Attitude Determination by Image Processing Algorithms

Hans J. Königsmann, Hans J. Rath, Hartmut Renken

ZARM, University of Bremen

P.O. Box 330440 2800 Bremen 33 Germany + 421 218 2940

To determine attitude or direction with star constellations is perhaps the oldest method in the field of navigation and control. Besides its accuracy, star constellations can give three axis information by using just a single sensor, a digital CCD camera. What is very easy for men looking at the stars turns out to be complex if done by an onboard computer. The process of star identification can be divided into two parts: object segmentation and, needing much more calculation time, the matching process. Matching can be explained as comparing a star catalog stored in a computer memory with identified objects in a picture.

A matching algorithm based on this principles has been developed and tested for the low cost satellite BREM-SAT. Tests have shown that at least four stars are necessary to identify a unambiguous attitude with 0.2° accuracy. Due to the comparably long computation time (20-120s), only the spin axis direction of the satellite is determined. This axis moves very slowly, but knowing its attitude is essential.

Different test methods, including a hardware-in-the-loop test with the sensor matching simulated constellations on a computer screen, will be presented. For future projects, several improvements has been found which will reduce the computation time and give full three axis identification

1. Introduction

Knowing the spacecrafts attitude is essential for most experiments and many subsystems, even if this attitude is passively or just partly (e.g. one-axis) controlled. Since most attitude sensor has restrictions concerning its field of view (FOV) and some physical restraints (e.g. possibility of only two-axis determination with sun, earth and magnetic field sensors), usually more than one sensor is necessary onboard. Within this context, star sensors have been mainly used to determine one-axis attitude to a high accuracy. This function is based on two different principles:

- A single star, which can be easily divided from other stars in the FOV due to its brightness (for instance Canopus), is tracked. Tracking can be done in two ways: The star sensor is mounted on top of a two-axis gimbal system and the rotation angle is measured (see fig. 1). Or, the position of the star is determined by its position in the focal plane. This can be done by algorithms based on object segmentation, which is very similar to the pre-processing algorithm used for our software.
- Two successive pictures are correlated to each other. The maximum of the correlation functions indicates a motion, which can be interpreted as a relative rotation. More complex algorithms allow the determination of the relative three-axis motion.

algorithms allow the determination of the relative three-axis motion.

It is obvious that both principles require additional sensors. For the first method, an initial attitude close to the nominal attitude has to be found. This type of sensor does not allow maneuveribility, since a "special" star has to be kept within the FOV (except some maneuveribility for the gimballed star sensor). The second method gives rotational rate information, which has to be integrated and initialised with inertial attitude information. For simplification, this might be done on ground.

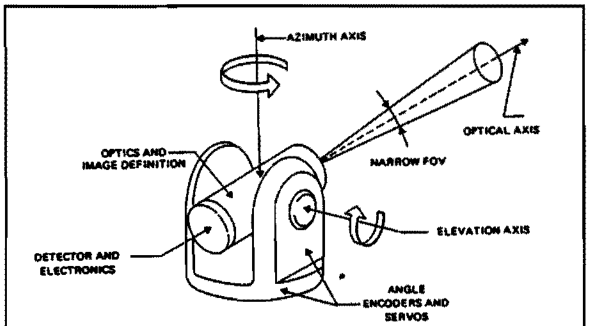


Fig. 1) Gimballed Star Sensor from (6)

The star-sensor software developed at ZARM differs in some points from these two principles. A digital image is processed by a computer in such a way, that all stars are extracted and stored in a list with their (x,y) position in the focal plane. These stars build a constellation, which is compared with all possible constellations of a star catalog. Whenever a constellation has been positively and definitely identified, the celestial coordinates of the FOV center can be calculated. Since the coordinates of the stars in the FOV are known by the star-catalog, two stars will indicate the rotation about the FOV center and a reference line. This, in turn, gives a three-axis attitude information. A first test in Space will be onboard of the BREM-SAT satellite (1), scheduled for launch in February, 1993.

2. Principle of Function

All examples are based on images of the constellation Cygnus (fig. 2), which can easily be found (or matched) in the Milky Way. Fig 2 is based on an actual image taken by the star sensor, whereas fig 3 shows the same sector of a star catalog (5).

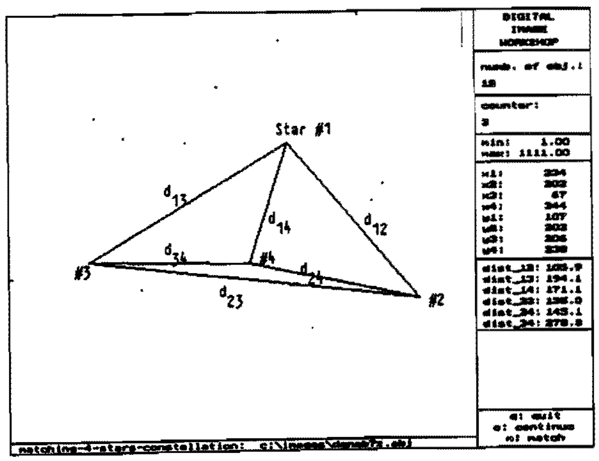


Fig. 2) Constellation Cygnus

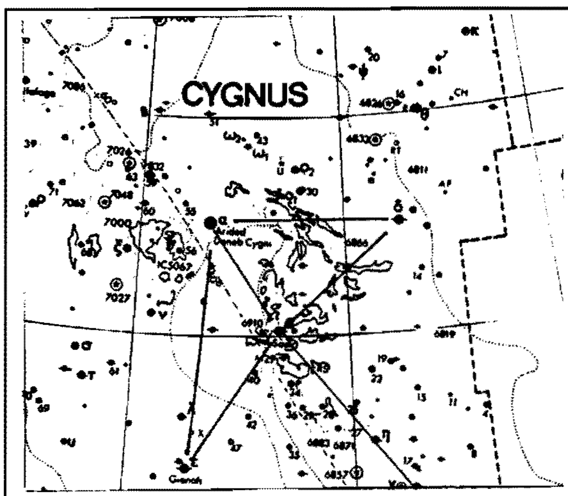


Fig. 3) Cygnus, star catalog (4)

This image is stored in the digital memory pixel by pixel. A star is identified by using a region-oriented segmentation algorithm (connectivity analysis). Stars are treated as objects and stored in a list with the (x,y) coordinates in the focal plane and the gray value. If stars cover more than one pixel (which happens at magnitudes below $m_V=2$), the center of the area and the accumulated gray value will be taken instead. Corrupted pictures are detected if their average gray value exceed a determined limit. Matching will only be done if at least three stars are segmented. If only three stars are identified, the attitude history is necessary to give an estimate. However, in most cases between four and six stars are used. The brightest stars are selected. All distances between these stars are calculated and stored in a list (see fig. 2).

For a four star constellation, six distances can be found. Since distances in the focal plane represent angles between stars at the sky unit sphere, an angle factor (degree/pixel) has to be applied. This angle factor depends on the camera optics and the pixel size in the focal plane. At that point, the six angles can be searched for in the star catalog. Beginning with the angle A_{12} for the given example, all angles between the first star of the star catalog and all other stars contained are determined and compared to it. Due to errors, a angle tolerance has to be taken into account. The smaller this tolerance can be defined, the less computation time will be used. If the first star of the catalog and all other stars do not match A_{12}, we proceed with the second catalog star. For our example, we assume that star #123 and star #567 matches the angle.

A third image star is added with the angles A_{23} to star #2 and A_{13} to star #1. Starting again with catalog star #1, the angle to the catalog star #123 is determined. This angle has to be compared to A_{13} **and** A_{23}. When one of these two angles can be found while scanning through the catalog again, the angle to the found star and to catalog star #567 must be matched also. If it does, we found the third star, if not, we have to proceed. Finding the third star we continue searching for the fourth by the same method. If available, the stars are verified with picture star #5 and #6.

In case the third star could not be identified, catalog star #123 and #567 have been wrong. Scanning through the star catalog will be continued with #123 and #568, #569 etc. for the angle A_{12}.

To reduce computation time, the declination of two stars is regarded at first. If the difference in declination is greater than our FOV, both stars cannot be contained in one picture. The angle tolerance has great impact on the computation time, too. The greater this tolerance, the more stars are matched. In turn, more angles have to be compared. The most effective method to reduce computation time is to reduce the star catalog size. This means that only stars up to a limited magnitude and in a certain area shall be included. As mentioned earlier, matching is not definite if either not enough stars are provided in the picture or if the angle tolerance is too large. With a tolerance corresponding to one pixel element we found that four stars are in 98% sufficient to identify the attitude unambiguous. This is demonstrated in fig. 4, 5 and 6, showing how the number of matched star constellations is reduced to a single solution using two, three and four stars for matching.

If the stars have been found, it is easy to find the coordinates where the camera is aligned at. Furthermore, because the coordinates of all stars contained in the picture are known from the star catalog, the direction of the x-axis of the focal plan can be determined. With all this information, the three-axis attitude can be determined.

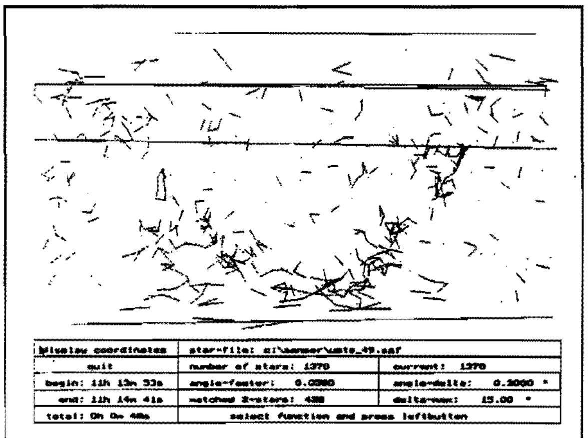


Fig 4) Matched star pairs with two star-matching

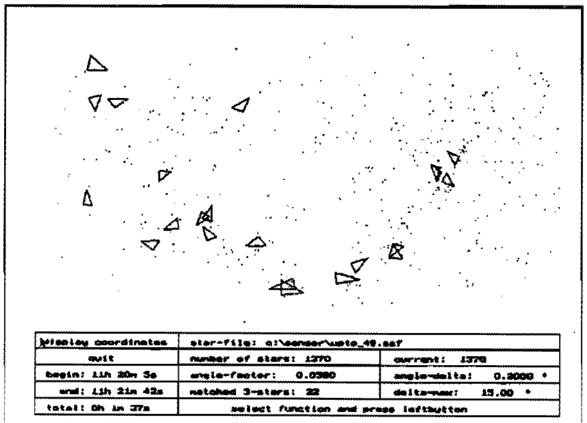


Fig 5) Matched stars with three star-matching

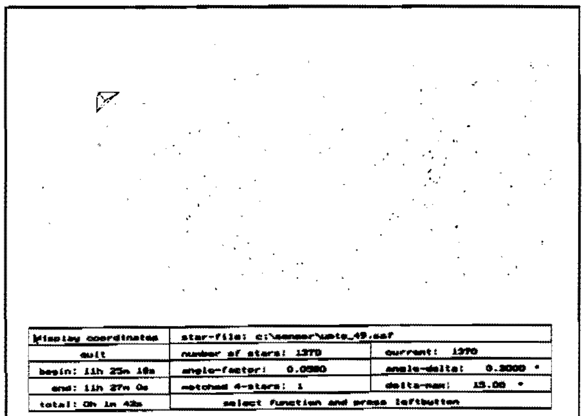


Fig. 6) Matched stars with four star-matching

3. Hardware Requirements

Because the matching software needs comparably long computation time, a powerful computer is necessary if time is critical for attitude control. If, on passively controlled spacecraft, only attitude determination is necessary, the matching process might be done on ground to reduce hardware costs. The sensor itself must be able to detect stars up to a certain magnitude, which strongly depends of it's field of view. The smaller the FOV is, the higher the sensitivity has to be. To keep the hardware as simple as possible, no gray values are necessary. Each pixel value shall be 0 for black background and 1, if a star is detected.

To determine the sensitivity for a given field of view, a short analysis has been done to demonstrate the dependencies. Fig. 7 shows how many stars a 10° x 10° FOV will contain, if the sensor can detect stars up to $m_V = 4.4$. It can be seen that in certain areas "black holes" with less than four stars exist, where matching is not definite. Increasing the sensitivity up to 4.8 (Fig. 8) shows an effective improvements. Instead of increasing the sensitivity, it is as well possible to increase the FOV with similar results.

Fig. 7) Stars distribution,

m_V<4.4

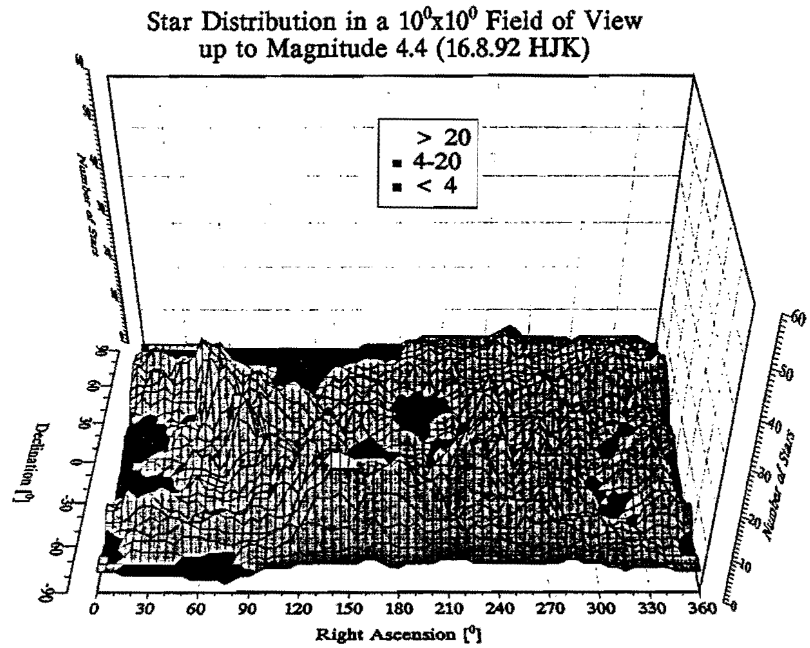
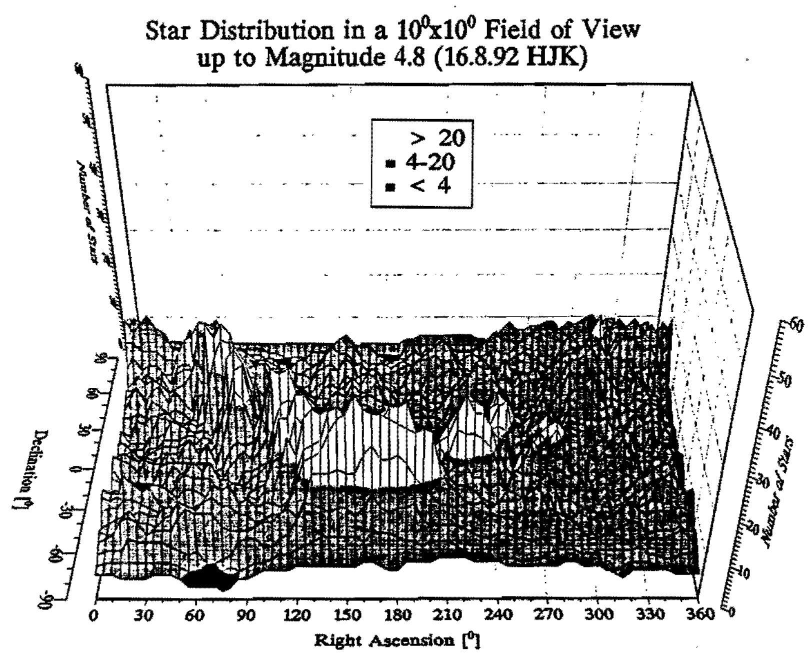


Fig. 8) Star distribution,

m_V<4.8



FOV and sensitivity cannot be increased without limits. Too high sensitivity needs a larger star catalog, which on one hand causes long computation times. Larger FOV´s, on the other hand, restrict the attitude because sun, earth and moon are not allowed in the FOV. In addition, the FOV is the most important factor with regard to the computation time, which can easily be seen in fig. 9, showing the required time vs. the FOV. As mentioned above, the angle tolerance is the third parameter influencing the time required.

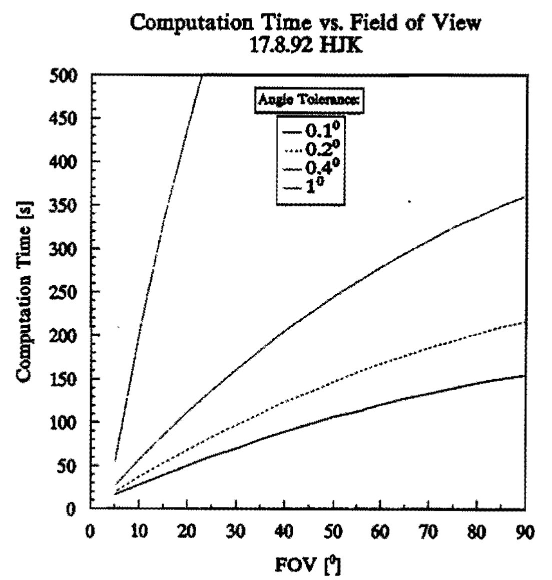


Fig. 9) Times have been measured with a 33 MHz 486. For a T800 Transputer, add 10-20%.

4. Error Sources

Images containing the sun, the earth or the moon cannot be used for matching. For this reason, as well as for planets, which are normally not included in the star catalog, the sensor cannot operate in the ecliptic plane. Beside this restriction, the sensor itself might introduce errors, which should be taken into account: Lens errors, which occur mainly at the images corners, have to be compensated with the angle tolerance. Furthermore, failure pixel have to be taken into account or, choosing qualified components, avoided. For the BREM-SAT project, the star catalog can be edited and reloaded by telecommand. This feature allows correction if the star catalog does not contain correct star coordinates. Bright planets (i.e. Venus and Jupiter) coordinates can be included, but have to be updated continuously. On very rare occasions, phenomenon like novae might confuse the matching software. In fig. 10, a novae of m_V= 4.3 in our example constellation Cygnus is shown (Compare to fig. 3!). With minor changes, this software can be used by observatories to detect novae automatically.

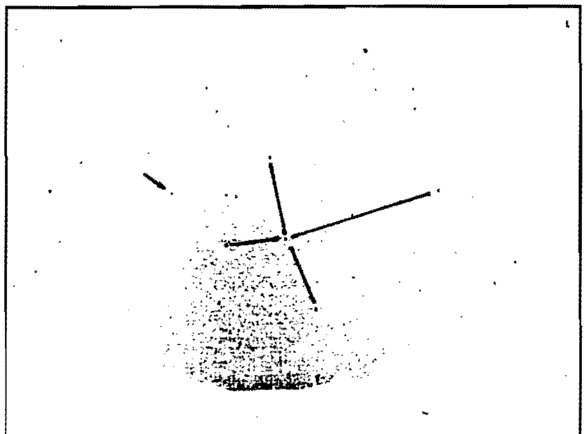


Fig. 10) Novae in constellation Cygnus (8)

5. Tests

Several approaches have been used to test the star sensor software. Real images, like the on of fig. 2, have always been matched unambiguously. However, testing images of the entire sky sphere consumes too much time and travel costs, and a method of matching artificial images from a computer screen has been developed instead. By that way, the sensor hardware was included in the test and a hardware-in-the-loop test could be made. After all hardware had been integrated in the BREM-SAT satellite, the sensor itself has been simulated by a PC.

6. Future Development

With the satellite project BREM-SAT this algorithm will be tested in Space. All BREM-SAT flight hardware passed recently through the vibration, thermal and EMC tests and is ready for flight. Since the stars segmented from the star sensor images can be send to the ground station, ground support software to verify the attitude must be developed. The main effort will be made to reduce the matching time. Vectorisation reduces the computation time significantly, and the first goal is to develop an autonomous attitude sensor capable to give three-axis information at 1 Hz.

Acknowledgements

The authors would like to thank the German Space Agency DARA which is funding the small satellite project BREM-SAT. Henning Bässmann from DIGSY/University of Bremen supervised great parts of this work and gave valuable hints. Many thanks also to OHB-System, responsible to for all major subsystems and EFK, responsible for onboard software. We appreciate the support from Schneider-Kreuznach, developing the sensor optics.

References

(1) A. Ginati, G. Lehmann, H. J. Königsmann, H. Iglseder, H. J. Rath
BREM-SAT, a Small Scientific Satellite
AIAA/USU Conference 1990

(2) A. Ginati
The TUBSAT-1 attitude control and stabilisation system
AIAA/USU Conference 1989

(3) H. Renken
Realisation eines bildverarbeitenden Systems zur Erkennung von Sternkonfigurationen
(Realisation of an image processing system to identify star constellations)
1992

(4) Vehrensberg-Blank
Handbook of the Constellations,
1987
Treugesell Verlag

(5) Wil Trion, Sky Atlas 2000
Cambridge University Press, 1989

(6) Wertz, J. R. (ed.)
Spacecraft Attitude Determination
D. Reidel, 1978

(7) Wertz, J. R., Wiley, J.L. (ed.)
Space Mission Design and Analysis
Kluwer Academic Publishers, 1991

(8) SKY & TELESCOPE, Aug.1992