Star Identification Using Delaunay Triangulation and Distributed Neural Networks

Saeideh Sadat Miri and Mohammad Ebrahim Shiri

Abstract—**In this paper, a new star recognition method based on Delaunay Triangulation (DT) algorithm and distributed neural networks was proposed to decrease the search space and increase the star recognition success rate for star sensors. It computed the DT of all stars in catalog. Then, it employed this method on stars in the captured image. It compared the generated DT graph of stars in the image with the catalog graph by using Relaxation By Elimination (RBE) method. RBE used an 'inverted' relaxation labeling method that found a good match of the input graph with the catalog graph. RBE was implemented by Correlation Matrix Memories (CMM). CMM was a kind of neural networks to store the constraints between the nodes of the graphs being searched. This algorithm relied on the positional relations and angular distance of stars in each triangle that is made by DT. The experimental results showed that when the position error was about 120 arc seconds, the identification success rate of this method was 89% while the identification method based on the matching probability was only 77%. In addition, the storage requirement of the algorithm was small.**

Index Terms—**Star identification, delaunay triangulation, graph matching, relaxation by elimination.**

I. Introduction

All spacecraft need to obtain their precise attitude. There are several sensors to determine the attitude relative to reference objects. Star sensors are the most effective among them, acquiring the attitude information by star map-processing methods and attitude determining algorithms. An autonomous star recognition method is one of the core technologies of spacecraft attitude measurements by a star sensor. According to the original star map data, obtained by the star sensor, the identification method transforms, transfers or combines the star points, which are included in a star map, and comes up with the characteristic information which reflects this star map as far as possible. Then, the information is compared with the Guidance-star database in order to complete the identification of the star map. The desired accuracy, functionality and reliability need to be considered when choosing a star identification method. Star identification algorithms are of two types: first, "lost-in-space" algorithms, in which no information regarding the attitude of the spacecraft is available, second, recursive algorithms, in which some information regarding the attitude is available. These techniques typically use inter-star angles (the angle between the line-of-sight of two stars from the perspective of a camera) and star the brightness and make some computations on these values to distinguish stars [1]. In this paper, an approach is presented with regard to the first type. In the last decade, many star map recognition methods were presented, which can be divided to three groups:

- The graph theory-based algorithms: they commonly create triangles from stars, calculate inter-angles of stars and compare these angles to catalog [2]-[6]. Some of these algorithms generate a vector of properties from stars, based on angular distance, magnitude, etc. to make a comparison. In [7], the k-vector approach for accessing the star catalog is presented. It provides a search-less means to obtain all the cataloged stars from the whole sky which could possibly correspond to a particular measured pair, in case the measured inter-star angle and the measurement precision are given. The Pyramid logic is built on the identification of a four-star polygon structure.
- Primary star-based algorithms: they commonly select the brightest star as primary star. For example in [8], an identification algorithm based on the matching probability is proposed. The brightest observed star is considered as the primary star and the radial geometry pattern is constructed by linking the primary star to other adjacent stars, then the pattern is matched with star database. The star which is most frequently visited is regarded as the correspondence of the primary star.
- The intelligence-based algorithms such as neural network and genetic algorithms: they commonly use pattern recognition methods. For example in [4], an approach based on Counter Propagation Neural Network is presented. In this method, input vector of network is made by the brightest stars as Guide stars and the angular separation of stars in a specified radius of guide stars as Secondary stars.

Generally, most of these algorithms select some of stars from the image, so they need a selection method. There is the possibility of choosing some non-star objects like planets, noise, etc. from the image. Therefore, sometimes, the selected stars may not be the correct stars. It means that these algorithms fail in this case. In the present paper, all stars are used for comparison. So, this algorithm does not fail in most of these cases.

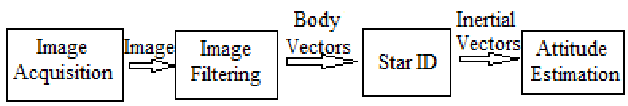

Fig. 1. Star- tracker image-based attitude determination flow

Attitude determination is done in several stages. As depicted in Fig .1, Star ID is one step in the process of

Manuscript received April 3, 2012; revised March 18, 2012.
The authors are with the Amirkabir University, Tehran, Iran (e-mail: miri@aut.ac.ir, shiri@aut.ac.ir).

determining the attitude of a spacecraft. First, the star image is acquired by a stellar camera with known calibrated parameters. Second, in Image Filtering step, stars are concentrated by a centroiding method. Stars are defocused in the image and spread on various pixels while being described in sub-pixel accuracy. Then, angular distances of pair stars are calculated based on the calibration parameters as Body Vectors. Third, in Star ID step, by comparing these body vectors of the stars with the known ones stored in a star catalogue, the colinearity equation of every star can be established. Fourth, camera and spacecraft attitudes in the celestial sphere coordinate are calculated based on the position of identified stars.

In this approach, the positional relations of stars were relied on. Positions of stars are not fixed; they have a little movement, so angular distances of pair stars change all the time and we cannot use angular distance only. Some methods fail because of increasing the position error using this feature. In the present method, however, distances were used for reducing the search space, then, correct stars were selected based on the relations in DT. This algorithm is capable of detecting stars correctly while DT of all stars does not change.

In the following sections, star pattern identification algorithm is described in detail. The new algorithm is coded and subjected to numerical simulation tests along with a reference algorithm based on matching probability. The results are compared in order to determine the advantages and limitations of the proposed new techniques.

II. The Star Recognition Method Based on DT Algorithm

In this paper, a DT algorithm is introduced to solve the problem of star identification. First, the Delaunay Triangulation (DT) of the set of star points was calculated based on their locations in the image (calculated locations by centroiding algorithm). For each edge in the triangles of DT graph, the angular distances of stars were computed. Delaunay Triangulation of stars in catalog was also generated because the graph of image was comparable to the catalog and this graph was saved in a star database.

After calculating the distances, a weighted graph was created. Weights of edges in two graphs were matched by a tolerance, and then the consistency of vertices in the candidate's edges was confirmed. If the algorithm assigned two IDs to some vertices, then the vertex would be checked in other triangles until just one ID was detected.

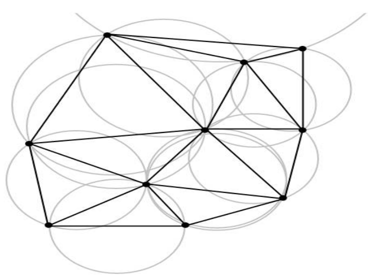

Fig. 2. A Delaunay triangulation in the plane with circumcircles shown

A. The Principle of the Delaunay Triangulation Algorithm

Delaunay Triangulation (DT) is a computational geometry algorithm which is used in many applications. A DT for a set P of points in the plane is a triangulation DT (P) such that no point in P is inside the circumcircle of any triangle in DT (P). Delaunay triangulations maximize the minimum angle of all the angles of the triangles in the triangulation; they tend to avoid skinny triangles. A sample of a DT is shown in Fig. 2. For a set of points, the Delaunay triangulation is unique [5]. So, the uniqueness property of DT was used to compare the stars with the database because a smaller graph (a graph DT of image) must be found exactly in the greater graph (a graph DT of sky).

The definition of the Delaunay Triangulation is based on the Voronoi diagram through the principle of duality. Voronoi diagrams with their application in a vast amount of fields are described in detail, mentioning a lot of original references in [9].

Voronoi Definition: Let $P=\{p_1,\cdots,p_k\}$ be a finite set of points in the n-dimensional space R^n and their location vectors $x_i \neq x_j \ \forall \ i\neq j$. For example in the 2-dimensional space, every point has two elements of location vector (x_1, x_2) in a way that no points have the same vectors (x, y). The region given by (1) is called Voronoi region (Voronoi box) which is associated with p_i.

$$V(p_i) = \{x \mid \|x-x_i\| \leq \|x-x_j\| \ \forall \ j\neq i\} \tag{1}$$

Based on (1), every point in Voronoi region of point p_i has the shortest distance from x_i (location vector of p_i) in relation to other x_j(location vecor of other points in P). ν(P) in (2) is said to be the Voronoi diagram of P.

$$\nu(P)=Uki=1\ V(pi) \tag{2}$$

B. Reasons of using Delaunay Triangulation

Notably, DT is used for matching applications. The generated graph from DT has some properties which are useful for matching the stars in the captured image with the stars in the catalog. These properties are as follows:

1) In the plane, each vertex has, on average, six surrounding triangles. Therefore, memory requirements are almost 6n (n is number of stars in the entire sky).
2) DT of one point set is unique if 4 points do not occur in a circle; therefore, catalog graph and input graph are unique.

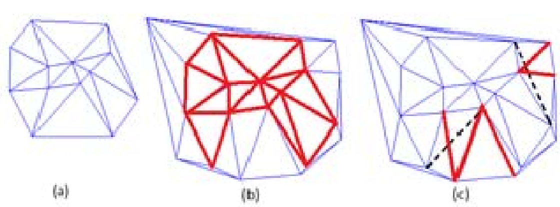

Fig. 3. (a) DT of a point set, (b) DT of other point set that contains the points in (a) and addition points. Bold edges show the similarity of DTs. (c) Some edges that link inner points to convex hull of points were deleted (Dashed edges) and some edges that link inner points to new points were added (Bold edges).

3) Incremental DT algorithm[1] definition: let P be a point set and D a DT of P. Insert one point to P and change the DT for new inserted point. By inserting new points, just regions corresponding to the adjacent new points will change. Stars in the image are a part of stars in the catalog; so, there are stars of catalog that are not in image. These stars are like additional points in the Incremental DT algorithm. If the DTs of image (Fig. 3-a) and catalog are calculated (Fig. 3-b), the difference between them will be:
   a) A part of convex hull of input graph is different with edges in catalog graph.(Fig. 3-c dash edges)
   b) Extra edges link DT of image to the DT of remaining stars in the catalog. The remaining stars exist in the catalog but does not exist in the image (Fig. 3-c bold edges).

Based on this reason, convex hull can be deleted from the edges' set and the inner edges can be used for matching graphs.

C. Graph Matching

In our method, we need to match DT of image to DT of catalog stars by using graph theories. Graph matching methods are almost NP-hard problems, but the graph of stars in catalog is fixed and planar, so the graph matching in our method is simple and need linear time to solve it.

D. Relaxation by Elimination

Our algorithm is based on an optimization technique, called Relaxation Labeling. The general idea behind relaxation labeling is to update individual mappings for graph vertices based not only on their feature measurements, but also by combining contextual evidence from their spatial environment. Relaxation by elimination (RBE) initially keeps all plausible solutions and iteratively removes implausible solutions below a defined similarity threshold. Consider an input graph Gi and a catalog graph Gc. A vertex vi of the input graph Gi is assigned a set of candidates $c_i = (c_{i1}, c_{i2}, ..., c_{i|ci|}, c_{ix} \in V_c, \quad x = 1, ..., |c_c|)$ that correspond to the currently plausible mappings from an input graph vertex to a set of catalog graph vertices[10].

To matching two very different graphs, the constraints on the nodes and arcs may only be similar. Thus the algorithm that matches one graph to another must not only find out which nodes and arcs match between the two graphs, but also find the ones that are most similar. One way to approach this problem is to use the idea of matching through the relaxation of constraints. In general, this approach attempts to find the minimum set of constraints needed to be withdrawn (relaxed) to allow the two graphs to match. These constraints can be unary i.e. single constraints such as found at a node (apparent magnitude of stars in nodes for example) and binary between two nodes such as is represented by an arc (angular separation between two stars for example). In RBE, the process of relaxation is achieved by elimination of constraints that are unsupported.

[1] A star identification method in [7] based on Incremental DT is presented; its method is Delaunay triangulation cutting algorithm. The result of this triangulation cutting method is unique from a random set of points. This triangulation algorithm was applied to identify star map for its particular character.

The match process aims to find a mapping between input graph Q and catalog stars graph D and thus find the match that minimizes the number of constraints that are violated. This represents typical data from an image analysis system, where the node information might be the type of vertex found at the node and the relations between these nodes are the distances between the edges. Note that the graphs contain some ambiguity in the node information meaning the relational information must be used to achieve a correct match.

The process of matching takes the following four stages:

1) Initialization by matching single nodes between graphs Q and D. this stage, takes each node in Q and attempts to matches to match this against graph D taking into account only the node constraints.

Computation of the relational support for the nodes. This stage uses the arc constraint information to calculate the support for each node and remove inconsistent matches. The process is summarized in pseudo code as: for each node X, in graph Qfor each node Y, in graph D that matches the node constraint on X for each arc A, from node Xcount C, the arcs from node Y with the same arc constraints as A.

1) The process of elimination of weakly support for the nodes. This stage is to eliminate the weakest supported nodes
2) The termination check. The process of elimination and calculation of support continues until no more nodes can be removed by elimination.

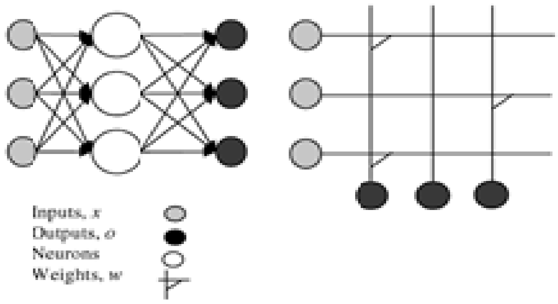

Fig. 4. Diagram of a CMM. On the left is a conventional diagram of a CMM, as a single layer network. Diagram On the right shows the same network as a matrix.

Repeat from 2 until termination.

These stages were described in [10] by an example completely. The next section describes how RBE is implemented on correlation matrix memories.

E. Binary correlation matrix memories

Correlation matrix memory is basically a single layer neural network that uses Hebbian learning. Fig. 4 illustrates a CMM as a binary matrix and a single layer network.

A CMM can be seen as an associative memory that, given a pattern vector x, recalls the associated pattern, o. the memory can be trained on large numbers of patterns to associate.

One of the benefits of RBE is that it can use the CMM to store the constraints in the graph and also for the mapping of each node in the graph Q to graph D. the following describes how the method is mapped onto CMMs.

The following sets out the steps in the process:

1) Step1: binary representation of node and arc constraints in Q and D. the value of a node (arc) constraint is represented as one bit set in a vector. This vector has a length equal to the number of possible values for the constraints in the system. To decrease the storage space, we use sparse vectors in order to large number of zero elements in vectors. To searching the position error and the apparent magnitude noise of stars, multiple elements of vector are set to 1. For example if node constraint is 2.34, the bit vector of this constraint will be a zero vector with 1 in 229th-239th element in this vector.

1) Step2: matrix of node and arc constraints

The node constraint vector and the arc constraint vector are combined into a correlation matrix by product.

2) Step3: conversion of the constraint matrix to a vector

3) Step 4: initialization of the support

   The initialization checks to see if the node constraints in the query match with any nodes in the catalog graph. If they are equal, then the relevant element in support matrix is set to 1.

4) Step 5: holding the support for each node

   The vectors representing the input query and the catalog can now be used to build the support list. Vector of each node in input graph is multiplied vector of each node in catalog graph. This number is support value. After calculation, support of last iteration is multiplied new support value.

5) Step 6: thresholding the support

   Matrix of support is thresholded to remove any value with poor support scores.

6) Step 7: termination of the process

   If the number of bits set fails to change in two iterations, then the process is terminated. Otherwise the process returns to step 5. By using this process, we can find best matches for any star. If in last iteration, some nodes remain with two or more matches, we can solve it by assigning stars to input graph and check relations and select correct stars.

III. Experiment Results

A. Experimental Data

The star catalogue Tycho-2 (J2000) [11]-[14] was obtained for the experiment. Its data types include star index, magnitude, the right ascension (hour, minute, second) and the declination (degree, arc minute, arc second), etc. There were 14581 stars in the initial star catalog. Considering the detecting ability of star sensor, the upper and lower limits of magnitude were 0 and 6.9 (mv). So, 8743 stars were left. The size of CCD was 512×512 pixels, and each pixel size was 13 um. In the star database, the angular distances' table had 17593 items and the basic star catalog had 8743 items.

B. Experiment and Analysis

In this paper, the FOV was set at 8° × 8°, and the resolution was set at 1024×1024 pixels. To validate this method, the fundamental star catalog was composed of 14,581stars, which were selected from the Tycho2 star catalog. The highest magnitude detected by the star sensor was 6.95. The range of star position error was set at less than 120" according to the precision of star centroid extraction. According to the FOV and the highest magnitude, It is known that there are about 20 stars in every star map.

For the every optic axis, there was a standard star map. And, for every standard star map (total of 10 star maps), the star position error, which was from 0" to 120", was added to the star map. There were 120 error points of star position for simulation. For every error point, 100 trials of the identification method were carried out and the mean value of the successful rate of identification was calculated as this error point's successful rate of identification. Finally, the simulation results of ten star maps were averaged according to the every error point. The average value of the successful rate is shown in Fig.5.

Fig. 5(a) shows the relationship between position error and success rate of the proposed identification. During the course of simulation, compared with the star recognition based on Delaunay cutting algorithm [7] (b) and the matching probability [8] (c), the results of simulation are as follows (the Delaunay cutting data and matching probability data were originated from reference [7]).

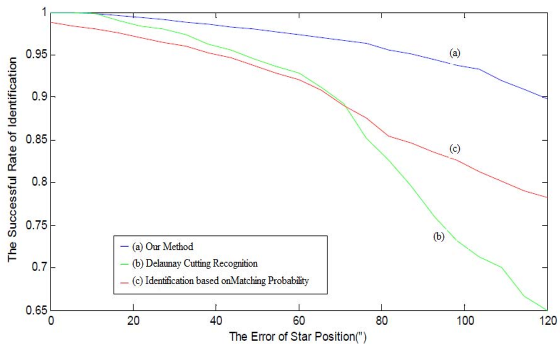

Fig. 5. Comparison under different position error

As seen in Fig. 5 when the position error is small (less than 12"), the identification success rate of this method is about 99.5% which is equivalent to the Delaunay cutting method. When the position error grows gradually, the accuracy of the Delaunay identification quickly falls down in order to prevent the of use information by adjacent triangles. This process is because the Delaunay cutting, like this method, adopts the angular distance information of three vertices for cutting the triangle in order to identify star map.

Also, in the matching probability algorithm, when the position error is small, the identification success rate of the method is approximately similar to the present method because the angular distance matching is accurate; however, when the position error gradually becomes larger, this method fails; since it relies on the angular distances only. However, the identification success rate of the star recognition based on the DT algorithm is not much sensitive to the position error while the DT of stars does not change and still keeps a high success rate of recognition. When the position error is about 50", the identification success rate of this method is about 98%. At the same time, the other two identification methods are only about 94%. As a whole, it can be seen from Fig. 5 that the recognition success rate of this method is higher than that of the Delaunay cutting method and the probability matching method.

In most cases that the algorithm fails, the number of stars is less than the specified threshold, but if the number of stars is greater than the threshold, then this algorithm will be reliable.

IV. Conclusions

A new star pattern identification algorithm based on Delaunay triangulation and correlation matrix memories was proposed in this paper. As each "Triangle" in the Delaunay Triangulation pattern includes the positional relations of stars and angular distance information. After generating two graphs from input image and star catalog, these graphs were represented to correlation matrix memories to removing minimum set of violated constraints by using relaxation by elimination method. Experimental results demonstrated that the efficiency of the proposed algorithm was higher, and the memory requirement was lower, compared with the traditional triangle matching algorithm. Identification success rate of this method still kept 98% while that of the matching probability algorithm was about 94%. So, the method had a high recognition success rate, and was fast and robust.

Acknowledgment

We thank Mr. M. Miri, the Iranian aerospace researcher. We also thank the Linux and GNU companies for providing their excellent software tools which were extensively used in this study.

References

[1] B. Spratling, and D. Mortari, "A Survey on Star Identification Algorithms," *Algorithms*, vol. 2, no. 1, pp. 93-107, 2009

[2] G. L. A. Rousseau, J. Bostel, and B. Mazari, "Star Recognition Algorithm for APS Star Tracker: Oriented Triangles," *IEEE Aerospace Electroic Systems Mag*, vol. 20, pp. 27-31, 2005.

[3] P. J. Roberts and R. A. Walker, "Application of a Counter Propagation Neural Network for Star Identification," *The American Institute of* Aeronautics *and Astronautics* **,** pp. 15-18, August 2005.

[4] M. DeBerg M., Ch. Otfried, V. K. Marc, and O. Mark, *Computational Geometry: Algorithms and Applications*, 3rd ed, Springer-Verlag, March 2008.

[5] W. Quan and J. Fang, "A Star Recognition Method Based on the Adaptive Ant Colony Algorithm for Star Sensors," in *Sensors*, vol. 10, pp. 1955-1966, 2010.

[6] G. J. Zhang, X. G. Wei, and J. Jiang,"Star Map Identification Based on a Modified Triangle Algorithm," *ACTA Aeronaut. ET Astronaut.* SINICA, vol. 27, pp. 1150-1154, 2006.

[7] Z. L. Wang and W. Quan, "An All-Sky Autonomous Star Map Identification Algorithm," *IEEE Aerospace Electroic Systems Mag*, vol. 19, pp. 10-14, 2004.

[8] X . Junfeng, J. Wanshou, and G. Jianya, "a new star identification algorithm based on matching probability," *IGARSS 3IEEE*, p. 1166-1169, 2008.

[9] B. Okabe, Boots, and K. Sugihara, *Spatial Tessellations - Concepts and Applications of Voronoi Diagrams*, 2nd ed, Wiley, 1992.

[10] S. Klinger and J. Austin, "Chemical Similarity Searching Using a Neural Graph Matcher," *In Proc. the 13th European Symposium on Artificial Neural Networks*, pp. 27-29, April 2005.

[11] StarCalc Home Page-Downloads. (2008). [Online]. Available: http://www.relex.ru/~zalex/files_eng.htm

[12] P. Fleischmann, "Mesh Generation for Technology CAD in Three Dimensions," in: *Dissertation*, December 1999.

[13] S. Zampelli, Y. Deville, C. H. Solnon, S. Sorlin, and P. Dupont, "Filtering for Subgraph Isomorphism," in: *Springer*, pp.728-742, 2007.

[14] J. Austin, "Relaxation Labelling Using Distributed Neural Networks", in: *Studies in Computational Intelligence*, vol. 210, pp. 111-138, 2009

Saeideh Sadat Miri was born in 1985. She received the B.S. degree in software engineering from the Shahid Abbaspour University,Tehran, Iran, in 2007 and M.S. degree in computer science from the Amirkabir University, Tehran, Iran, in 2011 .
She is currently a software engineer in the Mapsa Corporation

Mohammad Ebrahim Shiri is currently an Assistant Professor of computer science at Amirkabir University. He received his Ph.D. degree in computer science from University of Montreal, Montreal, Canada