ATTITUDE DETERMINATION USING STELLAR IMAGES

A Thesis

Submitted to the College of Graduate Studies and Research

in Partial Fulfillment of the Requirements for the Degree of

MASTER of SCIENCE

in the Department of Electrical Engineering

University of Saskatchewan

Saskatoon, Saskatchewan, Canada

By

TAMMY K. YEE

March 1999

PERMISSION TO USE

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

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

Head of the Department of Electrical Engineering
University of Saskatchewan
Saskatoon, Saskatchewan
S7N 5A9
Canada

ABSTRACT

Attitude determination systems using stellar images and star catalogs are used to obtain accurate pointing directions of rockets, satellites and other spacecraft. The task is to match image stars to reference stars in a star catalog. The reference stars have known positions (right ascension and declination). If the image stars match a minimum of three reference stars, the positions are used to calculate the pointing direction of the spacecraft.

The focus of this research is to develop an attitude determination algorithm and to investigate implementation on a microprocessor for rockets and micro-satellites. Rockets and micro-satellites have relatively small budgets compared to spacecraft missions and therefore this research is directed towards a low cost, reliable solution.

Many algorithms have been developed to match the image stars to reference stars, some requiring initial attitude information and others which are fully autonomous. Fully autonomous systems are highly desirable because they are not susceptible to the failure of secondary instrumentation and are of lower cost. None of the previous methods provide an ideal solution to the problem. Therefore, the goal is to develop an algorithm for determining the attitude of a rocket or micro-satellite by using some of the 'best' concepts of the previous methods.

This thesis develops a fully autonomous algorithm to determine the positions of stars within an image, attempts to match those image stars and if successful, calculates the pointing direction of the image using a minimum of three stars. The technique used to match the stars is the angular separation method along with the geometry of the triad. The algorithm has been tested on several images from two different cameras. The testing included processing and comparison of results for actual rocket image data from the OEDIPUS-C Sounding Rocket.

The initial development of the algorithm was performed using Microsoft Visual C++ on a personal computer running Microsoft Windows NT 4.0 with floating-point calculation capability. While floating-point calculations are easy to work with, it is more likely that an integer arithmetic microprocessor would be flown aboard a rocket or micro-satellite. Therefore, the floating-point calculations were converted to fixed-point calculations in order to use integer calculations. The algorithm was tested both on a Pentium II processor and the Motorola MC68360 microprocessor. The algorithm tested on the Motorola MC68360 microprocessor was compiled using the GNU C compiler on a personal computer running Linux.

ACKNOWLEDGMENTS

The author would like to thank Dr. R. J. Bolton for his supervision and guidance throughout this research. The author appreciates his enthusiasm and interest in astronomy. His Star Compass Sky Map has proved invaluable in confirming the results of the developed attitude determination software.

The author would also like to thank Dr. E. Norum and Mr. D. Hall for their assistance with the MC68360 microprocessor.

The author gratefully acknowledges the financial assistance provided by the Natural Sciences and Engineering Research Council of Canada.

TABLE of CONTENTS

LIST of TABLES

LIST of FIGURES

LIST of ABBREVIATIONS

CCD	Charge-Coupled Device
CSA	Canadian Space Agency
FOV	Field of View
GEMINI	General Excitation Mechanisms in Nightglow
MB	Mega Bytes
MHz	Mega Hertz
NASA	National Aeronautics and Space Administration
RAM	Random Access Memory

Chapter 1

INTRODUCTION

1.1 Spacecraft Attitude

> The attitude of a spacecraft is its orientation in space. Attitude determination is the process of computing the orientation of the spacecraft relative to either an inertial reference or some object of interest, such as the Earth. Attitude control is the process of orientating the spacecraft in a specified predetermined direction [1].

Attitude information is required on many spacecrafts. Data instruments may require information about its orientation for data analysis. Two examples are the proposed Danish micro-satellite "Oersted", the purpose of which was to map the magnetic field of the Earth in the early 1990s [2] and the Canadian Space Agency's GEMINI experiment to study oxygen chemistry in the atmosphere in 1994 [3]. Attitude control is necessary to avoid solar damage of instruments or components, to point instruments in the proper direction and to maintain existing orientation of the spacecraft. In order to be able to control the attitude of a spacecraft, current attitude must be known.

There are many instruments that can provide attitude information for rockets and satellites. These are sun sensors, horizon sensors, magnetometers, gyroscopes and star sensors. Table 1.1 lists various reference sources and their corresponding theoretical accuracies [4].

Table 1.1 Reference sources for attitude determination and their theoretical accuracies.

Reference Object	Potential Accuracy
Stars	1 arc second
Sun	1 arc minute
Earth (Horizon)	6 arc minute
Magnetometer	30 arc minute

1.1.1 Star Sensors

In general, the most accurate attitude determination systems are star sensors. There are three types of star sensing and tracking devices: star scanners, gimbaled star trackers and fixed head star trackers [5]. Star scanners are used on spinning spacecraft and each time a star passes a slit, a pulse is generated to signal a star's presence. The crossing time of the star in the first slit is proportional to the star's azimuth angle and the elapsed time between crossing the first slit and the second slit is proportional to the star's elevation. Gimbaled star trackers have a small field of view (FOV) and move to locate and track reference stars. The gimbal angle readout position then gives the star's position. Fixed head star trackers do not have any moving parts and have a FOV large enough to image several stars. Some stars in the FOV must be matched to a reference set of stars and once that has been accomplished, the spacecraft attitude is known.

> The underlying assumption related to stellar recognition is that the stars in the celestial sphere form unique patterns [6].

This research is focused on fixed head star trackers.

1.2 Literature Review

Stellar image features are discussed before reviewing many star identification methods. Star identification methods may be fully autonomous or require a priori information. In the review, fully autonomous methods are discussed.

1.2.1 Stellar Image Features

Measured features from a star image must be translationally and orientationally independent [7] and independent of re-scaling [8]. These features are star magnitudes and spectral types, angular separations between stars and geometry of star groups [7].

Star magnitudes are used in a number of methods [6, 9, 10, 11] but there can be problems with the accuracy of the measured intensity which must be converted to a visual magnitude. A monochrome imaging system is more sensitive to certain spectral types and thus instrument intensity can be misleading [7]. In addition to measurement errors of the instrument, double or multiple stars would give the impression of a single star. As well, there are variable stars, the intensity of which changes with time [12]. For a monochrome imaging system, information about spectral type is not available and spectral type is rarely chosen as a feature.

The most widely used method to match stars is a variation of the angular separation technique [13]. Simply, the angular separation method matches angular separations between stars in the image with reference stars in a catalog.

The geometry of star groups is often used in addition to the angular separations and the triangle is often used as the star group formation.

The use of groups of 4 or more stars would allow for more reliability in a unique match but overall identification performance is reduced since more identification features must be determined without error [7].

1.2.2 Review of Star Identification Algorithms

Many star identification algorithms have been developed. There are some that are fully autonomous [2, 3, 6, 7, 9, 10, 11, 14, 15, 16, 17] and some which require a priori attitude estimates [18, 19, 20, 21, 22]. In this review, only fully autonomous algorithms will be discussed. Fully autonomous algorithms can be divided into two types: ones that use the magnitude feature and ones that do not.

1.2.2.1 Fully Autonomous Algorithms that Use the Magnitude Feature

Rappaport et al. [6] have developed an algorithm which determines angular separations between a root star and up to eight nearby stars in the image. The root star is the brightest star in the image. A list of catalog star pairs that match the magnitudes of the first star pair and the first angular separation is stored. A list of catalog star pairs that match the magnitudes of the second star pair and the second angular separation is stored as long as one of the components has been previously stored for the first pair. This continues for up to eight star pairs, until either only one match is found or no matches are found. If one match is found, the algorithm has matched the root star or if no matches are found, the searching stops.

Scholl [9, 10] has developed an algorithm which constructs triangles out of the 10 brightest stars in a charge-coupled device (CCD) image. Each triangle consists of six features, magnitudes of the three image stars and the three angular separations

between them. Each image triangle is compared to a database of triangles and none, one or several possible matches within set tolerances can be found. This is repeated for all the triangles and in the end, the image stars should be matched to the same set of catalog stars. If not, tolerances for the magnitude and angular separations are tightened until a single set of catalog stars are matched to the image stars. The software was run on a SPARC II workstation and star fields could be identified in less than one second. This algorithm was tested by mounting the CCD camera on a telescope at Table Mountain Observatory and was successful for the 18 images taken.

Ketchum and Tolson [11] have developed an algorithm which does not require a large database. The database consists of the star catalog and special reference stars that attempt to evenly cover the sky. There is no database of angular separations. The five brightest stars in the image are selected and their visual magnitudes are determined. Angular separations are calculated between the brightest star and up to four other stars. All the special reference stars which agree within a given tolerance of the magnitude of the brightest image star are selected. For each special reference star and its neighbors within a specified FOV, angular separations are calculated and called a sub-catalog. The image angular separations and magnitudes are compared against the sub-catalogs of angular separations and magnitudes. A match is considered to be successful when the angular separations and magnitudes between the brightest image star and two of its neighbors agree with a sub-catalog of stars.

1.2.2.2 Fully Autonomous Algorithms that Do Not Use the Magnitude Feature

Sasaki and Kosaka [14] proposed a method to match image stars to catalog stars. The image stars are arbitrarily numbered and angular separations are calculated. For each angular separation, a list of matching catalog star pairs is made. For each image star, the relevant lists of angular separations are searched to obtain a list of possible matching catalog stars. A catalog star is considered a possibility as long as it is found in each relevant list. By forming an arbitrary image triangle and using the lists of matching catalog star pairs for the image triad, possible catalog stars for each image star in the triad are deleted if it cannot form a triangle. Using all image triangles, the list of possible matching catalog stars for each image star is reduced until one candidate remains.

A star camera system designed for the Danish micro-satellite "Oersted" used the angular separations to the first and second neighboring star and the angle between them [2]. The features given above are compared to a database of triangles.

In 1994, Moorhouse presented a thesis titled "An Attitude Video Camera Using Stellar Images for the GEMINI Project" [3]. Moorhouse developed a fully autonomous algorithm to determine the attitude of a rocket. The angular separation method is used to identify the stars. Initially, the algorithm attempts to match the three brightest stars in the image, looking for a unique set of catalog stars. If a unique set of catalog stars does not result, then a fourth star is added to the previous three stars and another attempt is made with stars 1, 2 and 4 and stars 1, 3 and 4. The three lists of triads are scanned for identical catalog stars for each image star.

The Star Tracker Stellar Compass developed by the Lawrence Livermore National Laboratory was flown on the Clementine mission in 1994 [15]. Once the star centroids have been determined, triangles are formed and these triangles are compared to an on-board database of triangles. Once a matching triangle is found, the rotation angle between the image and the catalog triangles are determined and saved. Using up to 12 of the brightest stars in the image, the process continues until enough rotation angles agree and the stars are now considered to be matched. In the information available, there is no mention how many rotation angles must agree.

On November 6, 1995, the OEDIPUS-C Sounding Rocket was launched to study natural and artificial waves in the ionospheric plasma [16]. Bristol Aerospace Limited developed an attitude determination algorithm using the method of angular separations. Angular separations are calculated for up to eight brightest stars in the image. If the angular separations of three stars matches the database of angular separations within a 0.1° tolerance and a fourth star confirms the fit, the stars are considered identified.

A grid algorithm has been developed by Padgett and Kreutz-Delgado [17]. A square grid is used in developing the database of patterns. A reference star is chosen and is centered in the middle of the grid with its nearest neighbor to the right. The stars within this grid make up a bit vector; if a bit contains a star, the bit is "on" otherwise the bit is "off". A chosen star in the image would then also form a bit vector and would be compared to the database of bit vectors. The best match is determined and if the number of bits in agreement is above a threshold, the reference star of the database pattern is considered to be matched with the chosen star in the image. This would be

repeated for several stars in the image. A final check is used to ensure that the matched stars can be imaged in the same FOV. The drawback of this method is that the database is large and matching is inefficient. By reorganizing the database pattern information, the database can be reduced and matching becomes more efficient.

Quine and Durrant-Whyte [7] proposed a method to rapidly identify triad star patterns. The target star is selected in the central portion of the image. There are two annuli drawn around the target star, an inner annulus and an outer annulus. The two brightest stars lying within the outer annulus are chosen. The dimmest star is rejected and a new one is chosen if any angular separations are smaller than the inner annulus. The three stars are ordered according to geometry and not by their magnitude by noting star 1 as the target star, star 2 as the star forming the anti-clockwise edge of the triad as you rotate about star 1 and star 3 as the star forming the clockwise edge of the triad as you rotate about star 1. Once the stars have been ordered, the angular separations are calculated. The reference catalog of triads is searched using a binary search technique rather than a sequential search to rapidly match the image triad to a catalog triad or triads. This is repeated for several image triads, which are ordered in terms of brightest target star.

1.2.2.3 Critique of Previous Methods

As stated in Section 1.2.1, there are problems associated with using star magnitude information. For those reasons, all fully autonomous methods which use the magnitude feature are not desirable. The fully autonomous methods which do not use the magnitude feature are critiqued in more detail.

The method proposed by Sasaki and Kosaka requires that for a catalog star to be an initial possibility for an image star, it must appear in all relevant star pair lists. For example, if there were five image stars, and the image star of interest is image star 2, then the same catalog star would have to appear in the list of star pairs for angular separations 1-2, 2-3, 2-4, and 2-5. A correct catalog star may be easily excluded if the tolerance on matching the star pairs was set too low for just one angular separation.

Liebe proposed a technique of matching the distances to the two nearest neighbors along with the angle between them. False or no identifications can occur if the nearest neighbors are missing or incorrectly identified, therefore to reduce this problem it is suggested that a large FOV be used in order to image as many stars. For example, Liebe has suggested a FOV of 30°. Camera systems that have a smaller FOV should not use this method.

The Star Tracker Stellar Compass developed by the Lawrence Livermore National Laboratory has an interesting method to matching the stars, that is, using rotation information. The database of triangles would possibly be slow to search depending on how it was arranged and what search technique was used because a database of triangles is considerably larger than a database of angular separations.

Both Moorhouse and Beattie described algorithms that match using three and four stars. This may be adequate, however if more stars are used, then reliability and confidence of the matches is increased.

In the search for attitude algorithms, only one method used a grid type algorithm. In order for matching to take place, the nearest neighbor must be correctly identified in the image. Or if the image star lies near to the edge of the FOV, the

nearest neighbor in the image may not be the nearest neighbor in the reference database. The possible problem of using the incorrect nearest neighbor can be improved somewhat by consideration of a second nearest neighbor if using the nearest neighbor fails to produce a match.

In Quine and Durrant-Whyte's method, star misidentification occurs when triad features and groups are incorrectly extracted from the image or the star catalog. For example, the target star may not be contained in the reference catalog, the recorded and catalog brightness vary causing different stars being chosen to make up the triad, or there may be foreign objects in the image, etc. It is noted in the paper that misidentification rarely prevents star determination because it is possible to weight triad identifications.

1.3 Research Motivation

Many of the attitude systems reviewed were designed for satellites or interplanetary travel. These systems would have large budgets associated with them. The Star Tracker Stellar Compass used for the Clementine mission to map the surface of the moon retails for approximately $250,000 US [23]. In comparison, rockets and micro-satellites would have smaller budgets. A low cost attitude system for rockets or micro-satellites is the intended purpose of this research.

For the GEMINI rocket in 1994, Moorhouse developed a stellar attitude system because of the need for a low cost system with a capability of determining the attitude to better than 2 minutes of arc. Moorhouse [3] states that the sounding rocket program in the United States has used star trackers and that they were extremely expensive.

In order to provide a low cost attitude system, the least amount of instrumentation is required. This can be achieved by developing a fully autonomous system. A priori attitude system requires coarse attitude information from other instruments and in addition to the increase in cost, failure of a coarse attitude instrument would not allow the a prior system to function. The argument may be made that a fully autonomous stellar attitude system could also fail and while this is obviously a possibility, it has been noted that fixed head star trackers do not contain moving parts.

There have been many fully autonomous attitude methods developed and no one method is ideal. Each method has shortcomings which were described in Section 1.2.2.3. Therefore, another attitude algorithm is investigated which uses some of the 'best' concepts from these previous methods.

The initial development platform of the algorithm is a personal computer which is capable of floating-point arithmetic. While this is easy to design and test with, the implementation aboard a rocket or micro-satellite would likely be an integer arithmetic microprocessor because of lower cost and power requirements. With this in mind, the floating-point implementation will be converted to fixed-point implementation in order to use integer arithmetic. Fixed-point implementation splits up an integer into two parts, a whole part and fractional part, with an 'imaginary' fixed point to separate the two parts. An advantage of using integer versus floating-point arithmetic is that integer arithmetic is faster that floating-point arithmetic.

1.4 Research Objectives

The objectives of the research work in this thesis are:

- To perform image processing on a stellar image to determine the locations of the stars.
- To investigate previous attitude determination algorithms.
- To develop a fully autonomous algorithm to perform pattern recognition of stars in an image.
- To modify the algorithm for implementation on a microprocessor that only supports integer arithmetic.

1.5 Outline of Thesis

Chapter 2 describes the attitude determination algorithm developed. There are three main parts: the method used to find the stars in the image, the identification of stars using pattern recognition and the calculations required to determine the attitude of the image. Chapter 3 explains the considerations required for implementation on a microprocessor that only supports integer calculations. This was investigated because an integer arithmetic microprocessor is likely the type of processor that will be used aboard a rocket or micro-satellite and integer calculations take less time to execute than floating-point calculations. Chapter 4 presents and discusses the results of processing several images. Chapter 5 concludes the work presented in the thesis along with recommendations for future work.

Chapter 2

ATTITUDE DETERMINATION ALGORITHM

2.1 Introduction

The algorithm described attempts to match the star field image with stars in a catalog. The angular separation technique and the geometry of triad groups will be used. If a minimum of three image stars can be matched to catalog stars, the attitude of the image can be determined. The attitude of the image is arbitrarily defined as any point on the image and is chosen as the center of the image.

2.2 Finding Stars in an Image

When a star is imaged by an optical system, the optical system causes the star image to be blurred. The point spread function of an optical system is defined as the system response to a point source input and this response causes the star image to be spread out. Using centroiding, the locations of the stars can be accurately determined. The centroids in the x and y direction are given by

$$\bar{x} = \frac{\sum_x \sum_y x f(x,y)}{\sum_x \sum_y f(x,y)} \tag{2.1}$$

$$\bar{y} = \frac{\sum_x \sum_y y f(x,y)}{\sum_x \sum_y f(x,y)} \tag{2.2}$$

where $f(x, y)$ is the pixel brightness at the position (x, y) and the sums are over all the pixels in the region of interest.

In order to process the image as quickly as possible, each pixel should only be evaluated once. This is accomplished using a pattern recognition technique called clustering [24]. Clustering is the grouping together of similar data. Considering Figure 2.1, it can be seen that the data would form two groups or clusters. In two-dimensional space, a simple way of clustering is to specify a circle of radius τ around a cluster center and any data that lies inside the circle is grouped together. If there is data that does not lie inside an existing cluster, a new cluster would be formed.

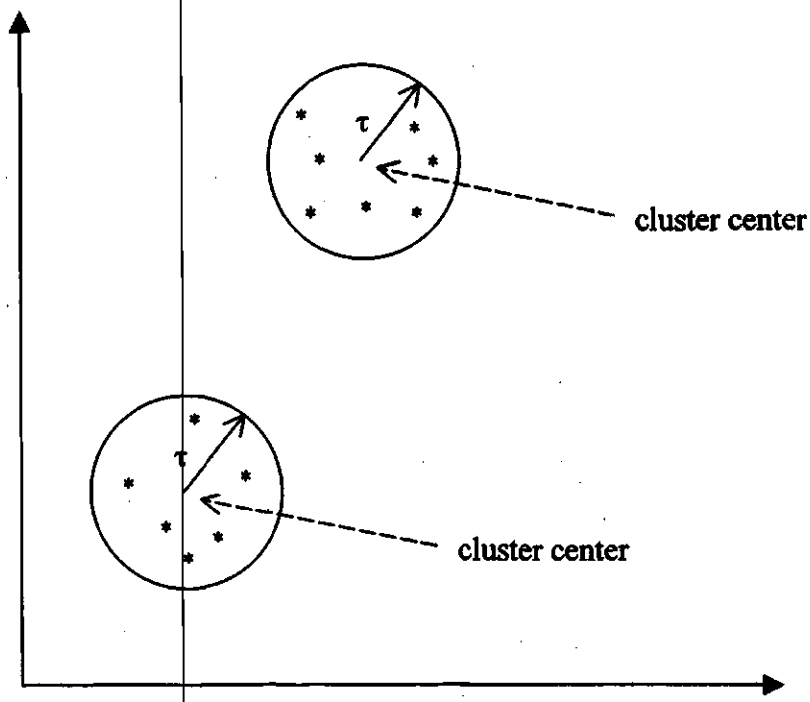

Figure 2.1 Example of data with two clusters.

To determine the centroids of stars, the clustering technique will be used. The algorithm begins evaluating pixels in the upper left corner of the image and moves to the right. When the end of a row is encountered, the left most pixel of the next row is evaluated and once again, the algorithm moves to the right. This process is repeated until each pixel in the image has been evaluated. When a pixel above some brightness threshold is encountered by the algorithm, it checks to see if it is near any existing cluster centers by a distance of τ. If it is, then it will be added to the cluster of existing pixels. If not, a new cluster center is formed. As pixels are grouped together, the centroid of each bright spot is automatically recalculated. τ is a user specified parameter.

After the centroids of all the stars in the image have been determined, any stars that are not composed of a minimum number of pixels are discarded. The remaining list is then sorted in descending order of the number of pixels making up the star. This is done because it is assumed that the probability of finding brighter image stars in the star catalog is greater than finding dimmer stars. Also, centroids of brighter stars can be more accurately determined than centroids of dimmer stars.

Figure 2.2 is an example of a star field image that has been processed to find the star centroids. This image was obtained from Moorhouse [3] and he identified the star labeled '1' to be Vega. The five brightest stars are labeled in this image.

2.3.2 Ephemeris

The Yale Bright Star Catalog [25] lists information for 9110 stars on the celestial sphere down to a visual magnitude of 8. Visual magnitude is discussed in Appendix A. It is interesting to note that the faintest stars that can be seen with the naked eye are magnitude 6 stars [26] and therefore, the majority of the stars in the Yale Bright Star Catalog are stars that can be seen with the naked eye. The fourth edition of the Yale Bright Star Catalog was compiled by Hoffleit in 1982 and was modified into a machine-readable version by Warren in 1982. Each star is given an identifying number known as the HR number which is an integer ranging from 1 to 9110. The position of each star is given by its right ascension and declination. Right ascension is the azimuth measured east from the vernal equinox along the celestial equator. Declination is the elevation measured from the celestial equator. Right ascension and declination are defined more detail in Appendix A. The accuracy of the star positions in the catalog are 0.1 seconds in right ascension and 1" in declination for the equinox date of 2000.

2.3.3 Databases

Two databases are required in the identification of stars in an image. The first database is a sub-catalog of stars for the section of sky that the rocket or micro-satellite would be in. For example, this could be the upper hemisphere. The second database is the database of angular separations between stars in the sub-catalog. This is required in order to match the angular separations in the image with angular separations in the database.

2.3.3.1 Sub-catalog of Stars

A sub-catalog of stars is formed from the Yale Bright Star Catalog. The parameters of the formation of this sub-catalog are the limiting visual magnitude, the right ascension and declination of the window center, and the spread in right ascension and declination of the window center.

The sub-catalog consists of a four column array: HR number, right ascension, declination and magnitude.

An example sub-catalog of stars is shown in Table 2.1 for the constellation Cassiopeia.

Table 2.1 Sub-catalog for the constellation Cassiopeia.

HR number	right ascension (hours) +/- 0.000028	declination (degrees) +/- 0.00028	visual magnitude
21	0.152944	59.14972	2.27
168	0.675111	56.53750	2.23
264	0.945111	60.71667	2.47
403	1.430250	60.23528	2.68
542	1.906556	63.67028	3.38

2.3.3.2 Angular Separation Database

The angular separation database is built from the sub-catalog of stars. The angular separation between two stars is calculated from [26]

$$\cos\theta = \cos(90° - \delta_1)\cos(90° - \delta_2) + \sin(90° - \delta_1)\sin(90° - \delta_2)\cos(\alpha_2 - \alpha_1) \quad (2.3)$$

where θ is the angular separation between the two stars, (α_1, δ_1) is the right ascension and declination of the first star in degrees and (α_2, δ_2) is right ascension and declination of the second star in degrees. The formula is discussed in Appendix A.

The maximum angular separation of the camera is a parameter in the formation of this database. Any angular separations which are greater than this maximum angular separation are not included in the database because the camera would not be able to simultaneously image two stars with an angular separation that is greater than the camera FOV.

The angular separation database consists of a three column array: angular separation in degrees, the HR number of the first delimiting star and the HR number of the second delimiting star. The database is sorted in ascending order based on the angular separations.

Using the sub-catalog of stars in the constellation Cassiopeia, an example angular separation database is shown in Table 2.2.

Table 2.2 Angular separation database for the constellation Cassiopeia.

angular separation (degrees)	HR number of the first delimiting star	HR number of the second delimiting star
3.6163	264	403
4.6789	168	264
4.7994	403	542
4.9149	21	168
6.1465	21	264
6.9811	168	403
7.3266	264	542
9.6946	21	403
11.5754	168	542
13.2602	21	542

2.3.4 Matching the Star Pattern

After the centroids of the image stars are determined and sorted according to apparent brightness, angular separations between all the image stars are calculated. The angular separations between two image stars is calculated from

$$\cos\theta = \cos\chi\cos\psi \tag{2.4}$$

where θ is the angular separation between the two stars, χ is the angular difference between the two stars in the x direction (horizontal) and ψ is the angular difference between the two stars in the y direction (vertical). In order to calculate χ and ψ, the scale factors (degrees per pixel) in the x and y direction must be known. These scale factors are determined by analysis of the camera FOV in the x and y directions and the number of image pixels in the x and y directions. See Appendix A for the derivation of Equation (2.4).

At this point, a maximum number of image stars will be considered for matching by the algorithm. This parameter is specified by the user. The number chosen should not be too large as computation time increases with the number of image stars that are attempted to be matched. The minimum value of this parameter is three. As stated previously, a minimum of three image stars must be identified in order to calculate the attitude of the image. For this discussion, twelve was chosen as the maximum number of image stars to be matched. This is the same value that is used in the Star Tracker Stellar Compass from Lawrence Livermore National Laboratory [15].

The calculated angular separations between a maximum of twelve brightest image stars are compared to the database of angular separations. For each calculated angular separation, a list of possible HR number pairs with matching angular

separations within a tolerance is found. This tolerance allows for distortion in the camera. The binary search method is used to find the minimum matching angular separation and star pairs are saved until the maximum angular separation is reached. There are sixty-six lists of possible HR number pairs for 12 stars.

Twelve image stars are taken three at a time to form 220 unique triangles. For example, let us assume the triangle we are considering is formed by image stars 1, 2 and 3. There are three angular separations between these stars and three lists of HR number pairs. The algorithm searches for triads of HR numbers that can be matched to image stars 1-2-3. Denote a possible catalog triad as HR1-HR2-HR3. Each triad is subjected to two tests before it is considered a possible match. The first test is the rotation angle test and the second is the enclosed angle test.

2.3.4.1 Rotation Angle Test

The rotation angle test is a test that rotates the image stars to match the catalog stars. In the image, two vectors are used, the vector from star 1 to star 2 and the vector from star 1 to star 3. For the catalog triad, the vector from HR1 to HR2 and the vector from HR1 to HR3 are constructed as if the spherical triangle was a planar triangle. This is done because once the algorithm is implemented in fixed-point format, the calculations are simpler. The angle to rotate image vector 1-2 to catalog vector HR1-HR2 is defined as rotation angle 1. The angle to rotate image vector 1-3 to catalog vector HR1-HR3 is defined as rotation angle 2. Rotation angle 1 is compared to rotation angle 2 and if these two angles agree within the rotation angle tolerance, the test is passed. If not, the catalog triad is discarded.

An example is shown for the Vega image and the triangle to consider is formed by image stars 1, 2 and 3. Figure 2.3 shows the image triangle 1-2-3 with the vectors 1-2 and 1-3 drawn in. The angle from the dotted horizontal line drawn out from image star 1 to vector 1-2 is 173.3° and the angle from the dotted horizontal line drawn out from image star 1 to vector 1-3 is 182.8°. This dotted horizontal line is chosen parallel to the x axis of the image.

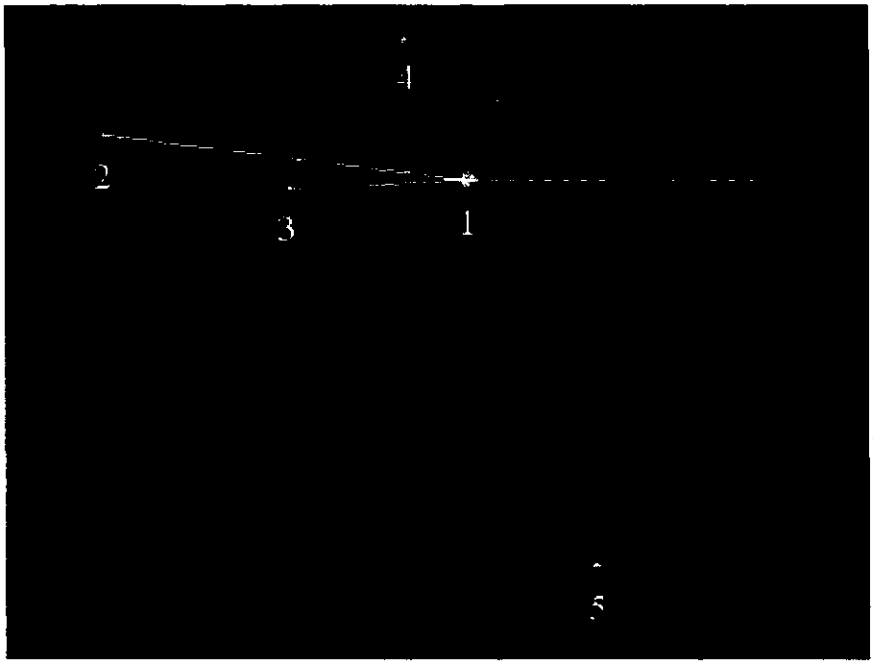

Figure 2.3 Vega image with two vectors shown for the rotation angle test.

Two possible catalog triads are shown below, one which passes the rotation angle test and one which does not. The first example shown is a triad that passes the rotation angle test. A possible catalog triad is formed by HR 7001, HR 7139 and HR 7056. Figure 2.4 is a Star Compass Sky Map [27] showing these three stars and the appropriate vectors. The two vectors used are the vectors from HR 7001 to HR 7139

and HR 7001 to HR 7056. The angle from the dotted horizontal line drawn out from HR 7001 to vector HR 7001-HR 7139 is 203.2° and the angle from the dotted horizontal line drawn out from HR 7001 to vector HR 7001-HR 7056 is 211.0°. This dotted horizontal line is chosen parallel to the declination axis of the Star Compass Sky Map. Rotation angle 1 is defined as the angle to rotate image vector 1-2 to catalog vector HR 7001-HR 7139 and rotation angle 2 is defined as the angle to rotate image vector 1-3 to catalog vector HR 7001-HR 7056. These angles are always measured in the counterclockwise direction. In this case, rotation angle 1 is equal to 29.9° (|203.2° – 173.3°|) and rotation angle 2 is equal to 28.2° (|211.0° – 182.8°|). The rotation angle tolerance was set at 5° for this image and therefore this catalog triad passed the rotation angle test.

The next example is of a catalog triad that does not pass the rotation angle test. This catalog triad is shown in Figure 2.5 and is composed of catalog stars HR 1063, HR 969 and HR 1001. The angle from the dotted horizontal line drawn out from HR 1063 to vector HR 1063-HR 969 is 36.2° and the angle from the dotted horizontal line drawn out from HR 1063 to vector HR 1063-HR 1001 is 22.2°. In this case, rotation angle 1 is equal to 222.9° (|36.2° – 173.3°|) and rotation angle 2 is equal to 199.4° (|22.2° – 182.8°|). The rotation angle tolerance was set at 5° for this image and therefore this catalog triad did not pass the rotation angle test. From this example, it can be seen that the rotation angle test accomplishes more than simply measuring the enclosed angle. This catalog triad matches the image triad in angular separations, but its orientation is a mirror image of the image triad.

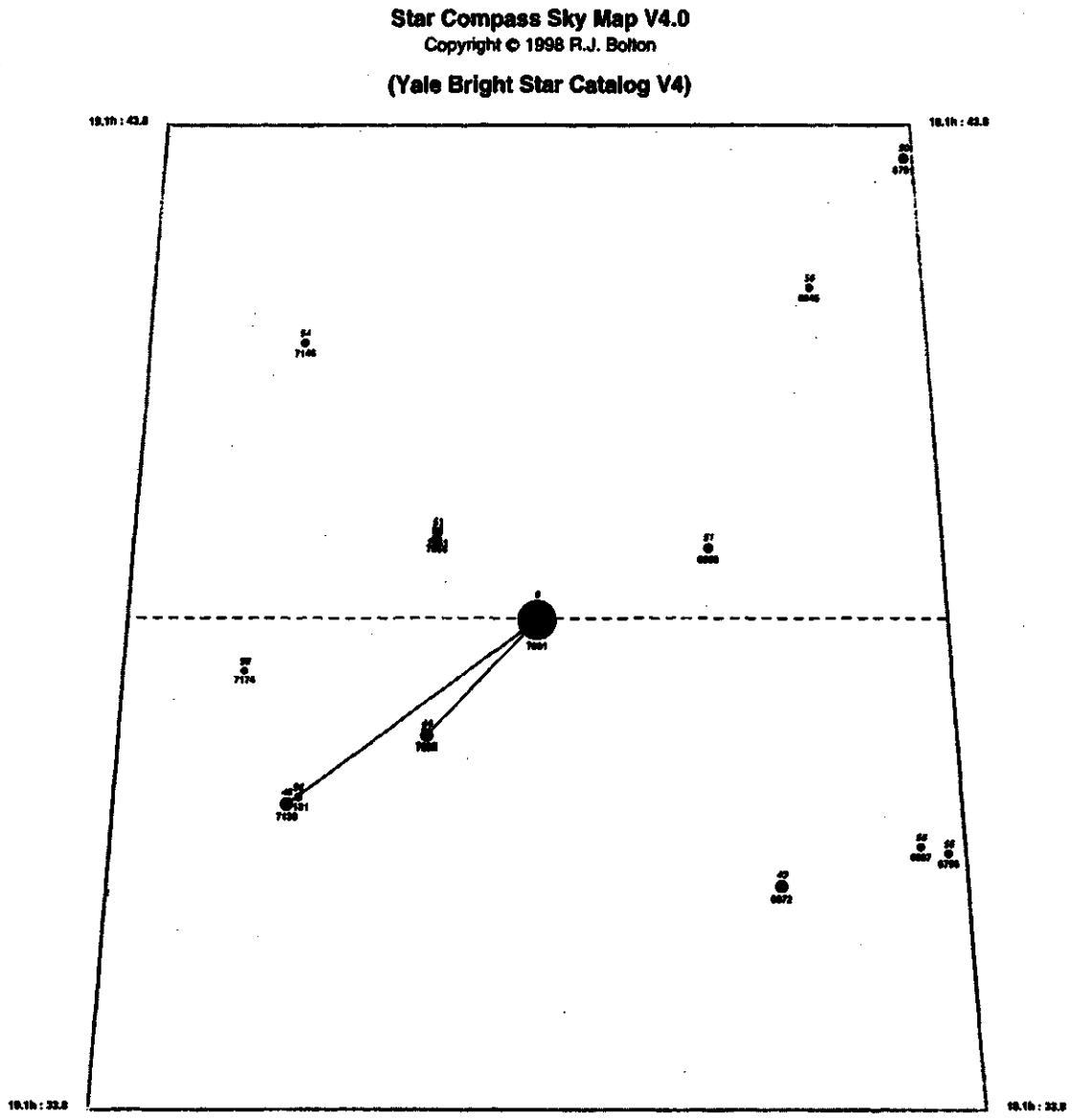

Figure 2.4 Star Compass Sky Map of catalog triad HR 7001, HR 7139 and HR 7056.

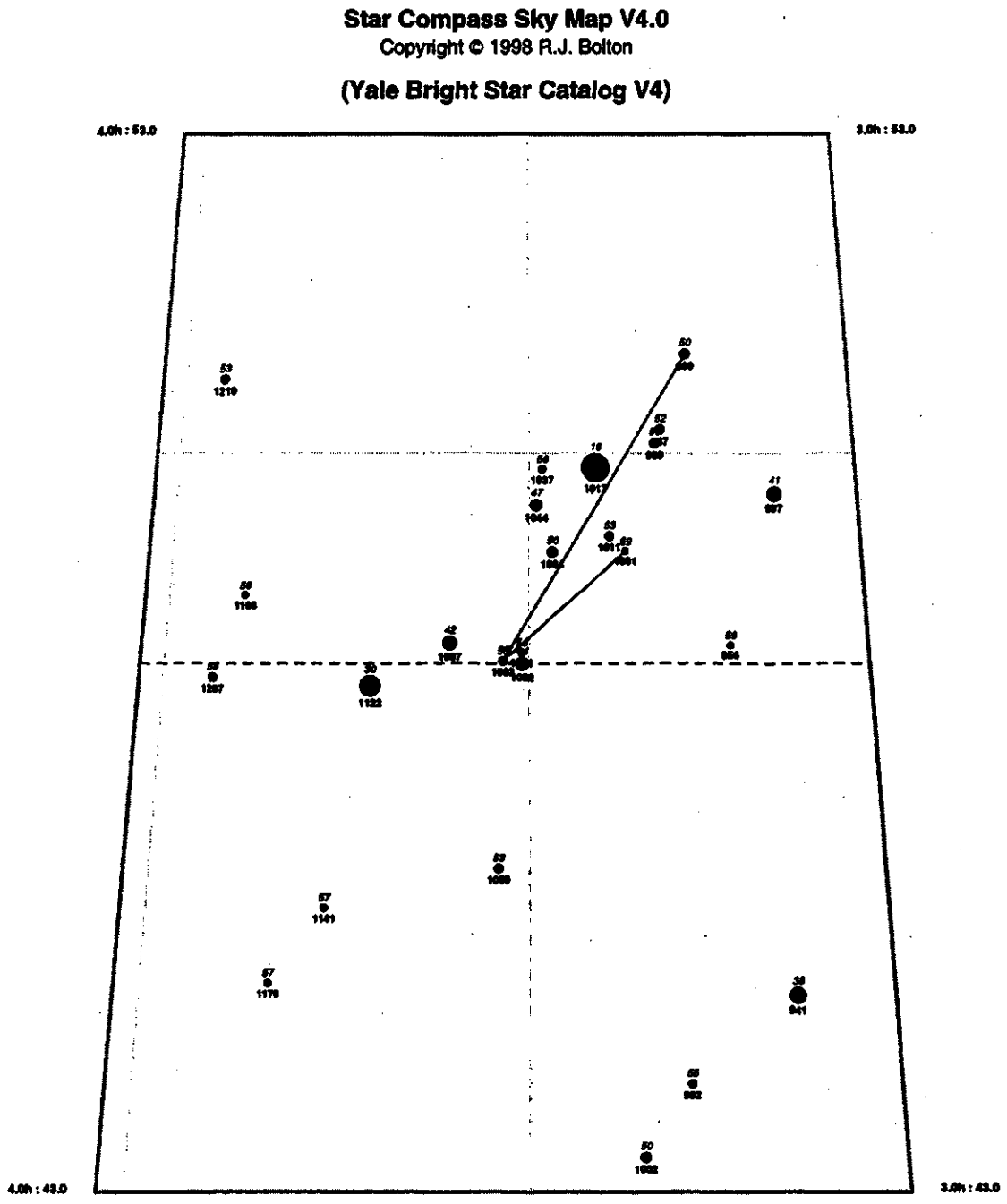

Figure 2.5 Star Compass Sky Map of catalog triad HR 1063, HR 969 and HR 1001.

2.3.4.2 Enclosed Angle Test

If the rotation angle test is passed, the catalog triad is subjected to the enclosed angle test. This test compares the enclosed or inner angles of the image triad against the catalog triad. Figure 2.6 shows a spherical triangle with its enclosed angles labeled. The equations used to calculate enclosed angles 'A', 'B' and 'C' are given by [26]

$$\cos a = \cos b \cos c + \sin b \sin c \cos A \tag{2.5}$$

$$\cos b = \cos a \cos c + \sin a \sin c \cos B \tag{2.6}$$

$$\cos c = \cos a \cos b + \sin a \sin b \cos C \tag{2.7}$$

where a is the angular separation between vertices B and C, b is the angular separation between vertices A and C, c is the angular separation between vertices A and B, A is the inner angle A (=∠BAC), B is the inner angle B (=∠ABC) and C is the inner angle C (=∠ACB). Note here that 'A', 'B' and 'C' represents inner angles and vertices.

Each angle must agree within the enclosed angle tolerance to pass the test. If the test is not passed, the catalog triad is discarded. If the test is passed, the catalog triad is saved by grouping it with other triads that have similar average rotation angles. The triads are grouped by average rotation angle because all image triads would be rotated approximately the same amount to match the correct catalog triads.

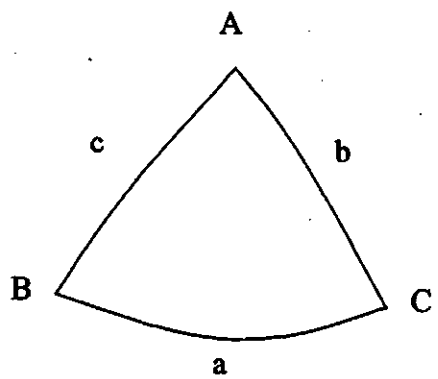

Figure 2.6 Enclosed or inner angles of a spherical triangle.

2.3.4.3 Saving Triads and Matching Each Image Star

An entry in a list of triads is shown below. This example is for image triad 1, 5 and 7 :

2349 0 0 0 2563 0 2908 0 0 0 0 0

The places where the HR numbers reside in the list indicate which stars they were matched to in the image. HR number 2349 is a possible match for image star 1, HR number 2563 is a possible match for image star 5 and HR number 2908 is a possible match for image star 7.

The algorithm continues this process of testing and saving successful triads for all possible catalog triads that are formed for each unique triad of image stars. It is assumed that the list(s) of triads with the largest number of entries are likely to contain the correct matches for the image stars.

For a list of triads with the largest number of entries, each image star is attempted to be matched to a catalog star. For each image star, a list is formed of possible HR numbers that could be matched to the image star and the frequency with which each HR number appeared. After the lists are formed, each list is scanned to see if any of the possibilities can be combined to one possibility because they are close enough to be imaged as one star. If two or more HR number possibilities are combined, the frequency of the HR numbers contributing are summed and the right ascension and declination of this new possibility is set to the brightest star making up this new possibility. Finally for each star, the HR number possibility that appeared with the

highest frequency is chosen as its match as long as the frequency is greater than or equal to a 'vote number' and if it is the only HR number possibility with the highest frequency. The 'vote number' allows the user to set a confidence level on a match.

The final test is to check that the angular separations of the catalog stars agree with the angular separations of the image stars. If some matches do not agree, they are discarded. To be able to determine the attitude of the image, three matched stars must remain at the end.

If there is more than one list of triads with the largest number of entries, the process discussed for a list of triads is performed for each list of triads until a minimum of three stars are matched.

Flowcharts of matching the star pattern are given in Appendix B.

2.4 Determining the Attitude of the Image

Once a minimum of three image stars have been matched to catalog stars, the right ascension and declination of any point on the image can be determined. This can be done once the right ascension and declination of the center of the image is determined. The pointing direction or attitude of the image is chosen to be the center of the image.

When an image of the celestial sphere is taken, the celestial sphere is mapped on to the planar surface of the photographic imaging device. In photographic astronomy, the position of an object in a rectangular coordinate system is given by standard coordinates (ξ, η). There are equations to convert these standard coordinates to equatorial coordinates (α, δ). The coordinates measured on an image are not standard

coordinates but are image coordinates. Therefore image coordinates must first be converted to standard coordinates before the equatorial coordinates can be determined.

2.4.1 Standard Coordinates

In photographic astronomy, standard coordinates (ξ, η) are used. Standard coordinates form a rectangular coordinate system on a photographic plate or CCD array [28]. The coordinate ξ increases with increasing right ascension and the coordinate η increases with increasing declination and are tangential to increasing right ascension and declination at the center of the image [26].

Standard coordinates are not generally the same as the measured image coordinates (x, y) and a transformation is required. Plate constants are used to perform the transformation.

2.4.1.1 Plate Constants

To convert between standard coordinates and image coordinates, six values known as plate constants are required. The plate constants correct for displacement of the origin, image rotation, non-perpendicularity of the image axes, scale errors due to focal-length calibration and tilt error of the photographic plate or CCD array [26]. The transformation between standard coordinates (ξ, η) and image coordinates (x, y) is given by [26]

$$\xi - x = ax + by + c \tag{2.8}$$

$$\eta - y = dx + ey + f \tag{2.9}$$

where a, b, c, d, e and f are the plate constants.

The plate constants are determined from the standard coordinates and the image coordinates for N known stars in an image using the method of least squares [26].

$$a\sum_{i=1}^{N}x_i^2 + b\sum_{i=1}^{N}x_i y_i + c\sum_{i=1}^{N}x_i = \sum_{i=1}^{N}x_i(\xi_i - x_i) \tag{2.10}$$

$$a\sum_{i=1}^{N}x_i y_i + b\sum_{i=1}^{N}y_i^2 + c\sum_{i=1}^{N}y_i = \sum_{i=1}^{N}y_i(\xi_i - x_i) \tag{2.11}$$

$$a\sum_{i=1}^{N}x_i + b\sum_{i=1}^{N}y_i + cN = \sum_{i=1}^{N}(\xi_i - x_i) \tag{2.12}$$

$$d\sum_{i=1}^{N}x_i^2 + e\sum_{i=1}^{N}x_i y_i + f\sum_{i=1}^{N}x_i = \sum_{i=1}^{N}x_i(\eta_i - y_i) \tag{2.13}$$

$$d\sum_{i=1}^{N}x_i y_i + e\sum_{i=1}^{N}y_i^2 + f\sum_{i=1}^{N}y_i = \sum_{i=1}^{N}y_i(\eta_i - y_i) \tag{2.14}$$

$$d\sum_{i=1}^{N}x_i + e\sum_{i=1}^{N}y_i + fN = \sum_{i=1}^{N}(\eta_i - y_i) \tag{2.15}$$

The two sets of equations require a minimum of three stars in the image in order to determine the plate constants. It is due to these equations that there is the requirement of three stars in the image that must be matched to be able to calculate the attitude.

2.4.2 Conversion Between Standard Coordinates and Equatorial Coordinates

The standard coordinates must be transformed to the equatorial coordinates on the celestial sphere. The transformations from equatorial coordinates to standard coordinates are given by [26]

$$\xi = \frac{\cos\delta \sin(\alpha - \alpha^*)}{\sin\delta^* \sin\delta + \cos\delta^* \cos\delta \cos(\alpha - \alpha^*)} \tag{2.16}$$

$$\eta = \frac{\cos\delta^* \sin\delta - \sin\delta^* \cos\delta \cos(\alpha - \alpha^*)}{\sin\delta^* \sin\delta + \cos\delta^* \cos\delta \cos(\alpha - \alpha^*)} \tag{2.17}$$

where α^* is the right ascension of the center of the image and δ^* is the declination of the center of the image.

The inverse transformations are given by [26]

$$\tan(\alpha - \alpha^*) = \frac{\xi}{\cos\delta^* - \eta \sin\delta^*} \tag{2.18}$$

$$\tan\delta = \frac{\sin\delta^* + \eta \cos\delta^*}{\cos\delta^* - \eta \sin\delta^*} \cos(\alpha - \alpha^*). \tag{2.19}$$

It should be noted that the right ascension and declination of the image center must be known to apply the transformations.

2.4.3 Finding the Right Ascension and Declination of the Image Center

To determine the right ascension and declination of the image center, an iterative procedure can be used to calculate the values [26]. This iterative procedure is explained in detail here. To make these calculations easier, the image coordinates of the image center are set to $x = 0$ and $y = 0$. If necessary, the image coordinates of the stars must be changed to reflect the location change of the origin of the image. For example,

until this point, the algorithm assumes the origin of the image is in the upper left corner, not the center of the image.

The initial values of the equatorial coordinates of the image center (α^*, δ^*) are estimated by linear interpolation. The initial value of the right ascension is an average of two values. The first value is found by correlating the x coordinate of each matched star to its right ascension and using the method of linear least squares to determine the right ascension of the center x pixel. The second value is found by correlating the y coordinate of each matched star to its right ascension and using the method of linear least squares to determine the right ascension of the center y pixel. The same method is used to find the initial value of the declination by substituting declination for right ascension in the above description.

Beginning with the initial values of the equatorial coordinates of the image center, the following calculations are performed iteratively to determine the image center. The iterations continue until two successive right ascensions and declinations of the image center are within an accepted tolerance (for example, 1 arc second).

- Equations (2.16) and (2.17) are used to calculate the standard coordinates (ξ, η) for each matched star using the values of the equatorial coordinates of the image center.
- Once those have been calculated, the plate constants a, b, c, d, e and f are determined.
- The standard coordinates of the center of the image are

$$\xi = c \tag{2.20}$$

$$\eta = f \tag{2.21}$$

because $x = 0$ and $y = 0$ at the center of the image.

- From the standard coordinates of the image center, new right ascension and declination estimates of the image center (α^*, δ^*) are calculated from Equations (2.18) and (2.19).

Chapter 3

INTEGER MICROPROCESSOR IMPLEMENTATION

3.1 Introduction

The algorithm described in Chapter 2 was developed on a personal computer with a processor that supported floating-point calculations. The implementation aboard a rocket or micro-satellite would likely be an integer arithmetic microprocessor because of lower cost and power requirements and therefore, modification of the code was investigated to change the floating-point calculations to integer calculations. This can be accomplished using fixed-point arithmetic. Fixed-point arithmetic uses ordinary integer operations to do arithmetic with integers that have a whole part and a fractional part [29]. It is called fixed-point because the binary or decimal point is in a fixed location.

3.2 Basic Fixed-point Calculations

Fixed-point addition, subtraction, multiplication and division are defined and examples are given. There are two types of fixed-point notation: decimal and binary. The notation used for specifying a fixed-point number is M.N. If decimal notation is used, M is the number of digits before the decimal point and N is the number of digits after the decimal point. If binary notation is used, M is the number of bits before the decimal point and N is the number of bits after the decimal point. For example, in

binary fixed-point notation, 10.6 would mean that there are 10 bits of integer and 6 bits of fraction. The format chosen is binary notation and in subsequent discussion, *fixed-point* should be taken as *binary fixed-point*.

In most cases, two's complement representation [30] will be used for the fixed-point formats. The most significant bit of the number serves as the sign bit. If the sign bit is zero, the number is positive and if the sign bit is one, the number is negative. For an n bit number, the range of integer values is $-(2^{n-1})$ to $(2^{n-1}-1)$. When the two's complement of a number is taken, the number is negated. To take the two's complement of a number, a bit by bit complement is performed on each bit and then a 1 is added.

In this discussion, if a fixed-point format is given, assume that two's complement representation is used unless "unsigned" is specifically noted. For example 4.4 fixed-point should be assumed to be 4.4 two's complement fixed-point notation.

For an M.N fixed-point number, the equivalent floating-point values range from $-(2^{M-1}).(0)$ to $(2^{M-1}-1).(1-2^{-N})$. For example, for a 4.4 fixed-point number, the equivalent floating-point values range from –8.000 to 7.9375.

3.2.1 Addition

Addition is straightforward, for two M.N fixed-point numbers the result is an M.N fixed-point number. In addition, if the signs of the addends are the same then the sign of the result must be the same [30]. If not, there is an addition overflow and the result is not correct.

To add an integer to a M.N fixed-point number, the integer must first be converted to the same format as the M.N fixed-point number by shifting the integer N bits to the left.

Example 1: Adding two 4.4 fixed-point numbers with the result in the range of a 4.4 fixed-point number.

The first addend is 0011.0000 binary (3.0000 floating-point) and the second addend is 0001.1000 binary (1.5000 floating-point). The result is 0100.1000 binary (4.5000 floating-point).

Example 2: Adding two 4.4 fixed-point numbers with the result that is not in the range of a 4.4 fixed-point number.

The first addend is 0100.1100 binary (4.7500 floating-point) and the second addend is 0101.0000 binary (5.0000 floating-point). The result using two's complement notation is 1001.1100 binary (-6.2500 floating-point). Obviously the result is incorrect because the result is negative and these two numbers cannot be added to form a result in 4.4 fixed-point notation.

3.2.2 Subtraction

Subtraction is also straightforward, for two M.N fixed-point numbers the result is an M.N fixed-point number. Rather than subtracting the subtrahend from the minuend, the subtrahend is negated by taking its two's complement and adding it to the minuend using the normal rules for addition [30].

To subtract an integer from a M.N fixed-point number or to subtract a M.N fixed-point number from an integer, the integer must be converted to the fixed-point notation by shifting the integer N bits to the left. Then both numbers are in M.N fixed-point format and they are subtracted as described above.

Example: Subtracting a 4.4 fixed-point number from a 4.4 fixed-point number with the result in the range of a 4.4 fixed-point number.

The minuend is 0011.0000 binary (3.0000 floating-point) and the subtrahend is 0001.1000 binary (1.5000 floating-point). First the two's complement of 0001.1000 binary is taken and is 1110.1000 binary (-1.5000 floating-point). Then the negated subtrahend is added to the minuend and the result is 0001.1000 binary (1.5000 floating-point).

3.2.3 Multiplication

The bits in the product depend on the bits in the multiplier and the multiplicand not on where the decimal points are in the multiplier or the multiplicand [29]. There are a few simple rules to note [29]:

1. The maximum number of bits you can have in the product is the sum of the number of bits in the multiplier and the multiplicand.
2. The number of bits of fraction in the product is the sum of the number of bits of fraction in the multiplier and the number of bits of fraction in the multiplicand.
3. An integer is a fixed-point number with 0 bits of fraction.

When multiplying a M.N fixed-point number by an integer (n bit), the variable which holds of the multiplication must be M+N+n bits long in order to accommodate the number of bits that result from the multiplication. Denote this intermediate result as result1. Then the value can be assigned back into a M.N fixed-point variable which loses the top n bits compared to result1. To check for overflow, result1 should be checked to see that its value fits into the range of the M.N fixed-point number. If not, then there has been overflow.

When multiplying two M.N fixed-point numbers, the result has 2N bits of fraction so that the result must be shifted right by N bits. Then the result is an M.N fixed-point number. This shifting causes error in fixed-point multiplication. When multiplying two M.N fixed-point numbers together, the result must go into a 2M.2N fixed-point variable and denote this first intermediate result as result1. After the result

has been shifted to the right by N bits (result2), it can be assigned to a M.N fixed-point variable (final result) which means that the upper M bits are also lost. The overflow check is a check that result2 has a value in the range of the M.N fixed-point format, if not, then the final result will be incorrect.

Example 1: Multiplying a 4.4 fixed-point number by a 4-bit integer with the result in range of a 4.4 fixed-point number.

The 4.4 fixed-point multiplicand is 0011.1000 binary (3.5000 floating-point) and the integer multiplicand is 2. Result1 is 00000111.0000 binary (7.000 floating-point). The final result is 0111.0000 binary (7.0000 floating- point).

Example 2: Multiplying a 4.4 fixed-point number by a 4-bit integer with the result that is not in range of a 4.4 fixed-point number.

The 4.4 fixed-point multiplicand is 0011.0000 binary (3.0000 floating-point) and the integer multiplicand is 4. Result1 is 00001100.0000 binary (12.0000 floating-point). At this point a check can be done to see if result1 is in the range of –8.0000 to 7.9375 floating-point, which it is not. From this an overflow has been detected.

Example 3: Multiplying a 4.4 fixed-point number by a 4.4 fixed-point number with the result in range of a 4.4 fixed-point number.

The 4.4 fixed-point multiplicand is 0011.0000 binary (3.0000 floating-point) and other 4.4 fixed-point multiplicand is 0001.1000 binary (1.5000 floating-point). Result1 is 00000100.10000000 binary (4.5000 floating-point), and shift result1 by 4 bits to the right to obtain result2, 00000100.1000 binary (4.5000 floating-point). Result2 is then assigned back to a 4.4 fixed-point variable for the final result of 0100.1000b (4.5000 floating-point).

Example 4: Multiplying a 4.4 fixed-point number by a 4.4 fixed-point number with the result that is not in range of a 4.4 fixed-point number.

The 4.4 fixed-point multiplicand is 1101.0000 binary (-3.0000 floating-point) and other 4.4 fixed-point multiplicand is 0111.1000 binary (7.5000 floating-point). Result1 is 11101010.10000000 binary (-22.5000 floating-point), and result1 is shifted to the right by 4 bits to obtain result2, 111111101010.1000 binary (-22.5000 floating-point). At this point a check can be done to see if result2

is in the range of –8.000 to 7.9375 floating-point, which it is not. From this an overflow has been detected.

3.2.4 Division

In dividing an M.N fixed-point number by an integer, the result is an M.N fixed-point number. No overflow checking is required, as the result is always smaller than the dividend.

In division, to divide a fixed-point number by a fixed-point number, the dividend must be shifted to the left by the number of bits of fraction that are required in the result before division takes place. For an M.N fixed-point number divided by an M.N fixed-point number and for a M.N fixed-point result, shift the dividend by N bits to the left and then divide by the divisor. To shift the dividend to the left by N bits, the word length must be able to accommodate a N bit left shift, otherwise bits can be shifted off the left side. In order to shift the dividend by N bits to the left, the variable that holds this amount must be M+N+N bits. Then after the division, the intermediate result (result1) is still in a M+N+N bit long variable ([M+N].N). For the final result, result1 is assigned back to a M.N fixed-point variable. To check for overflow, the value of result1 should be in the range of the M.N fixed-point variable and if it is not, overflow has occurred.

Example 1: The dividend is a 4.4 fixed-point number and the divisor is an integer.

The dividend is 0101.0100 binary (5.2500 floating point) and the divisor is integer 9. The result is 0000.1001 binary (0.5625 floating point).

Note that with 4 bits of fraction, the answer is only an approximation to the true answer of 0.5833 floating point.

Example 2: The dividend and divisor are both 4.4 fixed-point numbers and the result is a 4.4 fixed-point number.

The 4.4 dividend is 0101.0010 binary (5.1250 floating point) and the divisor is 0010.000 binary (2.0000 floating point). First the dividend is shifted to the left by 4 bits in order to have a 4.4 fixed-point final result, the shifted dividend is 01010010.0000 binary. The shifted dividend is divided by the divisor and result1 is 00000010.1001 binary (2.5625 floating point). Result1 is then checked for overflow, which is not the case and the final result is 0010.1001 binary (2.5625 floating point).

Example 3: The dividend and divisor are both 4.4 fixed-point numbers and the result is not in range of a 4.4 fixed-point number.

The fixed-point dividend is 0101.0010 binary (5.1250 floating point) and the divisor is 0000.1000 binary (0.5000 floating point). The dividend is shifted to the left by 4 bits is 01010010.0000 binary. This shifted dividend is divided by the divisor and result1 is 00001010.0100 binary (10.2500 floating point). Result1 is not in the range of –8.000 to 7.9375 floating-point and overflow has occurred.

3.3 Databases

As with the floating-point version of the program, the sub-catalog of stars and the angular separation database are required. The only changes in these are that the floating-point values in them must be converted to the equivalent integer value which when interpreted as the appropriate fixed-point number the correct value is obtained. For an M.N fixed-point format, the floating-point number is multiplied by 2^N and is saved in an integer variable. The accuracy of the fixed-point number is $1/2^N$. For example, floating-point number 5.6250 and a fixed-point format of 4.4. By multiplying the floating-point number by 2^4, the integer value obtained is 90. When this integer value is interpreted as a 4.4 fixed-point number, the value is 5.6250 floating-point (0101.1010 binary). A second example is floating-point value 1.98 and a fixed-point

format of 4.4. The integer value obtained is 31 and when interpreted as a 4.4 fixed-point number, the value is 1.9375 floating-point (0001.1111 binary).

Other databases are required when considerations are made for implementation using integer arithmetic and they are discussed below.

3.3.1 Cosine and Arc Cosine Look Up Tables for Calculating Angular Separations

Cosine and arc cosine tables are possibly required for calculating the angular separations between stars in the image. Possibly required because depending on the FOV, Pythagora's theorem can yield angular separations which are closer to the floating-point values. This is discussed in detail in the following section titled "Calculating Angular Separations", Section 3.4.

The cosine table depends on the format of the fixed-point value of x (M1.N1 two's complement) and the format of the fixed-point value of $\cos(x)$ (M2.N2 unsigned). The choice for the format of x depends on the maximum camera angle. For example for a maximum camera angle of 30°, there would need to be 6 bits of integer for two's complement notation. Even though the angular separations are always positive, two's complement notation is required because overflow of calculations involving angular separations need to be checked for. The format of $\cos(x)$ is unsigned since $\cos(x)$ would only be positive for maximum camera angles less that 90° and if this were not the case, then the format would be changed. If the format of x is M1.N1, then x begins at 0.0 floating-point and increases in increments of $(1/2^{N1})$ floating-point until $x = 2^{(M1-1)} - 1/2^{N1}$ floating point. For each floating-point value of x, $\cos(x)$ is calculated and then converted to the fixed-point format of M2.N2 by multiplying the floating-point value of

cos(x) by (2^{N2}). For this look up table, the value of x is the index into the table and the value at the index is cos(x). There are $2^{(M1+N1-1)}$ entries in the table.

The arc cosine table depends on the format of the fixed-point value of y (M2.N2 unsigned) and the format of the fixed-point value of arc cos(y) (M1.N1 two's complement). The value of y begins at 0.0 floating point, increases in increments of ($1/2^{N2}$) floating-point until y = 1.0 floating point. For each value of floating-point value of y, arc cos(y) is calculated and converted to the fixed-point format of M1.N1 by multiplying the floating-point value of arc cos(y) by (2^{N1}). This converted value is checked to see if the value is less than the maximum value allowed by the M1.N1 format. If it is, the value is saved to the arc cosine look up table. There are not ($2^{(M2+N2-1)} + 1$) entries in this table as some values are discarded. The index into the table is (y - number of values of arc cos(y) that were discarded) and the value at the index is arc cos(y).

3.3.2 Arc Cosine Look Up Table for the Rotation Angle Test

An arc cosine look up table is required in order to perform the rotation angle test. The arc cosine table needs to provide values for the first quadrant, that is 0° to 90°. The arc cosine table depends on the format of the fixed-point value of y (M3.N3 unsigned) and the format of the fixed-point value of arc cos(y) (M4.N4 two's complement). The value of y begins at 0.0 floating point, increases in increments of ($1/2^{N3}$) floating-point until y = 1.0 floating point. For each value of floating-point value of y, arc cos(y) is calculated and converted to the fixed-point format of M4.N4 by multiplying the floating-point value of arc cos(y) by (2^{N4}). The value is saved to the arc

cosine look up table and there are $(2^{(M3+N3-1)} + 1)$ entries in this table. The index into the table is y and the value at the index is arc cos(y).

3.3.3 Enclosed Angle Database

The enclosed angle test requires the rearrangement of Equation (3.1) [26] to isolate 'A',

$$\cos a = \cos b \cos c + \sin b \sin c \cos A \tag{3.1}$$

where a is the angular separation between vertices B and C, b is the angular separation between vertices A and C, c is the angular separation between vertices A and B and A is the inner angle A (=∠BAC). Note here that A represents the inner angle and the vertex.

The difficulty in using Equation (3.1) is that cosine, arc cosine and sine look up tables are required and thus precision is lost. In order to maintain accurate values for the enclosed angle test, a database of enclosed angles is required. This is done by considering all the possible catalog triads that can be formed using the sub-catalog of stars. As long as the angular separations between any three catalog stars are less than the maximum camera angle, the triad is formed and the enclosed angles are calculated using Equation (3.1). The floating-point enclosed angles must be converted to the proper fixed-point format.

The enclosed angle database consists of a six column array: HR number of the first delimiting star, HR number of the second delimiting star, HR number of the third delimiting star, enclosed angle 1, enclosed angle 2 and enclosed angle 3.

Using the sub-catalog of the stars in the constellation Cassiopeia, an example enclosed angle database was created. Table 3.1 shows the database with the enclosed angles in floating-point notation. Assuming that the fixed-point notation for the enclosed angles is 10.6 (two's complement), Table 3.2 shows the database with the enclosed angles in the proper integer format. When the integers are interpreted as 10.6 fixed-point notation, the correct values are obtained.

Table 3.1 Enclosed angle database with the enclosed angles in floating-point notation for the constellation Cassiopeia.

HR number of the first delimiting star	HR number of the second delimiting star	HR number of the third delimiting star	enclosed angle 1 (degrees)	enclosed angle 2 (degrees)	enclosed angle 3 (degrees)
21	168	264	48.55	79.71	51.94
21	168	403	43.35	108.00	28.94
21	168	542	59.74	99.11	21.64
21	264	403	5.21	165.98	8.86
21	264	542	11.19	159.57	9.38
21	403	542	16.39	129.30	34.62
168	264	403	28.29	114.05	37.80
168	264	542	19.40	148.49	12.27
168	403	542	8.89	158.24	12.97
264	403	542	34.45	120.45	25.24

Table 3.2 Enclosed angle database with the enclosed angles as integers for 10.6 fixed-point notation for the constellation Cassiopeia.

HR number of the first delimiting star	HR number of the second delimiting star	HR number of the third delimiting star	enclosed angle 1 (degrees)	enclosed angle 2 (degrees)	enclosed angle 3 (degrees)
21	168	264	3107	5101	3323
21	168	403	2774	6911	1852
21	168	542	3823	6342	1385
21	264	403	333	10622	566
21	264	542	716	10212	600
21	403	542	1049	8275	2215
168	264	403	1810	7299	2419
168	264	542	1241	9503	784
168	403	542	568	10127	830
264	403	542	2204	7708	1615

3.4 Calculating Angular Separations

As stated previously, there are two methods used in the fixed-point algorithm for calculating the angular separations between image stars. Deciding which method to use depends on the accuracy of the look up tables. Recall that the formula to calculate the angular separation between two image stars is

$$\cos\theta = \cos\chi\cos\psi \tag{3.2}$$

where θ is the angular separation between the two stars, χ is the angular difference between the two stars in the x direction and ψ is the angular difference between the two stars in the y direction.

The first method uses Pythagora's theorem which is quite accurate for small angular separations. The second method, uses the Equation (3.2) and the look up tables. Pythagora's theorem was investigated because, in the image, the distances between the

stars appear to be a planar value. Therefore, Pythagora's theorem was looked into and found to perform well for small angular separations. Pythagora's theorem is given by

$$\theta_1 = \sqrt{\chi^2 + \psi^2} \tag{3.3}$$

where θ_1 is the angular separation using the first method. The Newton-Raphson formula is used to take the square root of a fixed-point number [31].

An example will be used to explain how it is decided which method will be used. Consider a FOV with a maximum camera angle of 30°. The fixed-point format used for angular separations is 6.10 (two's complement). Now using Equation (3.2), calculate floating-point angular separations using the same angular difference in the x and y direction, beginning with $\chi = \psi = 0.25°$, increasing in increments of 0.25° and ending with $\chi = \psi = 22.5°$. With $\chi = \psi = 22.5°$, the maximum camera angle is approximately 30°. Using Pythagora's theorem, calculate an approximation to the floating-point values and using the look up table method, calculate a second approximation to the floating-point values. The values in the cosine look up table are 1.15 fixed-point format (unsigned) and the values in the arc cosine look up table are 6.10 fixed-point format (two's complement). Calculate the difference between the angular separation using Equation (3.2) and using Pythagora's theorem and the difference between using Equation (3.2) and using the look up table method. These two differences are plotted in Figure 3.1. From this figure, it can be seen that the error in angular separations is smallest for the Pythagorean theorem method until an angular separation of 11.65°. After that, the look up table method yields a better approximation. By using the two methods, the error in angular separation calculated using fixed-point notation has a maximum error of 0.02° compared to floating-point calculations.

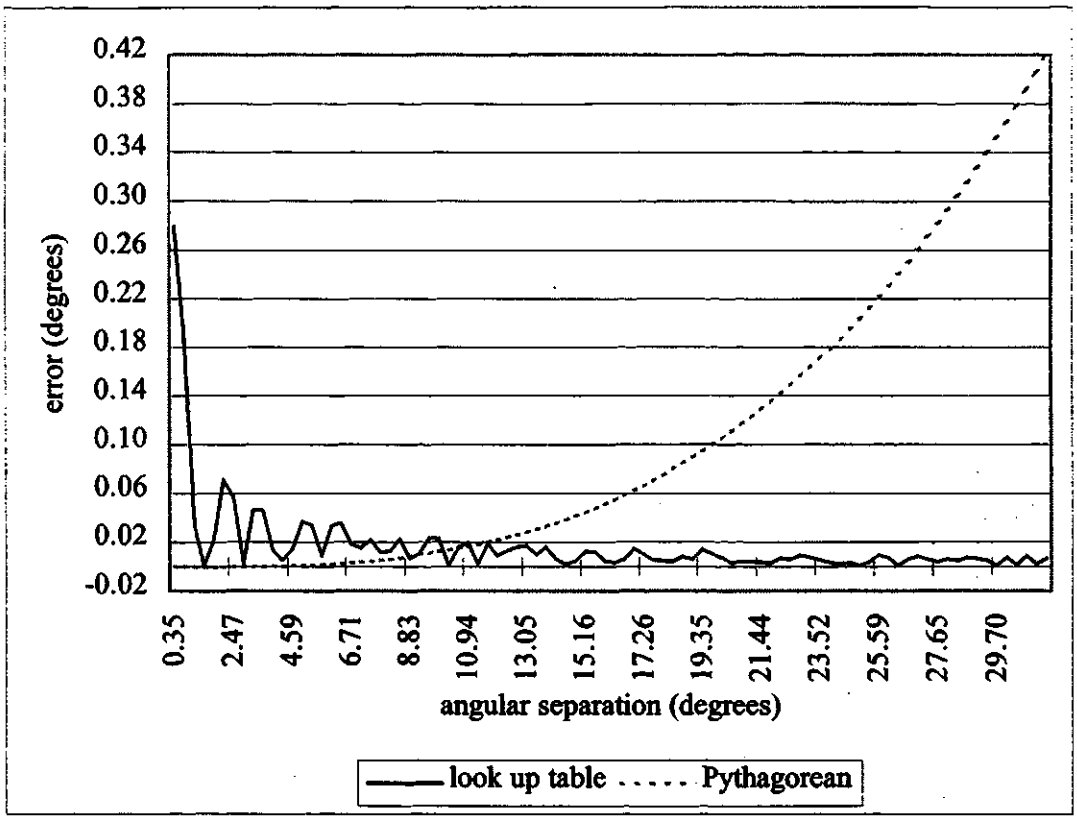

Figure 3.1 Comparison of Pythagora's theorem to the look up table method for calculating angular separation.

3.5 Determining the Attitude of the Image

The fixed-point implementation does not allow the attitude of the image to be determined, it can only determine the matches. With only 32 bits for the fixed-point version and a portion of those bits required for the fractional portion of a number, there would not be enough space to store the numbers required to calculate the plate constants.

Chapter 4

RESULTS AND DISCUSSION

4.1 Introduction

Results for two sets of images will be discussed. In the first set, there are three sample images obtained from Moorhouse [3]. The original images were taken with a Xybion intensified video camera (model ISS-250-GQR-3) and Moorhouse used them to verify his algorithm. The images used in this research were scanned in from the thesis and saved in Microsoft Windows bitmap form.

The results are first given in detail for the floating-point implementation. Then a comparison is made for the centroid and angular separation values between the floating-point and fixed-point implementation because the accuracy of these values have the most effect in the matching of the image stars. The execution times of the two implementations are compared.

The second set of images was obtained from a video recording of the OEDIPUS-C Sounding Rocket which was launched in November 6, 1995 from Poker Flat, Alaska [16]. This rocket was a joint Canadian Space Agency (CSA) and National Aeronautics and Space Administration (NASA) mission to study the natural and artificial waves in the ionospheric plasma as well as the dynamics of a spinning tethered space system. The camera used was a Xybion intensified video camera (model ISS-

250 low light level). The images from the camera were recorded at 30 frames per second on SVHS tape.

4.2 Vega, Deneb and Sadr Images

In each image, one star is identified by Moorhouse [3], as he had taken the images. The first image, Figure 4.1 contains the star Vega. The second image, Figure 4.2 contains the star Deneb. The third image, Figure 4.3 contains the star Sadr.

For the images, a sub-catalog of stars was created using the following parameters: the limiting magnitude was 6.00, the right ascension ranged from 0h to 24h and the declination ranged from 30° to 90°. The limiting magnitude was chosen to include the visible stars in the sky. The declination range was chosen based on knowing the declination of the brightest star in the image and for the three images, the declinations of the brightest star were above 30°. This resulted in a sub-catalog of 1266 stars. The database of angular separations was created using the sub-catalog of stars and a maximum angular separation of 12°. This value was chosen because the FOV of the camera was 9.0130° by 6.9049° [32]. There were 33,851 angular separations in the database.

For the Vega image, a detailed summary of the results will be given to show the steps of the attitude determination algorithm. For the Deneb and Sadr images, less detailed summaries will be given. The results were verified during the research with Star Compass Sky Maps [27].

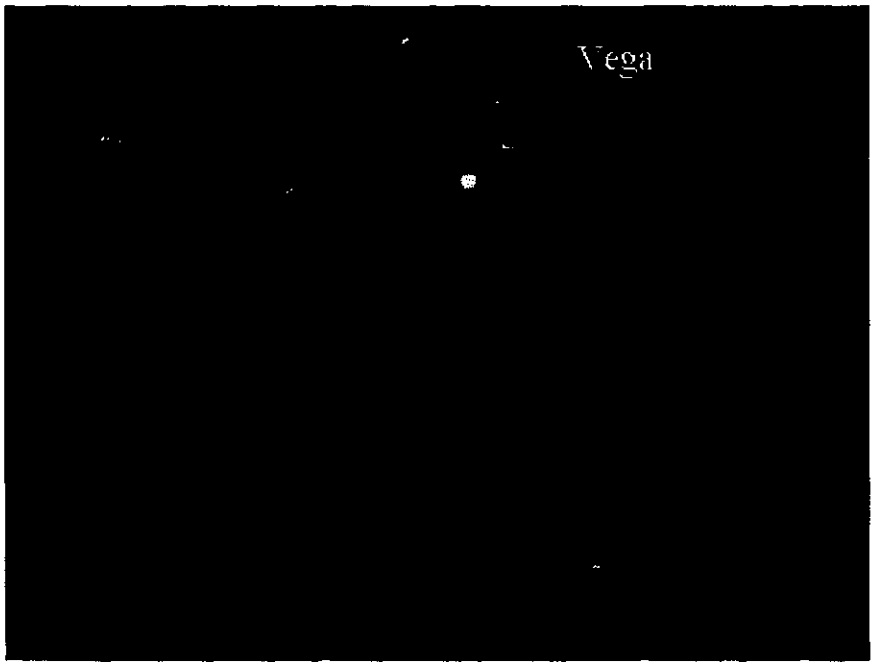

Figure 4.1 Vega image.

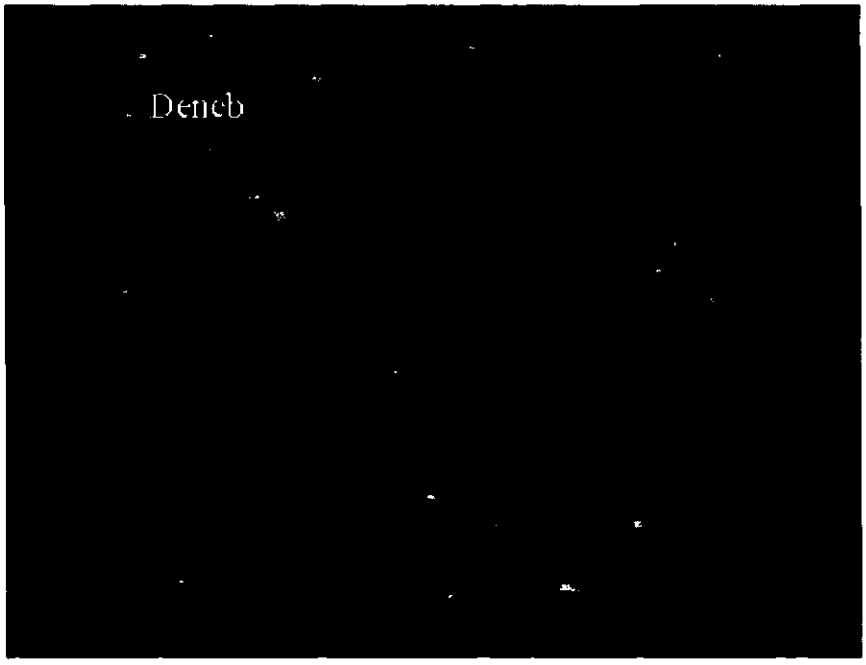

Figure 4.2 Deneb image.

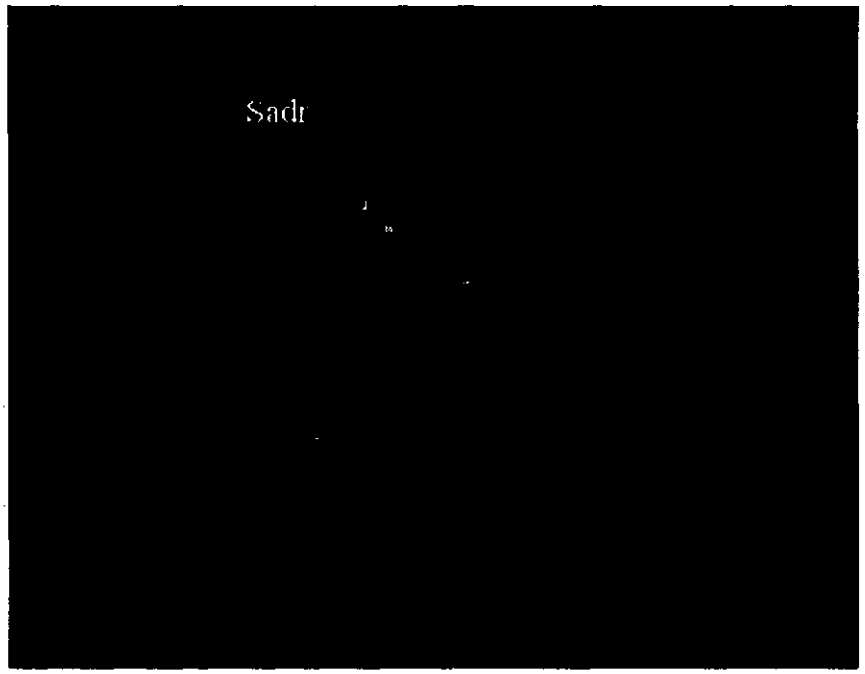

Figure 4.3 Sadr image.

4.2.1 Vega Results and Discussion

The image was stored in Microsoft Windows bitmap form. The algorithm read the image data into the image array. The Vega image is 427 pixels in width and 319 pixels in height. Each pixel has a grey scale intensity value varying from 0 to 255. Zero corresponds to black and 255 to white.

The image stars were found in the image as discussed in Section 2.2 and τ for clustering bright pixels together was set to 8. This was determined by experimentation, by starting at a small value of τ and increasing the value until all the pixels of the brightest star in the image were clustered into one star. The brightness threshold for pixels to be considered was set at 51. This was determined by looking at the intensity values of the dimmest pixels making up outer edges of stars. Only stars that are made up of a minimum number of 10 pixels were retained. There were five image stars that

passed the criteria and these were sorted according to apparent brightness as given by the number of pixels that made up the image star. These five image stars are labeled in Figure 4.4.

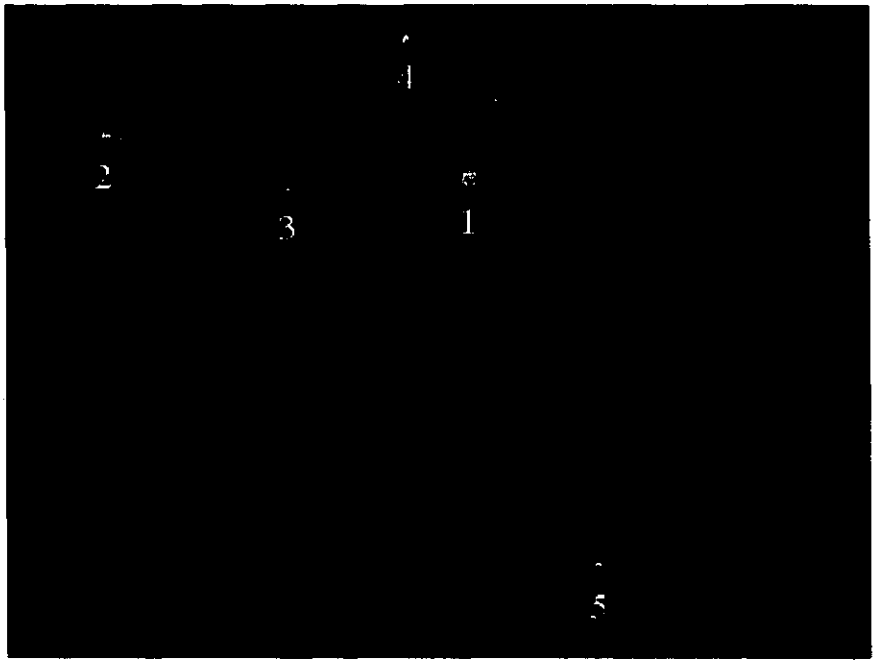

Figure 4.4 Vega: labeled image stars.

The angular separations between the image stars were calculated. For each calculated angular separation, a list of possible HR number pairs with matching angular separations within a tolerance was found. The tolerance was set to 0.15° to allow for distortion in the camera and error of the star centroids. Table 4.1 shows the calculated angular separations along with the number of catalog star pairs found.

Table 4.1 Vega: angular separations between image stars and the number of catalog star pairs found.

image star	image star	angular separation (degrees)	number of catalog star pairs
1	2	3.8045	526
1	3	1.8857	309
1	4	1.6315	291
1	5	4.2873	616
2	3	1.9725	305
2	4	3.2715	513
2	5	6.8250	1002
3	4	1.9924	321
3	5	5.1148	742
4	5	5.9112	878

The image stars were taken three at a time to form 10 unique triangles: 1-2-3, 1-2-4, 1-2-5, 1-3-4, 1-3-5, 1-4-5, 2-3-4, 2-3-5, 2-4-5 and 3-4-5. For each triangle, using the lists of HR number pairs, possible catalog triangles were formed and tested with the rotation angle test and enclosed angle test. The rotation angle tolerance was set to 5° and the enclosed angle tolerance was set to 1°. If a catalog triad passed both tests, it was grouped with other triads based on the average angle the image must be rotated to match the catalog triad. This was repeated for all possible catalog triads for all unique image triangles. The tolerance on this grouping was set to 5°. All successful catalog triangles for the Vega image are shown in Table 4.2. Recall that the places where the HR numbers reside in the list indicate which stars they were matched to in the image and a zero was put in the spaces that are not to be considered. The last two columns show the rotation angles needed to rotate the image vector to the catalog vector.

Table 4.2 Vega: successful catalog triads.

catalog triad number	possible HR number for : image star 1	image star 2	image star 3	image star 4	image star 5	rotation angle 1 (degrees)	rotation angle 2 (degrees)
	group 1	(average rotation angle = 195.8 degrees)					
1	3624	3391	3531	0	0	197.5	194.1
	group 2	(average rotation angle = 30.5 degrees)					
2	0	7131	7057	0	6872	31.4	34.7
3	0	7131	7056	0	6872	31.6	34.7
4	0	7139	7057	7054	0	31.8	29.1
5	0	7139	7057	7053	0	31.8	29.1
6	0	7139	7057	7051	0	31.8	29.5
7	0	7139	7056	7054	0	32.0	29.1
8	0	7139	7056	7053	0	32.0	29.1
9	0	7139	7056	7051	0	32.0	29.5
10	7001	7131	7056	0	0	30.0	28.2
11	7001	7131	7057	0	0	30.0	28.3
12	7001	7139	7056	0	0	29.9	28.2
13	7001	7139	7057	0	0	29.9	28.3
	group 3	(average rotation angle = 5.1 degrees)					
14	7262	7395	0	0	7192	4.1	6.2
	group 4	(average rotation angle = 272.6 degrees)					
15	8028	8023	0	0	8143	274.4	270.8
	group 5	(average rotation angle = 231.8 degrees)					
16	915	0	0	854	1046	229.3	234.3
	group 6	(average rotation angle = 36.9 degrees)					
17	0	0	7056	7053	6872	35.3	36.5
18	0	0	7056	7054	6872	35.3	36.5
19	0	0	7057	7054	6872	35.1	36.6
20	0	0	7057	7053	6872	35.1	36.6
21	0	0	7056	7051	6872	35.1	36.5
22	0	0	7057	7051	6872	34.9	36.6
23	7001	0	0	7053	6872	41.1	39.0
24	7001	0	0	7054	6872	41.1	39.0
	group 7	(average rotation angle = 257.1 degrees)					
25	0	9094	7	0	146	259.1	255.1
	group 8	(average rotation angle = 313.0 degrees)					
26	0	0	1122	1052	1135	312.0	314.1

There were eight groups of successful catalog triangles, each group with a different average rotation angle. Group 2 was the group with the largest number of entries, containing 12 entries. It can be noted that group 6 also contained the same set of HR numbers found in group 2 though the average rotation angles were different. The reason for this is that when the rotation angle test is performed, the catalog triangles are considered to be planar triangles rather than spherical triangles. This introduces an error in the rotation angles. The reason that the catalog triangles are treated as planar triangles is that for a microprocessor that does not support floating-point calculations, the calculations for planar triangles are simpler. If a microprocessor could support floating-point calculations then group 2 and group 6 would form one group when the catalog triangles are considered to be spherical. Whether or not planar triangles versus spherical triangles are used, the image stars are still matched to the same HR numbers. Another solution would be to increase the tolerance for grouping triangles together, for example 10 degrees.

By examining the results in group 2, something interesting can be noticed. That is, image stars 2, 3 and 4 are very likely multiple stars. If a close look is taken at Figure 4.4, image stars 2 and 4 are not very round but appear to be stars made up of more than one star. Image star 3 does not look like it is made up of more than one star.

The group with the largest number of entries is expected to most likely contain the correct matches for the image stars. Therefore, group 2 was chosen. For each image star, a list of the possible HR numbers is generated and is shown in Table 4.3.

Table 4.3 Vega: list of possible HR numbers for each image star.

image star	HR number	frequency of the HR number	visual magnitude of the HR number
1	7001	4	0.03
2	7139	8	4.30
	7131	4	5.58
3	7056	6	4.36
	7057	6	5.73
4	7051	2	5.06
	7053	2	5.14
	7054	2	5.37
5	6872	2	4.33

For each image star, the list of possible HR numbers was scanned to see if any possibilities could be combined to form another possibility because the stars are so close in angular separation that they would be imaged as one star. The value for this was calculated to be 0.1732°. This calculation is simply done by using Equation (2.4) along with $\tau = 8$. If two pixels are separated by less than or equal to 8 pixels, they would be clustered together as one star. First consider a separation of 8 pixels in the x direction and 0 pixels in the y direction. With these values calculate χ and substitute it into Equation (2.4). The result is an angular separation of 0.1689°. Then consider a separation of 0 pixels in the x direction and 8 pixels in the y direction. The angular separation is 0.1732°. The larger of these two values is taken.

For image star 2, HR 7139 and HR 7131 are separated by an angular separation of 0.1714°. A new distinct HR number was created to indicate that several catalog stars

were close enough to be imaged as one star by the camera. The HR number that was created is 17139. The last four digits of the new HR number indicate which catalog star was the brightest of the stars making up the new possibility. The right ascension and declination of the combined star were set to be the right ascension and declination of HR 7139. The frequency of the combined star was the sum of the frequency for HR 7139 and HR 7131.

HR number possibilities were also combined for image star 3. HR 7056 and HR 7057 are separated by an angular separation of 0.0121°. The same was done for image star 4. HR 7051 and HR 7053 are separated by an angular separation of 0.0578°. HR 7051 and HR 7054 are separated by an angular separation of 0.0578°. HR 7053 and HR 7054 are separated by an angular separation of 0.0000° (when the angular separation is given to 4 decimal places).

The new list of possible HR numbers is shown in Table 4.4 which includes the combined possibilities. The image stars are matched to the HR number with the highest frequency as long as that frequency is equal to or greater than a parameter called 'vote number'. This parameter exists so that the user can set a confidence level for matches. Here this parameter was set to 2. By examining Table 4.4, image star 1 was matched to HR 7001, image star 2 was matched to HR 17139, image star 3 was matched to HR 17056, image star 4 was matched to HR 17051 and image star 5 was matched to HR 6872. By checking the remarks file [33] of the Yale Bright Star Catalog, it can be confirmed that Vega is HR 7001.

Table 4.4 Vega: new list of possible HR numbers for each image star.

image star	HR number	frequency of the HR number	visual magnitude of the HR number
1	7001	4	0.03
2	7139	8	4.30
	7131	4	5.58
	17139	12	4.30
3	7056	6	4.36
	7057	6	5.73
	17056	12	4.36
4	7051	2	5.06
	7053	2	5.14
	7054	2	5.37
	17051	6	5.06
5	6872	2	4.33

A final check is performed to ensure that the angular separations between the matched stars agree within tolerance to the image stars. The equatorial coordinates of the image center are: right ascension of 18h 33m 49.7s (278.45703°) and declination of 37.24007° for an iteration tolerance of 1 arc second (0.00028°). While the error in the right ascension and declination of the image center cannot be directly determined, it can be determined indirectly using the identified stars in the image. Once the image center is determined, the right ascension and declination of any point on the image can be calculated. For a given *(x, y)* coordinate, the standard coordinates can be calculated using Equations (2.8) and (2.9). The plate constants used are the plate constants of the last iteration used to determine the image center. Equations (2.18) and (2.19) are used to calculate the right ascension and declination of the point on the image. Therefore,

using the *(x, y)* coordinates of the identified stars and calculating the right ascension and declination of the star, the calculated equatorial coordinates can be compared to the equatorial coordinates in the star catalog. Table 4.5 compares the calculated equatorial coordinates to the star catalog equatorial coordinates. The maximum error in right ascension and declination is approximately 0.02° or 1.2 arc minutes. Therefore the right ascension and declination of the image center are 18h 33m 49.7s ± 4.8s and 37.24° ± 0.02°.

Table 4.5 Vega: comparison of calculated and star catalog equatorial coordinates.

HR number	star catalog right ascension (degrees)	calculated right ascension (degrees)	difference right ascension (degrees)
7001	279.23415	279.23476	0.00061
17139	283.62542	283.61611	0.00931
17056	281.19292	281.21146	0.01853
17051	281.08458	281.07801	0.00657
6872	274.96499	274.96173	0.00326
HR number	star catalog declination (degrees)	calculated declination (degrees)	difference declination (degrees)
7001	38.78361	38.80490	0.02129
17139	36.89889	36.89968	0.00080
17056	37.60500	37.60544	0.00044
17051	39.67000	39.65362	0.01637
6872	36.06445	36.05834	0.00610

4.2.2 Deneb Results and Discussion

The Deneb image is 424 pixels in width and 320 pixels in height. All the parameters mentioned in discussing Vega's results are the same for Deneb unless otherwise mentioned. Nine stars were found in the image to be a minimum of 10 pixels in size and are labeled in Figure 4.5.

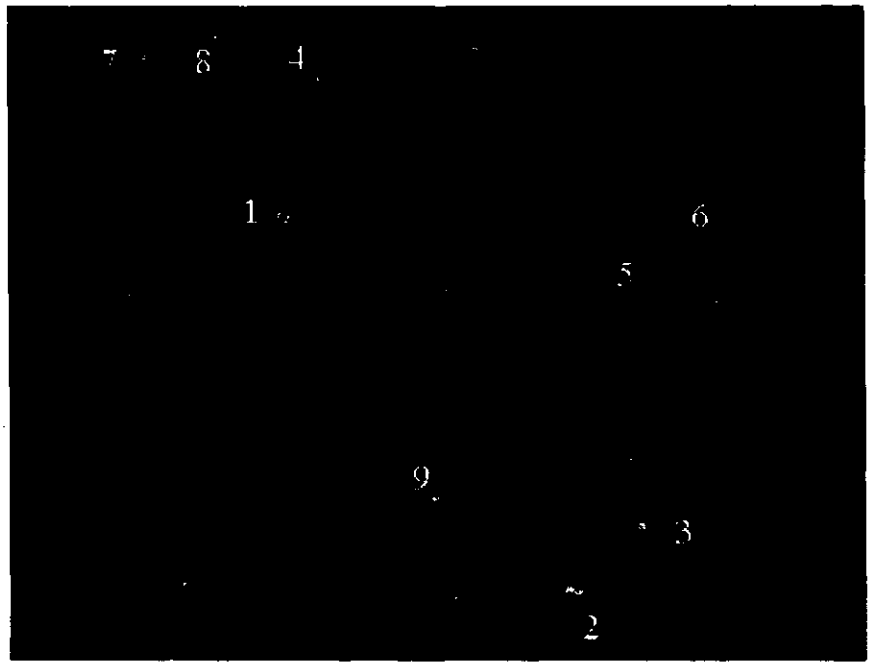

Figure 4.5 Deneb: labeled image stars.

The calculated angular separations and number of catalog star pairs found for each calculated angular separation within a tolerance of 0.15° are shown in Table 4.6.

Table 4.6 Deneb: angular separations between image stars and the number of catalog star pairs found.

image star	image star	angular separation (degrees)	number of catalog star pairs
1	2	4.9825	721
1	3	4.9943	725
1	4	1.5168	234
1	5	4.0603	592
1	6	4.2114	636
1	7	2.2311	341
1	8	2.0516	315
1	9	3.3728	530
2	3	0.9983	168
2	4	6.0231	895
2	5	3.4923	529
2	6	3.8077	531
2	7	7.2091	1065
2	8	6.9657	1034
2	9	1.7465	297
3	4	5.8163	848
3	5	2.6878	415
3	6	2.9814	467
3	7	7.2035	1058
3	8	6.8609	1006
3	9	2.2037	340
4	5	4.1650	616
4	6	4.2013	633
4	7	1.8496	302
4	8	1.1979	190
4	9	4.5963	686
5	6	0.3294	65
5	7	5.9189	878
5	8	5.3552	765
5	9	3.3962	532
6	7	5.9896	879
6	8	5.3976	767
6	9	3.7190	510
7	8	0.7602	121
7	9	5.5732	821
8	9	5.4115	771

The 9 image stars taken three at a time formed 84 unique triangles. The number of successful catalog triads which passed the rotation angle test and the enclosed angle test was 96. This is too much information to include here and only the relevant information will be given. Table 4.7 is a summary of the all the triad groups created, including the number of triads in each group and the average rotation angle for each group. There were a total of 36 groups and group 1 contained the largest number of triads. The 23 catalog triads in group 1 are shown in Table 4.8.

For each image star, a list of possible HR numbers was generated. For each image star, the list of possible HR numbers was scanned to see if any possibilities could be combined to form another possibility because the stars are so close in angular separation that they would be imaged as one star. The value for this was calculated to be 0.1726°. This calculation is simply done by using Equation (2.4) along with $\tau = 8$. If two pixels are separated by less than or equal to 8 pixels, they would be clustered together as one star. First consider a separation of 8 pixels in the x direction and 0 pixels in the y direction. With these values calculate χ and substitute it into Equation (2.4). The resulting angular separation is 0.1701°. Then consider a separation of 0 pixels in the x direction and 8 pixels in the y direction. The angular separation is 0.1726°. The larger of these two values is taken. For image star 2, HR 7735 and HR 7730 were combined to a new possibility as they are separated by 0.0936°. The possibilities are shown in Table 4.9.

Table 4.7 Deneb: summary of the catalog triad groups formed.

group number	number of triads	average rotation (degrees)
1	23	63.2
2	2	22.4
3	4	208.5
4	2	245.0
5	2	32.1
6	1	141.1
7	2	293.4
8	7	225.9
9	3	53.4
10	2	200.1
11	2	177.9
12	3	266.7
13	1	184.1
14	1	273.9
15	2	319.5
16	3	95.5
17	3	256.8
18	1	303.9
19	2	69.8
20	3	46.1
21	1	250.5
22	4	108.8
23	2	126.4
24	1	79.5
25	3	311.6
26	1	38.7
27	1	131.9
28	2	340.4
29	2	215.5
30	1	3.5
31	1	331.6
32	1	13.7
33	2	287.1
34	2	120.8
35	1	158.2
36	2	115.4

Table 4.8 Deneb: successful catalog triads in group 1.

catalog triad number	possible HR number for: image star 1	image star 2	image star 3	image star 4	image star 5	image star 6	image star 7	image star 8	image star 9	rotation angle 1 (degrees)	rotation angle 2 (degrees)
1	0	0	0	0	0	7851	8001	8003	0	60.7	60.4
2	0	0	0	0	7844	0	8001	8003	0	60.6	60.5
3	0	0	0	7977	0	0	8001	8003	0	65.3	62.6
4	0	0	0	7977	0	7851	0	8003	0	59.8	62.6
5	0	0	0	7977	7844	0	0	8003	0	60.1	62.6
6	0	0	7751	0	0	0	8001	8003	0	62.5	63.6
7	0	0	7751	7977	0	0	0	8003	0	64.7	63.6
8	0	0	9052	135	0	0	0	153	0	68.9	67.3
9	0	0	7751	7977	0	0	8001	0	0	64.7	62.5
10	0	7735	0	0	0	0	8001	8003	0	64.4	65.8
11	0	7735	0	7977	0	0	0	8003	0	67.5	65.8
12	0	7735	0	7977	0	0	8001	0	0	67.5	64.4
13	0	91	96	298	0	0	0	0	0	59.2	60.7
14	7924	0	0	0	0	0	8001	0	7798	66.1	67.4
15	7924	0	0	0	0	7851	8001	0	0	61.4	66.1
16	7924	0	7751	0	0	7851	0	0	0	60.9	61.4
17	7924	0	7751	0	7844	0	0	0	0	60.9	60.4
18	7924	7730	0	0	0	0	0	0	7798	64.1	67.4
19	7924	7735	0	0	0	0	8001	0	0	63.7	66.1
20	7924	7735	0	0	0	7851	0	0	0	63.7	61.4
21	7924	7735	0	0	7844	0	0	0	0	63.7	60.4
22	1078	936	0	0	1002	0	0	0	0	61.2	56.6
23	7924	7735	7751	0	0	0	0	0	0	63.7	60.9

Table 4.9 Deneb: list of possible HR numbers for each image star.

image star	HR number	frequency of the HR number	visual magnitude of the HR number
1	1078	1	5.81
	7924	9	1.25
2	936	1	2.12
	7730	1	4.83
	91	1	5.57
	7735	7	3.79
	17735	8	3.79
3	96	1	5.74
	9052	1	6.00
	7751	6	3.98
4	298	1	5.99
	135	1	5.93
	7977	7	4.84
5	1002	1	4.95
	7844	4	4.95
6	7851	5	5.44
7	8001	10	4.78
8	153	1	3.66
	8003	9	5.45
9	7798	2	5.58

Image star 1 was matched to HR 7924, image star 2 was matched to HR 17735, image star 3 was matched to HR 7751, image star 4 was matched to HR 7977, image star 5 was matched to HR 7844, image star 6 was matched to HR 7851, image star 7 was matched to HR 8001, image star 8 was matched to HR 8003 and image star 9 was

matched to HR 7798. The angular separations between these HR numbers agreed within tolerance of the angular separations between the image stars.

Table 4.10 Deneb: comparison of calculated and star catalog equatorial coordinates.

HR number	star catalog right ascension (degrees)	calculated right ascension (degrees)	difference right ascension (degrees)
7924	310.35750	310.37383	0.01633
17735	303.40751	303.38585	0.02166
7751	303.86750	303.87514	0.00763
7977	312.23416	312.21948	0.01468
7844	307.51416	307.52551	0.01135
7851	307.82790	307.82381	0.00409
8001	313.31125	313.31479	0.00353
8003	313.32707	313.32068	0.00639
7798	305.52209	305.53016	0.00807
HR number	star catalog declination (degrees)	calculated declination (degrees)	difference declination (degrees)
7924	45.28028	45.28283	0.00255
17735	46.74139	46.75783	0.01644
7751	47.71444	47.71561	0.00117
7977	46.11417	46.11785	0.00369
7844	48.95167	48.94919	0.00248
7851	49.22055	49.21985	0.00070
8001	44.38722	44.39854	0.01132
8003	45.18195	45.17177	0.01018
7798	45.79500	45.77316	0.02184

The equatorial coordinates of the image center are: right ascension of 20h 31m 37.7s (307.90724°) and declination of 46.43192° for an iteration tolerance of 1 arc second (0.00028°). In the same manner as described earlier for Vega, the error in the

image center is found indirectly. From Table 4.10 it can be seen that the maximum error in right ascension and declination is approximately 0.02°. The right ascension and declination of the image center are 20h 31m 37.7s ± 4.8s and 46.43° ± 0.02°.

4.2.3 Sadr Results and Discussion

The Sadr image is 469 pixels in width and 360 pixels in height. All the parameters mentioned in discussing Vega's results are the same for Sadr unless otherwise mentioned. Three stars were found in the image to be a minimum of 10 pixels in size and are labeled in Figure 4.6.

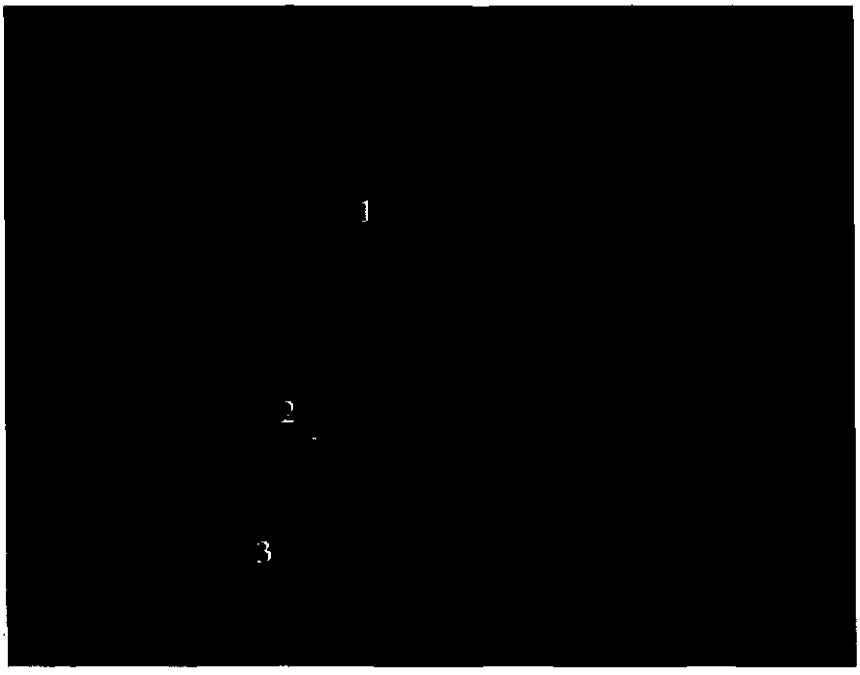

Figure 4.6 Sadr: labeled image stars.

The calculated angular separations and number of catalog star pairs found for each calculated angular separation within a tolerance of 0.15° are shown in Table 4.11.

Table 4.11 Sadr: angular separations between image stars and the number of catalog star pairs found.

image star	image star	angular separation (degrees)	number of catalog star pairs
1	2	2.3159	349
1	3	3.6805	521
2	3	1.3705	216

The 3 image stars taken three at a time formed 1 unique triangle. There were two successful triads which are shown in Table 4.12.

Table 4.12 Sadr: successful catalog triads.

catalog triad number	possible HR number for : image star 1	image star 2	image star 3	rotation angle 1 (degrees)	rotation angle 2 (degrees)
	group 1 (average rotation angle = 46.2 degrees)				
1	7796	7763	7736	46.2	46.3
	group 2 (average rotation angle = 177.2 degrees)				
2	8079	8053	8047	31.4	177.6

There were no matches because the frequency of any possible HR number is only 1 when there are only three image stars and the minimum requirement was 2 for a match. A possible addition to the algorithm would be to include special case of only 3 image stars. This special case would allow for a match if there was a unique catalog triangle found. This may seem attractive, but may not be a good solution because there is no other information to back up the possible match. In this case, the special clause

would not be used as two successful catalog triangles were found and it is impossible to say which is the correct match without a priori information.

With the parameters defined as stated for Vega, the algorithm was not able to match the 3 image stars, but as stated previously, Moorhouse [3] had identified the images. In the Yale Bright Star Catalog [25], Sadr is HR 7796. And from that information, image stars 2 and 3 can be checked for their matches which are HR 7763 and HR 7736 respectively. Therefore catalog triangle number 1 is the correct triangle but cannot be matched for reasons stated above.

Many trials were investigated in order to see what the parameters would need to be set to in order to match a minimum of three image stars. The parameters of interest are the minimum number of pixels that are required for a bright spot to be considered a star, the tolerance for matching angular separations, the rotation angle tolerance, the enclosed angle tolerance and the tolerance for grouping successful catalog triangles based on average rotation angle.

The first parameter to reduce is the minimum number of pixels that are required for a bright spot to be considered a star. By looking at intermediate information that was generated by the program, before screening the bright spots, 10 possible stars were found. The distribution was as follows: 3 bright spots made up of 1 pixel, 2 bright spots made up of 2 pixels, 1 bright spot made up of 7 pixels, 1 bright spot made up of 8 pixels, 2 bright spots made up of 10 pixels and 1 bright spot made up of 27 pixels. The bright spots made of one or two pixels should be disregarded because they could be noise or very dim stars. If bright spots of 7 pixels or more were considered, then there

would be 5 stars in the image to attempt to match. Therefore the parameter was reduced from 10 pixels to 7 pixels for a bright spot to be considered a star.

This second trial was not successful at matching three image stars. The 5 image stars are labeled in Figure 4.7. The successful catalog triads are shown in Table 4.13. From there it can be seen that group 1 was the group with the largest number of triads and contained the correct triads. But with a minimum frequency of 2 required for each HR number, once again no matches could be made. As the brightest star was identified by Moorhouse [3], image star 4 is HR number 7708 and image star 5 is HR 7759.

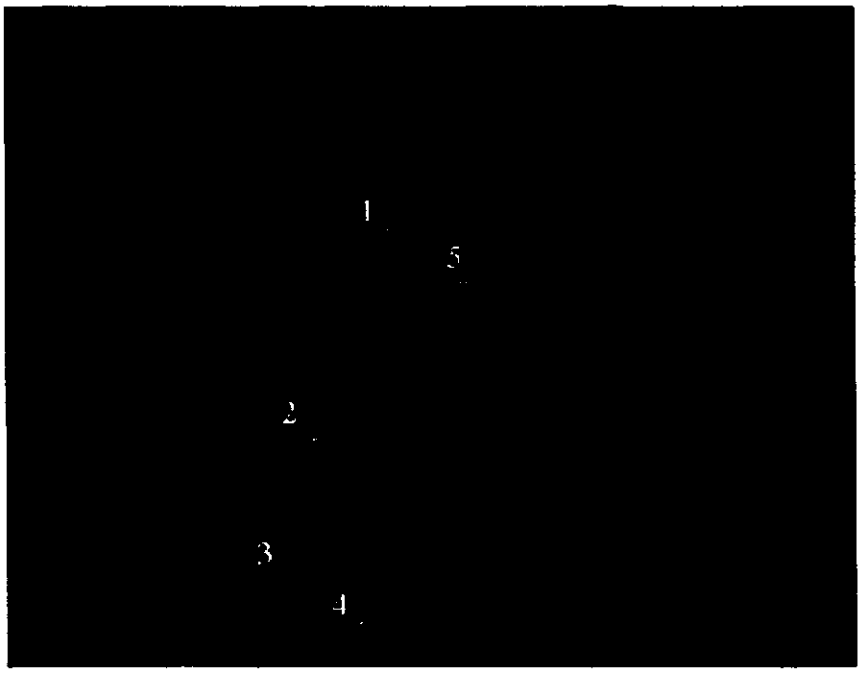

Figure 4.7 Sadr: labeled image stars with the minimum number of pixels reduced to 7.

Table 4.13 Sadr: successful catalog triads for trial 2.

catalog triad number	possible HR number for : image star 1	image star 2	image star 3	image star 4	image star 5	rotation angle 1 (degrees)	rotation angle 2 (degrees)
	group 1 (average rotation angle = 46.4 degrees)						
1	7796	7763	0	7708	0	46.2	46.9
2	7796	7763	7736	0	0	46.2	46.3
	group 2 (average rotation angle = 177.2 degrees)						
3	8079	8053	8047	0	0	176.8	177.6
	group 3 (average rotation angle = 234.4 degrees)						
4	8976	9003	0	9053	0	233.6	235.2
	group 4 (average rotation angle = 55.7 degrees)						
5	464	0	417	390	0	55.9	55.4
	group 5 (average rotation angle = 19.1 degrees)						
6	7800	0	0	7769	7767	18.7	19.6
	group 6 (average rotation angle = 61.9 degrees)						
7	0	153	124	96	0	64.3	59.4
	group 7 (average rotation angle = 251.9 degrees)						
8	0	4278	0	4333	4256	250.3	253.4

In the remaining trials, the minimum number of pixels for a bright spot to be considered a star is 7. For trials three and four, the tolerance for matching angular separations was increased from 0.15° to 0.20° and 0.25°. This obviously changed the overall number of successful catalog triangles, but did not change the results in the group with the most triads. Once again, this group contained only 2 entries, the same ones as catalog triangle numbers 1 and 2 in Table 4.13. The remaining groups contained one catalog triangle each. Therefore it was determined that the tolerance for

matching angular separations was not the parameter that needed to be altered and was set back to 0.15°.

The next parameter to adjust was the rotation angle tolerance. It was increased from 5° to 6°, 7°, 8°, 9° and 10°. When the rotation angle tolerance was set to 6°, the results were the same as Table 4.13. As it was increased to 7°, image stars 1, 2 and 4 were matched. For rotation angle tolerances of 8°, 9° and 10°, the five image stars were successfully matched.

The enclosed angle tolerance was also adjusted while the rotation angle tolerance was set at 5°. It was increased from 1° to 1.5°, 2.0° and 2.5°. With the enclosed angle tolerance at 1.5°, the results were the same as trial 2. With the enclosed angle tolerance at 2.0° and 2.5°, image stars 1, 2, 3 and 4 were matched.

In summary, to be able to match at least 3 stars in the Sadr image, the minimum number of pixels for a bright spot to be considered a star should be decreased to 7, the rotation angle tolerance should be increased to a minimum of 7° and the enclosed angle tolerance should be increased to a minimum of 2°.

4.2.4 Comparison of Some Results Using Floating-point Calculations versus Fixed-point Calculations

There are two implementations of the algorithm. The development platform was the Pentium II processor and the algorithm was initially designed using floating-point calculations. The processor aboard a rocket or micro-satellite would likely be an integer arithmetic microprocessor because of lower cost and power requirements and a second implementation was investigated using fixed-point notation. This fixed-point notation

implementation can be executed on the Pentium II processor and the Motorola MC68360 microprocessor. The MC68360 is a 32-bit integer microprocessor and the fixed-point implementation was developed based on 32 bits.

In this section, two important results are compared using the floating-point and fixed-point notations. The first being the centroid values and secondly, the angular separations. In order to successfully match the stars in the image, the angular separations using fixed-point notation must be accurately calculated. Therefore, the centroid and angular separation results using fixed-point notation are compared to the floating-point results.

The fixed-point notations need to be chosen in order to accommodate the maximum integer needed and enough accuracy in the fractional portion. Each format was chosen to be 16 or 32 bit integers.

The comparisons will be made for the images Vega, Deneb and Sadr. These comparisons are made between the floating-point and fixed-point implementations executed on the Pentium II processor.

4.2.4.1 Centroids

With a maximum image size of 512 pixels by 512 pixels, the fixed-point format for the centroids was chosen to be 10.22. The maximum integer centroid value range is 0 to 511 and therefore 9 bits of integer is required. A sign bit is added on in order to check for overflow in calculations to bring the total to 10 bits of integer. As stated each format was chosen to be 16 or 32 bits of integer. With 16 total bits of integer, the fractional accuracy is limited to $1/2^6$ (0.015625). This amount of fractional accuracy

may not be sufficient and therefore 22 bits of fraction was chosen. Table 4.14 compares the accuracy of the fixed-point centroids with the floating-point centroids for Vega, Table 4.15 for Deneb and Table 4.16 for Sadr. The zero point in the image was chosen to be the center of the image. While the accuracy to the centroids would normally not be quoted to 5 decimal places, it was in this case to show that the fixed-point centroids values are very close to the floating-point value of the centroids.

Table 4.14 Vega: comparison of centroid values.

image star	floating-point x centroid (pixels)	fixed-point x centroid (pixels)	difference x centroid (pixels)
1	16.13144	16.13140	0.00004
2	-162.81865	-162.81863	0.00002
3	-73.08759	-73.08756	0.00003
4	-15.73050	-15.73047	0.00003
5	79.27619	79.27616	0.00003
image star	floating-point y centroid (pixels)	fixed-point y centroid (pixels)	difference y centroid (pixels)
1	74.39691	74.39689	0.00002
2	95.40933	95.40930	0.00003
3	69.96350	69.96348	0.00002
4	143.07092	143.07091	0.00001
5	-113.87619	-113.87616	0.00003

Table 4.15 Deneb: comparison of centroid values.

image star	floating-point x centroid (pixels)	fixed-point x centroid (pixels)	difference x centroid (pixels)
1	-76.22261	-76.22258	0.00003
2	67.43734	67.43728	0.00006
3	101.86992	101.86988	0.00004
4	-57.83884	-57.83881	0.00003
5	112.81633	112.81629	0.00004
6	121.36667	121.36662	0.00005
7	-144.14078	-144.14077	0.00001
8	-109.93023	-109.93020	0.00003
9	-0.88889	-0.88887	0.00002
image star	floating-point y centroid (pixels)	fixed-point y centroid (pixels)	difference y centroid (pixels)
1	56.43993	56.43991	0.00002
2	-126.10052	-126.10048	0.00004
3	-94.63821	-94.63817	0.00004
4	124.35950	124.35949	0.00001
5	29.45918	29.45916	0.00002
6	42.18889	42.18887	0.00002
7	135.27670	135.27666	0.00004
8	145.53488	145.53486	0.00002
9	-81.14444	-81.14441	0.00003

Table 4.16 Sadr: comparison of centroid values.

image star	floating-point x centroid (pixels)	fixed-point x centroid (pixels)	difference x centroid (pixels)
1	-24.02893	-24.02894	0.00001
2	-64.56410	-64.56414	0.00004
3	-80.65517	-80.65519	0.00002
image star	floating-point y centroid (pixels)	fixed-point y centroid (pixels)	difference y centroid (pixels)
1	58.39256	58.39259	0.00003
2	-55.32051	-55.32050	0.00001
3	-124.93103	-124.93101	0.00002

4.2.4.2 Angular Separations

The FOV of the images allows for a maximum angular separation of 12°. It was chosen so that the angular separations in the fixed-point program would be allowed a maximum angular separation of 30°, and the fixed-point format required was 6.10. The method used to determine this is the same as the one used to determine the fixed-point format for the centroids. Table 4.17 compares the accuracy of the fixed-point angular separation with the floating-point angular separations for Vega, Table 4.18 for Deneb and Table 4.19 for Sadr. In Section 3.4, an example was given to decide the cut off point for calculating angular separations using Pythagora's theorem versus using the look up table method. The example given was for an angular separation fixed-point format of 6.10 and a maximum angular separation of 30°. The cut off value was determined to be 11.65°. Since all the angular separations in the images are less than 11.65°, Pythagora's theorem is used to calculate the angular separations in the fixed-

point version of the program. The differences appear in the third decimal place or higher. This is adequate for matching the angular separations within a tolerance of 0.15° as stated for this set of images.

Table 4.17 Vega: comparison of angular separations.

image star	image star	angular separation		
		floating-point (degrees)	fixed-point (degrees)	difference (degrees)
1	2	3.8045	3.8027	0.00176
1	3	1.8857	1.8848	0.00089
1	4	1.6315	1.6309	0.00066
1	5	4.2873	4.2861	0.00121
2	3	1.9725	1.9717	0.00078
2	4	3.2715	3.2715	0.00001
2	5	6.8250	6.8281	0.00317
3	4	1.9924	1.9912	0.00117
3	5	5.1148	5.1152	0.00042
4	5	5.9112	5.9102	0.00101

Table 4.18 Deneb: comparison of angular separations.

image star	image star	angular separation		
		floating-point (degrees)	fixed-point (degrees)	difference (degrees)
1	2	4.9825	4.9814	0.0011
1	3	4.9943	4.9941	0.0002
1	4	1.5168	1.5156	0.0011
1	5	4.0603	4.0586	0.0017
1	6	4.2114	4.2099	0.0015
1	7	2.2311	2.2295	0.0016
1	8	2.0516	2.0508	0.0008
1	9	3.3728	3.3721	0.0007
2	3	0.9983	0.9971	0.0012
2	4	6.0231	6.0234	0.0003
2	5	3.4923	3.4912	0.0012
2	6	3.8077	3.8066	0.0011
2	7	7.2091	7.2119	0.0028
2	8	6.9657	6.9678	0.0021
2	9	1.7465	1.7461	0.0004
3	4	5.8163	5.8174	0.0010
3	5	2.6878	2.6865	0.0013
3	6	2.9814	2.9805	0.0009
3	7	7.2035	7.2060	0.0025
3	8	6.8609	6.8633	0.0023
3	9	2.2037	2.2021	0.0015
4	5	4.1650	4.1641	0.0010
4	6	4.2013	4.2002	0.0011
4	7	1.8496	1.8486	0.0010
4	8	1.1979	1.1963	0.0016
4	9	4.5963	4.5947	0.0016
5	6	0.3294	0.3271	0.0022
5	7	5.9189	5.9189	0.0001
5	8	5.3552	5.3555	0.0002
5	9	3.3962	3.3955	0.0008
6	7	5.9896	5.9902	0.0006
6	8	5.3976	5.3974	0.0002
6	9	3.7190	3.7188	0.0003
7	8	0.7602	0.7588	0.0014
7	9	5.5732	5.5732	0.0001
8	9	5.4115	5.4111	0.0003

Table 4.19 Sadr: comparison of angular separations.

image star	image star	angular separation		
		floating-point (degrees)	fixed-point (degrees)	difference (degrees)
1	2	2.3159	2.3145	0.0015
1	3	3.6805	3.6797	0.0009
2	3	1.3705	1.3691	0.0014

4.2.4.3 General Discussion of Final Results with the Fixed-point Implementation

There are some differences in the number of generated triads between the floating-point algorithm and the fixed-point algorithm. This is due to the difference in centroid values and angular separations between the two algorithms and the possible difference in the values of the parameters. Obviously, the angular separation database for the fixed-point algorithm has small differences between the floating-point angular separations and the fixed-point angular separations.

As expected and as required, the matches between the catalog stars and the image stars are the same for the fixed-point and floating-point versions of the algorithm.

4.2.5 Comparison of Execution Times Between Programs on a Pentium II versus the MC68360

Two images were used to compare execution times. These images were Vega and Deneb.

The Pentium II was a 266 Mega Hertz (MHz) processor with 64 Mega Bytes (MB) of random access memory (RAM) and the operating system was Microsoft Windows NT 4.0. The compiler used was Microsoft Visual C++ 4.0 and 5.0. The integer microprocessor used was the Motorola MC68360 (model

#MC68EN360RC25C), with 32-bit arithmetic capability. The MC68360 development board was designed and built at the Linear Accelerator Laboratory at the University of Saskatchewan. The development board has available 4 MB of RAM to hold the program and provide heap space for the program execution. The compiler for the program run on the MC68360 was the GNU C compiler on a personal computer with the Linux operating system.

There are two main versions of the program: a floating-point version and a fixed-point version. There are two versions of the fixed-point code, denoted here by fixed-point version A and fixed-point version B. The floating-point version is run on the Pentium II processor. Fixed-point version A is run on the Pentium II processor and fixed-point version B is run on both the Pentium II processor and the Motorola MC68360 microprocessor.

Various execution times were determined. The floating-point version was run with and without the function used to determine the equatorial coordinates of the image center. This was done so that fixed-point version A of the program run on the Pentium II processor could be compared with the floating-point version without the calculation of the image center. Recall from Section 3.5 that the fixed-point version of the program cannot determine the attitude of the image. It is important to note that for the floating-point version of the program, the enclosed angles between stars from the star catalog are calculated as need, but for the fixed-point version of the program, a database is used. (This was discussed in Section 3.3.3). For the following parameters: limiting magnitude of 6.00, the right ascension ranging from 0h to 24h and the declination ranging from 30° to 90°, the number of enclosed angles was 1,175,901.

In order to execute the fixed-point program on the MC68360, the fixed-point version A of the code was slightly modified to fixed-point version B. There are several differences between the two versions, the first being that instead of being able to read in the databases required and the bitmap file from the hard drive, version B had the databases and image data initialized into arrays. Secondly, on the MC68360 development board, the maximum amount of RAM available is 4 MB and there is not enough space to store all the information about the enclosed angles (approximately 5 MB). The enclosed angle test helps to rule out many triads but is not necessarily needed for identification of the stars and is not used in fixed-point version B. Thirdly, with the amount of RAM available, instead of allowing there to be up to 12 stars in the image, the program was tailored for each image and therefore for only 5 image stars were allowed Vega and only 9 stars for Deneb. For Deneb, even with reducing the number of stars to 9, there was still not enough RAM available for the saving of the star pair lists. In order to reduce the amount of memory required, instead of using the HR numbers as the numbering for the catalog stars, the stars were renumbered. This allowed a parameter called 'STARS_IN_BSC' [34] to be reduced to the number of stars contained within the sub-catalog of stars from the number of stars in the Yale Bright Star Catalog. Once again, this number reduces the amount of RAM required for saving the star pair lists.

If a program required databases to be read in from a file, the time to do this was omitted from the execution times considered. The only output from the program was to display the matches of the image stars on the screen. Additionally, if the equatorial coordinates of the image center were determined, then they were also displayed. A

repetition loop of 100 times was chosen in order to determine execution time. One-hundred was chosen because it was long enough that any random processing time for uncontrollable events would not substantially affect the overall time measurement and short enough that execution time would not take too long to run. The length of time to run was not a factor of concern on the Pentium II but was for the MC68360. This process was repeated 10 times in order to obtain an average. In general, relative comparison times are more important that absolute comparisons.

In determining execution time on the Pentium II processor, another factor to consider is the version of the compiler used. The project was begun using Microsoft Visual C++ 4.0 and ended with version 5.0. As shown below, there are significant differences in execution times.

Table 4.20 shows the summary for the Vega images, for one instance of processing the image. Table 4.21 shows the summary for the Deneb images. For details about execution times of specific trials, see Appendix C.

Table 4.20 Vega: run-time.

program version	Pentium II Visual C++ 4.0 (seconds) +/- 0.001	Pentium II Visual C++ 5.0 (seconds) +/- 0.001	MC68360 (seconds) +/- 0.01
floating-point - with the image center	0.398	0.232	n/a
floating-point - without the image center	0.364	0.197	n/a
fixed-point version A	0.243	0.197	n/a
fixed-point version B	0.189	0.145	9.41

Table 4.21 Deneb: run-time.

program version	Pentium II Visual C++ 4.0 (seconds) +/- 0.001	Pentium II Visual C++ 5.0 (seconds) +/- 0.001	MC68360 (seconds) +/- 0.01
floating-point - with the image center	2.317	0.644	n/a
floating-point - without the image center	2.286	0.596	n/a
fixed-point version A	1.430	0.622	n/a
fixed-point version B	n/a	0.464	23.66

Before going into specifics for each image, there are several general comparisons that can be made. As expected, in comparing the floating-point execution time with and without finding the image center, there is a small amount of time required to determine the image center. Surprisingly, there is a large difference in time between compilation of the programs using Visual C++ 4.0 and 5.0. The improvement in execution time is greater for the floating-point version than the fixed-point version. Therefore, in comparing floating-point version without determining the image center and fixed-point version A, using Visual C++ 5.0, there is no longer a significant improvement in using integer calculations (fixed-point version) versus floating-point calculations. As expected, in comparing fixed-point version B executed on the Pentium II versus the MC68360, it takes quite a lot longer on the MC68360.

Comparing the execution time of the programs run on the Pentium II using Visual C++ 4.0 versus 5.0, the newer version has obviously gained significant improvement. For the floating-point versions of the program, for Vega, the improvement was a factor of 1.7 to 1.8. For Deneb, the improvement factor was 3.6 to

3.8. For the fixed-point versions of the program, for Vega, the improvement was a factor of 1.2 to 1.3. For Deneb, the improvement was 2.3. For Deneb, there was no comparison between fixed-point version B using the two versions of Visual C++, because the renumbering of the HR numbers was not done until version 5.0 had been installed and with its installation, version 4.0 was replaced and no longer available.

For Vega, fixed-point version B executed on the MC86360 took approximately 50 - 65 times longer than on the Pentium II depending whether Visual C++ 4.0 or 5.0 was used. For Deneb, fixed-point version B executed on the MC86360 took approximately 50 times longer than on the Pentium II with Visual C++ 5.0.

4.3 OEDIPUS-C Images

The second set of images was obtained from a video recording of the OEDIPUS-C Sounding Rocket which was launched in November 6, 1995 from Poker Flat, Alaska. Beattie et al. [16] identified stars in an image as belonging to two constellations, Cassiopeia and Cepheus. As the rocket spun, various stars from the two constellations came into and out of view.

The OEDIPUS-C tape was obtained from the Physics and Engineering Physics Department at the University of Saskatchewan. Eighty images were digitized from this tape. An example of an image is shown in Figure 4.8. It is image 6. Stars from Cassiopeia are on the right side and stars from Cepheus are on the left side. Note that the tether can be seen in the image. The video quality was not very good and it can be seen that the image is fairly noisy. A second image, Figure 4.9 is given to show that the rocket is spinning. This second image is image 7.

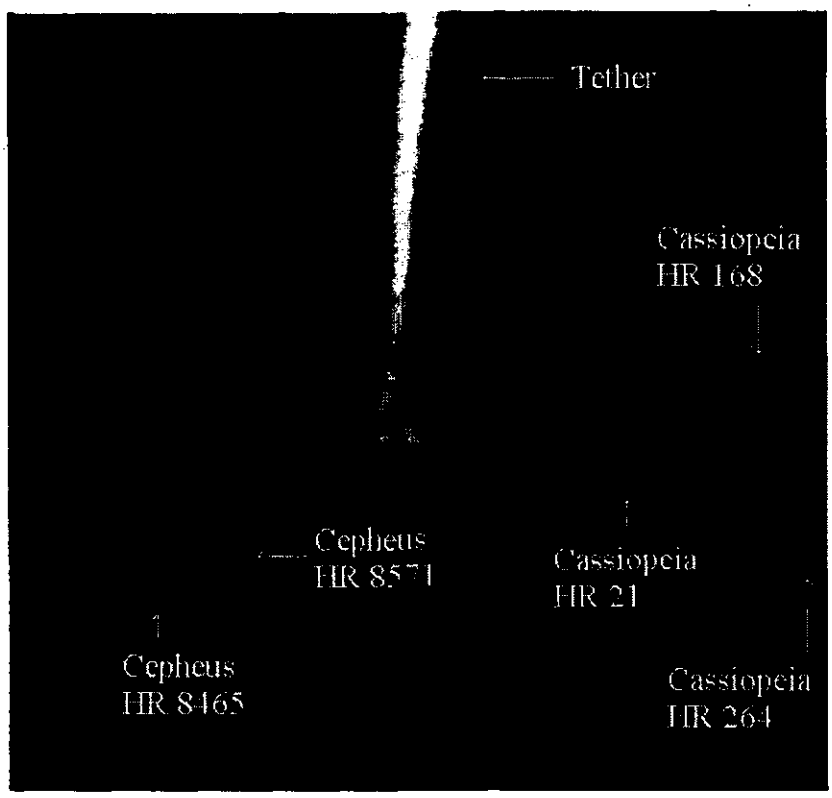

Figure 4.8 OEDIPUS-C image 6.

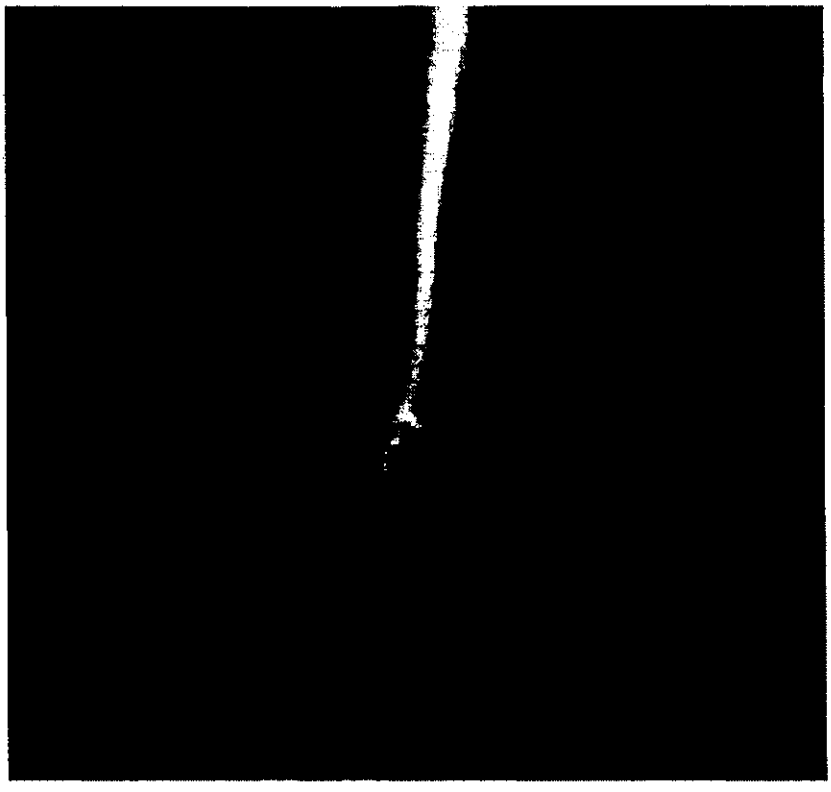

Figure 4.9 OEDIPUS-C image 7.

For the images, a sub-catalog of stars was created using the following parameters: the limiting magnitude was 5.00, the right ascension ranged from 0h to 24h and the declination ranged from 50° to 90°. This resulted in a sub-catalog of 175 stars. In analyzing the OEDIPUS-C data, Beattie et al. chose a limiting magnitude of 5.00 and the declination range of the analyzed data was above 50° [16]. The database of angular separations was created using the sub-catalog of stars and a maximum angular separation of 40°. This value was chosen because the FOV of the camera was 29.6716° by 23.5869°. There were 8,823 angular separations in the database. The FOV was not given explicitly in the paper, but enough information was given so that it could be calculated. This calculation is given in Appendix D.

For the images that were digitized, it was found that the FOV of 29.6716° by 23.5869° did not yield the correct angular separations between known stars in the image. The difference between angular separation of the image stars and the catalog stars were quite large, varying linearly with increasing angular separation. For OEDIPUS-C image 6, Figure 4.10 shows the linear relationship. The actual values used to determine the difference between the true and calculated angular separations are given in Appendix D.

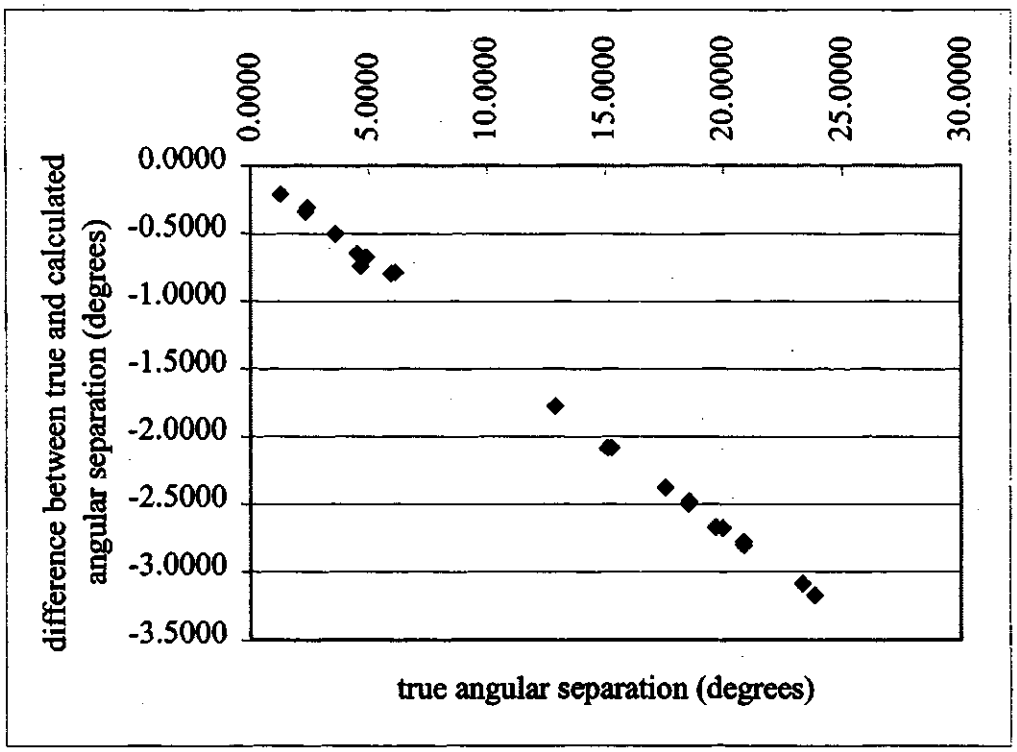

Figure 4.10 Difference between true and calculated angular separations for stars in OEDIPUS-C image 6.

The linear relationship showed that both the x FOV and y FOV were too large by some factor. It was hypothesized that when the images were digitized from the video tape that pixels were cropped off. A comparison was made between the array size given in by Beattie et al. (558.02 pixels by 557.08 pixels [16]) and the array size of the digitized images (512 pixels by 480 pixels). Using the ratio of digitized array size to array size given in the paper, a new FOV size was calculated to be 25.8356° by 20.3319°. The angular separations were calculated for stars in OEDIPUS-C image 6 and compared to the true angular separations and the differences were less than 0.34°.

The parameters in the program to determine attitude were set to the following values: $\tau = 15$, the brightness threshold for pixels to be considered = 100, only stars made up of a minimum of 10 pixels were retained, the tolerance for matching angular

separations = 0.35°, the maximum number of stars to match = 7 (this was reduced from 12 because by looking at the images, only about 7 out of each image were bright enough to be stars), rotation angle tolerance = 10°, enclosed angle tolerance = 5°, the tolerance for grouping successful triads together based on rotation angle = 10°, the value used to determine if two HR possibilities are close enough to be combined into one new possibility = 0.8693° and 'vote number' = 2. In general, the values are larger than what was used for the first set of images. The values were found through trial and error.

The first 25 images were analyzed because one revolution of the spinning payload was approximately 25 images. The tether was removed from the images using a drawing program. While the correct stars could be determined for many of the images, there were instances of incorrect matches or the program was unable to determine any matches. The final results are summarized in Appendix D. When correct matches were made, the right ascension and declination were calculated. These values of declination versus right ascension are plotted in Figure 4.11. These results agree with the results given by Beattie et al. [16] for the aft payload.

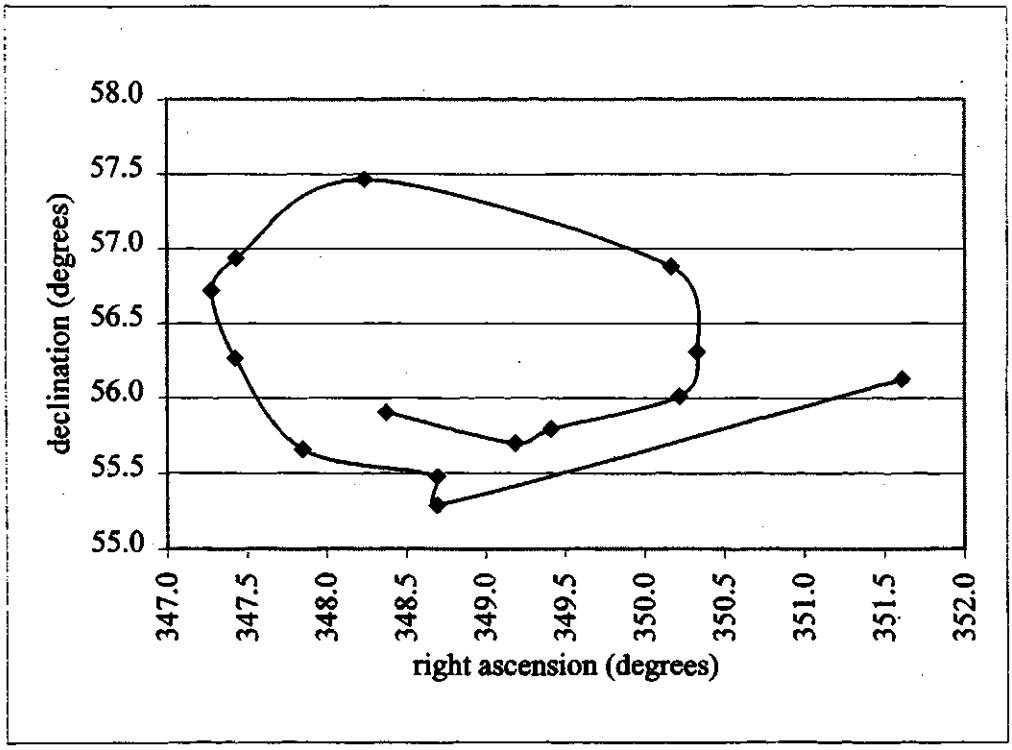

Figure 4.11 OEDIPUS-C images: declination versus right ascension.

Chapter 5

CONCLUSIONS

5.1 Conclusions

The method of clustering to find the stars in an image works well. It is efficient, as the image array must only be scanned once. The drawback is that τ for clustering bright spots together must be set to cluster large stars into one centroid and at the same time not too large as to incorrectly group two separate stars together. As it stands, τ is a static value and has been determined by trial and error. Several images should be used for each camera FOV to determine τ.

The algorithm for matching the star pattern uses the angular separation technique, the formation of triads and the use of rotation angles and enclosed angles. The algorithm is able to handle binary, triple or multiple stars rather than preprocessing the star catalog to combine the multiple stars into one star. The method has been tested on several images and correctly identifies the stars in the majority of images. Once the stars have been identified, the pointing direction of the image can be determined using the coordinates of the image stars and their matched celestial coordinates. If the stars were correctly identified, then the pointing direction was arrived at through a few iterations.

Considerations were made to convert the floating-point version of the code to a fixed-point version because it is likely that an integer microprocessor would be used in

attitude system flown on a rocket or micro-satellite. With a 32-bit integer processor, this can be accomplished with comparable accuracy to the floating-point results.

Execution times were measured to compare the floating-point and fixed-point versions of the program. With the Pentium II processor and Microsoft Visual C++ 5.0, there appears to be little difference in execution time and no significant advantage is gained by using integer versus floating-point calculations. Execution time was also compared for a fixed-point version of the program run on the Pentium II and an integer microprocessor. The Motorola MC68360 was used as a test integer microprocessor for the fixed-point code. Execution times are 50 – 65 times slower compared to the Pentium II processor. Different microprocessors should be investigated to decrease execution times. Also, if the enclosed angle test is to be performed, more RAM is required. For example, 16 MB would be sufficient for the databases created for Vega and Deneb.

Note that the software is avaliable in a separate report titled 'Software Report for Attitude Determination Using Stellar Images' [34].

5.2 Recommendations for Future Work

From the development and tests of the attitude determination algorithm using several images, recommendations for further work are suggested:

- There are several parameters that need to be set in the algorithm. Trial and error were used determine many of these parameters. A drawback of having many parameters to satisfy is that they must all be set so that correct recognition is likely. Values that are too small may exclude correct matches

while values that are too large may allow incorrect matches to overshadow the correct matches.

- An adaptable τ should be considered in finding stars in the image. As the number of pixels for a centroid increased, τ could also increase. If a static τ is used, the algorithm should check if any centroids are close enough in the image to be combined into one.
- Each catalog triad is subjected to two tests before being considered a possibility, the rotation angle test and the enclosed angle test. The rotation angle test is an important part of the algorithm. The enclosed angle test should be reconsidered and possibly removed. Firstly, since the database for the fixed-point program is very large, there needs to be sufficient memory to store this database. Secondly, the enclosed angles are calculated using the angular lengths between the stars and since all the angular lengths would agree within a tolerance, it seems that this test does not exclude that many possibilities.
- If only three stars are found in an image and a unique catalog match is found, the current algorithm cannot match the stars because each catalog star must appear a minimum of 'vote number' of times. It may be desired, in this specific case, to ignore the minimum frequency and match the stars. At the same time, it may be undesirable as reliance of other star matches is unavailable and incorrect matching can occur.

- The minimum frequency that a successful catalog match may occur with is 'vote number'. Currently, this value is static and has been kept relatively low. This value could be a function of the number of image stars.
- Only the groups of triads having the largest number of triads are considered to contain the correct matches for the stars. It is possible that the correct matches are not in the triad group(s) with the largest number of triads. Therefore, if the algorithm is unable to match image stars using the groups with the largest number of triads, the algorithm could continue checking other groups in order of decreasing number of triads.
- The developed algorithm is capable of to recognizing binary and other multiple stars when they are possibilities for a single image star. If a multiple star is found, a new star entry is created whose frequency is the sum of the catalog stars making it up. This may not be the best method of handling this case because the confidence level may appear larger than it actually is. A better method would be to reprocess the triads to remove the multiple star references. Also, preprocessing the catalog may be done so that multiple stars are removed and formed into a single star.
- In order to determine attitude, a minimum of three image stars must be matched. In the developed method, all matched image stars are used to determine attitude. Another method used by Beattie et al. [16] is to use only three matched stars at a time to determine attitude and to then average all of the attitudes. This method may be examined to see if the attitude is calculated to more accuracy.

- There are two main versions of the code, a floating-point version and a fixed-point version. Each version is separate from the other. This has made making changes longer than necessary. In some cases, the changes in code are exactly the same, and in other cases they are similar, but the fixed-point code may require extra considerations. In the cases where the code is exactly the same, it takes double the time. Therefore only one program should exist with special statements in the code to indicate which version should be used and which statements belong to each version.
- While a 32-bit integer microprocessor is adequate for achieving accuracies for centroid values and angular separation values compared to the floating-point values, it is recommended that a 64-bit integer microprocessor be used. One example of why this is recommended is that the centroid values require 32 bits and when adding a new pixel to a centroid, two 32-bit integers need to be multiplied resulting in a 64-bit integer as an intermediate step. With a 32-bit microprocessor, this is not possible and the calculation had to be split up into smaller chunks in order to not violate the maximum 32 bit integer restriction.
- As stated before, on the MC68360 development board, 16 RAM would have been useful.

REFERENCES

[1] Wertz, J. R., "Introduction," in *Spacecraft Attitude Determination and Control*, Editor Wertz, J. R., D. Reidel, Boston, Massachusetts, U.S.A., pp. 1-3, 1978.

[2] Liebe, C. C., "Pattern Recognition of Star Constellations for Spacecraft Applications," *IEEE Aerospace and Electronic Systems Magazine*, Vol. 7, pp. 34-41, June 1992.

[3] Moorhouse, P. G., "An Attitude Video Camera Using Stellar Images For the Gemini Experiment," Master of Science Thesis, College of Graduate Studies and Research, Department of Physics and Engineering Physics, University of Saskatchewan, Saskatoon, Saskatchewan, Canada, August 1994.

[4] Fortescue, P. and Stark, J., *Spacecraft System Engineering*, John Wiley and Sons Limited, Sussex, England, 1991.

[5] Fallon, L. (III), "Star Sensors," in *Spacecraft Attitude Determination and Control*, Editor Wertz, J. R., D. Reidel, Boston, Massachusetts, U.S.A., pp. 184-195, 1978.

[6] Rappaport, B., Dunning, T., Jordan, J., Phillips, K. and Stanton, R., "Autonomous Star Identification for Spacecraft Attitude Control," in *Proceedings of the Conference in Astronomy from Large Databases: Scientific Objectives and Methodological Approaches*, pp. 239-244, October 1987.

[7] Quine, B. and Durrant-Whyte, H. F., "Rapid Star Pattern Identification," in *Proceedings of SPIE – The International Society for Optical Engineering*, Vol. 2739, pp. 351-360, 1996.

[8] Murtagh, F., "A New Approach to Point-Pattern Matching," *Astronomical Society of the Pacific*, Vol. 104, pp. 301-304, April 1992.

[9] Scholl, M. S., "Experimental Demonstration of a Star-Field Identification Algorithm," *Optics Letters*, Vol. 18, pp. 402-404, March 15, 1993.

[10] Scholl, M. S., "Star-Field Identification for Autonomous Attitude Determination," *Journal of Guidance, Control, and Dynamics*, Vol. 18, pp. 61-65, January-February 1995.

[11] Ketchum, E. A. and Tolson, R. H., "Onboard Star Identification Without A Priori Attitude Information," *Journal of Guidance, Control, and Dynamics*, Vol. 18, pp. 242-246, March-April 1995.

[12] Mattei, J. A., "Variable Stars," in *Observer's Handbook 1995*, Editor Bishop, R. L., University of Toronto Press Incorporated, Toronto, Ontario, Canada, pp. 202-206, 1995.

[13] Gottlieb, D. M., "Star Identification Techniques," in *Spacecraft Attitude Determination and Control*, Editor Wertz, J. R., D. Reidel, Boston, Massachusetts, U.S.A., pp. 259-266, 1978.

[14] Sasaki, T. and Kosaka, M., "A Star Identification Method for Satellite Attitude Determination Using Star Sensors," in *Proceedings of the Fifteenth International Symposium on Space Technology and Science*, pp. 1125-1130, May 1986.

[15] Preist, B., "Star Tracker Stellar Compass (STSC)," 1995. Available from the WWW at http://www.llnl.gov/sensor_technology/STR45.html.

[16] Beattie, D., Krabel, E. and Whitehead, B., "Attitude Data Reduction for the Oedipus-C Sounding Rocket Using Stars," Bristol Aerospace Limited, Winnipeg, Manitoba, Canada, 1996.

[17] Padgett, C. and Kreutz-Delgado, K., "A Grid Algorithm for Autonomous Star Identification," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 33, pp. 202-213, January 1997.

[18] Junkins, J. L., White, C. C. (III) and Turner, J. D., "Star Pattern Recognition for Real Time Attitude Determination," *The Journal of the Astronautical Sciences*, Vol. 25, pp. 251-270, July-September 1997.

[19] Strikwerda, T. E. and Fisher, H. L., "A CCD Star Camera Used For Satellite Attitude Determination," in *Proceedings of the 1988 Summer Computer Simulation Conference*, Edited by Barnett, C. C. and Holmes, W. M., pp. 241-246, 1988.

[20] Strikwerda, T. E., Fisher, H. L., Kilgus C. C., and Frank, L. J., "Autonomous Star Identification and Spacecraft Attitude Determination with CCD Star Trackers," in *Proceedings of the First ESA International Conference on 'Spacecraft Guidance, Navigation and Control Systems,'* pp. 195-200, 1991.

[21] Fisher, H. L., Strikwerda, T. E., Kilgus, C. C., Frank, L. J. and Shuster, M. D., "Autonomous, All-Stellar Attitude Determination Experiment: Ground Test Results," *Advances in the Astronautical Sciences*, Vol. 74, pp. 69-84, 1991.

[22] Kosik, J. C, "Star Pattern Identification Aboard an Inertially Stabilized Spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 14, pp. 230-235, 1991.

[23] Kordas, J. F., Lawrence Livermore National Laboratory, University of California, Livermore, California, U.S.A., December 2, 1996, Private Communication to Bolton, R. J.

[24] Meisel, W. S., *Computer-Oriented Approaches to Pattern Recognition*, Academic Press, New York, New York, U.S.A., 1972.

[25] Hoffleit, D. and Jaschek, C., The Bright Star Catalogue, 4th revised edition, Yale University Observatory, New Haven, Connecticut, U.S.A., 1982.

[26] Green, R. M., *Spherical Astronomy*, Cambridge University Press, New York, New York, U.S.A., 1985.

[27] Bolton, R. J., Star Compass Sky Map, Star Compass Sky Map Software Package, Saskatoon, Saskatchewan, Canada, 1998.

[28] Taff, L. G., *Computational Spherical Astronomy*, John Wiley and Sons, Toronto, Ontario, Canada, 1981.

[29] Pendleton, B., "Doing It Fast," 1993. Available from the WWW at http://www.gameprogrammer.com/4-fixed.html.

[30] Wakerly, J. F., *Digital Design Principal and Practices Second Edition*, Prentice Hall, Englewood Cliffs, New Jersey, U.S.A., 1994.

[31] Pendleton, B., Fixed.cpp, 1993. Available from the WWW at http://www.gameprogrammer.com/4-fixed.html.

[32] Moorhouse, P. and Llewellyn, E. J., "Characterization, Calibration and Archiving Software for the GEMINI Attitude Video Camera: Final Report to the Canadian Space Agency," Institute of Space and Atmospheric Studies, Department of Physics and Engineering Physics, University of Saskatchewan, Saskatoon, Saskatchewan, Canada, March 31, 1994.

[33] Warren, W. H. Jr., The Bright Star Catalogue, 4th revised edition, Remarks File, National Space Science Data Center (NSSDC)/World Data Center A for Rockets and Satellites (WDC-A-R&S), National Aeronautics and Space Administration, Goddard Space Flight Center, Greenbelt, Maryland, U.S.A., May 1982.

[34] Yee, T. K., "Software Report for Attitude Determination Using Stellar Images," Department of Electrical Engineering, University of Saskatchewan, Saskatoon, Saskatchewan, Canada, March 1999.

[35] Menzel, D. H., *A Field Guide to the Stars and Planets*, Houghton Mifflin, Boston, Massachusetts, U.S.A., 1964.

[36] Wertz, J. R., "Coordinate Systems," in *Spacecraft Attitude Determination and Control*, Editor Wertz, J. R., D. Reidel, Boston, Massachusetts, U.S.A., pp. 24-31, 1978.

[37] Wertz, J. R., "The Spacecraft-Centered Celestial Sphere," in *Spacecraft Attitude Determination and Control*, Editor Wertz, J. R., D. Reidel, Boston, Massachusetts, U.S.A., pp. 22-24, 1978.

Appendix A

ASTRONOMY AND SPHERICAL ASTRONOMY

A.1 Visual Magnitudes of Stars

Star intensity is given as a magnitude value. It is important to note that the magnitude of a star is inversely proportional to its intensity. For example a 5^{th} magnitude star is dimmer than a 4^{th} magnitude star.

More specifically, each unit of magnitude signifies a difference in brightness of 2.512 times [35]. A star of magnitude 1 is 2.512 times brighter as a star of magnitude 2. A star of magnitude 1 is 100 (=2.512 × 2.512× 2.512 × 2.512 × 2.512) times brighter than a star of magnitude 6. There are stars with negative magnitudes which are brighter than stars with a positive magnitudes. The faintest stars that can be seen with the naked eye are magnitude 6 stars [26].

A.2 Equatorial Coordinates

The equatorial coordinates of a star on the celestial sphere are given by its right ascension and declination in the equatorial coordinate system [26]. This is analogous to a position on the earth being given by its latitude and longitude.

The right ascension is the azimuth measured along the celestial equator to the east from the vernal equinox [26]. The vernal equinox is a fixed point on the celestial equator and corresponds to the sun's position at the northern vernal equinox when the sun crosses the celestial equator from south to north. It is given the following symbol, γ. The most common units of right ascension are hours (h), minutes (m) and seconds (s)

of time and ranges from 0 to 24 hours. Right ascension can also be given in degrees (°) and ranges from 0 to 360°. Note that 1 hour = 15°, 1 minute = $1/60^{th}$ hour = 0.25° and 1 second = $1/60^{th}$ minute = 0.04166°. Minutes and seconds of right ascension are not equivalent to minutes and seconds of arc, even along the equator [36].

Declination is the elevation measured from the celestial equator [36]. The units of declination are degrees, minutes and seconds. To the north, it is positive and ranges from 0 to 90°. To the south, it is negative and ranges from 0 to –90°.

The right ascension and declination of a star is shown in Figure A.1.

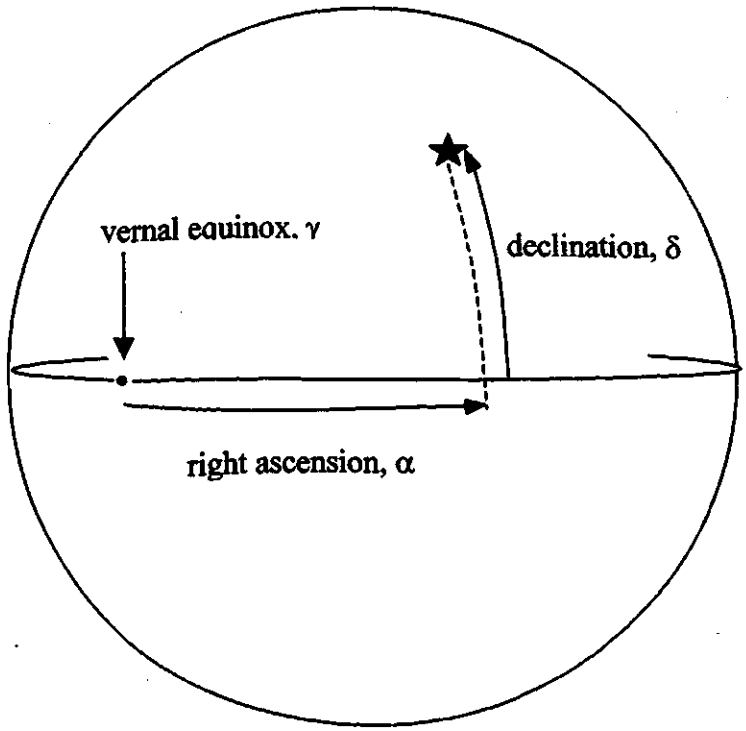

Figure A.1 Celestial coordinates.

It should be noted that the celestial coordinate system is not truly inertial [36]. Due to the gravitational force of the Moon and Sun, the earth bulges at the equator. Over a period of 26,000 years, the celestial equator slowly changes position causing a slow change in the right ascension and declination of stars. Therefore a date is given along with celestial coordinates to accurately define the position of the vernal equinox. Examples are 1900, 1950 and 2000.

A.3 Angular Separation

A.3.1 Cosine Formula for a Spherical Triangle

The lengths of the sides of a spherical triangle are called arc lengths or angular separations [37].

For the spherical triangle shown in Figure A.2, the cosine formula for a spherical triangle is given by [26]

$$\cos a = \cos b \cos c + \sin b \sin c \cos A \tag{A.1}$$

where a is the angular separation between vertices B and C, b is the angular separation between vertices A and C, c is the angular separation between vertices A and B, and A is the inner angle A (=$\angle$BAC). Note that A represents an inner angle and a vertex.

A.3.2 Angular Separation Between Two Stars on the Celestial Sphere

It follows that for a spherical triangle with celestial coordinates as shown in Figure A.3, the angular separation between two stars is calculated from the following expression [26]

$$\cos\theta = \cos(90° - \delta_1)\cos(90° - \delta_2) + \sin(90° - \delta_1)\sin(90° - \delta_2)\cos(\alpha_2 - \alpha_1) \tag{A.2}$$

where θ is the angular separation between the two stars, (α_1, δ_1) is the right ascension and declination of the first star in degrees, and (α_2, δ_2) is right ascension and declination of the second star in degrees.

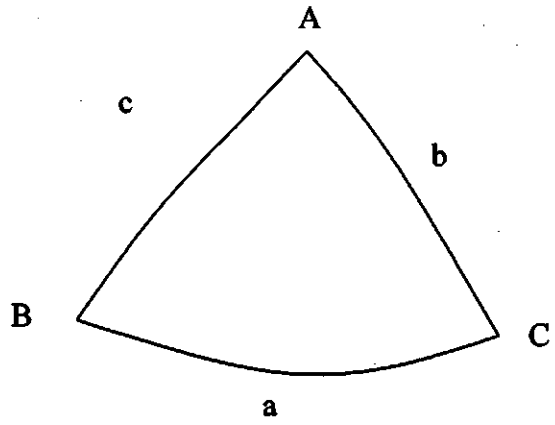

Figure A.2 Spherical triangle.

A.3.3 Angular Separation Between Two Image Stars

Calculation of angular separation between two image stars is given by

$$\cos\theta = \cos\chi \cos\psi \tag{A.3}$$

where θ is the angular separation between the two stars, χ is the angular difference between the two stars in the x direction (horizontal) and ψ is the angular difference between the two stars in the y direction (vertical).

This formula can be easily derived from the cosine formula, Equation (A.1). For stars in an image, inner angle A is 90°, and by substituting this value into Equation (A.1), the following result is

$$\cos a = \cos b \cos c. \tag{A.4}$$

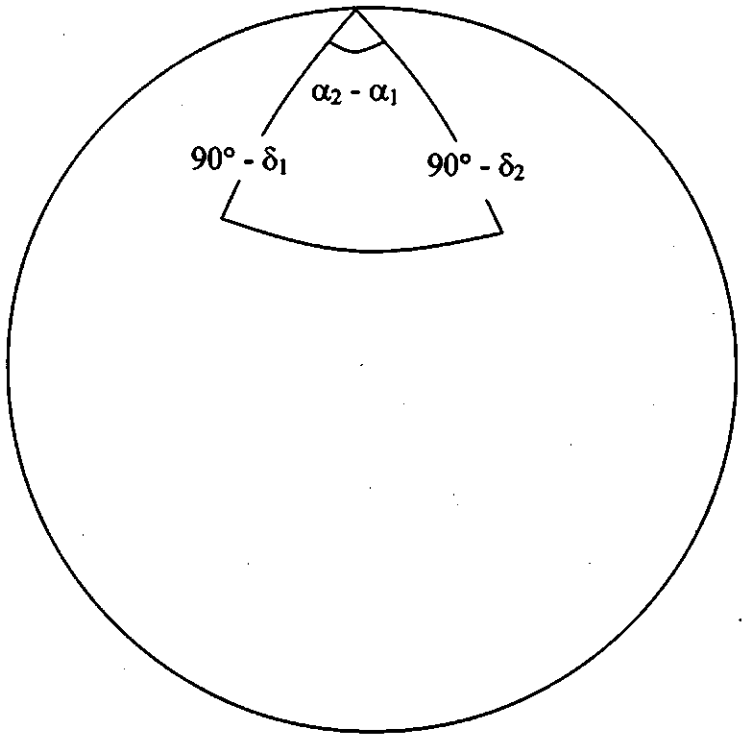

Figure A.3 Spherical triangle with celestial coordinates.

Appendix B

FLOWCHARTS OF MATCHING THE STAR PATTERN

There are three flowcharts to explain the matching of the star pattern. The first flowchart, Figure B.1 begins with calculating the angular separations between stars in an image and finishes with the lists of triads that were generated. The second flowchart, Figure B.2 continues from the end of the first flowchart and describes the matching of the stars once the lists of triads have been generated. The third flowchart, Figure B.3 expands on the box in the second flowchart labeled as 'Attempt to match each individual star'.

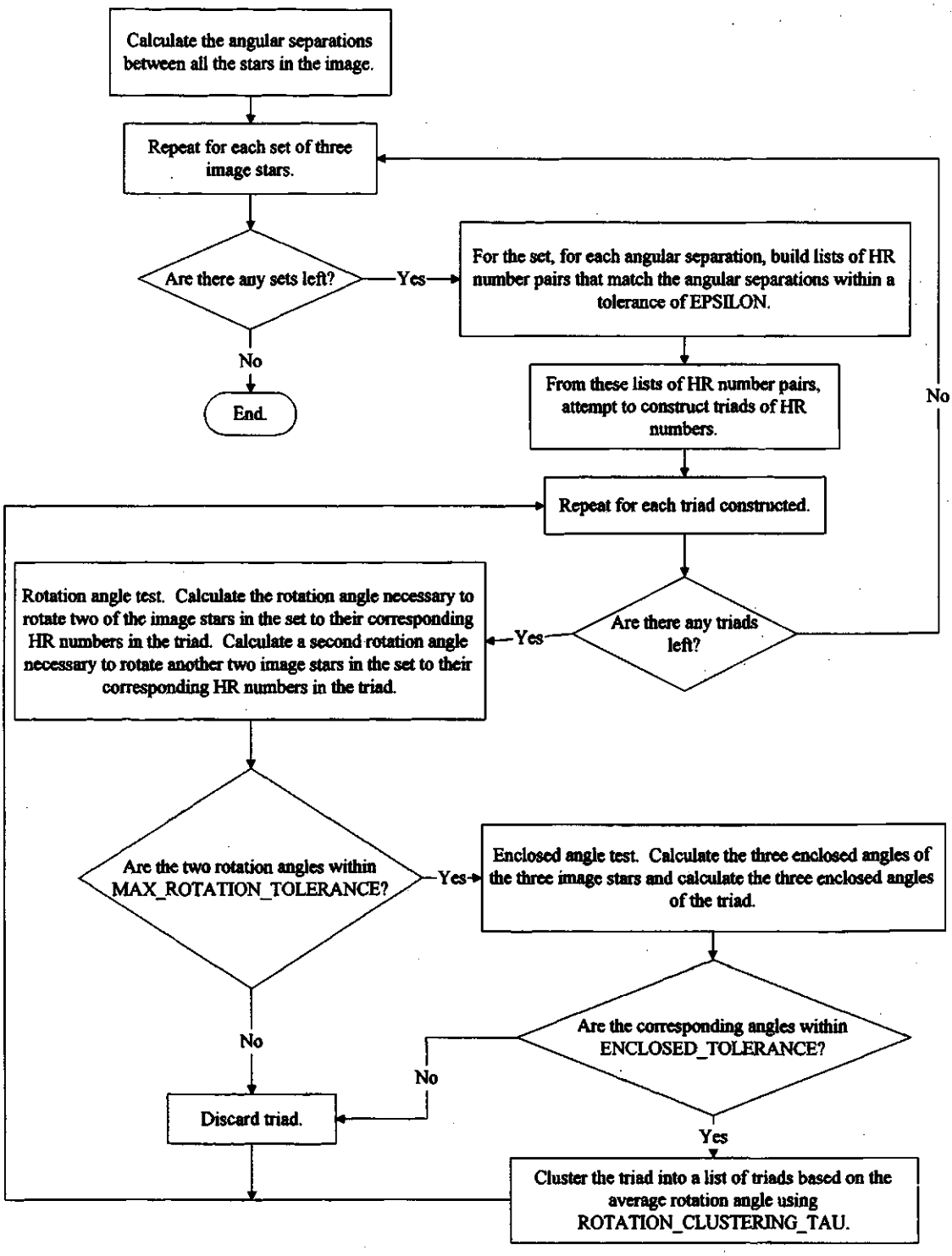

Figure B.1 Flowchart of matching the angular separations and generating triads.

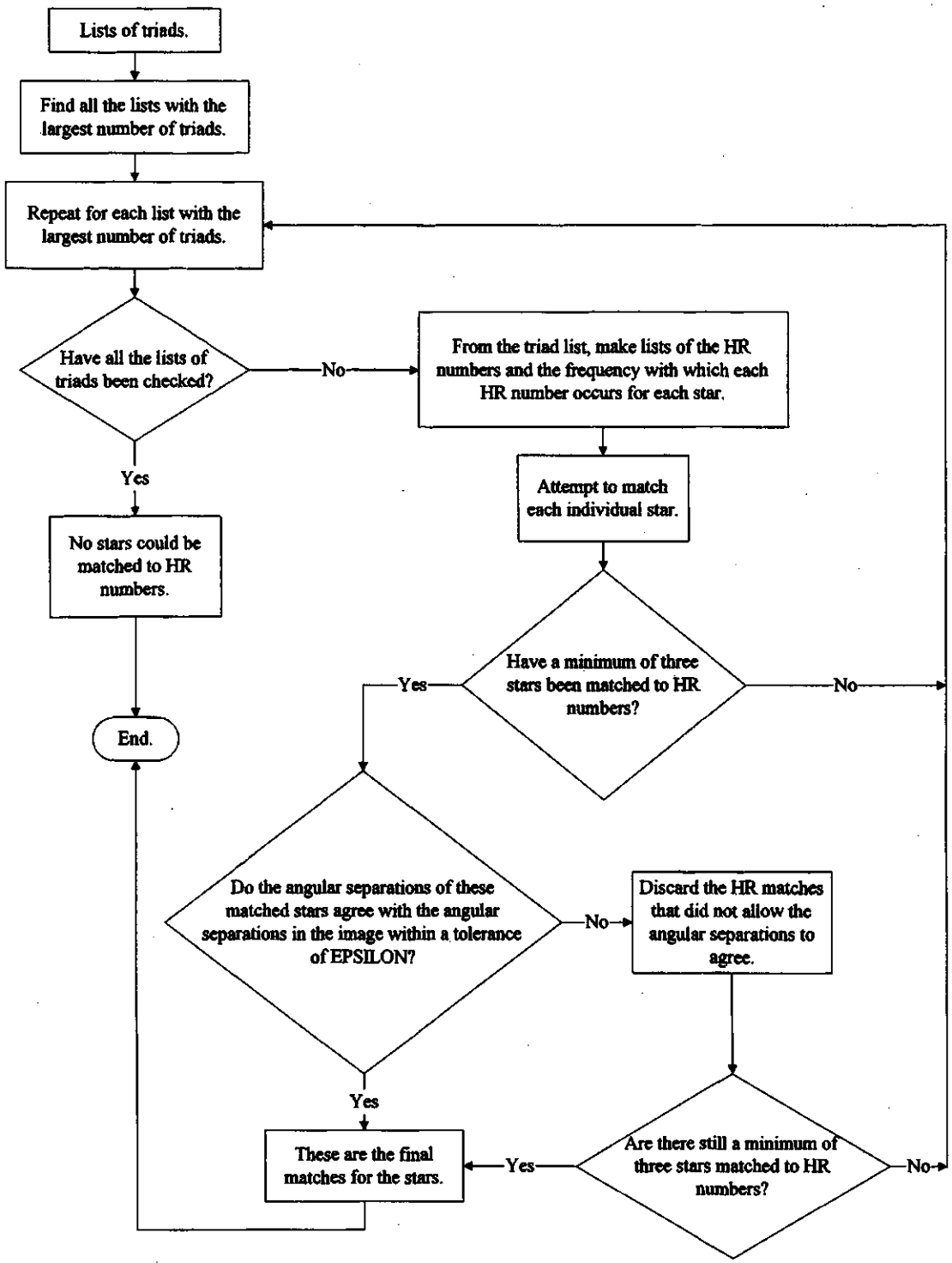

Figure B.2 Flowchart of matching the stars once the lists of triads have been generated.

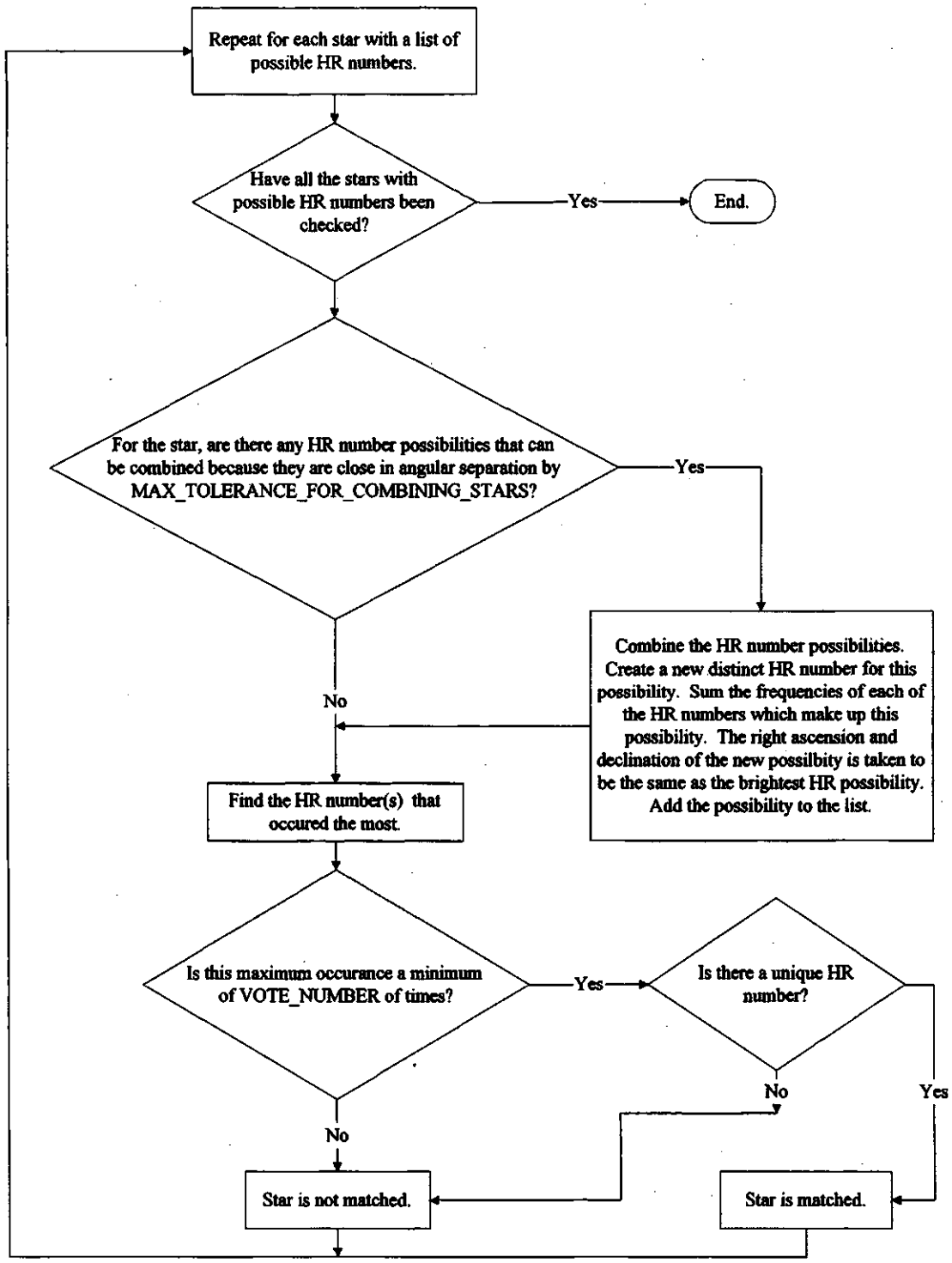

Figure B.3 Flowchart of matching each individual star.

Appendix C

EXECUTION TIMES

These are the details of the execution times of images Vega and Deneb. For each trial listed below, recall that a repetition loop of 100 times was used. Note that the final row in each table is the individual run-time. Unless otherwise mentioned, some important parameters that were not static or otherwise mentioned in Chapter 4 are: MAX_STARS = 12 [34], except for fixed-point version B where MAX_STARS = 5 for Vega and MAX_STARS = 9 for Deneb; and STARS_IN_BSC = 9999 [34]. If other values for these parameters were used they appear as a note below the table of interest.

Table C.1 Vega: execution times on the Pentium II with Visual C++ 4.0.

trial	floating-point		fixed-point	fixed-point
	with the image center (seconds) +/- 0.001	without the image center (seconds) +/- 0.001	version A (seconds) +/- 0.001	version B (seconds) +/- 0.001
1	37.424	34.610	23.875	19.387
2	40.358	36.282	23.985	20.610
3	40.598	36.482	24.436	19.508
4	39.918	35.972	24.486	19.548
5	40.568	36.713	23.854	18.537
6	37.775	36.913	24.425	18.277
7	39.928	37.174	24.375	18.266
8	40.378	36.462	24.655	18.267
9	40.438	36.931	24.446	18.296
10	40.268	36.703	24.575	18.256
Average	39.765	36.424	24.311	18.895
Run-Time	0.398	0.364	0.243	0.189

Table C.2 Vega: execution times on the Pentium II with Visual C++ 5.0.

trial	floating-point		fixed-point	fixed-point
	with the image center (seconds) +/- 0.001	without the image center (seconds) +/- 0.001	version A (seconds) +/- 0.001	version B (seconds) +/- 0.001
1	23.083	19.628	19.628	14.581
2	23.223	19.749	19.718	14.451
3	23.063	19.768	19.689	14.490
4	23.193	19.759	19.659	14.431
5	23.224	19.768	19.648	14.481
6	23.214	19.718	19.689	14.491
7	23.223	19.708	19.699	14.510
8	23.213	19.719	19.718	14.460
9	23.133	19.688	19.648	14.450
10	23.114	19.768	19.668	14.450
Average	23.168	19.727	19.676	14.480
Run-Time	0.232	0.197	0.197	0.145

Table C.3 Vega: execution times on the MC68360.

trial	fixed-point version B (seconds) +/- 0.01
1	941.03
2	941.01
3	941.01
4	941.05
5	941.04
6	941.02
7	941.02
8	940.99
9	941.00
10	941.01
Average	941.02
Run-Time	9.41

Table C.4 Deneb: execution times on the Pentium II with Visual C++ 4.0.

trial	floating-point		fixed-point	fixed-point
	with the image center (seconds) +/- 0.001	without the image center (seconds) +/- 0.001	version A (seconds) +/- 0.001	version B (seconds) +/- 0.001
1	229.701	228.198	142.085	n/a
2	231.102	227.587	143.186	n/a
3	231.172	229.360	143.879	n/a
4	231.613	229.189	143.215	n/a
5	233.206	227.768	143.477	n/a
6	230.842	229.089	142.415	n/a
7	230.912	230.281	143.857	n/a
8	233.396	228.218	141.874	n/a
9	232.184	228.309	143.647	n/a
10	232.435	228.338	142.185	n/a
Average	231.656	228.634	142.982	n/a
Run-Time	2.317	2.286	1.430	n/a

Table C.5 Deneb: execution times on the Pentium II with Visual C++ 5.0.

trial	floating-point		fixed-point	fixed-point
	with the image center (seconds) +/- 0.001	without the image center (seconds) +/- 0.001	version A (seconds) +/- 0.001	version B (seconds) +/- 0.001
1	64.904	59.135	61.879	46.316
2	64.403	59.736	62.240	46.337
3	64.392	59.706	62.199	46.327
4	64.383	59.706	62.239	46.246
5	64.343	59.896	62.269	47.338
6	64.222	60.056	62.210	46.277
7	64.222	59.476	62.229	46.276
8	64.253	59.476	62.249	46.316
9	64.302	59.466	62.240	46.317
10	64.343	59.306	62.279	46.247
Average	64.377	59.596	62.203	46.400
Run-Time	0.644	0.596	0.622	0.464

Table C.6 Deneb: execution times on the MC68360.

trial	fixed-point version B (seconds) +/- 0.01
1	2366.96
2	2366.89
3	2366.98
4	2366.96
5	2366.89
6	2366.98
7	2355.85
8	2366.94
9	2366.93
10	2366.93
Average	2365.83
Run-Time	23.66

For Tables C.5 and C.6, the HR numbers were renumbered for fixed-point program B and STARS_IN_BSC = 1266 [34].

Appendix D

OEDIPUS-C DETAILS

D.1 Calculation of FOV of OEDIPUS-C Camera

The FOV was not given explicitly by Beattie et al., rather the number of pixels of the image were given as 558.02 by 557.08 [16]. As well as the scaling factor of number of pixels per mm was given as 44.3991 pixel/mm for x and 53.3618 for y [16]. A formula used in the paper to calculate angular separation is [16]

$$\cos\theta = \cos(s1)\cos(s2) + \sin(s1)\sin(s2)\cos\beta \tag{D.1}$$

where

$$s1 = \tan^{-1}\left(\frac{\sqrt{x_1^2 + y_1^2}}{\text{focal length}}\right) \tag{D.2}$$

$$s2 = \tan^{-1}\left(\frac{\sqrt{x_2^2 + y_2^2}}{\text{focal length}}\right) \tag{D.3}$$

$$\beta = \tan_2^{-1}\left(\frac{y_2}{x_2}\right) - \tan_2^{-1}\left(\frac{y_1}{x_1}\right) \tag{D.4}$$

θ is the angular separation, (x_1, y_1) is the pixel coordinate of star 1 in mm, (x_2, y_2) is the pixel coordinate of star 2 in mm, focal length = 25mm, s1 is the spherical distance from the image center to star 1, s2 is the spherical distance from the image center to star 2, β is the angle between the stars and $\tan_2^{-1}$ is the four quadrant solution of $\tan^{-1}$.

To calculate the x FOV, consider star 1 to have the coordinates of (-6.6220, 0.0000) and star 2 to have the coordinates of (6.6220,0.0000) in units of mm. Substitute the values into Equation (D.1) and the angular separation or x FOV is calculated to be 29.6716°.

To calculate the y FOV, consider star 1 to have the coordinates of (0.0000, -5.2198) and star 2 to have the coordinates of (0.000, 5.2198) in units of mm. Substitute the values into Equation (D.1) and the angular separation or y FOV is calculated to be 23.5869°.

D.2 Determination of Incorrect FOV

The values of calculated and true angular separation for stars in OEDIPUS-C image 6 using a FOV of 29.6716° by 23.5869° are given in Table D.1. These values were used to create Figure 4.10 which shows the linear relationship between true angular separation and the difference between the true and calculated angular separation.

D.3 Summary of Matched Stars for OEDIPUS-C Images 1-25

The detailed results for OEDIPUS-C images are given in Table D.2. Do not look for a common HR number to correspond to an image star, that is, the HR numbers in each column are not supposed to match. Rather, for each image, the image stars are all numbered differently based on the number of pixels making up the star. What is important to note is that the same HR numbers are found throughout the table. Using the matched image stars, the right ascension and declination were calculated and are shown in the table. The error was determined by using the calculated center in the same manner as described for Vega in Section 4.2.1. For each image, the maximum error was noted and then averaged for all the images. For the successful images, Figure 4.11 was created by plotting declination versus right ascension.

Table D.1 The calculated and true angular separations of stars in OEDIPUS-C image 6.

image star	image star	calculated angular separation (degrees)	HR number	HR number	database angular separation (degrees)	difference between database and calculated angular separation (degrees)
1	2	4.1196	8316	8465	3.6182	-0.5014
1	3	26.4279	8316	168	23.3426	-3.0853
1	4	27.0294	8316	264	23.8575	-3.1719
1	5	21.0464	8316	21	18.5455	-2.5009
1	6	5.1726	8316	8494	4.5292	-0.6434
1	7	6.7468	8316	8571	5.9523	-0.7945
2	3	22.6566	8465	168	19.9812	-2.6754
2	4	23.6393	8465	264	20.8618	-2.7775
2	5	17.3605	8465	21	15.2812	-2.0793
2	6	1.4974	8465	8494	1.2858	-0.2116
2	7	2.7259	8465	8571	2.4147	-0.3112
3	4	5.4179	168	264	4.6789	-0.7390
3	5	5.5859	168	21	4.9149	-0.6710
3	6	22.3577	168	8494	19.6891	-2.6686
3	7	19.9533	168	8571	17.5743	-2.3790
4	5	6.9322	264	21	6.1465	-0.7857
4	6	23.6668	264	8494	20.8625	-2.8043
4	7	21.0526	264	8571	18.5731	-2.4795
5	6	17.1917	21	8494	15.1083	-2.0834
5	7	14.6783	21	8571	12.9027	-1.7756
6	7	2.6729	8494	8571	2.3325	-0.3404

Table D.2 Summary of OEDIPUS-C results.

image	HR numbers of matched image stars image stars 1	2	3	4	5	6	7	right ascension (degrees) +/- 0.1	declination (degrees) +/- 0.1
1		21	8571			8494	9045	348.4	55.9
2		21	8316	8571		8494		349.2	55.7
3	8465	168	21	8571		219	8494	349.4	55.8
4	no stars were matched								
5	8465	8316	8571	21		10264		350.2	56.0
6	8316	8465	168	10264	21	8494	8571	350.3	56.3
7	no stars were matched								
8	21	168	10264	8465	8571			350.2	56.9
9	no stars were matched								
10	no stars were matched								
11	not enough stars in the image								
12	21	8465	8571			8494		348.2	57.5
13	no stars were matched								
14	21	168	8465	8571		8316		347.4	56.9
15	21	168	8465	8316		8571		347.3	56.7
16	21	168		8465	8571	8316		347.4	56.3
17	no stars were matched								
18	21	168	8465	8494	8316	8571		347.8	55.7
19	168	21	8494	8465	8571			348.7	55.5
20	168	21	8465	8571	8538			348.7	55.3
21	no stars were matched								
22		6688	7180	7352	6945			see	below
23	no stars were matched								
24		8571		8465		9045		see	below
25	168	8465	21		8571	8316	8334	351.6	56.1

For image 22, these stars are the incorrect matches and therefore the attitude was not correct. For image 24, while the stars were matched correctly, the program was unable to determine the image center. The initial estimate of the image center was off by several degrees and diverged after that. If the program could match star 1 to HR 21, the attitude could be calculated (351.1°, 56.0°).