Robustness Study of Non-Dimensional Star Pattern Recognition for a Typical Star Tracker

Jafar Roshanian[1], Shabnam Yazdani[2], Masoud Ebrahimi[3]
1- K.N.Toosi university of technology Tehran, Iran.
Email: roshanian@kntu.com
2- K.N.Toosi university of technology Tehran, Iran.
Email: shabnamyazdani@yahoo.com (Corresponding Author)
3- K.N.Toosi university of technology Tehran, Iran.
Email: ebrahimi_k_m@yahoo.com

Received: February 2012 Revised: July 2012 Accepted: August 2012

ABSTRACT:
Non-Dimensional star pattern recognition uses planar angles as its recognition feature. This feature is independent of image focal length and optical axis offset. However, this independency doesn't mean that the algorithm conserves its robustness in presence of any type of errors. These errors arise from poor hardware calibration and software inaccuracy that causes the angles to differ from their true amounts stored in the database. In order to evaluate the effect of angle differences on algorithm performance, overall disposition of bright point centers is modeled. The monte-carlo simulation method is then used to evaluate the algorithm's performance for different amounts of error. Results demonstrate that 0.1 pixel size error is admissible in conserve the trade-off between desired update frequency, hardware accuracy and algorithm's robustness.

KEYWORDS: Non-Dimensional star pattern recognition algorithm, K-vector search method, Star tracker.

1. INTRODUCTION

In the past few decades, it has been proven that one of the most accurate attitude determination tools in space navigation is the star tracker. Star trackers use the information of recognized stars available in the field of view (FOV) in order to determine the spacecraft attitude with respect to Earth Inertial Reference frame. For this purpose, a photo of stars in FOV is taken, processed and recognized; afterwards, the relation between the photo frame and star inertial information is derived to determine the attitude.

Up to date, several star pattern recognition algorithms have been developed, which use different types of recognition features. These features are mainly related to star arrangements and their numeral configuration in the image plane. For example, the area of a triangle (which is created using 3 stars on the image) [1], triangle's polar moment or the most popular feature, the inter-star angle between stars in three-dimensional space, with focal length being the third dimension.

After 1976 that Junkins et al. [1] introduced the inter-star angle as the main feature in pattern recognition, the scientists have been looking for a pattern feature which is totally independent of prior attitude information and has the least dependency on the interference parameters such as image plane size, focal length, optical axis offsets, star magnitude or centroiding algorithm accuracy.

In 2006, Samaan et al. [1] introduced the Non-dimensional algorithm which uses the planar angles of a star triangle as a pattern feature. It is obvious that a planar angle is independent of focal length, and optical axis offset variations. While the algorithm doesn't need the calibration information, it is capable of determining the focal length as well. [1]

However, the independency of the camera parameters doesn't conclude in right recognition under all circumstances. Camera calibration parameters are not the only parameters affecting algorithm's performance. If for any reason, the input information such as centroiding data is not valid enough and the angles computed, differ from the inertial combination, the algorithm might return no match or even worse, the wrong match. So a close study on the reasons manipulating the errors is necessary.

To be brief, the overall error budget of a star tracker can be listed as below [1]:

1) Optic errors including Ground Calibration, Thermal Distortion, Chromaticity, Point Spread Function (PSF) Distortion.

2) Centroiding error including Pixel Non-Uniformity, Quantization error, Centroid algorithm Uncertainty.

3) Noise Equivalent Angle: errors including Read out Noise, Dark Current Noise, Stray Light Noise and Photon Noise.

To be able to study the effect of each error sources mentioned above, one must have accurate knowledge of their behavior and various parameters related to them which is more than this paper's intention.

The overall effect of the hardware and software errors is the difference between the inertial angles computed from star catalog and planar angles computed from image information. This difference is caused by various reasons but the goal here, is to determine the algorithms robustness in the presence of a certain amount of planar angle variation.

So instead of modeling the error sources individually, which won't provide us any fruitful information as well, we assume that a finite source of error has caused an angle difference and the algorithm is supposed to distinguish the real combination from its database. Similar error analyze has been performed such as [1] for Pyramid algorithm which uses inter-star angle as its recognition feature.

In the second section there will be a brief explanation of Non-Dimensional star pattern recognition performance, in the third section database considerations are discussed, in the fourth section a brief explanation of search method and error analysis is presented and in the fifth section the conclusions can be seen.

2. NON-DIMENSIONAL STAR PATTERN RECOGNITION ALGORITHM

As mentioned before, the non-dimensional algorithm uses the planar angles of star triangles on the image plane to recognize the star combination. It is proved that the star triangles from inertial coordinates are relevant to the star triangles on the image plane and by benefiting this relation, a star pattern can be collated to inertial reference without accurate calibration of camera.

The algorithm starts by comparing the smallest angle of a typical star triangle generated on the image plane. The comparison is done using k-vector search technique [1,2], a novel search method proposed by Mortari et al. to accelerate the comparison of elements.

While the search method provides the algorithm with a range of certain triangle combinations, the biggest planar angle of the star triangle on the image plane is compared. This leads the algorithm to a unique answer.

A small portion of the database created for 15 degree square FOV and stars brighter than magnitude 5 using Hipparcos catalogue [4], can be seen in Table.1

Hipparcos catalogue contains 5044 stars brighter than magnitude 6. While trying to develop a database such as the portion shown in Table 1, the number of valid triangles exceeded 10 million after perusing only 1884 stars! And by valid, the probability of triangle's existence in FOV and having angles more than 1 degree is meant. Meanwhile it must be noticed that in order to create the database, we assume that in each loop, the star tracker boresight is pointed at each one of the 5044 stars of the sky and the triangles are formed using the boresight star as a fixed vertex.

By reviewing a few commercial star tracker brochures, one can easily realize that industrial star tracker databases usually contain less than 500,000 elements depending on the pattern feature their algorithm uses. This is beside the fact that "the bigger the database gets, the slower the recognition is performed", meanwhile the recognition will be less accurate because of the closeness of planar angles as shown in Table 1.

Table 1. A Portion of Non-dimensional Method Database

Triangle Index	Smallest Angle	Biggest Angle	Star ID	Star ID	Star ID
263185	28.60397	92.51451	61199	47391	58867
263186	28.604	94.80227	63724	67234	60260
263187	28.60408	115.8236	92041	88635	87936
263188	28.60409	112.2487	31125	30788	25859
263189	28.60415	122.2483	92818	94481	95372
263190	28.60425	97.75082	7588	2484	17440
263191	28.60495	81.76284	44816	47175	38827
263192	28.605	119.8081	90139	88128	88765
263193	28.60506	100.1218	40091	40706	37819
263194	28.60535	113.4421	87220	90595	86736
263195	28.60536	77.09538	86170	86736	89642
263196	28.60546	109.1302	80763	78104	83574
263197	28.60556	110.0415	25142	27511	22549
263198	28.60558	89.95953	73165	75379	69269
263199	28.60572	113.8923	27989	25142	25247

3. DATABASE DEVELOPMENT CONSIDERATIONS

In order to make sure that in any orientation, at least 5 stars exist in a typical field of view, Monte-carlo sky simulation is used. Calculating the number of stars in each FOV and specified magnitude is done using ten thousand random boresights (Iterations) and self developed sky simulator software. Choosing 5 stars is the consequence of the fact that recognizing only one triangle (3stars in FOV) is not considered reliable in any pattern recognition procedure so there must be at least one or two more stars present in the FOV to validate the first recognition. In this research having at least 5 stars within FOV was inspired by the same usage in patterns like Pyramid [1] which is reduced to four in further considerations. The results show that for 15 degree square FOV, the stars having magnitude 6 or less must be used to create database,

The histogram showing the number of stars for 15 deg square FOV and magnitudes 5, 5.5 and 6 are shown in Figures 1 to 3. The probability of having at least 5 stars in the mentioned FOV, for magnitude 6, is about 98%. But still decreasing the database volume is prior.

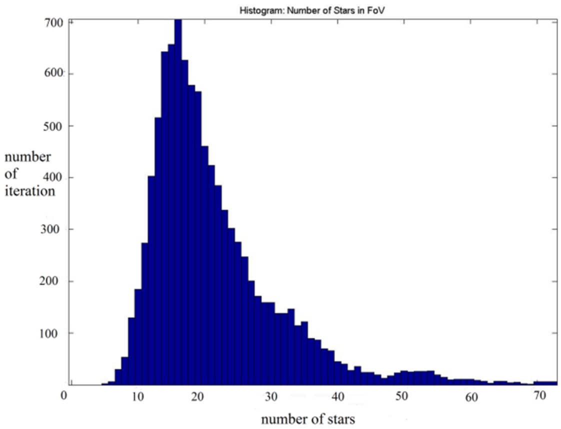

Figure 1. Histogram for 15 degree FOV and magnitude 6

For decreasing the database content, there exist a few solutions in two categories:

- **Reduction of catalogue stars:**
  1. Eliminating double stars.
  2. Reduction of magnitude.
  3. Sky uniformity.
- **Reduction of database elements:**
  1. Eliminating triangles with planar angles less than 1 degree.
  2. Eliminating triangles with small area.

By double star, two stars having small inter-star angle that they appear as one star in night sky imaging is meant. Their combination is as bright as sum of the both magnitudes. For magnitude 6, the Hipparcos catalogue contains 582 double stars that eliminating them doesn't help much in reducing the database content. But the results of magnitude reduction accompanied by eliminating double stars are shown in Table 2.

Table 2. Number of database triangles

Magnitude	FOV	No. of stars	No. of Triangles
5.5	15	2851	3581282
5.5 without double stars	15	2492	1220462

It can be seen that neither the magnitude reduction nor the double star elimination doesn't reduce the database content enough, plus the fact that elimination of double stars is not as helpful as it might sound. For example consider the Orion constellation; The Orion belt contains three stars of magnitude 2.4 (Mintaka), 1.65(Alnilam) and 1.85(Alnitak). But two of these three stars are considered as double stars; Mintaka accompanied by a star of magnitude 6.8 and Alnitak accompanied by a star of magnitude 9.5. As you can see, the second stars aren't even included in the mission catalogue but by eliminating the stars holding a double flag, we are actually eliminating stars that could result in a good triangle leading the algorithm to a true, fast and accurate recognition!

So regardless of double star elimination, the database content reduction is continued using the magnitude reduction solution;

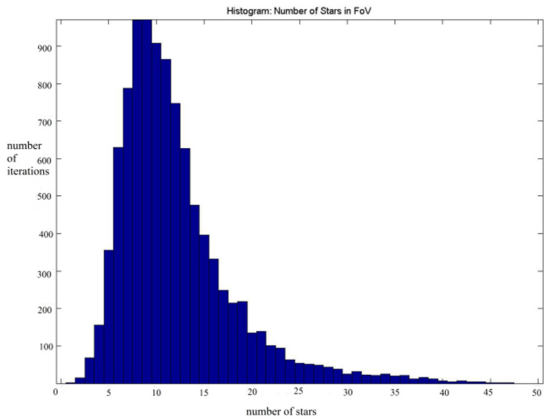

Figure 2. Histogram for 15 degree FOV and magnitude 5.5

If we consider that an accurate recognition can also be made by at least 4 stars in FOV, then the magnitude reduction can be more useful. In Figure 1, the probability of having 4 stars in FOV is about 99.8%. For magnitude 5.5 or less, as it can be seen in

Figure 2, the percentage reduces to 97% and for magnitude 5 or less, the probability is about 90.11%.

Hipparcos catalogue contains 1627 stars brighter than magnitude 5. This amount of stars, results in a database of 436646 triangles which appears to be acceptable.

It must be noticed that triangles with angles more than 1 degree are accepted in all above database information.

Uniforming the sky is also implied to the catalogue with magnitude less than 6 in two ways, using Spherical patches method [1]:

- 5 closest stars to center of each FOV which returned 2135 valid stars
- 5 Highest magnitudes in each FOV which returned 2885 valid stars.

The first combination returned a database of 2656695 triangles and the second one returned 2544479 triangles.

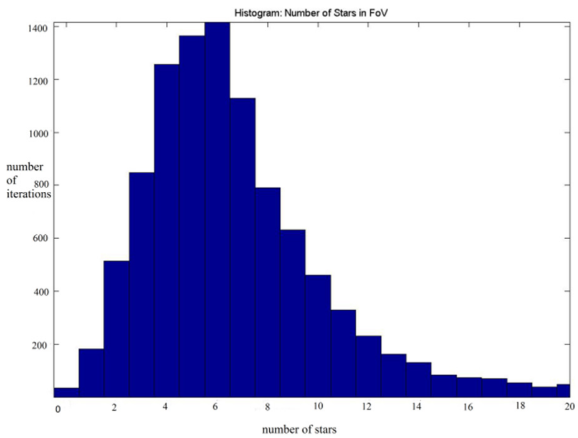

Figure 3. Histogram for 15 degree FOV and magnitude 5

So far the best result is produced by reducing the magnitude to 5 and FOV of 15 degrees.

4. SEARCH METHOD

A star pattern recognition performance beside the pattern feature is very dependent on the search method. To be able to compare a pair of data with its equivalent pair in database, the k-vector search method creates a margin around the smallest angle and returns a range of possible triangles. This margin is produced by adding a specific scalar, "αerror" to the smallest angle. For a small amount of αerror, the range will be small and comparison is done faster but the critical issue here is the error on the angle itself.

It can be seen in Table.1 that the small angles are barely different in the third decimal. The very close distribution of angles of totally different triangles makes the action of αerror more critical. Consider a specific amount of error on the angles calculated from image information, if αerror is less than the mean variation of angles in the database, the true combination will not be placed in search range and by comparing the biggest angle, algorithm will either end up in No match or even worse returns a Wrong match.

As mentioned in section 1 the error sources will eventually affect the planar angles. So instead of modeling the errors, random error on the triangle's vertices is implied which causes the planar angles to vary from their true amount.

It must be noticed that the amount an angle might vary according to this random error is dependent on the size of the triangle and the spot the vertices are located with respect to the image. To clarify this subject let's assume both big and small typical triangles in a typical sized image (although the concept can be extended to any image size but it must be mentioned that a 512*512 pixel image plane with pixel size of 9 micrometers was assumed which equals to a 4.6 mm square image plane and is a realistic assumption). By big and small, the area of a typical triangle is assumed. If a random error (in means of moving the vertex of triangle sideways randomly) is implied in thousand iterations, the variation of triangle shapes (and though the planar angles) can be seen in Figure 4. Please notice that for more resolution, Figure 4 was produced with less than thousand runs but the data used to produce the following histograms is the result of mentioned amount of iterations.

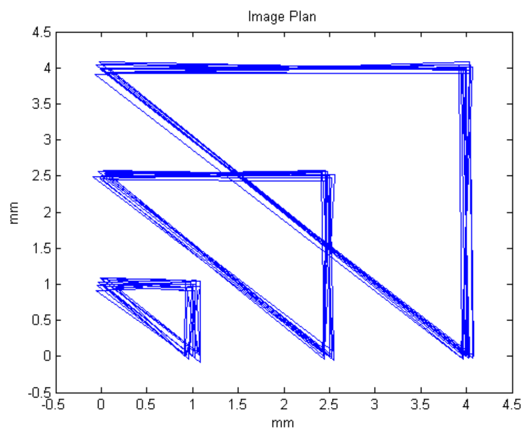

Figure 4. Angle variation for a typically big triangle.

It must be also mentioned that Figure 4 is resulted from moving the vertices for about 0.5 pixel (using normal random error) around their current position.

Figure 4 demonstrates that 0.5 pixel size error on bright point positions causes the angles of a typically big triangle to vary less than a typically small triangle. Afterwards the difference between the produced angle and the true amount is calculated and its normal distribution is plotted according to Figures 5 and 6. These figures demonstrate that 3σ of this Probability Distribution Function (PDF) is about 0.04 degrees. It means that for 0.5 pixel random error implied on a typically large triangle vertex, the planar angles vary for about 0.04 degrees from their true amount.

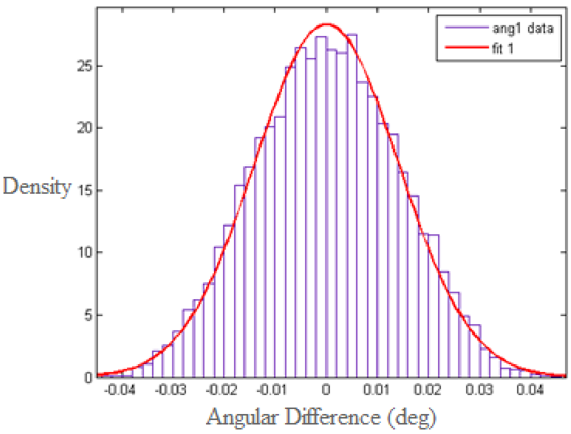

Fig. 5. Effect of 0.5 pixel size error on angle of a typically big triangle

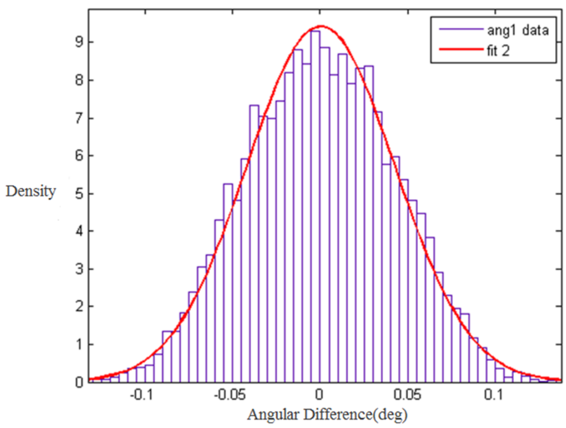

Fig. 6. Effect of 0.5 pixel size error on angle of a typically small triangle

The same procedure is repeated for a typically small triangle showed in Figure 4. 3σ of the current PDF is about 0.1 degrees which is more than big triangle's variation.

According to these results, bigger triangles (relative to image size) destabilize the algorithm performance less than small triangles but choosing a bigger triangle requires the stars on the image plane to be far from each other. In other words stars chosen would be closer to the image borders where the lens distortion can affect their true position systematically.

Meanwhile, 0.5 pixel error for determining the centroids of bright points on the image plane is considered a big value. The centroiding algorithms today, have reported accuracy of less than 0.1 pixel sizes.

Table 3. Monte-carlo test results

No match %	Wrong %	Right %	Angle difference between measured and true angle (deg)	Bright point position error (pixels)
0.0135	0.46	99.5265	0.001	0.02
0.0138	0.9028	99.0834	0.002	0.04
0.0138	2.7317	97.2544	0.006	0.1
0.0143	4.4959	95.4898	0.01	0.12
27.467	5.7714	66.7616	0.012	0.13
32.4542	6.0468	61.499	0.0125	0.135
36.8992	6.2518	56.849	0.013	0.14
38.5535	6.3277	55.1188	0.0132	0.142
50.2435	6.9079	42.8486	0.015	0.2

For the rest of the analysis, a moderate sized triangle is considered. The size is considered between big and small triangles mentioned above to include both effects.

Table 3 shows the algorithm's performance in the presence of different pixel size errors. The monte-carlo test results for 10 rounds of 1000 runs can be seen in Table 3.

The algorithm's performance is shown in Figure 7.

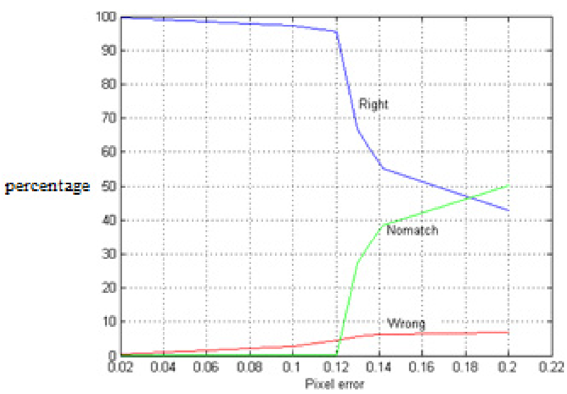

Fig. 7. Monte-carlo 10000 runs test results.

For 0.1 pixel error, the same procedure causes 0.006 degree error on planar angles which leads to 97% true recognition.

Table 4 shows the computing time required for 10 iterations of 1000 monte-carlo rounds. The computer characteristic is a PC Intel(R) Core™ i7 CPU 12Gbyte RAM. It can be seen from Figure 8 that the less accurate the bright point positions gets, the more computing time is required for triangle recognition.

So far it is showed that the Non-dimensional star pattern recognition algorithm can recognize true star combinations in the presence of a *total* error of 0.1 pixel size error.

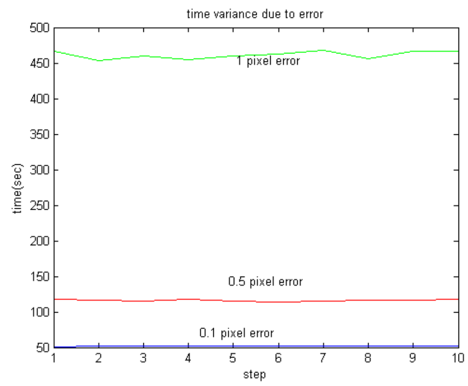

Fig. 8. Time consumption with respect to pixel error

5. CONCLUSION

In this paper Non-Dimensional star pattern recognition performance is studied. Although by increasing the total hardware and software accuracy, one can obtain more accurate bright point positions from centroiding algorithms but this overall improvement can increase the final production cost of a typical star tracker. By using an accuracy of 0.1 pixel size of centroiding algorithms, it has been shown that the non-dimensional algorithm is robust and conserves the trade-off between required accuracy and hardware capacity.

6. ACKNOWLEDGEMENT

The authors would like to thank Dr. Malak Samaan for his sincere cooperation in answering our questions regarding to his method, Mr. Mahdi Hassani for providing the sky simulation results of Figures 1to3.

Table 4. Time variance with respect to pixel size error

CPU time (sec)			
1	0.5	0.1	Pixel error / Runs
466.06	117.26	51.32	**1**
453.11	116.42	51.45	**2**
460.05	115.44	51.84	**3**
455.09	117.2	51.89	**4**
459.24	115.52	51.52	**5**
462.78	114.35	52.2	**6**
467.23	115.71	51.63	**7**
455.4	115.83	52.16	**8**
466.54	116.54	51.44	**9**
466.06	117.26	51.37	**10**

REFERENCES

[1] Craig L. Cole, John L. crassidis, **"Fast Star-Pattern Recognition Using Planar Triangles"**. J. Guid. Control. Dynam, V.29,pp. 64-71, 2006.

[2] Benjamin B. Spratling, IV and Daniele Mortari **"A Survey on Star Identification Algorithms"** *Algorithms 2009*, ISSN 1999-4893.

[3] Malak A. Samaan, Daniel Mortari and John Junkins, **"Nondimensional star Identification for uncalibrated star cameras"**, *The Journal of the Astronautical Sciences,* Vl.54, No.1, Januray-March 2006.

[4] Michel John Swanzy **"Analysis and demonstration: Aproof of concept compass star tracker"** *Master thesis,Texas A&M university* December 2006

[5] Eisennman, Carl Christian Liebe **"The Advancing State-of-the-art in Second Generation Star Trackers"**

[6] Malak Samaan, S. Theil **"Development of a low cost star tracker for shefex mission"** *Aerospace science and technology* 2011.

[7] D. Mortari, **"Search-less algorithm for star pattern recognition,"** *J. Astronaut. Sci*, vol. 45, pp. 179–194, 1997.

[8] D. Mortari, **"K-vector range searching techniques, "** *Adv. Astronaut. Sci*, vol. 105, pp. 449–464, 2000

[9] Perryman M.A.C., Lindegren L., Kovalevsky J., Hog E., Bastian U., Bernacca P.L., Creze M., Donati F., Grenon M., Grewing M., van Leeuwen F., van der Marel H., Mignard F., Murray C.A., Le Poole R.S., Schrijver H., Turon C., Arenou F., Froeschle M.,

[10] Petersen C.S., **"The Hipparcos Catalogue"** 1997

[11] Daniele Mortari, Malak A. Samaan, Christian Bruccoleri, and John L. Junkins "The Pyramid Star Identi¯cation Technique" Tchnical Report, Texas University

[12] M. Samaan, C. Bruccoleri, D. Moratri, J.Junkins **"Novel techniques for creating nearly uniform star catalog"** *paper AAS 03-609*.