New Star Identification Algorithm using Labelling Technique

Sangkyun Kim[a,*], Mengu Cho[a]

[a]*Laboratory of Spacecraft Environmental Interaction Engineering, Kyushu Institute of Technology, Kitakyushu-shi, Japan*

Abstract

A new star identification algorithm is proposed for the attitude determination of a star sensor in the lost-in-space case, where prior attitude information is not available. The algorithm is based on a labelling technique, which uses label values to represent each group of stars. Using label values, multiple stars are simultaneously identified without repetition of search work. This labelling algorithm allows for a fast identification speed with efficiency, and provides the capability of more reliable identification by redundant confirmation. The proposed algorithm was verified by simulation study under various conditions.

Keywords: Star identification, Labelling

1. Introduction

Star sensors are the most accurate sensors for spacecraft attitude determination and are becoming essential devices for many space missions, which require accurate attitude control [1, 2]. A star sensor estimates the attitude by combining the star vector information of captured images and the matched vector information of stars on reference data stored in memory. A sequence of procedures is required in order to estimate the attitude information from the captured image of white dots on a black background, as shown in Fig. 2. The

*Corresponding author
Email address: kim.sangkyun571@mail.kyutech.jp (Sangkyun Kim)
URL: http://laseine.ele.kyutech.ac.jp/english/ (Sangkyun Kim)

star positions on the image are extracted from the image at first [1], and their corresponding vector information is calculated with respect to the sensor frame. After that, star identification is required to match a number of stars in the image with stars in the stored reference data. After the identification work, the attitude can be estimated using more than two identified star vectors between two different frames, the sensor frame (measured vector information) and the inertia frame (reference vector information in memory) [3, 4]. Among those procedures, star identification is usually the most complicated and time consuming. Especially, star identification becomes more difficult in the lost-in-space case, with no priori information about the star sensor attitude.

For the lost-in-space case, many excellent algorithms have already been introduced [5, 6]. Those algorithms make use of various methods and can be categorized into angle-based algorithms and pattern-based algorithms, including experimental methods [7, 8]. However, all algorithms have the same goal, namely, to extract certain aspects from the sensor image and find uniquely identified stars from the data stored in memory. Each algorithm has its own advantages. Angle-based algorithms mainly use the precise angular distances between stars. Among them, the representative one is the triangle algorithm [9], with which the brightest stars in the image are chosen to make a triangle. The three sides that form the triangle and the brightness of each star are then compared with those in the database. Other algorithms have also been developed in this category and show improved robustness and speed compared to the triangle algorithm. The most successful of those is the pyramid algorithm because of its speed and robustness[10, 11]. Pattern-based algorithms use a strategy that locates the most similar image by comparing the entire image pattern with pattern data stored in memory. The grid algorithm is one of the most famous algorithms in this category because of its intuitiveness and performance [12]. Some algorithms make use of artificial intelligence, such as neural network algorithms and genetic algorithms [13, 14, 15]. And the singular value decomposition method has also been proposed for use with star identification [16].

Both angle-based and pattern-based algorithms are already being used in orbit and show their relative advantages. Generally, angle-based algorithms are faster when a very accurate optic system is available. And pattern-based algorithms are more robust with regard to individual star position errors because an entire star pattern is used for identification. However, angle-based algorithms provide robustness with regard to false stars because they use only some of the stars on the image. False stars on a star sensor image include planets such as Neptune, debris in orbit, or broken pixels on the image sensor [17]. Such false stars are not merely annoying but represent a very critical problem for star identification because a false identification result is more dangerous than an empty result. The pyramid algorithm is one of the most famous angle-based star identification algorithms for that reason. It provides very quick identifications because of the K-vector technique, and improved robustness against false stars is also guaranteed. The improved robustness comes from multiple triangle comparisons. The best way to avoid the false star problem is to check as many multiple cases as possible. However, the redundancy in comparison takes more time, and the limited computation power of an onboard star sensor computer should be considered. Moreover, star identification is not the only work a star sensor is tasked with in orbit. The credibility of results is important for a star identification algorithm, but the time consumption should also be minimized. That is why faster star identification algorithms have been proposed in many studies to date. One of the useful approach is to accelerate the efficiency by the simultaneous identification with the large database, And, Planar Triangles algorithm characterizes multiple stars using its area and moment value to accelerate the identification speed[18].

In this study, a new star identification algorithm is proposed for faster star identification. The proposed algorithm is based on a labelling technique in order to simultaneously identify multiple stars without a repeat of search process. The labelling technique uses the simple idea of having a label value that defines a group of stars. The label values are stored in the star sensor memory in the sorted database format. Label values are calculated from the ratio values between the distances of star positions on the image. Because multiple stars are represented by one label value, finding one label value on the database identifies multiple stars at the same time. And, this labelling technique shows improved robustness against false stars and focal length errors. A labelling algorithm does not use the brightness information of stars for better robustness. It is very tempting to use star brightness for a quicker identification speed; however, those values are not fixed on both sides of the image sensor and in the stored database. Variable stars are good examples, and with other stars it is also difficult to accurately define instrumental magnitude. Excluding star brightness provides greater robustness, and this is also very convenient for night sky viewing tests on the ground.

2. Labelling

The labelling technique proposed in this document is based on the uniqueness of the label value for a group of stars. At first, a group of stars is selected and the distances between each star on the image are calculated. When s stars are selected, ${}_SC_2$ combinations appear for the group. The distance combinations reveal the uniqueness of the group of stars. And the distances are not directly used for the calculation of the label value. The ratio values are calculated using the largest distance, and it provides greater robustness against focal length error in orbit. When a group is composed of a larger number of stars, the label value has a stronger uniqueness. However, the limitations of the computing resources should be considered for a practical star sensor. Also, the optical system should be able to support the total number of available stars. Three stars begin to show a unique label value when the measured position is precise enough, but it is very risky to rely on precise position information. Actually, there are many tolerances with regard to the measurements of star position on the image, such as optical distortion, environmental changes, and so on [19]. Four stars have six distances, and usually sufficient uniqueness is guaranteed, as suggested in the earlier discussion of the pyramid algorithm. Five or six stars will give stronger unique label values for a group of stars and would be a good choice if the star sensor has sufficient computing resources and a powerful optical system.

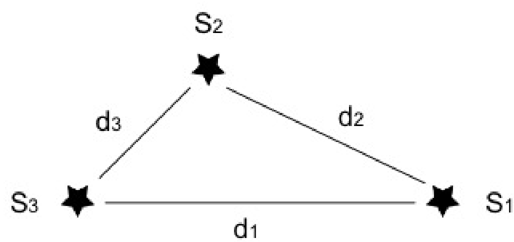

Fig. 1. Stars and distances on the image

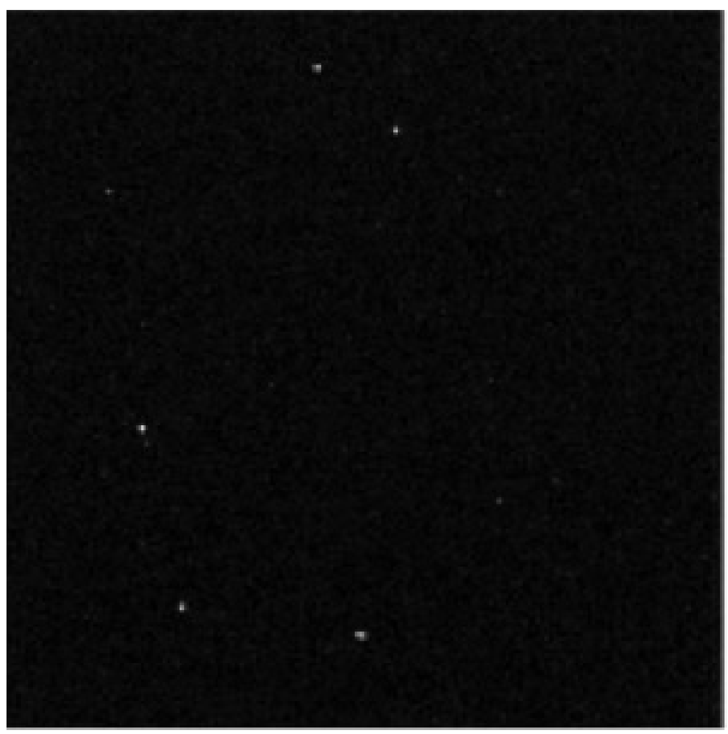

Fig. 2. Example of actual star image from star sensor

2.1. Sub-label

For a label, sub-labels need to be calculated first. A sub-label is a ratio value between each distance and the largest distance in the combination of star positions on the image. A sub-label is calculated to acquire a value between 0 and 1, as in the following equation when the number of stars is s, and the longest distance is d_1:

$$l_{1i} = \frac{d_i}{d_1}, i = 2..n, n = \binom{s}{2} \tag{1}$$

Those sub-labels are a set indicating the characteristics of the group of stars. However, it is not convenient to compare each sub-label to find matched stars on the database.

2.2. Label

The sub-labels are combined to create a single label value to characterize the whole group. When sub-labels are combined, it is important that they not be mingled with each other to preserve the information of each. Because of that, each sub-label is rounded after multiplication with some scale number to create a small-digit number. Then, a label is calculated by the summation of these sub-labels with a different digit order using the scale number. The following equations explain how to calculate the label value:

$$L_{1i} = [\alpha l_{1i}], \alpha = 10^r \tag{2}$$

$$L = \sum_{i=2}^{n} \alpha^{n-i} L_{1i} \tag{3}$$

The proper scale number of r should be carefully selected. This will depend on the resources available in the star sensor, such as the accuracy of the optical system, available memory, and computational power. Unlike sub-labels, label values do not need to be stored with precise values. When a label is calculated with the measured star position, its value has tolerance caused by the measurement error. Actually, several groups of stars can have the same label value because of the similar sub-labels. In section 4, we will explain how to identify stars from multiple candidates with the same label value.

Table 1: Example of label calculation

items	values
s, Number of stars	3
n, $\binom{s}{2}$, Number of combinations	3
d_1, d_2, d_3, Distances between stars with [pxl] of image	452, 385, 276
l_{12}, l_{13}, Sub-labels	0.85177, 0.61061
r, Scale number	2
L, Label	8561

A hypothetical example can be given to explain how to calculate the label value when a group has three stars, as in Fig. 1. Table 1 shows the label value when the three stars have assumed distances and parameters for the label calculation. Assuming the three stars of the group have distances represented in pixels as 452, 385, 276, the sub-labels become 0.85177 and 0.61061. Because there are two scale numbers, 85 (from 0.85177) and 61 (from 0.61061), they are combined to create a label value of 8561. That label value represents this group of three stars.

2.3. Data structure in memory

The calculated labels are stored in the star sensor memory as a reference database. The progress being made in electronics is dramatic, even now, and the memory capacity of onboard computers is increasing. However, there are still restrictions when compared to ground systems. Therefore, a rule to minimize the memory usage is necessary. A label is stored in memory in the following simple form:

$$Label, ID_1, .., ID_s \tag{4}$$

Star IDs should be kept in sequence by a determined rule. The attached distances between stars are used for this purpose. Each star is a specific distance from the other stars in the group, the summation values of the distances are used to sort the IDs, and the IDs are stored in descending order by their summation values. Any unique identification number can be used for a star ID. In this study, Hipparcos catalog IDs are used to identify the stars. The calculated label values are sorted to minimize the search work. Usually, a database with label values exhibits a curve, as shown in Fig. 3.

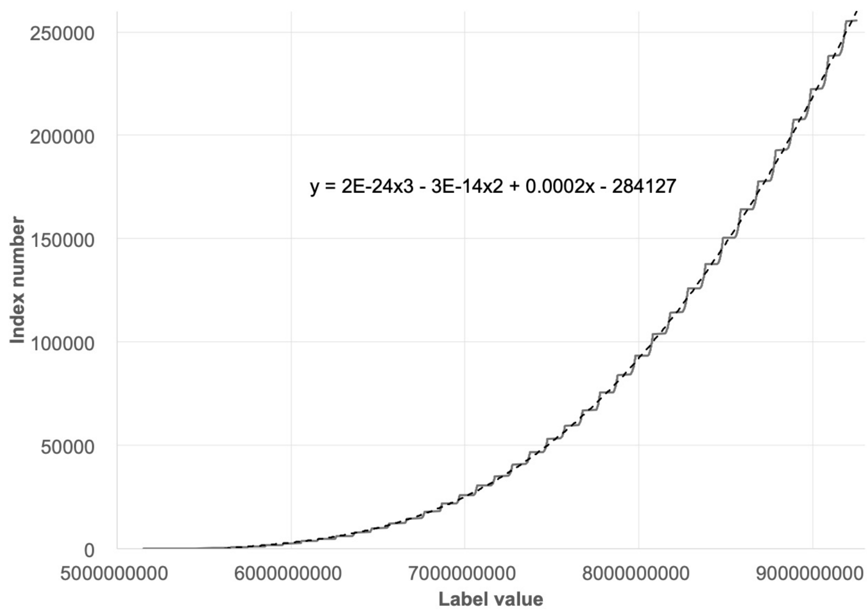

Fig. 3. Label value, Index number, and Trend line

In the sorted label values database, there are many ways to accelerate the speed in finding a specific label. In this study, a polynomial trend line equation is used for that purpose. Using that equation, the near index number is calculated from the target label value.

3. Onboard database generation

There are several open star catalogs with which to create a database for reference. For the database of this study, the Hipparcos star catalog constitutes the base set of star groups. Several preprocesses are required to make a suitable database from the catalog. At first, it is necessary to remove stars that are too dark because they are of a greater visual magnitude than the sensitivity of the considered star sensor. When the star sensor takes an image of stars, stars that are too close to one another appear as one because of optical limitations. Those close stars have to be merged as one in the database. After preprocessing, a number of stars are selected from the catalog assuming a boresight direction with the FOV of the star sensor. The number of stars depends on the available computational power of the database-generating computer. In this study, eight stars in the FOV are selected to make a database, and four-star combinations are made from them. Generally, brighter stars of lower visual magnitude are selected because they are easier to detect in the image. After selection, combinations of distances, sub-labels, and the label are calculated. Then, the label value and IDs are stored in the database. After the database has been generated for all combinations in the FOV, the boresight moves 1.0 deg, and the database generation is repeated again until all area of the celestial sphere have been scanned.

When the entire celestial sphere area has been scanned, the database is sorted by label values in ascending order and is stored in memory, keeping the IDs in sequence. In this study, four stars are selected to make combinations from the eight stars, and the scale number is 2. The darkest star of the database has a visual magnitude of 5.2 and the FOV is assumed to be 24 deg. For the whole area of the celestial sphere, 258,474 combinations were created. Each combination has a label value of 8 bytes and four IDs of 8 bytes. As a result, each combination requires 16 bytes, and 4.1 MB of memory capacity is required to store the entire database. Also, this labelling technique needs a star catalogue for the final confirmation of the minimum angle error. In this study, a star catalog with 2055 stars is used, requiring 53 KB of memory. Such a huge database size is one major disadvantage of labelling algorithms as like other algorithms which have same approach[18]. Figure 4 shows how much memory is required for each identification algorithm, pyramid, grid, and labelling. Under the same conditions, the pyramid algorithm and the grid algorithm need around 0.2 MB only.

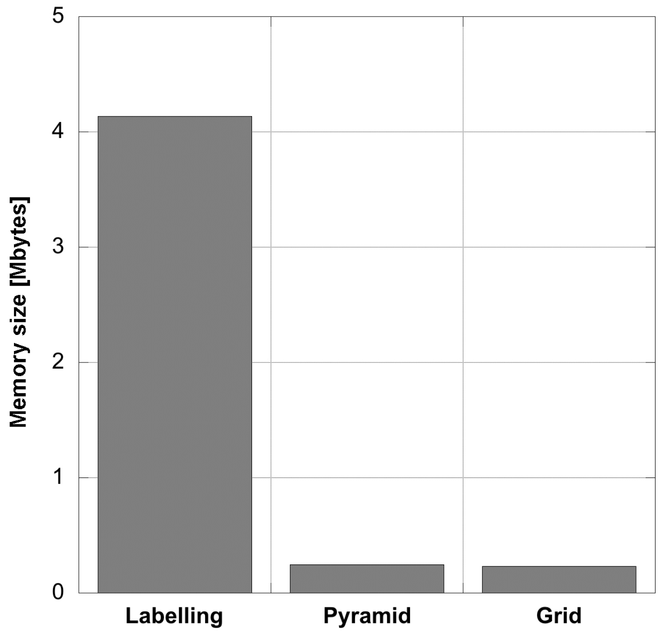

Fig. 4. Database size between Labelling, Pyramid, and Grid algorithms

4. Star Identification Using Labelling Algorithm

The labelling algorithm uses a suitable identification flow to search the labels on the database with efficiency. This flow is shown in Fig. 5. First, several bright stars are chosen and the the label value is calculated. Second, using the polynomial trend line equation, the nearest index is calculated. Third, candidates of the same label on the database have to be found, and the same label value usually appears on the database within several search steps from the nearest index. It is easy to expect that only one unique data has same label value but, in reality, many candidates have the same label value. The label value absorbs the tolerance of star positions, and multiple candidates appear with the same label value. Moreover, even in the case of a unique candidate with the same label value, the angle error between the measured star positions and the star position in the star catalog has to be checked for the final confirmation. In the case of multiple candidates, a candidate with a minimum angle error must be found. If the angle error is small enough, then the stars are identified.

The angular distances between stars can be calculated by the dot product of each unit vectors of stars. The data in the star catalog have angle information in inertia frames, which can be converted to unit vector information. In the case of the stars in images, the unit vector is calculated from the measured star position and the focal length of the star sensor. The angle error is calculated by the summation of the errors between the measured angles (θ_{mea}) and the reference angles (θ_{cat}) in the catalog, as follows:

$$\theta_{err} = \sum_{i=2}^{n} |\theta_{i,mea} - \theta_{i,cat}|, n = \binom{s}{2} - 1 \tag{5}$$

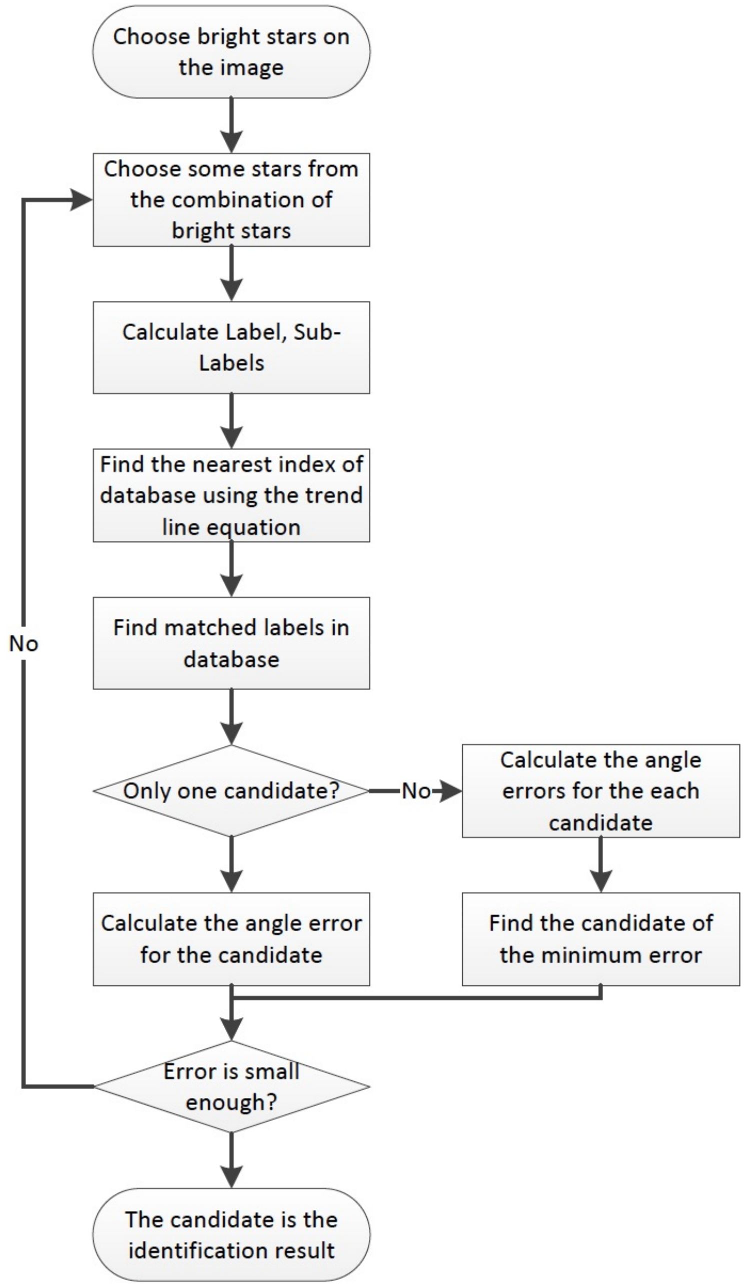

Fig. 5. Flow chart for star identification using labelling algorithm

5. Simulation Result

Simulations were repeated to confirm the performance of the proposed labelling algorithm under various conditions. The configurations of the star sensor for the simulation are based on the actual star sensor under development. The optical system has a 24 x 24 degree FOV with an image sensor of 1200 x 1200 pixels. The minimum sensitivity of the sensor was set to the apparent stellar magnitude of 5.2. The simulation image has an intentional star position error with one sigma value of 38 arcsec with assumption of a half pixel random error.

For this study, we chose the pyramid algorithm and the grid algorithm to compare results because those algorithms are one of the most successful star identification algorithms representing the angle-based algorithm and the pattern-based algorithm. They have efficiency, robustness, and most importantly they have already shown excellent practical performance in orbit. It is not easy to accurately compare identification performance between different algorithms. Even when the same hardware is used for the algorithms, they have their own characteristics and their parameters should be optimized for the best performance. For proper comparison, the parameters are optimized with caution for all algorithms, and 100,000 simulated images were applied with randomly chosen boresight directions. The simulation is performed on a personal computer for quick development. Its main processor is a 2.5-GHz Intel Core i7, with 8 GB of memory, using the Microsoft Visual Studio 2013 C# platform.

5.1. Identification speed

First, the average values of the identification speeds for each algorithm are compared. Table 2 shows the results, and Fig. 6 shows the results with % values. In simulation, the labelling algorithm needs just 21% of the time required by the grid algorithm for identification under the same conditions. The pyramid algorithm is faster than the grid algorithm, however it needs 84% of identification time. This fast identification speed of labelling algorithm is very important in many ways. The faster identification speed makes it possible to

Table 2: Execution time for the identification

Identification algorithm	Execution time [msec]	Execution time [%]
Grid algorithm	3.69	100
Pyramid algorithm	3.09	84
Labelling algorithm	0.76	21

output attitude information more frequently, and the attitude control accuracy can be improved. Also, the fast data rate provides the additional capability of using the star sensor for secondary functions such as angular velocity estimation.

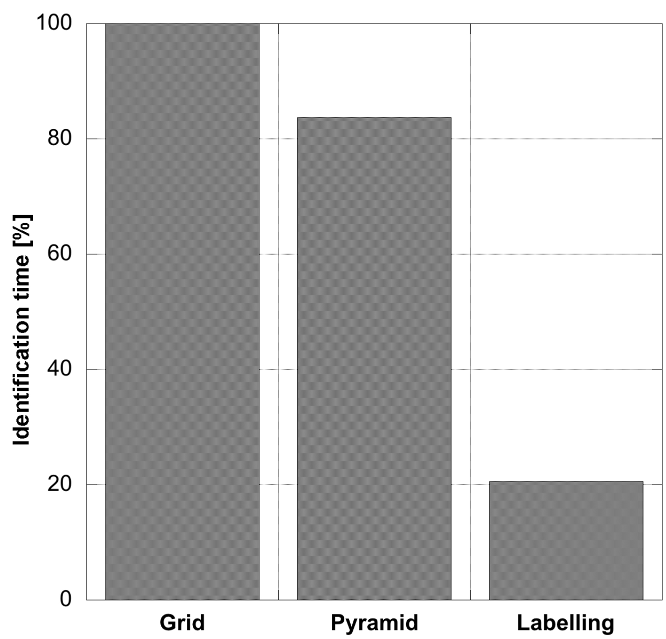

Fig. 6. Execution time between Grid, Pyramid, and Labelling algorithms

5.2. Robustness against focal length error

One of the major advantages of the labelling algorithm is its robustness against focal length error. Actually, focal length is a very critical parameter in star sensors because it determines the measured star vector information.

Usually, the star sensor optical system is designed to minimize focal length tolerance, but it still has some risk of error, such as temperature changes, launch vibration, and so on. The labelling algorithm uses the ratio values from the distances between star positions, and it shows really strong robustness when its focal length has errors. In Fig. 7, the labelling algorithm shows no failure rate until the focal length has 1% maximum error. And the labelling algorithm has no significant time changes for the identification, as shown in Fig. 8. The focal length error is enough to create serious problems in the pyramid algorithm, which uses the angular distances. The pyramid algorithm shows a poorer success rate than 40%, and its identification time rapidly increases with focal length errors. The grid algorithm of pattern-based algorithm has small changes of the identification time, however it shows the same problem of success rate with the change of focal length too, because its measured pattern has big changes by the error of focal length.

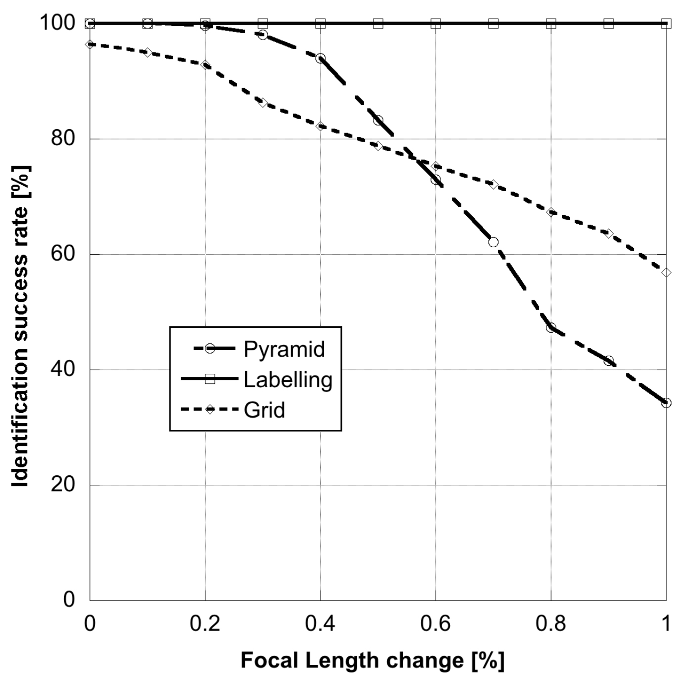

Fig. 7. Success rate between Grid, Pyramid and Labelling algorithms by FL change

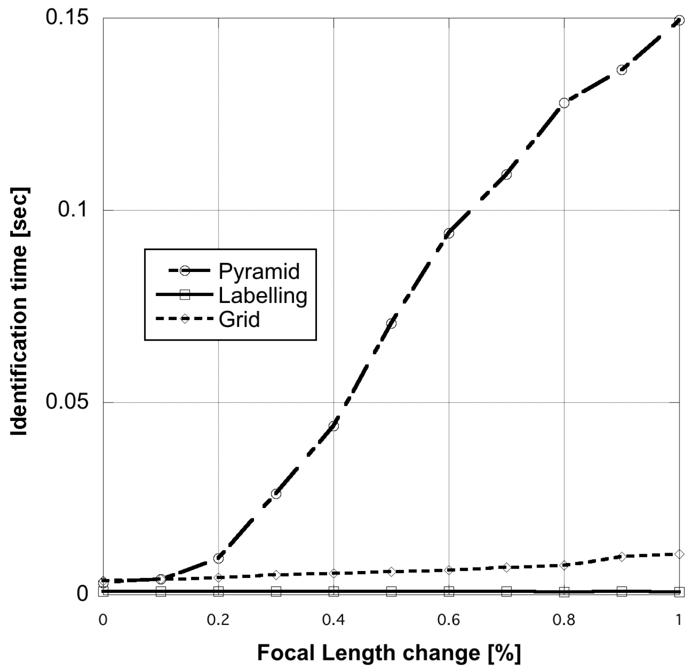

Fig. 8. Identification time between Grid, Pyramid and Labelling algorithms by FL change

5.3. Identification time with false stars

Basically, a long identification time is required to avoid false stars because of redundant confirmation. The problem is that a long identification time is not easily supported by the limited computational power of a star sensor. The labelling algorithm simultaneously identifies multiple stars within a short time when the label value is confirmed. It offers advantages to robustness against false stars. Figure 9 shows the identification speed with the number of false stars. Each algorithm needs a lot of identification time as the number of false stars increases. However, the identification time of the labelling algorithm shows a smaller time consumption, and it is able to deal with more false stars within the same identification time required by the pyramid algorithm. The grid algorithm has slower identification speed compare to the labelling algorithm when the number of false star is zero or one, however it has very small changes of identification time against the number of false star because it uses an entire star pattern. The disadvantage of grid algorithm is the success rate of identification when the image has false stars as shown in Fig. 10. The two angle-based algorithms, the labelling algorithm and the pyramid algorithm, keep the success rate more than 95%, however the identification success rate of grid algorithm become rapidly poor to 85% when the number of false stars is increasing.

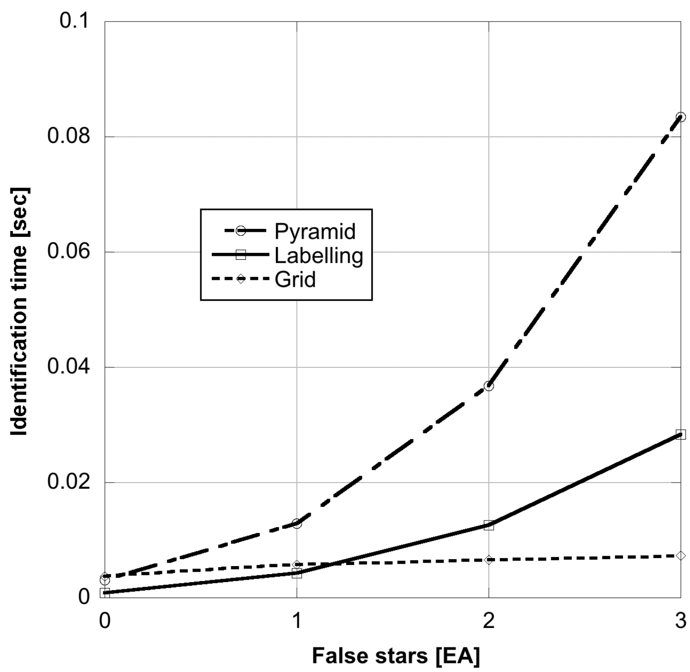

Fig. 9. Identification time with false stars

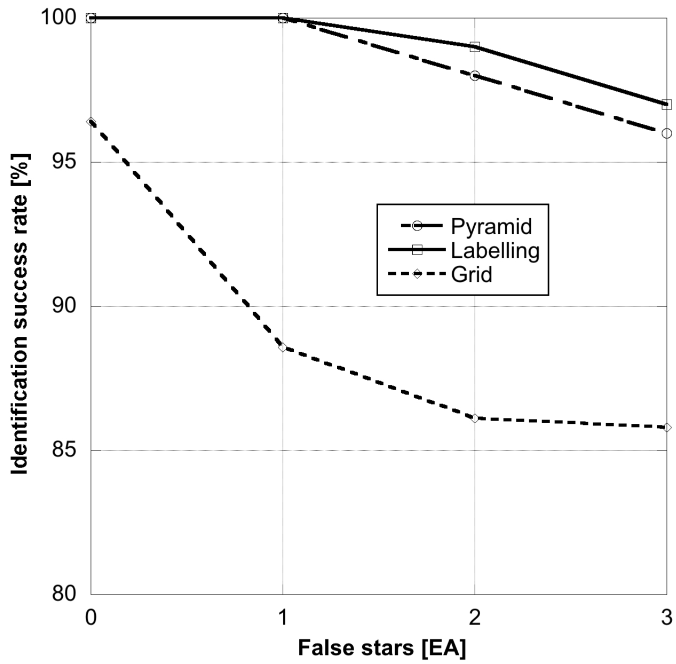

Fig. 10. Identification success rate with false stars

6. Conclusion

In this study, a new star identification algorithm was proposed using a labelling technique. This technique contains several important new features. The first is its ability to identify multiple stars with one label value of simple calculation, instead of repeating search work to find suitable star pairs. The label database is built a priori for some given working magnitude threshold and the FOV of the star sensor. Essentially, the database is a structural of all groups of stars that could possibly fit in the star sensor from the celestial sphere. The groups of stars are sorted in ascending order of label values on the database. And the polynomial equation of the trend line gives the nearest index value to the target label value. Sometimes, the label value represents several groups of stars, not a unique candidate. When the label has several candidates for the same label value, the angle error is used to select the candidate with the least error.

The second new feature is to use the ratio of distances, not to use distances themselves. Distance measurement has some error in a practical star sensor because of optical errors and environmental changes. The labelling technique provides with improved robustness against those, especially for changes in focal length. Also, this algorithm uses only distance information for the identification work, and does not use star brightness information. Even though it creates some extra complexity for the search work and data structure, it provides advantages for robustness in avoiding brightness variance, and convenience for night sky view testing. The simulation study confirmed its fast identification speed and its robustness was also confirmed for the focal length error and false stars.

References

[1] C. Liebe, Accuracy performance of star trackers-a tutorial, IEEE Transactions on Aerospace and Electronic Systems 38 (2) (2002) 587–599.

[2] A. Davies, A. Holt, Use of autonomous star trackers in modern attitude and orbit control systems, in: 5th ESA International Conference on Spacecraft Guidance, Navigation and Control Systems, 2002.

[3] M. Shuster, S. Oh, Three-axis attitude determination from vector observations, Journal of Guidance, Control, and Dynamics 4 (1) (1981) 70–77.

[4] D. Mortari, Euler-q algorithm for attitude determination from vector observations, Journal of Guidance, Control, and Dynamics 21 (2) (1998) 328–334.

[5] K. Ho, A survey of algorithms for star identification with low-cost star trackers, Acta Astronautica 73 (2012) 156–163.

[6] B. B. Spratling, D. Mortari, A survey on star identifications algorithms, Algorithms 2 (2009) 93–107.

[7] P. S. L. Y. L. B. Yoon, H., H. Lee, New star pattern identification with vector pattern matching for attitude determination, IEEE Transactions on Aerospace Electronic Systems 49 (2) (2013) 1108–1118.

[8] L. Y. Yoon, H., H. Bang, New star-pattern identification using a correlation approach for spacecraft attitude determination, Journal of Spacecraft and Rockets 48 (1) (2011) 182–186.

[9] C. Liebe, Star trackers for attitude determination, IEEE AES Systems Magazine (1995) 10–16.

[10] D. Mortari, B. Neta, K-vector range searching techniques, in: 10th Annual AIAA/AAS Space Flight Mechanics Meeting, American Astronautical Society, 2000.

[11] J. J. L. Mortari, D., M. A. Samman, Lost-in-space pyramid algorithm for robust star pattern recognition, in: Guidance and Control Conference, American Astronautical Society, 2001.

[12] C. Padgett, D. K. K., A grid algorithm for star identification, IEEE Transactions on Aerospace and Electronics Systems 33 (1997) 202–213.

[13] K. T. Kim, H. Bang, Reliable star pattern identification technique by using neural networks, Journal of the Astronautical Sciences 52 (2004) 239–249.

[14] J. Hong, J. A. Dickerson, Neural-network-based autonomous star identification algorithm, Journal of Guidance, Control, and Dynamics 23 (2000) 728–735.

[15] Z. F. Li, L. H., T. Lin, An all-sky autonomous star map identification algorithm based on genetic algorithm, Opto-Electronic Engineering 27 (5) (2000) 15–18.

[16] K. H. Y. Juan, J. N., J. L. Junkins, An Efficient and Robust Singular Value Method for Star Recognition and Attitude Determination, NASA, nasa/tm-20030212142 Edition (2003).

[17] J. L. Lauer, M., S. Kielbassa, Operational experience with autonomous star trackers on esa interplanetary spacecraft, in: 20th International Symposium on Space Flight Dynamics, 2007.

[18] J. L. C. Craig L. Cole, Fast star-pattern recognition using planar triangles, Journal of Guidance, Control, and Dynamics 29 (1) (2006) 64–71.

[19] P. J. M. A. J. I. K. S. J. G. Muruganandan, V. A., Development of arc second pico star tracker(apst), in: The 2016 Asia-Pacific International Symposium on Aerospace Technology, 2016.