An Experimental Survey of Evaluation Strategies for Constellation Queries

Glenn Galvizo
glennga@hawaii.edu
University of Hawaii at Manoa
Honolulu, Hawaii

Lipyeow Lim
lipyeow@hawaii.edu
University of Hawaii at Manoa
Honolulu, Hawaii

ABSTRACT

Given a set of query points within an image coordinate system, constellation queries identify the matching points in a database of known points within a standard coordinate system. Constellation queries are an integral part of orientation determination systems used in spacecrafts to orient and navigate themselves. The query points are bright spots in an image captured by a camera on the spacecraft and the database contains known celestial objects in a celestial coordinate system. This paper studies six existing constellation query processing strategies (Angle, Interior Angle, Spherical Triangle, Planar Triangle, Pyramid, Composite Pyramid) using a unified algorithmic framework and presents experimental evaluation of the six strategies. We find that the Pyramid strategy in its simplified form has the best accuracy to runtime ratio given simulated images with false positives, false negatives, and Gaussian noise.

CCS CONCEPTS

• **Information systems → Information retrieval**; **Information retrieval query processing**; **Retrieval effectiveness**; **Retrieval efficiency**.

ACM Reference Format:
Glenn Galvizo and Lipyeow Lim. 2019. An Experimental Survey of Evaluation Strategies for Constellation Queries. In *31st International Conference on Scientific and Statistical Database Management (SSDBM '19), July 23–25, 2019, Santa Cruz, CA, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3335783.3335806

1 INTRODUCTION

With the advent of commercial space industry, we are seeing an increasing number of spacecrafts, both manned and unmanned, launched into space. One important function on a spacecraft is the ability to determine its orientation in space quickly using the images captured by a camera on the spacecraft – this is known as the *Lost-in-space* problem. For example, consider the design of low Earth orbit (LEO) spacecrafts. In order for the spacecraft to point a payload, direct its thrusters, or orient its solar panels, an accurate *attitude* (another term for orientation) must be known within a reasonable amount of time. There are a few known landmarks in space where some attitude can be extracted (e.g. the Earth, the Sun), but most attitude determination systems instead use multiple stars within the field of view of a camera to determine their orientation.

The images capture a rectangular region of space – think of an image of the sky at night from an Earth-centric perspective. Each image is essentially a collection of bright spots (celestial objects) and their location in the image coordinate system. To find the attitude of the spacecraft, we first have to match the pattern of celestial objects in the image to a database of known celestial objects. We refer to such database queries as *Constellation Queries*: Given a set of query points in the image coordinate system, find the set of points, i.e. constellations, in a database of known points (in a standard coordinate system) that matches a subset of the query points.

Hypothetically, if we could transform the query points and database points to a common coordinate system, then constellation queries would be a simple database lookup for each query point. Unfortunately, evaluating constellation queries are complicated by several issues. First the query points and the points in the database are in two different coordinate systems and the transformation between the two coordinate system is not known *a priori*. In fact, attitude determination *is* finding the transformation between the two coordinate systems. Second, the query points are affected by transient celestial objects (e.g. meteors) resulting in spurious points in the image, obstructions resulting in missing points, and camera characteristics resulting in deviations of the query point's true position.

The constellation queries studied in this paper can be viewed as a type of *subgraph isomorphism query* that aims to find some 1-to-1 mapping between the vertices in two graphs (i.e. the reference database and the image) if it exists [5]. The subgraph isomorphism formulation is often juxtaposed with *pattern recognition approaches*, which map larger sets of points within a defined field-of-view across some reference table and image [13]. We also focus on *non-recursive* constellation queries - queries that do not process results from a *prior* query. In recursive searches such as the SP-Search and SNA by Samaan [15], an additional filter can be applied to the query that limits the possible database points that map to our image set. We are also interested in indexing strategies for constellation queries. We indexed our reference relations solely by some statistic of a collection of star positions (e.g. the interstar angle between two points, the area between three points), rather than indexing by star brightnesses (see Scholl[16], Ketchum [8], and more recently Zhang et. al [19]). There is also much prior work on optimizing the data accesses in constellation queries, most notably Motari's

SSDBM '19, July 23–25, 2019, Santa Cruz, CA, USA

ACM ISBN 978-1-4503-6216-0/19/07…$15.00
https://doi.org/10.1145/3335783.3335806

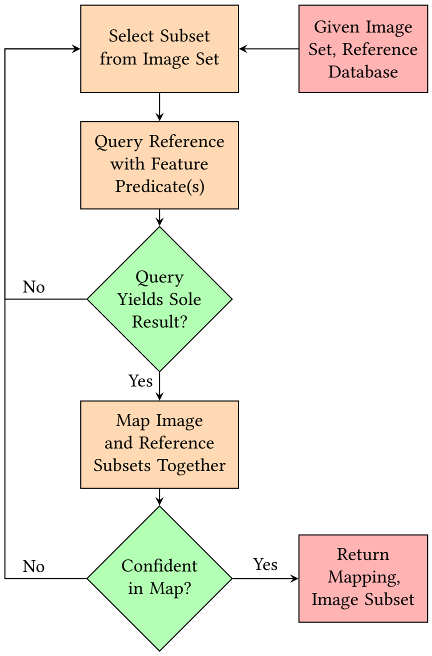

Figure 1: Flowchart depicting the unified identification framework which all strategies here follow. Given a query image, this process returns a map that pairs of subset of image points with points in a reference database. In the event all image subsets are exhausted, an error is raised and no map is returned (not depicted).

SLA (Search-Less Algorithm) [10]. Our focus is on the end-to-end constellation query and not just on the data access aspects.

This paper is an empirical survey of six existing constellation query processing strategies in the literature and in the industry. To the best of our knowledge, no systematic survey exists for these constellation query processing algorithms, no attempts have been made to study them in a unified framework, and no empirical evaluation of these algorithms have been done even though many of these algorithms have been deployed in spacecrafts and satellites (SAS-3, HTSE, ISC [6, 11]). Additional survey papers have been published by Spratling [7] and Brätt [3], but our research focuses on the specific area of constellation queries (not image processing, attitude determination, or database searching) and describes each strategy within a unified framework.

2 QUERY PROCESSING STRATEGIES

2.1 Unified Identification Strategy Framework

Each constellation query is presented with a set of points from an image IMG of size n as well as a database containing two relations: REF holds the collection of all known points indexed by some identifier, and $\texttt{REF}_{\texttt{STRATEGY}}$ holds sets of these known points indexed by one or more strategy specific features of the set itself. Elements in $\texttt{REF}_{\texttt{STRATEGY}}$ are henceforth referred to as r sets, with $|r| = d$. The resultant of the constellation query is a d-sized subset of points from the image known as the b set, as well as a mapping from this set to the database. Identification of all points in each image is not the focus.

To start the query, all strategies begin by choosing a b set from the image IMG. r sets are then retrieved from $\texttt{REF}_{\texttt{STRATEGY}}$ using some predicate dependent on the current b. If there exists only one r as a result, then we proceed. Otherwise, we loop back to the first step and select a different subset of d stars. Certain methods may also choose to invoke a secondary database retrieval utilizing a new database set to filter the current r before deciding to choose a different b altogether. Once we have found a unique r set, we determine a mapping between this set and b. If we are confident in this map, we return this function along with the current b set. Otherwise, we loop back to the start and choose a new & distinct b set. This process is depicted in Figure 1. In the event we exhaust all possible b sets, an error is raised and no map is returned.

2.2 Overview of Strategies

We discuss all six strategies in this section: the Angle strategy, the Interior Angle strategy, the Spherical Triangle strategy, the Planar Triangle strategy, the Pyramid strategy, and the Composite Pyramid strategy.

The *Angle* strategy (abbreviated as ANG) is composed of a naive image subset decision, angular features of point pairs, and an exhaustive optimal map determination procedure. Given a point set from the image IMG, our query starts by selecting $d = 2$ distinct points b_1, b_2 to represent the b set. To naively select some b is to fix the point b_1 for n image subset selections, while constantly changing b_2 for every new b choice. With an image subset selected the strategy proceeds by querying $\texttt{REF}_{\texttt{ANG}}$, a relation holding tuples of all combinations of two REF identifiers, indexed by the angular separation between both REF points. The Angle method performs a range search across $\texttt{REF}_{\texttt{ANG}}$ for all r sets such that the angular separation defined in $\texttt{REF}_{\texttt{ANG}}$ is close to the separation between the image subset [3]. If our database search yields a single r set, we proceed to the last step: finding a mapping through a process known as the direct-match test. We follow Tappe's implementation of this procedure, which iterates through all possible maps, transforms the image to the standard coordinate system using this pairing, and chooses the map with the most amount of image points close to some database point [12, 17]. In the event both maps possess no matching database points other than those in r, we return to the image subset selection step.

The *Interior Angle* strategy (abbreviated as INT) is composed of a non-exhaustive "clustered" image subset decision, a feature set possessing two interstar angles and one interior angle, and an asymmetric permutation store for map determination. Given a set of points from the image IMG, the Interior Angle strategy chooses a central point from the image b_c and the two nearest points b_1, b_2 to the central point [9]. Choosing another b set involves only choosing a new b_c from IMG. Two out of three points are dependent on the current choice of the central point, meaning that this strategy avoids exploring $\binom{n}{3} - n$ combinations in contrast to the exhaustive approach of the Angle strategy. With an image subset selected the Interior Angle strategy proceeds by querying $\texttt{REF}_{\texttt{INT}}$, a relation holding tuples of all *permutations* of three REF identifiers, indexed by the angular separations from the central point to the two nearest points & the angle between the two nearest points with the central point as the vertex (i.e. the interior angle). The Interior Angle strategy performs a three-dimensional range search for all r sets such that the aforementioned features of r are close to the features of b [3]. If our database search yields a single r set, we proceed to the last step: map determination. We follow RezaToloei's implementation of the Interior Angle strategy which sacrifices storage to avoid performing an optimal map determination procedure (e.g. the direct-match test) by storing permutations of REF instead of combinations like the Angle strategy [14]. When this permutation store is used in conjunction with an asymmetry rule, "the first interstar angle must be less than the second", we store the mapping of r as well [2]. Thus, to determine a map here is to use the pairing: $\{(b_c, r_c), (b_1, r_1), (b_2, r_2)\}$.

The *Spherical Triangle* strategy (abbreviated as SPH) is composed of a naive image subset decision, spherical area and moment features, a "pivoting" process to jointly utilize 2+ database searches, and an exhaustive optimal map determination process. Given a set of points from the image IMG, our query starts by selecting $d = 3$ distinct points to represent the b set in the similar manner to our Angle strategy. With an image subset selected, the Spherical Triangle strategy proceeds by querying REF_{SPH}, a relation holding tuples of all combinations of three REF identifiers, pairwise indexed by the spherical area and moment between all three REF points. The Spherical Triangle strategy performs a two-dimensional range search for all r sets such that the spherical area and moment of b is close to the spherical area and moment of r. If our database search does not yield a single r set, we perform another range search with a new image subset that differs from our original b by only one point. The results of the secondary search are then used to filter the initial set by removing all sets in our initial search that do not share two points with our secondary search. This "pivot" is repeated to filter the initial search further until a single r is found or all n additional image subsets have been exhausted [4]. Once a sole r set is obtained, we perform a direct match test to either obtain a map or choose another b set and repeat the process altogether.

The *Planar Triangle* strategy (abbreviated as PLN) is identical to their Spherical Triangle strategy, with the exception that each image trio is represented as a planar triangle instead of a spherical one. This results in the computation of a planar area and moment, operations that do not require a recursive formula in contrast to the spherical area and moment.

The *Pyramid* strategy (abbreviated as PYR) is composed of a false star avoidance image subset decision, a custom voting based identification process for star trios, and a voting based confidence check. Given a set of points from the image IMG, the Pyramid strategy selects $d = 3$ points in a such a way to avoid the persistence of misleading points for more than a few combinations [11]. What follows is a voting-based process which utilizes the relation REF_{ANG}. Per image subset, three of the database range searches used in the Angle strategy are performed to find database pairs for all distinct image pairs in b. To determine database points for a single image point in b is to take the intersection between both sets of points of the database pairs containing b in their query [18]. Similar to the Interior Angle method, an optimal map determination procedure is not required because the map is implicitly formed to determine r. To establish confidence in this map, another voting strategy is used which repeats a process similar to obtain r with the inclusion of an additional star from IMG. Only if there exists a sole result associated with this additional star do we return the map. Otherwise, we choose another b set and repeat the process.

The *Composite Pyramid* strategy (abbreviated as COM) is composed of the Pyramid strategy's image subset decision, planar area and moment features, and a voting based confidence check, and an exhaustive optimal map determination process. This method can be thought of as a combination of the Planar Triangle strategy and the Pyramid strategy, borrowing the features, database range search, and optimal map determination strategy of the former while using the image subset decision and confidence check of the latter.

3 EMPIRICAL EVALUATION

In this section we discuss our analysis of the six strategies in terms of their runtime and accuracy response to typical errors that may occur when capturing an image of the sky. For our testing, the astronomical catalog used to populate all relations in the reference database was the Hipparcos Input Catalogue [1]. The REF relation was shrunk to 1,471 elements dependent on each star's brightness ($m < 4.5$) to reduce the size of the $\text{REF}_{\text{STRATEGY}}$ relations (and consequently the running time) for each strategy. All simulations were performed 2,000 times on a Raspberry Pi Model 3B+, implemented in C++14 without optimization, and utilized SQLite as the embedded SQL database engine. Image data was generated as points randomly (but uniformly) rotated from the database coordinate system to remove the discrepancies that may arise from the image processing component. Three types of errors were introduced to the image data from here: false positives, false negatives, and points whose position is misrepresented from the database (i.e. Gaussian noise). The width of each strategy's range search was determined using a grid search, exploring boundaries from 1×10^{-1}, 1×10^{-2}, $\ldots$, to 1×10^{-10}. The exact implementation is available at the following link: https://github.com/glennga/hoku.

In a clean image (introducing no errors), all strategies are always able to produce a correct mapping from some image subset to the REF table. The Pyramid and Interior Angle strategies are the fastest methods here (6.293±0.167ms), followed by the Angle strategy (81.249ms), with all methods possessing triangular features last (122.479±1.940ms). Both the Pyramid and Interior Angle strategies do not utilize an optimal map determination procedure, which significantly reduces their runtime. The Angle strategy, though still dependent on the direct-match test, has the smallest $\text{REF}_{\text{STRATEGY}}$ relation (2-combinations vs. 3-combinations) resulting in a slight speedup from the triangular featured strategies.

In an image of three to six false positives, the unmodified methods that perform the most accurately are the Angle, Planar Triangle, and Spherical Triangle strategies. The source of error for all other methods are a result of exhausting all b sets, as opposed to returning an incorrect map. Removing the voting based confidence check eliminated this error for both the Pyramid and Composite Pyramid strategies, raising both accuracies to near 100% with a runtime faster than the Angle, Planar Triangle, and Spherical Triangle strategies. This difference in speed also demonstrates the efficacy of the pyramid b decision in contrast to the naive approaches. For the Interior Angle strategy, replacing its non exhaustive image

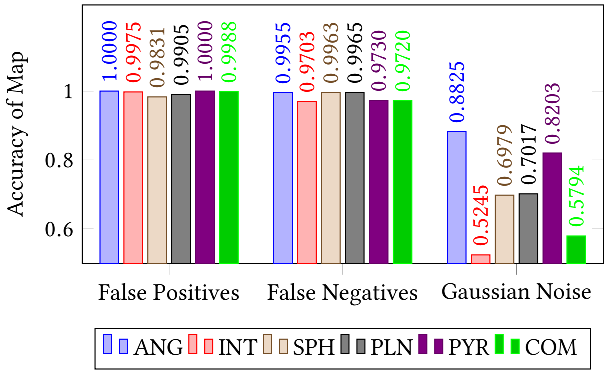

Figure 2: Depicts the average accuracy of the resulting map of the unaltered Angle and triangle strategies, an Interior Angle strategy with a naive image subset decision, and pyramid strategies without their verification steps given images with false positives, false negatives, and Gaussian noise.

subset decision with an exhaustive naive approach increased its accuracy to the near 100%. When dealing with this type of noise, not exploring all possible combinations increases the chance that a trio without false positives is never handled, thus lowering overall accuracy. Figure 2 depicts the accuracy of each strategy in its most optimal configuration under varying types of noise.

In an image of one to three light blocking blobs (resulting in false negatives), all strategies except the unaltered pyramid strategies perform with similar accuracy and runtime. These errors stem as a result of an image possessing too few points to perform identification altogether. The unaltered pyramid strategies require a minimum of four points, raising the chance of having too little stars to perform identification and decreasing the overall accuracy. Removing the verification step for the pyramid methods raises the Composite Pyramid accuracy to a similar range of the rest and does not significantly affect the Pyramid accuracy, suggesting that this verification step is too aggressive of a filter overall. In terms of speed the Interior Angle method is the fastest of the six, followed closely by Pyramid method (5.791ms vs. 6.050ms).

In an image with Gaussian noise of $\sigma = 0.0001°$ to $\sigma = 0.0002°$, the Angle strategy is the most accurate but runs nearly 15 times as long as the 2nd most accurate & fastest overall strategy: the Pyramid strategy without a verification step (208.552ms vs. 13.801ms). Unlike the results of the previous simulations, we now see incorrect maps returned by strategies instead of returning with an error. Though small, these maps are only produced by the strategies with triangular features (Planar Triangle, Spherical Triangle, Composite Pyramid). Interstar angles appear to be the most effective feature set against Gaussian noise.

To conclude, the Angle strategy is the most accurate strategy under all types of noise, but is the slowest of the six. The Pyramid strategy without a verification step is the next best strategy in terms of accuracy, and is the fastest overall method. The Interior Angle strategy was improved by using a naive image subset decision and has the lowest floor in terms of speed, however this strategy handles Gaussian noise the worst of the six. The triangular strategies lie between the Angle and Interior Angle methods in terms of accuracy, but are a faster alternative to the Angle method. The Composite Pyramid strategy attempts to utilize a different feature set with the effectiveness of the Pyramid core but ultimately inherits the worst of each strategy.

ACKNOWLEDGMENTS

We would like to thank Dr. Miguel Nunes, Eric Pilger, and Yosef Ben Gershom from the Hawaii Space Flight Laboratory for providing input toward the creation of software for a first generation star tracker.

REFERENCES

[1] M A. C. Perryman, Lennart Lindegren, J Kovalevsky, E Hoeg, U Bastian, P L. Bernacca, M Crézé, Francesco Donati, M Grenon, M Grewing, Floor van Leeuwen, Hans Marel, F Mignard, CA Murray, Rudolf Le Poole, H Schrijver, C Turon, Frédéric Arenou, M Froeschlé, and C S. Petersen. 1997. The Hipparcos Catalogue. 323 (06 1997), L49–L50.

[2] D. S. Anderson. 1991. *Autonomous Star Sensing and Pattern Recognition for Spacecraft Attitude Determination.* Ph.D. Dissertation. Texas A&M University.

[3] Steven Paul Brätt. 2013. Analysis of Star Identification Algorithms due to Uncompensated Spatial Distortion. (2013), 9–10, 12–23, 33–34 pages.

[4] Craig L. Cole. and John L. Crassidis. 2004. Fast Star Pattern Recognition Using Spherical Triangles. *AIAA/AAS Astrodynamics Specialist Conference and Exhibit* (08 2004), 3–10.

[5] Scott Fortin. 1996. *The graph isomorphism problem.* Technical Report. University of Alberta. 1–4 pages.

[6] D.M. Gottlieb. 1978. *Spacecraft Attitude Determination and Control.* Kluwer, Chapter 7.7, 259–266.

[7] Benjamin B. Spratling IV and Daniele Mortari. 2009. A Survey on Star Identification Algorithms. *Algorithms* 2 (03 2009), 94–95.

[8] Eleanor A Ketchum and Robert H Tolson. 1995. Onboard star identification without a priori attitude information. *Journal of Guidance, Control, and Dynamics* 18, 2 (1995), 242–246.

[9] Carl Christian Liebe. 1995. Star Trackers for Attitude Determination. *IEEE Aerospace and Electronic Systems Magazine* 10, 6 (Jun 1995), 13–15.

[10] Daniele Mortari and Jonathan Rogers. 2013. A k-vector Approach to Sampling, Interpolation, and Approximation. *The Journal of the Astronautical Sciences* 60, 3-4 (2013), 686–706.

[11] Daniele Mortari, Malak A. Samaan, Christian Bruccoleri, and John L. Junkins. 2014. The Pyramid Star Identification Technique. *Navigation* 51, 3 (2014), 8,13–18,31–32.

[12] David Needelman, Rongsheng Li, and Yeong-Wei Wu. 2005. Recent Advances in Stellar Attitude Acquisition (SAA) Algorithms and Procedures. *AIAA Guidance, Navigation, and Control Conference and Exhibit* (2005), 1–17.

[13] Curtis Padgett and Kenneth Kreutz-Delgado. 1997. A grid algorithm for autonomous star identification. *IEEE Trans. Aerospace Electron. Systems* 33, 1 (1997), 202–213.

[14] Ali RezaToloei, Morteza Shayan, and Manouchehr Abaszadeh. 2014. A New Composite Algorithm for Identifying the Stars in the Star Tracker. *International Journal of Computer Applications* 102 (09 2014), 30–31.

[15] Malak A Samaan, Daniele Mortari, and John L Junkins. 2005. Recursive mode star identification algorithms. *IEEE Trans. Aerospace Electron. Systems* 41, 4 (2005), 1–5.

[16] MS Scholl. 1995. Star-field identification for autonomous attitude determination. *Journal of Guidance, Control, and Dynamics* 18, 1 (1995), 61–65.

[17] Jack A. Tappe. 2009. Development of Star Tracker System for Accurate Estimation of Spacecraft Attitude. (2009), 9–14, 19–22 pages.

[18] V. Tichý, R. Fullmer, D. Fowler, and M. Hromčík. 2011. Preliminary tests of commercial imagers for nano-satellite attitude determination. In *Optical Sensors 2011; and Photonic Crystal Fibers V (Proceedings of SPIE)*, Vol. 8073. 1–9. https://doi.org/10.1117/12.886743

[19] Peng Zhang, Qile Zhao, Jingnan Liu, and Ning Liu. 2014. A brightness-referenced star identification algorithm for aps star trackers. *Sensors* 14, 10 (2014), 18498–18514.