithm to attempt registration.

Gaussian noise was added only to the rotated and translated object, which was then registered to the original object without noise. The Gaussian noise was generated using a normally distributed set of random numbers with a mean of zero, variance of one, and standard deviation of one. This set of randomly generated numbers was then multiplied by some greater standard deviation to achieve the desired signal to noise ratio (SNR). The SNR was calculated using the ratio of root mean square amplitudes of the signal and the noise [64]:

$$SNR = \left(\frac{A_{signal}}{A_{noise}} \right)^2 = \frac{\sum_{i=1}^{n} x_i^2}{\sum_{i=1}^{n} k_i^2}$$

where A is the root mean square amplitude, x_i is the voxel intensity at each point in the object, and k_i is the intensity of the noise added at each point in the object. The SNR can also be expressed in decibels:

$$SNR = 10 \log_{10} \left(\frac{\sum_{i=1}^{n} x_i^2}{\sum_{i=1}^{n} k_i^2} \right)$$

As the test model, we used the knee MRI object with a rotation of 10, 20 and 30 degrees and a translation of -3, 7 and 0 voxels. The first test used a variance of 100 to achieve a SNR of 4.74, or 13.50 dB. Successful registration occurred in an elapsed time of 34 seconds. The second test used a variance of 5000 to achieve a SNR of 0.67, or -3.48 dB. Successful registration occurred in an elapsed time of 46 seconds.

Note that there is noise induced in each of these tests through the initial rotation process in addition to the Gaussian noise induced.

This algorithm uses the second- and third-level resolution wavelet coefficients for registration. The first-level resolution wavelet coefficients contain information relating to the smallest scale details; much of the noise is contained in this level. This makes the algorithm quite robust to noise, as demonstrated by the examples herein.
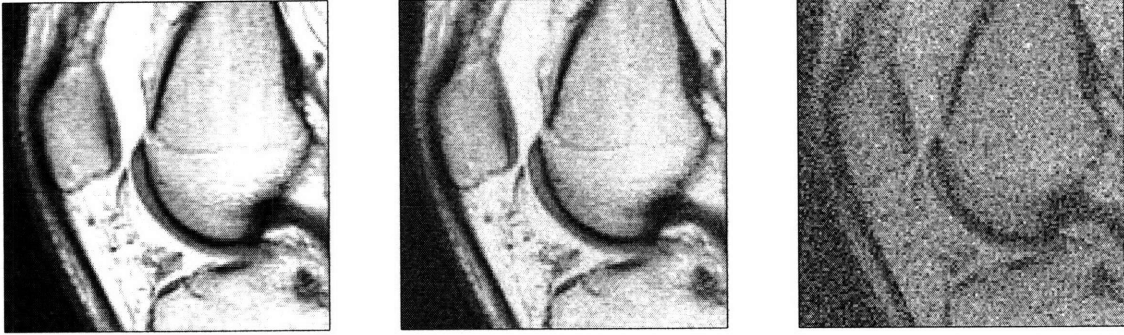
Figure 5-5: Knee MRI scan with Gaussian noise added. Left image without noise, center image with SNR 4.74, right image with SNR 0.67.

| Angles | | | Δ_r | Vote | Translation | | | Δ_t | Level 2 | | Level 3 | |
θ	ϕ	ψ			i	j	k		+	-	+	-
-10	-21	-30	1	2059	-3	8	0	2	10	5	16	17
-11	-20	-28	3	1568	-3	9	0	2	10	5	14	18
-8	-20	-31	2	1549	-4	9	-1	3	10	5	14	17
-11	-19	-30	2	1446	-4	8	2	3	10	5	16	17
-12	-21	-29	4	1174	-3	8	0	1	10	5	16	17
-8	-18	-30	4	1150	-3	7	2	3	11	5	15	17
-11	-19	-28	4	1085	-1	9	3	6	10	5	15	18
-12	-22	-30	4	897	-2	6	0	2	10	5	16	17

Table 5.5: Registration results for Knee MRI with Gaussian noise in descending order of vote. SNR = 3.32. The correct rotation in pitch-roll-yaw Euler angles is -10, -20, and -30 degrees, with a translation of -3, 7 and 0 voxels.

5.4 Occlusion and Clutter

This example was constructed using Paul Debevec's knee MRI. A full model of the knee was rotated and translated in three dimensions, then the original and rotated versions were cropped so the majority of the object was the same but the edges contained different parts of the knee; some portions were rotated out of the box while

74

Angles θ	φ	ψ	Δ_r	Vote	Translation i	j	k	Δ_t	Level 2 +	-	Level 3 +	-
-10	-20	-30	1	2131	-3	8	0	1	6	2	15	16
-11	-20	-28	3	1848	-3	8	0	2	6	2	15	17
-10	-19	-30	2	1768	-4	7	2	3	6	2	16	16
-11	-19	-28	4	1431	-1	9	3	7	7	2	14	16
-8	-20	-30	2	1417	-3	8	-1	2	6	2	14	16
-9	-18	-30	4	1373	-3	7	2	2	7	2	15	16
-12	-21	-28	5	1036	0	10	1	7	5	2	14	16
65	-15	-22	89	174	14	11	-38	59	9	1	2	0

Table 5.6: Registration results for Knee MRI with Gaussian noise of in descending order of vote. SNR = 0.67. The correct rotation in pitch-roll-yaw Euler angles is -10, -20, and -30 degrees, with a translation of -3, 7 and 0 voxels.

others were rotated in. You can see in Figure 5-6, for example, that the top right edge of the knee is visible in the top row slice but not in the bottom row slice. Similarly, variations in the tibia are visible at the bottom edge of the slice in the bottom row but not in the top row. Successful registration was achieved in 18 seconds; results are shown in Table 5.7.

Angles θ	φ	ψ	Δ_r	Vote	Translation i	j	k	Δ_t	Level 2 +	-	Level 3 +	-
-6	-6	6	3	924	5	-3	1	2	9	15	4	17
-6	-7	6	1	857	5	-3	1	1	9	15	4	17
-4	-6	6	5	768	5	-3	1	1	9	15	4	16
-4	-8	6	3	706	5	-3	2	1	9	15	4	17
-6	-3	6	5	506	4	-3	0	3	7	15	4	17
-4	-3	6	8	458	5	-3	1	1	7	13	4	17
-8	-10	7	4	280	5	-3	2	1	8	14	3	17
1	-1	5	16	141	2	-3	2	3	5	13	1	17

Table 5.7: Registration results for Knee MRI containing occlusion and clutter. The correct rotation in pitch-roll-yaw Euler angles is -6, -8 and 7 degrees, with a translation of -5, 3 and -2 voxels.
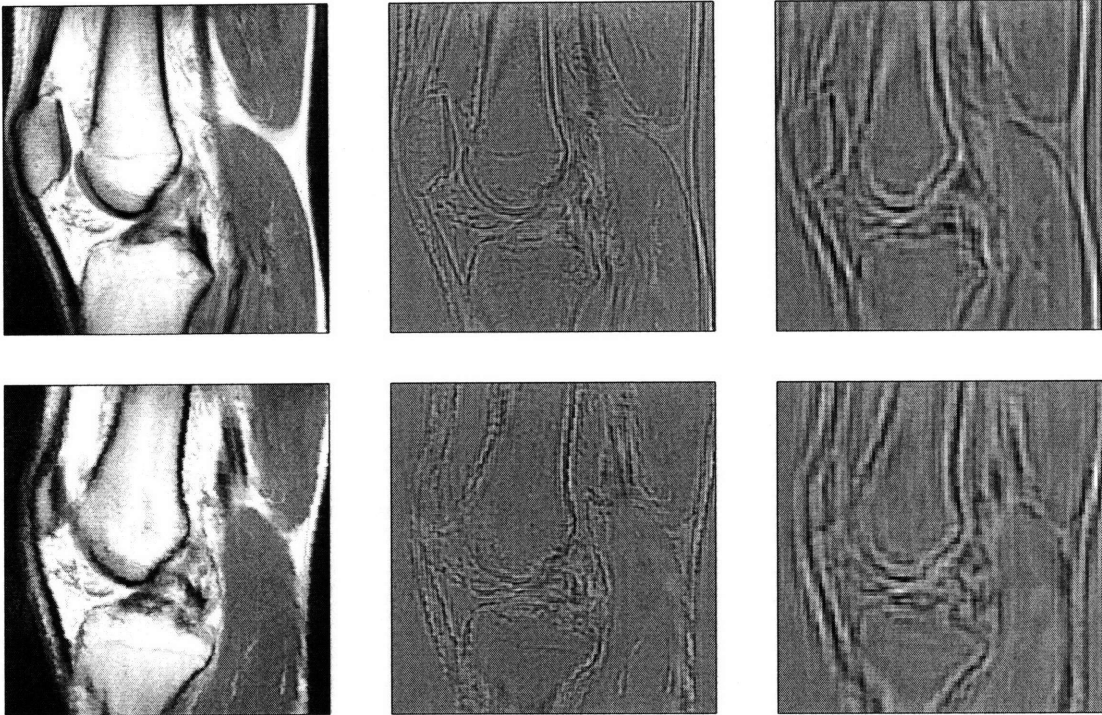
Figure 5-6: Occlusion and clutter. The left column is a single slice of the original object. The middle and right columns are single slices of the reconstructed wavelet coefficients at the second and third levels of resolution respectively. The top row is model A which has been cropped. The bottom row is model B, which was rotated before cropping.

5.5 Temporal

The final example consists of two MRI scans taken of Bruce Gooch at different times, one on June 5 and one on July 13 [19]. The other examples in this section are concocted by rotating and translating an object then comparing to a copy of its original self; therefore, the rotation and translation are known exactly. In this example, we are comparing two different objects of unknown initial orientation.

For the first test, we rotate the June 5 scan through a large rotation, then attempt registration with the July 13 scan. Results are shown in Table 5.8. The resulting rotation and translation are not exactly equal to the induced ones. It can be seen in Figure 5-7 that the post-alignment registration is improved over the initial slightly unaligned version.

For the second test, the June 5 and July 13 scans are registered as is, with no induced rotation or translation. Results are shown in Table 5.9. Note that the recommended rotation and translation are very close to the difference between the induced and recommended rotation and translation in the first test.

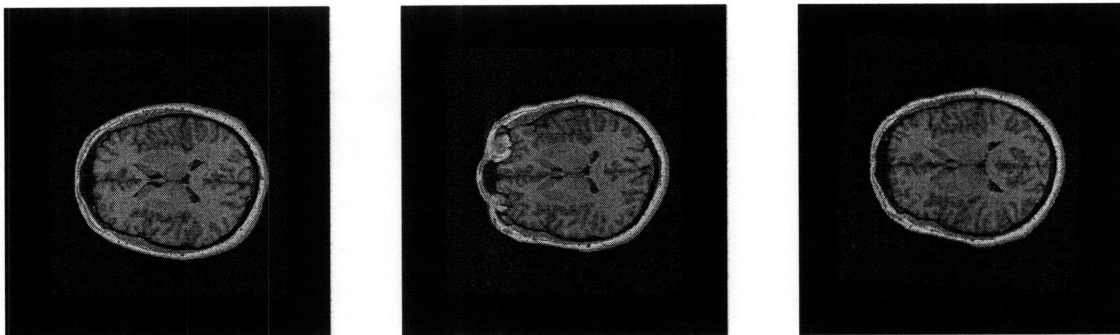Figure 5-7: Temporal brain MRI scan comparison. The left image was taken on July 13. The center image was taken on June 5. The right hand image is the June 5 image after a large rotation is induced then removed through registration with the July 13 image; note that the alignment is improved after registration despite the large initially induced angle. For example, the top of the eye is no longer visible after registration.

| Angles | | | Δ_r | Vote | Translation | | | Δ_t | Level 2 | | Level 3 | |
θ	ϕ	ψ			i	j	k		+	-	+	-
1	-3	-73	10	3700	8	4	-2	14	17	16	25	9
0	-5	-72	7	3453	7	4	-2	13	18	15	25	9
-1	-4	-75	10	3257	7	5	-1	13	19	17	25	13
0	-2	-73	10	3143	7	3	-2	12	19	16	25	9
1	-3	-75	12	3065	7	4	-1	13	19	17	24	13
-1	-3	-75	11	2988	6	5	-1	12	19	17	24	11
0	-2	-75	12	2718	6	4	-1	11	19	17	24	11
0	-7	-72	9	2432	9	7	2	18	19	3	25	9

Table 5.8: Registration results for temporal Brain MRI in descending order of vote. The induced rotation in pitch-roll-yaw Euler angles is -5, -5, and -70 degrees, with no translation.

| Angles | | | Δ_r | Vote | Translation | | | Δ_t | Level 2 | | Level 3 | |
θ	ϕ	ψ			i	j	k		+	-	+	-
3	2	-3	8	2879	7	-4	0	11	21	14	25	13
6	1	-3	10	2530	7	-6	-1	13	21	14	25	13
4	1	-2	7	2353	7	-4	0	12	21	13	25	12
5	1	-1	8	2322	7	-6	0	13	21	14	25	12
3	3	-3	9	2290	4	-4	-1	9	21	14	25	13
3	1	-5	10	2137	7	-4	-1	12	21	13	25	12
5	3	-3	12	1984	4	-6	-1	11	21	13	25	13
1	1	-4	6	1937	5	0	0	5	21	14	25	13

Table 5.9: Registration results for temporal Brain MRI in descending order of vote. There was no initial induced rotation or translation.

Chapter 6

Conclusions and Recommendations

6.1 Conclusions

We have demonstrated an algorithm to conduct rigid registration of voxelized three-dimensional objects using the wavelet transform. The method performs registration of objects that are grossly misaligned and brings them closely into alignment. Global registration is successfully achieved without need of initial alignment information. Further fine registration may be conducted at this point using elastic registration methods or other fine alignment methods such as ICP.

We explored the effects of variations in the algorithm such as choice of feature point type and quantity, rotation representation, and voting scheme. Choices are explained and quantified.

We presented several applications and demonstrate efficacy on both fully three dimensional models in which the intensity varies throughout and on surface models in which the interior is a constant value. The algorithm is robust to noise, occlusion and clutter, and can operate on models that vary in time.

The method compares favorably in speed to other geometric feature registration methods and is signifiantly faster than voxel intensity registration methods, thus making its use as a pre-processing step for voxel intensity methods feasible. It operates on voxelized objects, which many geometric feature registration methods do not, and solves the problem of most voxel intensity methods in that it achieves global

registration without settling into a local minimum.

6.2   Recommendations for Future Work

The directions in which this research could go next are numerous and interesting. A partial matching application could allow the alignment of smaller data sets to larger ones. Adding scale variation could allow the alignment of higher resolution data sets in areas of interest.

Within the algorithm itself, the exploration of different size filters or different shape wavelets that still meet the DTCWT criteria is possible and may lead to improvements in precisely locating feature points.

Cross modal registration would allow the comparison of objects from different types of scans, such as MRI and CT data for medical applications or acoustical and optical data for security applications such as mine detection.

Appendix A

DTCWT Shift Invariance - Aliasing Energy Ratio

A quantitative analysis of the shift-invariance of Kingsbury's Dual-Tree Complex Wavelet Transform (DTCWT) is provided in [37], and is summarized here.

Figure A-1 graphically represents the decomposition of a signal using the DTCWT. The level one decomposition uses a discrete wavelet; subsequent levels use a q-shift wavelet and its reverse to achieve the proper interleaving of samples between the trees. A decomposition and recomposition of a signal using the DTCWT is graphically represented by Figure A-1, feeding into a reverse of the graph in Figure A-1. If only one level of wavelets were retained, e.g. d_{3a} and d_{3b}, the decomposition and recomposition could be represented by Figure A-2, where, e.g., $A(z) = H_{0a}(z)H_{00a}(z^2)H_{001a}(z^4)$ and $M = 8$.

Downsampling and upsampling a signal by some number of samples, M, is shown in the bottom leg of Figure A-2. Using multi-rate analysis, this results in [37]:

$$D(z) = \frac{1}{M} \sum_{k=0}^{M-1} B(W^k z)$$

where $W = e^{j2\pi/M}$.

Therefore, the mathematical representation of Figure A-2 is:

$$Y(z) = Y_a(z) + Y_b(z)$$
$$= \frac{1}{M} \sum_{k=0}^{M-1} X(W^k z)[A(W^k z)C(z) + B(W^k z)D(z)].$$

Aliasing terms occur when $k \neq 0$. In order to avoid aliasing, one must design filters such that $A(W^k z)C(z)$ and $B(W^k z)D(z)$ are small or cancel one another.

An indication of the degree of shift invariance is the aliasing energy ratio [37]:

$$R_a = \frac{\sum_{k=1}^{M-1} \mathcal{E}\{A(W^k(z)C(z) + B(W^k z)D(z)\}}{\mathcal{E}\{A(z)C(z) + B(z)D(z)\}}$$

where $\mathcal{E} = \sum_r |u_r|^2$ and $U(z) = \sum_r u_r z^{-r}$. This is the ratio of the total energy of the unwanted aliasing transfer functions to the energy of the wanted transfer function. Aliasing energy ratios, listed in dB, are compared in Table A.1 for various versions of the DTCWT and for the DWT at several levels of decomposition. Note that the aliasing energy ratio for the DWT is many orders of magnitude greater than that of any of the DTCWTs.

Filters	A	B	C	DWT
Wavelet				
Level 1	$-\infty$	$-\infty$	$-\infty$	-9.40
Level 2	-31.40	-29.06	-14.11	-3.54
Level 3	-27.93	-25.10	-11.00	-3.53
Scaling fn.				
Level 1	$-\infty$	$-\infty$	$-\infty$	-9.40
Level 2	-32.50	-30.17	-15.93	-9.38
Level 3	-35.88	-29.21	-20.63	-9.37

Table A.1: Aliasing Energy Ratios in dB ($10 \log_{10} R_a$), where A indicates (13,19)-tap near-orthogonal filters at level 1 with 18-tap Q-shift filters at levels ≥ 2, B indicates (13,19)-tap near-orthogonal filters at level 1 with 14-tap Q-shift filters at levels ≥ 2, C indicates (5,3)-tap LeGall filters at level 1 with 6-tap Q-shift filters at levels ≥ 2, and DWT indicates a (13,19)-tap DWT. Adapted from [37].
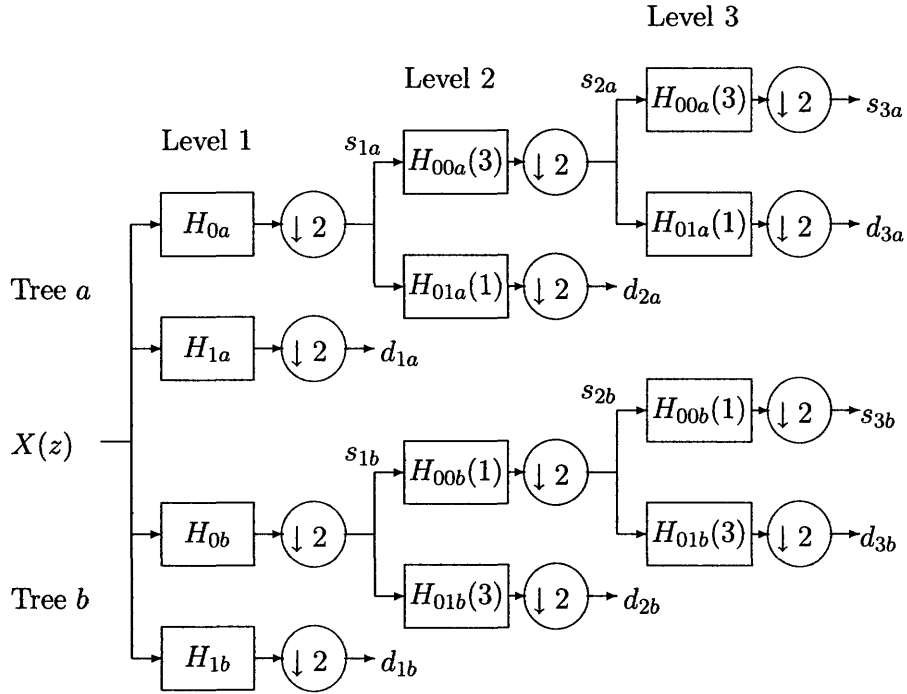
Figure A-1: DTCWT using q-shift wavelets. H_1 and H_0 are the first level discrete wavelet and scaling function filters respectively. H_{01} and H_{00} are the q-shift wavelet and scaling function filters; (1) indicates a one-quarter shift and (3) indicates a three-quarter shift. Adapted from [37].
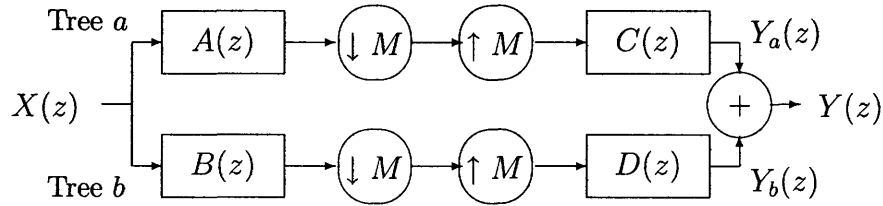

Figure A-2: Decomposition and recomposition of a single level of detail coefficients, using the DTCWT. If the third level of wavelets were retained, i.e. d_{3a} and d_{3b}, then $A(z) = H_{0a}(z)H_{00a}(z^2)H_{001a}(z^4)$ and $M = 8$. $B(z)$, $C(z)$, and $D(z)$ can similarly be determined from Figure A-1.

Appendix B

Probability of a Random Match

When using the wavelet transform to select feature points in an image, the question arises whether the wavelet transform extrema actually select feature points, or whether randomly selected points could achieve similar results. It is shown below that the probability of randomly selecting matching points, even with noise considered, is quite low.

B.1 Basic Probability

First, we begin with some basic probability concepts.

1. The probability of randomly selecting one specific item from a set of n items is $1/n$ [74]. For example, given a bowl of 10 marbles numbered 1 through 10, the probability of selecting marble number 3 in one draw is $1/10$.

2. The probability of randomly selecting one of m specific items from an set of n items is m/n [74]. For example, given a bowl of 10 marbles, 6 blue and 4 red, the probability of selecting a red marble in one draw is $4/10$.

3. The probability of an event is the number of combinations that produce the event divided by the total number of combinations [74].

4. Given a set of n items, the number of possible combinations of m items within that set, if order does not matter, is [74]

$$c = \binom{n}{m} = \frac{n!}{m! \, (n-m)!},$$

which is known as "n choose m." For example, given 6 marbles, 4 of which are red, there are $\binom{6}{4} = 15$ possible combinations of the marbles, assuming there is no differentiation among red marbles or among blue marbles; i.e., order of the marbles other than color does not matter so the sequence Red1-Red2-Blue1 is the same as Red2-Red1-Blue1.

B.2 Probability of a Random Match

Based on these ideas of basic probability, the following points applicable to our research have been developed.

B.2.1 Choosing less than the full amount

Assume you have a set of n items, m of which are desired. If you draw d items from the total, what is the probability that k of those d will be the desired items?

From item 3 above, the probability of such an event is the number of correct combinations divided by the total number of possible combinations. From item 4 above, the total number of possible combinations is $\binom{n}{d}$.

The number of correct combinations is the number of combinations of m items in k chances times the number of combinations of the incorrect/undesired items $(n-m)$ in the remaining positions $(d-k)$:

$$s = \binom{m}{k} \binom{n-m}{d-k}.$$

For example, six marbles, numbered 1 through 6, drawn 3 at a time, produce $\binom{6}{3} = 20$ possible combinations. If the desired marbles are numbered 1, 2 and 3, then

the number of combinations which produce two desired marbles is:

$$s = \binom{m}{k}\binom{n-m}{d-k} = \binom{3}{2}\binom{6-3}{3-2} = \binom{3}{2}\binom{3}{1} = 3 \cdot 3 = 9.$$

Thus, the probability of drawing exactly two desired marbles in three draws is $s/c = 9/20 = 45\%$. See Figure B-1.

Correct Combinations										Incorrect Combinations									
1	1	1	1	1	1	2	2	2	1	1	1	1	2	2	2	3	3	3	4
2	2	2	3	3	3	3	3	3	2	4	4	5	4	4	5	4	4	5	5
4	5	6	4	5	6	4	5	6	3	5	6	6	5	6	6	5	6	6	6
2									3	1									0

Figure B-1: Possible combinations of 3 marbles drawn from a set of 6 marbles numbered 1 through 6. The desired marbles are numbered 1, 2, and 3. The bottom row shows the number of desired marbles in each combination. Of 20 possible combinations, 1 has three desired marbles, 9 have two desired marbles, 9 have one desired marble, and 1 has zero desired marbles. For our example, the 'correct' combinations are those that have exactly two desired marbles.

B.2.2 Pixel Rings

To compensate for noise, we allow a choice that is 'close' to the specific desired item to meet our requirements. We define close as within a p-pixel ring in 2D or a p-voxel ring in 3D, where p is the number of concentric sets of points included. We define q as the number of points contained in the sets of rings. For a 2-pixel ring, $q = 25$. See Figure B-2.

The pixel ring concept introduces the difficulty that q varies with the location of the chosen sample. For example, randomly selecting a voxel within one pixel of the correct one in a 2D array results in $q = 9$ for all interior points, $q = 6$ for edges, and $q = 4$ for corners. See Figure B-3. In this assessment, we assume that q is the largest possible value, thus overestimating the probability of a random match.

In addition, we assume that the pixel rings do not overlap, thus again overestimating the probability of a random match.

```
X  X  X  X  X  X  X
X  -  -  -  -  -  X
X  -  O  O  O  -  X
X  -  O  X  O  -  X
X  -  O  O  O  -  X
X  -  -  -  -  -  X
X  X  X  X  X  X  X
```

Figure B-2: Pixel rings. The desired point is the red x. The one-pixel ring includes all blue o's. The two-pixel ring includes all magenta -'s. A voxel ring is similarly constructed in 3D; the one-voxel ring would include the blue o's shown plus those adjacent one plane in and out of the page, for a total of 26 blue o's.

```
X  X  X  X  X  O  X  O  X
X  O  O  O  X  O  O  O  X
X  O  X  O  X  X  X  X  X
X  O  O  O  X  X  X  X  X
X  X  X  X  X  X  X  O  O
X  X  X  X  X  X  X  O  X
```

Figure B-3: Number of samples in a pixel ring. Note that for a centered item, the number of points in the 1-pixel ring is $q = 9$; for an edge point, $q = 6$, and for a corner, $q = 4$.

B.2.3 Probability of choosing with noise

Given the formulation of Section B.2.1, assume that for each good point, m, there are q possible options. The number of correct combinations is still the number of combinations of desired items in k chances times the number combinations of undesired items in the remaining positions, $d - k$.

The number of combinations of desired items in k positions is $q^k \binom{m}{k}$. For example, assume two options $(q = 2)$ for each of three correct marbles $(m = 3)$, numbered 1a, 1b, 2a, 2b, 3a, 3b. The number of combinations of these marbles in three positions $(k = 3)$ is $q^k \binom{m}{k} = 2^3 \binom{3}{3} = 8$, as shown below:

```
1a   1a   1a   1a   1b   1b   1b   1b
2a   2a   2b   2b   2a   2a   2b   2b
3a   3b   3a   3b   3a   3b   3a   3b
```

The number of undesired items to be possibly placed in the remaining positions is the total number of items that are not in any pixel rings $(n - qm)$ plus the number

87

of items that are in the pixel rings that have already been chosen that are not higher priority $((q-1)k/2)$. This results in the following number of correct combinations:

$$s = q^k \binom{m}{k} \binom{n - qm + (q-1)\frac{k}{2}}{d - k}.$$

To ensure we have an integer within the *choose* statement we will round $k/2$ up, which again maximizes the probability of a random match.

As an example, using the marbles listed above (1a, 1b, 2a, 2b, 3a, and 3b), assume there is a fourth set of desirable marbles (4a and 4b), plus several non-desired marbles (5, 6, 7, etc.). If four draws are allowed $(d = 4)$, but only three desired marbles are drawn $(k = 3)$, then there are $q^k \binom{m}{k} = 2^3 \binom{4}{3} = 32$ combinations of desired items (those shown above plus several more using marbles 4a and 4b).

Take one of the possible combinations: 1a, 2b, 3b. The marble in the fourth position must meet the following requirements:

- It cannot be either 4a or 4b, or k would equal 4 (there would be four desired items in four draws).

- It can be any of the non-desired marbles (5, 6, 7, etc.); there are $n - qm$ of these.

- It can be 1b since drawing both 1a and 1b counts as choosing only one desired item (both are from the same pixel ring).

- It could be 2a for the same reason as it can be 1b, except that the combination falls under "1a, 2a, 3b, x" instead of "1a, 2b, 3b, x" in our listing scheme. Similarly, 3a is listed earlier.

B.3   Examples

As examples we present a 2D image that is 128 pixels on a side, for a total of $n = 16,384$ pixels. Probabilities for matching k of m points are shown in Table B.1.

m	k	Probability
20	20	1.87×10^{-28}
20	15	5.94×10^{-15}
20	10	5.03×10^{-7}
20	5	0.022
20	1	1.288
6	5	1.25×10^{-8}
6	3	2.71×10^{-4}
6	1	0.155

Table B.1: Probabilities for choosing k of m correct items out of 16,384 possible points.
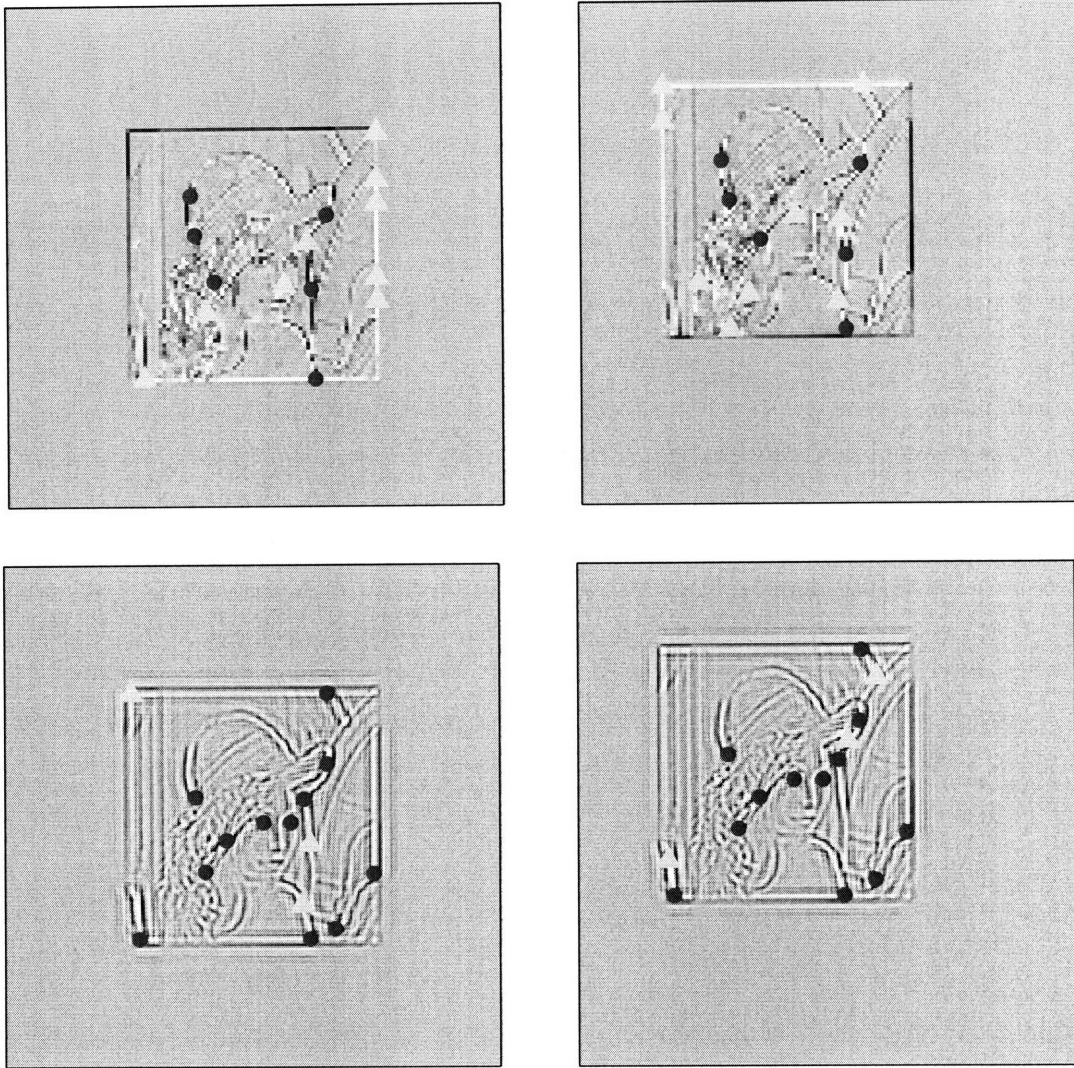
Figure B-4: Translation Variance. The image size is 128 x 128. The top row is the normal DWT: 6 of the top 15 extrema correspond in location. Circles denote matching extrema; triangles denote non-matching extrema. The probability of randomly matching 6 of 20 points within a 4-pixel ring is 1.52×10^{-4}. The bottom row is the DTCWT: 12 of the top 15 extrema correspond in location. The probability of randomly matching 12 of 15 points within a 4-pixel ring is 1.85×10^{-14}.
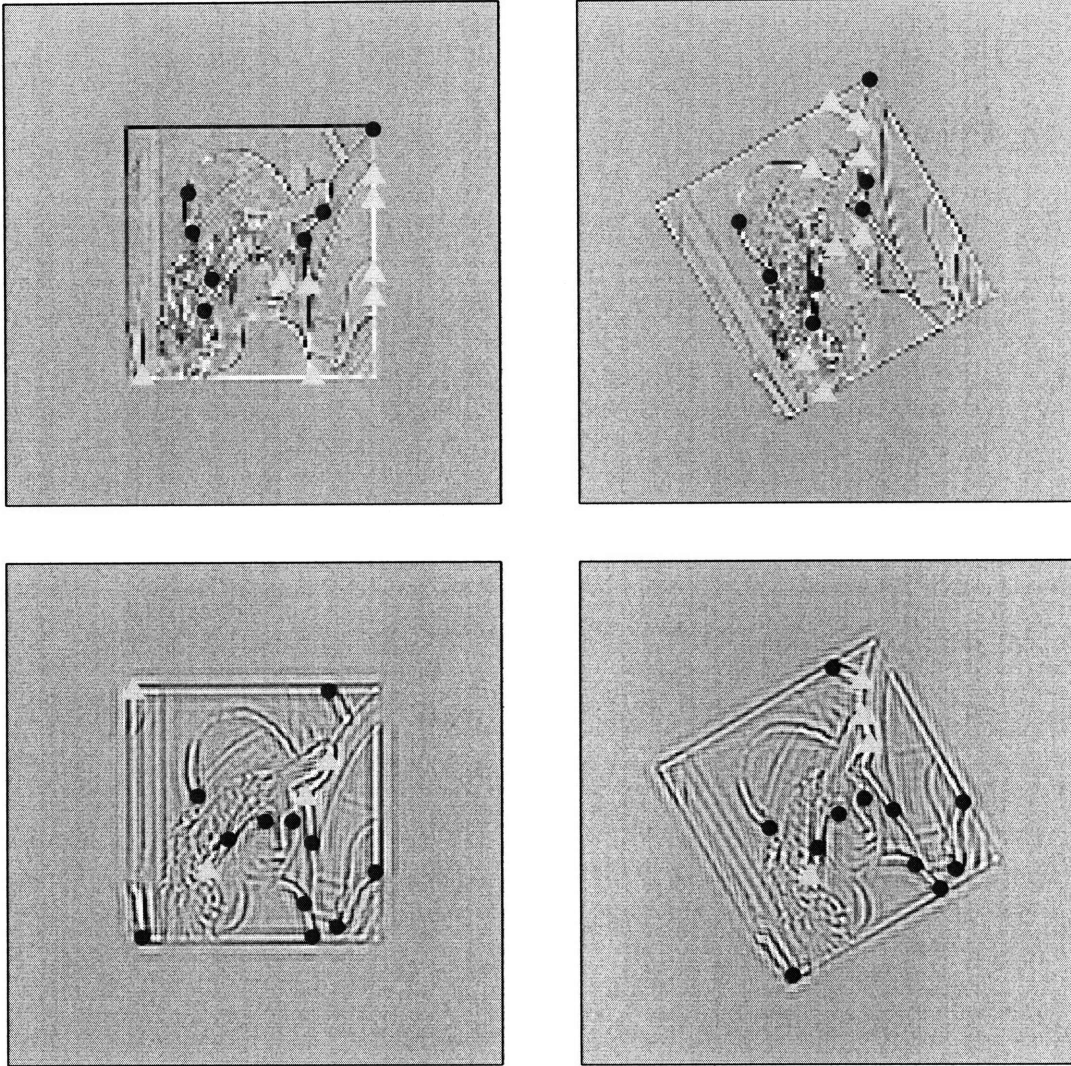
Figure B-5: Rotation Variance. The image size is 128 x 128. The top row is the normal DWT: 7 of the top 15 extrema correspond in location. Circles denote matching extrema; triangles denote non-matching extrema. The probability of randomly matching 7 of 15 points within a 4-pixel ring is 9.45 x 10^{-6}. The bottom row is the DTCWT: 11 of the top 15 extrema correspond in location. The probability of randomly matching 11 of 15 points within a 4-pixel ring is 2.66 x 10^{-12}.

Appendix C

Rotation Representations

This appendix provides details of several representations of rotation, along with conversion methods between representations.

C.1 Orthonormal Rotation Matrix

The rotation of a point, $\mathbf{p}$, through an angle, θ, can be represented using an orthonormal rotation matrix [60]:

$$\mathbf{p}' = \mathbf{R}\mathbf{p} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix}. \tag{C.1}$$

An orthogonal rotation matrix produces a rigid rotation. If the determinant of $\mathbf{R}$ is equal to one, this is a proper rotation. If the determinant of $\mathbf{R}$ is equal to -1, equation (C.1) produces an inversion with a rotation [94].

The following rotation matrix produces a counterclockwise rotation of an object in two dimensions about the origin through an angle, θ, while the axes remain fixed [94]:

$$\mathbf{R}_{\theta} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}.$$

Note that, although there are four terms in this rotation matrix, the terms are not independent; there is only one degree of freedom. Similarly, a three-dimensional rotation matrix has nine terms and three degrees of freedom.

C.2 Euler Angles

One convention of representing three dimensional rotations is the use of Euler angles, in which a three dimensional rotation may be represented as a sequential rotation about each of three different axes. We will use the pitch, roll, yaw convention, which rotates an object about the x, y, and z axes by angles θ, ϕ and ψ respectively, and in that order. Thus, the rotation matrix becomes [27]:

$$
\mathbf{R} = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}
$$

$$
= \begin{bmatrix} \cos\phi\cos\psi & \sin\theta\sin\phi\cos\psi - \cos\theta\sin\psi & \cos\theta\sin\phi\cos\psi + \sin\theta\sin\psi \\ \cos\phi\sin\psi & \sin\theta\sin\phi\sin\psi + \cos\theta\cos\psi & \cos\theta\sin\phi\sin\psi - \sin\theta\cos\psi \\ -\sin\phi & \sin\theta\cos\phi & \cos\theta\cos\phi \end{bmatrix} \cdot
$$

The pitch, roll and yaw angles, taken in that order, can be extracted from the rotation matrix above as follows:

$$
\begin{aligned}
\theta &= \arctan(R_{32}/R_{33}) \\
\phi &= -\arcsin(R_{31}) \\
\psi &= \arctan(R_{12}/R_{11})
\end{aligned}
\tag{C.2}
$$

Note that the order of rotation matters. For example, if the object is rotated

about the z axis first, then the y axis and then the x axis, the rotation matrix is:

$$
\mathbf{R} =
\begin{bmatrix}
1 & 0 & 0 \\
0 & \cos\theta & -\sin\theta \\
0 & \sin\theta & \cos\theta
\end{bmatrix}
\cdot
\begin{bmatrix}
\cos\phi & 0 & \sin\phi \\
0 & 1 & 0 \\
-\sin\phi & 0 & \cos\phi
\end{bmatrix}
\cdot
\begin{bmatrix}
\cos\psi & -\sin\psi & 0 \\
\sin\psi & \cos\psi & 0 \\
0 & 0 & 1
\end{bmatrix}
$$
$$
=
\begin{bmatrix}
\cos\phi\cos\psi & -\cos\phi\sin\psi & \sin\phi \\
\sin\theta\sin\phi\cos\psi + \cos\theta\sin\psi & -\sin\theta\sin\phi\sin\psi + \cos\theta\cos\psi & -\sin\theta\cos\phi \\
-\cos\theta\sin\phi\cos\psi + \sin\theta\sin\psi & \cos\theta\sin\phi\sin\psi + \sin\theta\cos\psi & \cos\theta\cos\phi
\end{bmatrix},
\tag{C.3}
$$

and the yaw, roll and pitch angles, taken in that order, are then:

$$
\begin{aligned}
\psi &= \arctan(-R_{12}/R_{11}) \\
\phi &= \arcsin(R_{13}) \\
\theta &= \arctan(-R_{23}/R_{33}).
\end{aligned}
\tag{C.4}
$$

Note that the inverse sine and inverse tangent functions are ambiguous; for a given value there are two possible corresponding rotation angles in the region $-\pi$ to π; either may be possible and the entire rotation matrix must be verified to ensure the correct value has been chosen.

C.2.1 Gimbal Lock

Using Euler angles has the complication that rotations of 90° can cause the once-unique axes to coincide, reducing the degrees of freedom described by the system from three to two. This is called gimbal lock. When using an Euler angle representation, care must be taken to avoid such situations.

During the first moon landing, a three-gimbal system was used to control the lunar lander. The following conversation was recorded between Command Module Pilot Mike Collins and CapCom Owen Garriott during the landing [34]:

Garriott: Columbia, Houston. We noticed you are maneuvering very close to gimbal lock. I suggest you move back away. Over.

Collins: Yeah. I am going around it, doing a CMC Auto maneuver to the Pad values of roll 270, pitch 101, yaw 45.

Garriott: Roger, Columbia. (Long Pause)

Collins: (Faint, joking) How about sending me a fourth gimbal for Christmas.

C.3   Axis-Angle

A fairly intuitive representation of rotation is the axis-angle representation, which consists of rotation of an angle, α, about a unit vector $\mathbf{n}$.

There are several difficulties in applying this method. Note that if α is zero, the axis vector is arbitrary. Also, there is an ambiguity in that the same rotation can be represented in two manners: if $R(\mathbf{n}, \alpha)$ is a rotation of angle α about unit vector $\mathbf{n}$, then $R(\mathbf{n}, \alpha) = R(-\mathbf{n}, -\alpha)$. If this is restricted by limiting α to rotation angles between zero and 180 degrees, then there is a singularity near zero and 180 degrees where the axis can jump significantly with a small change in rotation. A third difficulty is that two subsequent rotations cannot be combined to find a total rotation without converting to matrices or quaternions.

C.4   Quaternions

Quaternions are vectors with the representation [26]

$$\dot{q} = q_0 + q_1 \mathbf{e}_1 + q_2 \mathbf{e}_2 + q_3 \mathbf{e}_3, \qquad q_i \in R,$$

where the basis vectors, $\mathbf{e}_i$, observe the following convention:

$$\mathbf{e}_1^2 = \mathbf{e}_2^2 = \mathbf{e}_3^2 = -1,$$

$$\mathbf{e}_1 \mathbf{e}_2 = \mathbf{e}_3, \quad \mathbf{e}_2 \mathbf{e}_3 = \mathbf{e}_1, \quad \mathbf{e}_3 \mathbf{e}_1 = \mathbf{e}_2,$$

$$\mathbf{e}_2 \mathbf{e}_1 = -\mathbf{e}_3, \quad \mathbf{e}_3 \mathbf{e}_2 = -\mathbf{e}_1, \quad \text{and} \quad \mathbf{e}_1 \mathbf{e}_3 = -\mathbf{e}_2.$$

This has a simple correlation with the axis-angle representation as follows [27]:

$$
\begin{aligned}
q_0 &= \cos(\alpha/2) \\
q_1 &= \sin(\alpha/2)\cos(\beta_x) \\
q_2 &= \sin(\alpha/2)\cos(\beta_y) \\
q_3 &= \sin(\alpha/2)\cos(\beta_z),
\end{aligned}
$$

where β_x, β_y and β_z are the direction cosines of the unit vector $\mathbf{n}$ described above.

Rotation using a quaternion is formulated as follows [27]:

$$
\mathbf{p}' = \dot{q}\mathbf{p}\dot{q}^*
$$

where $\dot{q}^* = q_0 - q_1\mathbf{e}_1 - q_2\mathbf{e}_2 - q_3\mathbf{e}_3$ is the quaternion conjugate. From this, one can derive the orthogonal rotation matrix [26]:

$$
\mathbf{R} =
\begin{bmatrix}
1 - 2(q_2^2 + q_3^2) & 2(q_1q_2 - q_0q_3) & 2(q_0q_2 + q_1q_3) \\
2(q_1q_2 + q_0q_3) & 1 - 2(q_1^2 + q_3^2) & 2(q_2q_3 - q_0q_1) \\
2(q_1q_3 - q_0q_2) & 2(q_0q_1 + q_2q_3) & 1 - 2(q_1^2 + q_2^2)
\end{bmatrix}.
\tag{C.5}
$$

From this equation it becomes obvious that q and $-q$ produce the same rotation.

It follows from equations (C.2) and (C.5) that pitch-roll-yaw Euler angles to rotate about the z, then y, then x axes respectively are thus:

$$
\begin{aligned}
\theta &= \arctan\left(\frac{R_{3,2}}{R_{3,3}}\right) = \arctan\left(\frac{2(q_0q_1 + q_2q_3)}{1 - 2(q_1^2 + q_2^2)}\right) \\
\phi &= \arcsin\left(-R_{3,1}\right) = \arcsin\left(2(q_0q_2 - q_1q_3)\right) \\
\psi &= \arctan\left(\frac{R_{2,1}}{R_{1,1}}\right) = \arctan\left(\frac{2(q_1q_2 + q_0q_3)}{1 - 2(q_2^2 + q_3^2)}\right).
\end{aligned}
$$

Quaternions have several advantages over other methods. They avoid the gimbal lock problem of Euler angles. They can be combined easily as can rotation matrices, but it is easier to find a unit quaternion close to an un-normed quaternion than it is to find an orthogonal matrix near a non-orthogonal one.

Bibliography

[1] Paul S. Addison. The Illustrated Wavelet Transform Handbook: Introductory Theory and Applications in Science, Engineering, Medicine and Finance. Institute of Physics Publishing, Bristol, 2002.

[2] Pankaj K. Agarwal, Sariel Har-Peled, Micha Sharir, and Yusu Wang. Hausdorff distance under translation for points and balls. In Proceedings of the 30th International Conference on Computer Graphics and Interactive Techniques - SIGGRAPH 2003, pages 282–291, San Diego, CA, June 8-10 2003. ACM.

[3] G. Barequet and M. Sharir. Partial surface and volume matching in three dimensions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(9):929–948, September 1997.

[4] P. J. Besl. The free–form surface matching problem. In H. Freeman, editor, Machine Vision for Three–Dimensional Scenes, pages 25–71. Academic Press, 1990.

[5] P. J. Besl and N. D. McKay. A method for registration of 3D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(2):239–256, February 1992.

[6] P. Cachier and X. Pennec. 3D non-rigid registration by gradient descent on a Gaussian-windowed similarity measure using convolutions. In Proceedings of the IEEE Workshop on Mathematical Methods in Biomedical Image Analysis, 2000, pages 182 – 189, Hilton Head Island, SC, 2000. IEEE.

[7] Carlos J. Camacho and Sandor Vajda. Protein-protein association kinetics and protein docking. Current Opinion in Structural Biology, 12:36–40, 2002.

[8] P. Campisi, A. Neri, G. Panci, and G. Scarano. Robust rotation-invariant texture classification using a model based approach. IEEE Transactions on Image Processing, 13:782–791, 2004.

[9] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. In Proceedings of the 1991 IEEE International Conference on Robotics and Automation, pages 2724–2729, Sacramento, California, April 1991.

[10] C. S. Chua and R. Jarvis. 3D free-form surface registration and object recognition. International Journal of Computer Vision, 17(1):77–99, 1996.

[11] A. A. Cole-Rhodes, K. L. Johnson, J. LeMoigne, and I. Zavorin. Multiresolution registration of remote sensing imagery by optimization of mutual information using a stochastic gradient. IEEE Transactions on Image Processing, 12:1495–1511, 2003.

[12] A. Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Suetens, and G. Marchal. Automated multimodality image registration using information theory. In Yves Bizais, Christian Barillot, and Robert Di Paola, editors, Proceedings of the 14th International Conference on Information Processing in Medical Imaging: IPMI '95, pages 263–274, Ile de Berder, France, June 26-30, 1995.

[13] W. R. Crum, L. D. Griffin, D. L. G. Hill, and D. J. Hawkes. Zen and the art of medical image registration: Correspondence, homology, and quality. NeuroImage, 20:1425–1437, 2003.

[14] Ingrid Daubechies. Ten Lectures on Wavelets. Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania, 1992.

[15] Paul Debevec. Knee, 1993. http://www.debevec.org/Knee/.

[16] T. K. Dey, J. Giesen, and S. Goswami. Shape segmentation and matching with flow discretization. In Frank Dehne, Jörg Rüdiger Sack, and Michiel Smid, editors, Proceedings of the 8th International Workshop on Algorithms and Data Structures (WADS), volume 2748 of Lecture Notes in Computer Science, pages 25–36, Ottawa, Canada, 2003.

[17] M. N. Do and M. Vetterli. The Contourlet transform: an efficient directional multiresolution image representation. IEEE Transactions on Image Processing, 14(12):2091–2106, December 2005.

[18] C. Dorai and A. K. Jain. COSMOS-a representation scheme for 3D free-form objects. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(10):1115–1130, October 1997.

[19] Bruce Gooch Computer Scientist. Bruce Gooch's Brain, 2001. http://www.cs.utah.edu/~bgooch/BruceBrain.html.

[20] C. Guerra and V. Pascucci. Line-based object recognition using Hausdorff distance: from range images to molecular secondary structures. Image And Vision Computing, 23:405–415, 2005.

[21] S. Gundimada and V. Asari. Face detection technique based on rotation invariant wavelet features. In Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04), volume 2, pages 157–158, Las Vegas, Nevada, April 2004. IEEE.

[22] Inbal Halperin, Buyong Ma, Haim Wolfson, and Ruth Nussinov. Principles of docking: An overview of search algorithms and a guide to scoring functions. Proteins: Structure, Function, and Genetics, 47:409–443, 2002.

[23] Peter Hastreiter and Thomas Ertl. Integrated registration and visualization of medical image data. In Proceedings of Computer Graphics International (CGI), pages 78–85, Hannover, Germany, 1998. IEEE.

[24] Derek L. G. Hill, Philipp G. Batchelor, Mark Holden, and David J. Hawkes. Medical image registration. Physics in Medicine and Biology, 46:R1–R45, 2001.

[25] P. R. Hill, D. R. Bull, and C. N. Canagarajah. Rotationally invariant texture features using the dual-tree complex wavelet transform. In Proceedings of the International Conference on Image Processing (ICIP), pages 901–904, Vancouver, BC, Canada, September 10-13, 2000. IEEE.

[26] B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. Journal of the Optical Society of America, Series A, 4(4):629–642, 1987.

[27] J. Hoschek and D. Lasser. Fundamentals of Computer Aided Geometric Design. A. K. Peters, Wellesley, MA, 1993. Translated by L. L. Schumaker.

[28] P. W. Huang and S. K. Dai. Design of a two-stage content-based image retrieval system using texture similarity. Information Processing & Management, 40:81–96, 2004.

[29] Etienne Huot, Hussein Yahia, Isaac Cohen, and Isabelle Herlin. Matching structures by computing minimal paths on a manifold. Journal of Visual Communication and Image Representation, 13:302–312, 2002.

[30] Charles E. Jacobs, Adam Finkelstein, and David H. Salesin. Fast multiresolution image querying. In Proceedings of SIGGRAPH '95, the 22nd Annual Conference on Computer Graphics and Interactive Techniques, pages 277–286, Los Angeles, CA, August 6 – 11, 1995.

[31] B. Javidi, I. Moon, S. Yeom, and E. Carapezza. Three-dimensional imaging and recognition of microorganism using single-exposure on-line (SEOL) digital holography. Optics Express, 13:4492–4506, 2005.

[32] M. Jenkinson and S. Smith. A global optimisation method for robust affine registration of brain images. Medical Image Analysis, 5:143–156, 2001.

[33] A. E. Johnson and M. Hebert. Recognizing objects by matching oriented points. In Proceedings of the 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), pages 684–689, San Juan, Puerto Rico, June 1997.

[34] Eric M. Jones and Paul Fjeld. Gimbal angles, gimbal lock, and a fourth gimbal for Christmas, 2000. http://www.hq.nasa.gov/alsj/gimbals.html.

[35] A. Karasaridis and E. P. Simoncelli. A filter design technique for steerable pyramid image transforms. In Proceedings of the International Conference on Acoustics, Speech and Signal Processing, pages 2389–2392, Atlanta, GA, May 1996. IEEE.

[36] Nick Kingsbury. Image processing with complex wavelets. Philosophical Transactions of The Royal Society of London Series A-Mathematical Physical and Engineering Sciences, 357:2543–2560, 1999.

[37] Nick Kingsbury. Complex wavelets for shift invariant analysis and filtering of signals. Applied and Computational Harmonic Analysis, 10(3):234–253, 2001.

[38] K. H. Ko, T. Maekawa, and N. M. Patrikalakis. An algorithm for optimal free-form object matching. Computer-Aided Design, 35(10):913–923, September 2003.

[39] K. H. Ko, T. Maekawa, N. M. Patrikalakis, H. Masuda, and F.-E. Wolter. Shape intrinsic properties for free-form object matching. Journal of Computing and Information Science in Engineering, 3(4):325–333, 2003.

[40] Pavel Krsek, Tomáš Pajdla, and Václav Hlaváč. Differential invariants as the base of triangulated surface registration. Computer Vision and Image Understanding, 87:27–38, 2002.

[41] Hamid Laga, Hiroki Takahashi, and Masayuki Nakajima. Spherical wavelet descriptors for content-based 3D model retrieval. In Proceedings of the IEEE

International Conference on Shape Modeling and Applications, Shape Modeling International (SMI 2006), page 15, Matsushima, Japan, 2006. IEEE.

[42] Moon-Chuen Lee and Chi-Man Pun. Rotation and scale invariant wavelet feature for content-based texture image retrieval. Journal of the American Society for Information Science and Technology, 54(1):68–80, 2003.

[43] D. G. Lowe. Distinctive image features from scale-invariant keypoints. International Journal Of Computer Vision, 60:91–110, 2004.

[44] H. X. Luan, F. H. Qi, Z. Xue, L. Y. Chen, and D. G. Shen. Multimodality image registration by maximization of quantitative-qualitative measure of mutual information. Pattern Recognition, 41:285–298, 2008.

[45] L. Ma, T. N. Tan, Y. H. Wang, and D. X. Zhang. Efficient iris recognition by characterizing key local variations. IEEE Transactions on Image Processing, 13:739–750, 2004.

[46] J. B. A. Maintz and M. A. Viergever. A survey of medical image registration. Medical Image Analysis, 2:1–36, 1998.

[47] U. Malsch, C. Thieke, P. E. Huber, and R. Bendl. An enhanced block matching algorithm for fast elastic registration in adaptive radiotherapy. Physics In Medicine And Biology, 51:4789–4806, 2006.

[48] Matlab, 2007. The Mathworks, http://www.mathworks.com.

[49] Patrick Min. Binvox, 2006. http://www.google.com/search?q=binvox or http://www.cs.princeton.edu/ min/binvox/.

[50] X. Ming, T. Xu, and Z. X. Wang. Using multi-matching system based on a simplified deformable model of the human iris for iris recognition. In David Zhang and Anil K. Jain, editors, Proceedings of the First International Conference on Biometric Authentication, ICBA, volume 3072 of Lecture Notes in Computer Science, pages 434–441, Hong Kong, China, 2004.

[51] G. Mori, S. Belongie, and J. Malik. Efficient shape matching using shape contexts. IEEE Transactions On Pattern Analysis And Machine Intelligence, 27:1832–1837, 2005.

[52] P. Mrázek and J. Weickert. Rotationally invariant wavelet shrinkage. In B. Michaelis and G. Krell, editors, Proceedings of the German Association for Pattern Recognition (DAGM), volume 2781 of Lecture Notes in Computer Science, pages 156–163. Springer, 2003.

[53] David C. Munson, Jr. A note on Lena. IEEE Transactions on Image Processing, 5(1):3, 1996.

[54] A. Natsev, R. Rastogi, and K. Shim. WALRUS: A similarity retrieval algorithm for image databases. IEEE Transactions on Knowledge and Data Engineering, 16:301–316, 2004.

[55] A. Nelleri, U. Gopinathan, J. Joseph, and K. Singh. Three-dimensional object recognition from digital Fresnel hologram by wavelet matched filtering. Optics Communications, 259:499–506, 2006.

[56] N. S. Netanyahu, J. Le Moigne, and J. G. Masek. Georegistration of LANDSAT data via robust matching of multiresolution features. IEEE Transactions on Geoscience and Remote Sensing, 42:1586–1600, 2004.

[57] F. S. Nooruddin and G. Turk. Simplification and repair of polygonal models using volumetric techniques. IEEE Transactions on Visualization and Computer Graphics, 9:191–205, 2003.

[58] T. Pajdla and L. Van Gool. Matching of 3-D curves using semi-differential invariants. In 5th International Conference on Computer Vision, pages 390–395, Cambridge, MA, 1995. IEEE Computer Society Press.

[59] K. Park, D. Metaxas, and L. Axel. A finite element model for functional analysis of 4D cardiac-tagged MR images. In Randy E. Ellis and Terry M. Peters, editors, Medical Image Computing and Computer-Assisted Intervention - MICCAI

2003, Pt 1, volume 2878 of Lecture Notes in Computer Science, pages 491–498, 2003.

[60] N. M. Patrikalakis and T. Maekawa. Shape Interrogation for Computer Aided Design and Manufacturing. Springer-Verlag, Heidelberg, February 2002.

[61] V. Pekar, E. Gladilin, and K. Rohr. An adaptive irregular grid approach for 3d deformable image registration. Physics In Medicine And Biology, 51:361–377, 2006.

[62] Maurizio Pocci. 3D Valley, 2006. http://www.3dvalley.com/models.shtml.

[63] Chi-Man Pun and Moon-Chuen Lee. Extraction of shift invariant wavelet features for classification of images with different sizes. IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(9):1228–1233, 2004.

[64] Lawrence R. Rabiner and Charles M. Rader. Digital Signal Processing. IEEE Press, New York, 1972.

[65] Scott Reed, Yvan Petillot, and Judith Bell. An automatic approach to the detection and extraction of mine features in sidescan sonar. IEEE Journal of Oceanic Engineering, 28(1):90–105, January 2003.

[66] Phil Schewe and Ben Stein. MRI with 80-nanometer resolution. Physics News Update, 680:1, 2004.

[67] H. Schneiderman and T. Kanade. Object detection using the statistics of parts. International Journal of Computer Vision, 56:151–177, 2004.

[68] David W. Scott. Averaged shifted histograms: Effective nonparametric density estimators in several dimensions. The Annals of Statistics, 13(3):1024–1040, 1985.

[69] Y. Shan, H. S. Sawhney, B. Matei, and R. Kumar. Shapeme histogram projection and matching for partial object recognition. IEEE Transactions On Pattern Analysis And Machine Intelligence, 28:568–577, 2006.

[70] R. Shekhar, V. Walimbe, S. Raja, V. Zagrodsky, M. Kanvinde, G. Y. Wu, and B. Bybel. Automated 3-dimensional elastic registration of whole-body pet and ct from separate or combined scanners. Journal Of Nuclear Medicine, 46:1488–1496, 2005.

[71] H. Shimazaki and S. Shinomoto. A method for selecting the bin size of a time histogram. Neural Computation, 19:1503–1527, 2007.

[72] E. P. Simoncelli and W. Freeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. In Proceedings of the International Conference on Image Processing (ICIP), pages 444–447, Washington, DC, 1995. IEEE.

[73] B. J. Son, J. H. Ahn, J. H. Park, and Y. B. Lee. Identification of humans using robust biometric features. Structural, Syntactic, and Statistical Pattern Recognition, Proceedings, 3138:903–911, 2004.

[74] Murray R. Spiegel. Schaum's Outline Series, Theory and Problems of Statistics, 2nd Edition. McGraw-Hill, Inc., New York, 1988. Eighth printing, 1994.

[75] T. Srinark, C. Kambhamettu, and M. Stone. A hierarchical method for 3D rigid motion estimation. In P.J. Narayanan, Shree K. Nayar, and Heung-Yeung Shum, editors, Proceedings of the Asian Conference on Computer Vision - ACCV 2006, Pt II, volume 3852 of Lecture Notes in Computer Science, pages 791–800, Hyderabad, India, 2006.

[76] Stanford University. The Stanford 3D Scanning Repository, 2005. http://graphics.stanford.edu/data/3Dscanrep/.

[77] F. Stein and G. Medioni. Structural indexing: Efficient 3-D object recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(2):125–145, February 1992.

[78] Eric J. Stollnitz, Tony D. DeRose, and David H. Salesin. Wavelets for Computer Graphics. Morgan Kaufmann Publishers, Inc., San Francisco, California, 1996.

[79] G. Strang and T. Nguyen. Wavelets and filter banks. Wellesley-Cambridge Press, Wellesley, Massachusetts, 1996.

[80] C. Studholme, D. L. G. Hill, and D. J. Hawkes. Automated three-dimensional registration of magnetic resonance and positron emission tomography brain images by multiresolution optimization of voxel similarity measures. Medical Physics, 24:25–35, 1997.

[81] M. Styner, G. Gerig, J. Lieberman, D. Jones, and D. Weinberger. Statistical shape analysis of neuroanatomical structures based on medial models. Medical Image Analysis, 7:207–220, 2003.

[82] Yiyong Sun, Joonki Paik, Andreas Koschan, David L. Page, and Mongi A. Abidi. Point fingerprint: A new 3-D object representation scheme. IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics, 33(4):712–717, 2003.

[83] Johan W.H. Tangelder and Remco C. Veltkamp. Polyhedral model retrieval using weighted point sets. In Proceedings of the International Conference on Shape Modeling and Applications, pages 119–129, Seoul, Korea, May 12-15, 2003. IEEE.

[84] C. J. Tymczak, Anders M. N. Niklasson, and Heinrich Röder. Separable and nonseparable multiwavelets in multiple dimensions. Journal of Computational Physics, 175(2):363–397, 2002.

[85] M. Urschler, J. Bauer, H. Ditt, and H. Bischof. SIFT and shape context for feature-based nonlinear registration of thoracic CT images. Computer Vision Approaches To Medical Image Analysis, 4241:73–84, 2006.

[86] Ilya A. Vakser, Omar G. Matar, and Chan F. Lam. A systematic study of low-resolution recognition in protein-protein complexes. Proceedings of the National Academy of Sciences, 96:8477–8842, July 1999.

[87] D. Van De Ville, T. Blu, and M. Unser. Isotropic polyharmonic B-splines: Scaling functions and wavelets. IEEE Transactions on Image Processing, 14(11):1798–1813, November 2005.

[88] P. A. Viola. Alignment by maximization of mutual information. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 1995.

[89] F. Wang, B.C. Vemuri, M. Rao, and Y Chen. A new & robust information theoretic measure and its application to image alignment. In Chris Taylor and J. Alison Noble, editors, Proceedings of the Conference on Information Processing in Medical Imaging : IPMI03, volume 2732 of Lecture Notes in Computer Science, pages 388–400, Ambleside, UK, July 2003.

[90] James Ze Wang, Jia Li, Gio Wiederhold, and Oscar Firschein. System for screening objectionable images. Computer Communications, 21(15):1355–1360, 1998.

[91] James Ze Wang, Gio Wiederhold, Oscar Firschein, and Sha Xin Wei. Wavelet-based image indexing techniques with partial sketch retrieval capability. In Proceedings of the Fourth International Forum on Research and Technology Advances in Digital Libraries, pages 13–24. IEEE Computer Society, 1997.

[92] Y. J. Wang and C. S. Chua. Face recognition from 2D and 3D images using 3D Gabor filters. Image And Vision Computing, 23:1018–1028, 2005.

[93] Yongmei Wang, Bradley S. Peterson, and Lawrence H. Staib. 3D brain surface matching based on geodesics and local geometry. Computer Vision and Image Understanding, 89:252–271, 2003.

[94] Eric W. Weisstein. Rotation Matrix, 2006. From MathWorld–A Wolfram Web Resource. http://mathworld.wolfram.com/RotationMatrix.html.

[95] W.M. Wells, P. Viola, H. Atsumi, S. Nakajima, and R. Kikinis. Multi-modal volume registration by maximization of mutual information. Medical Image Analysis, 1:35–52, 1996.

[96] R. P. Woods, J. C. Mazziotta, and S. R. Cherry. MRI-PET registration with automated algorithm. Journal Of Computer Assisted Tomography, 17:536–546, 1993.

[97] R. Xu and Y. W. Chen. Wavelet-based multiresolution medical image registration strategy combining mutual information with spatial information. International Journal Of Innovative Computing Information And Control, 3:285–296, 2007.

[98] Z. Xue, D. G. Shen, and C. Davatzikos. Correspondence detection using wavelet-based attribute vectors. In Randy E. Ellis and Terry M. Peters, editors, Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003, Pt 2, volume 2879 of Lecture Notes in Computer Science, pages 762–770, 2003.

[99] S. M. Yamany and A. A. Farag. Surface signatures: An orientation independent free-form surface representation scheme for the purpose of objects registration and matching. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(8):1105–1120, 2002.

[100] X. Z. Yang, G. Pang, and N. Yung. Discriminative training approaches to fabric defect classification based on wavelet transform. Pattern Recognition, 37:889–899, 2004.

[101] S. Yeom and B. Javidi. Three-dimensional object feature extraction and classification with computational holographic imaging. Applied Optics, 43:442–451, 2004.

[102] S. Yeom, B. Javidi, Y. J. Roh, and H. S. Cho. Three-dimensional object recognition using x-ray imaging. Optical Engineering, 44(2):27201.1–23, 2005.

[103] Mitsuo Yokokawa, Ken'ichi Itakura, Atsuya Uno, Takashi Ishihara, and Yukio Kaneda. 16.4-Tflops direct numerical simulation of turbulence by a Fourier spectral method on the Earth Simulator. In Proceedings of the 2002 ACM/IEEE

Conference on Supercomputing, pages 1–17. IEEE Computer Society Press, 2002.

[104] H. Yoshida. Local contralateral subtraction based on bilateral symmetry of lung for reduction of false positives in computerized detection of pulmonary nodules. IEEE Transactions on Biomedical Engineering, 51:778–789, 2004.

[105] Q. Z. Yu, J. W. Tian, and H. Liu. A novel contour-based 3D terrain matching algorithm using wavelet transform. Pattern Recognition Letters, 25:87–99, 2004.

[106] Y. P. Zhang and R. L. Wang. Neural network combines with a rotational invariant feature set in texture classification. In Chengqi Zhang, Hans W. Guesgen, and Wai K. Yeap, editors, Proceedings of the Pacific Rim International Conference on Artificial Intelligence PRICAI 2004: Trends in Artificial Intelligence, volume 3157 of Lecture Notes in Computer Science, pages 436–444, 2004.

[107] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. International Journal of Computer Vision, 13(2):119–152, 1994.

[108] Hui Zhou, Takashi Takenaka, and Toshiyuki Tanaka. Three-dimensional reconstruction of a shallowly buried mine using time-domain data. Microwave and Optical Technology Letters, 39(4):276–280, 2003.