An Experimental Survey of Evaluation Strategies for Constellation Queries

Glenn Galvizo
University of Hawaii at Manoa
Honolulu, Hawaii

glennGa@hawaii.edu

Lipyeow Lim
University of Hawaii at Manoa
Honolulu, Hawaii

lipyew@hawaii.edu

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:

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
Additional survey papers have been known as the resultant of the constellation query is a tagifier, and holds the collection of all known points indexed by some identifier. Each constellation query is presented with a set of points from an image 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 REFSTRATEGY 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 b1, b2 to represent the b set. To naively select some b is to fix the point b1 for n image subset selections, while constantly changing b2 for every new b choice. With an image subset selected the strategy proceeds by querying REFANG, 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 REFANG for all r sets such that the angular separation defined in REFANG 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 IMG and the two nearest points b1, b2 to the central point [9]. Choosing another b set involves only choosing a new b3 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 REFINT, 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.
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 \( \text{REF} \) 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: \((b_0, r_0), (b_1, r_1), (b_2, r_2)\).

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 \( \text{IMG} \), 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 \( \text{REF}_{\text{SPH}} \), a relation holding tuples of all combinations of three \( \text{REF} \) identifiers, pairwise indexed by the spherical area and moment between all three \( \text{REF} \) 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 \( \text{IMG} \), 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 \( \text{REF}_{\text{IMG}} \). 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 \( b \) it 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 \( \text{IMG} \). 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 \( 1 \times 10^{-1} \), \( 1 \times 10^{-2} \), \ldots, to \( 1 \times 10^{-10} \). The exact implementation is available at the following link: https://github.com/glennaga/hoku.

In a clean image (introducing no errors), all strategies are always able to produce a correct mapping from some image subset to the \( \text{REF} \) 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.

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 Gershon from the Hawaii Space Flight Laboratory for providing input toward the creation of software for a first generation star tracker.

REFERENCES


