

# ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА

УДК 621.3.087.9

## АНАЛИЗ ОСНОВНЫХ ХАРАКТЕРИСТИК АЛГОРИТМОВ ОБРАБОТКИ ИЗОБРАЖЕНИЙ ДЛЯ ЗВЁЗДНОГО ДАТЧИКА

С. А. Волков

В данной работе приведён общий анализ основного компонента звёздного датчика – математических алгоритмов обработки изображений и идентификации звёзд. В механизме работы звёздного датчика заложена процедура анализа изображения, полученного со светочувствительной матрицы и записанного в оперативную память математического процессора. Далее процессор, выполняя алгоритм и обращаясь к базе данных (звёздный каталог), осуществляет поиск верного решения. Существует множество методик и алгоритмов решений проблемы идентификации звёзд и установления координат космического аппарата. В работе описана общая классификация алгоритмов, приведено описание нескольких из алгоритмов идентификации звёзд, разработанных ранее, и показан обобщённый механизм работы.

**Ключевые слова:** звёздный датчик, алгоритм, быстродействие, идентификация звезд, анализ.

Одним из способов ориентации космического аппарата является использование звёздного датчика. Он позволяет выполнить угловую ориентацию космического аппарата относительно некоторой выбранной оси с применением изображения звёздного неба. Изображение фиксируется с помощью светочувствительной матрицы.

После получения информации со светочувствительной матрицы её необходимо

обработать. Работа звёздного датчика заключается в получении относительных координат трёхмерного положения космического аппарата, используя на информационном входе лишь двухмерное изображение. Для решения этой проблемы используются различные математические методики обработки изображений и отождествления пар. На рисунке 1 показан обобщённый механизм, связывающий реальное изображение и выходные данные.



Рис. 1. Обобщенная схема работы алгоритма

© Волков С. А., 2020.

Волков Сергей Александрович (serega.volkov1234@gmail.com), студент IV курса факультета электроники и приборостроения Самарского университета, 443086, Россия, г. Самара, Московское шоссе, 34.

Поиск соответствий по существующей базе данных осуществляется за счет перебора по заранее собранному каталогу звёзд. Все алгоритмы возможно классифицировать по нескольким типам [1]:

1. Рекурсивные алгоритмы – выполняется предположение о положении датчика;
2. Нерекурсивные (lost-in-space) – не используется никаких предположений;
3. Алгоритмы с использованием информации о яркости звёзд;
4. Использование особой методики выбора звёзд – произвольный перебор каталога или выбор с использованием какого-либо из записанных параметров;
5. Калибровочная инвариантность – инвариантность по отношению к дисторсиям первого порядка.

Алгоритмами может быть предусмотрена возможность пополнения информации звёздного каталога. Такая возможность применяется для формирования определённого каталога по маршруту следования аппарата или же формирования каталога с новыми свойствами или относительно некоторой новой оси координат.

После окончания поиска звёзд выполняется проверка истинности значений. Возможны случаи, когда блики от космического мусора будут создавать ложные звёзды. Обычно проверка выполняется на основе предыдущих значений: новые координаты должны находиться на небольшом удалении от прежних. Если же ошибка подтверждается, создается запрос на получение нового снимка и выполняется новый цикл работы программы.

У всех алгоритмов возможно выделить

несколько важных характеристик.

Первым из всех параметров выделяется быстродействие алгоритма обработки. Важно понимать, что общая скорость работы звёздного датчика зависит не только от алгоритма: влияние вносит светочувствительность матрицы (скорость зарядки ячеек), скорость считывания заряда ячеек матрицы, скорость работы математического процессора и искомая точность. Сама работа системы основана на двух факторах: решении различных систем уравнений и сравнении результатов с каталогом. Скорость поиска по каталогу зависит от количества данных для сравнения и скорости микроконтроллерной системы.

Следующим параметром, имеющим большее значение, является объём звёздного каталога. В звёздный каталог записываются относительные координаты звёзд, яркость блеска, относительное расстояние между ними или другие характеристики, в зависимости от выбранного алгоритма. Звёздный каталог вынесен отдельно, так как его объём напрямую влияет на точность и быстродействие системы. Чем больше в каталоге звёзд, тем больше требуется времени на поиск верного решения системы.

Устойчивость алгоритма определяет то, как сильно будут влиять помехи на изображении (ложные звёзды) на быстродействие алгоритма или возможность применения алгоритма как такового. Для повышения вероятности успешной обработки изображения предварительно фильтруются с использованием ЦОС. В таблицу 1 занесены характеристики некоторых категорий алгоритмов [2].

Рассмотрим несколько алгоритмов идентификации звезд.

Таблица 1

Относительные характеристики алгоритмов распознавания звёзд

Название алгоритма	Объем каталога	Суммарное время обращения к каталогу	Время расчёта характеристик	Устойчивость к появлению ложных звёзд
Переборные алгоритмы	$F(N*S)$	$F(S*b)$ $b=3$ или $4$	$F(2*S)$	устойчив
Сеточные алгоритмы	$F(N)$	$F(\ln(S))$	$F(S)$	неустойчив
Алгоритмы сопоставления групп	$F(N*S)$	$F(S*\ln(N))$	$F(S)$	устойчив
Геометрические алгоритмы	$F(N)$	$F(\ln(N))$	$F(S)$	неустойчив

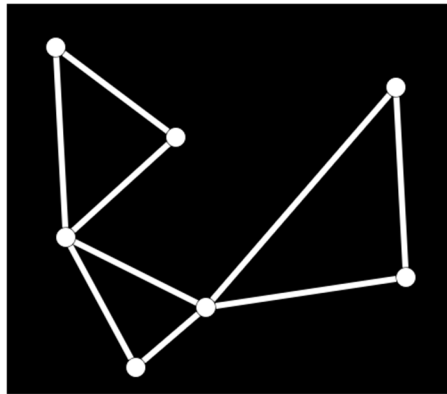


Рис. 2. Пример треугольников, соединяющих объекты

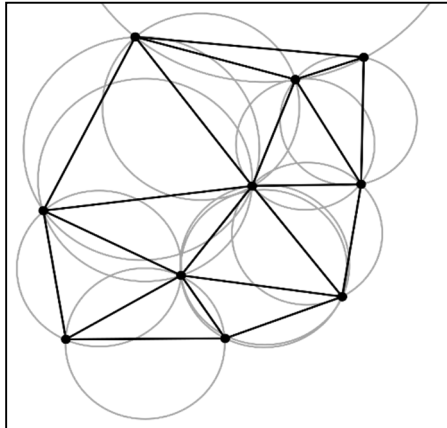


Рис. 3. Пример использования триангуляции Делоне

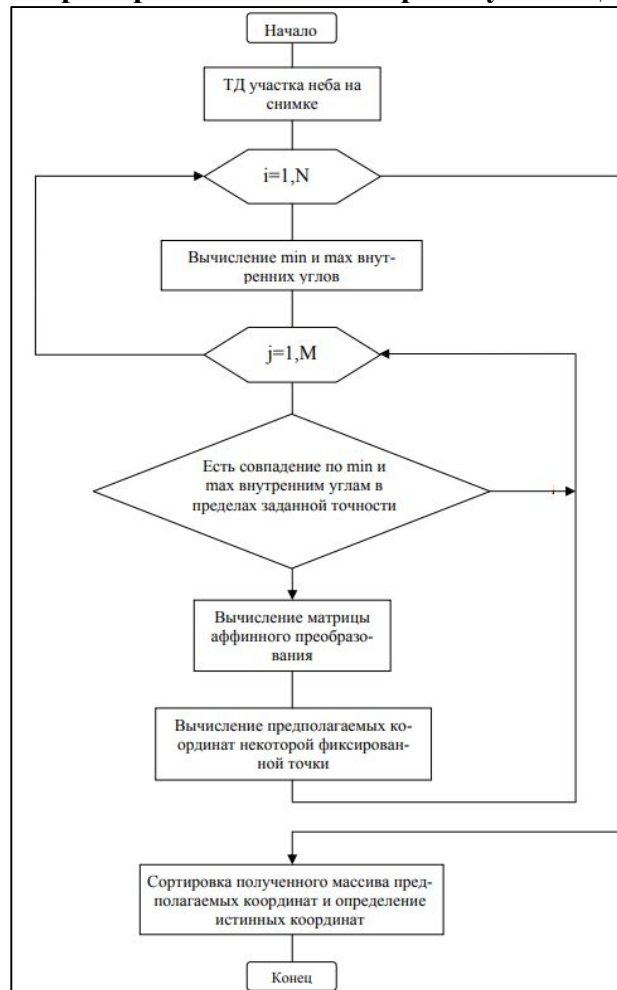


Рис. 4. Блок-схема алгоритма с триангуляцией Делоне

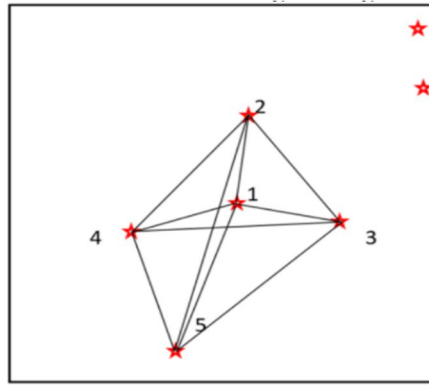


Рис. 5. Пример выбранных триплетов

Первый алгоритм относится к геометрическим с поиском определённых характеристик в каталоге и его работа основана на составлении треугольников, вершинами которых являются звёзды (рис. 2). У треугольника вычисляются наибольший и наименьший угол. В каталог занесены данные о звёздах, составляющих треугольники и соответствующие углы.

Критическим недостатком такого алгоритма является количество звёзд на изображении: чем больше звёзд (включая ложные звёзды), тем больше будет треугольников, и, как следствие, на вычисление уйдёт значительно больше времени. Количество треугольников можно вычислить по формуле (1):

$$C_3^n = \frac{n!}{3! \cdot (n-3)!} \quad (1)$$

Если объектов 4-5 (4-10 треугольников), то это замедлит вычисление, но работа продолжится в штатном режиме. При количестве объектов в кадре более 10, треугольников станет более 120, что может стать фатальным для скорости работы системы. В таком случае количество объектов, записанных в каталог, станет огромным, что повлияет на быстродействие алгоритма.

Для решения этой проблемы было предложено использование триангуляции Делоне [1] (далее – ТД). Наиболее простым описанием ТД является следующее: триангуляция, в которой для любого треугольника верно, что внутри описанной около него окружности не находится точек из исходного множества.

Использование ТД позволяет значительно сократить количество треугольников на изображении, упорядочивает данные и, как следствие, ускоряет расчёты при больших количествах объектов. При наличии хотя бы

нескольких идентифицированных треугольников алгоритм остаётся устойчивым. В худшем случае, процесс поиска по алгоритму будет составлять  $F(S^2)$ .

На рисунке 4 указана блок-схема алгоритма. Работа делится на этапы:

- Получение изображения;
- Выполнение ТД для снимка;
- Установление наибольшего и наименьшего углов у найденных треугольников;
- Сравнение со звёздным каталогом: если есть совпадение, то вычисляются возможные координаты, иначе процесс повторяется;
- После проверки всех углов и составления массива возможных координат выполняется сортировка координат и вычисление конечного значения координат.

Как уже было сказано выше, благодаря использованию ТД компенсируются недостатки простой методики работы геометрического алгоритма. Быстродействие повышается за счёт меньшего количества треугольника и углов. Звёздный каталог становится меньше. Из недостатков можно выделить только сложность реализации алгоритма – для меньших потерь быстродействия со стороны процессора необходимо мощное вычислительное устройство и хорошо оптимизированный код.

Триангуляция Делоне является не единственным решением идентификации звёзд. Следующий алгоритм [3] проводит анализ вблизи звезды, близкой к центру FOV (field of view – поле зрения):

1. Выбирается звезда, ближайшая к центру поля зрения;
2. Выбирается  $n-1$  ближайших к ней звёзд;

3. Комбинирование выбранных звёзд во все возможные комбинации триплетов;

4. Вычисление всех плоских углов и формирование матрицы углов  $C_3^n$ ;

5. Вычисление абсолютной ошибки для всех комбинаций матрицы из кадра и всех матриц из памяти;

6. Выбор матрицы с минимальной абсолютной погрешностью: матрица содержит информацию об искомым звёздах;

Звёздный каталог предварительно формируется аналогично пунктам 2-4 для каждой звезды в каталоге. Каталог состоит из набора матриц, в которых заключена информация о звездах и плоских углах.

Недостатком такого алгоритма является степенная сложность: добавление одной звезды на кадр увеличивает время работы в 2-3 раза. Хорошие результаты достигаются только при относительно высоком количестве анализируемых звезд на кадре (6-7 штук) и относительно высоких разрешениях светочувствительной матрицы (свыше 1024x1024).

Алгоритм Stars Pattern Hash Table [4] похож на предыдущий: в нем используются триплеты плоских углов. Отличие заключается в методе хранения звездного каталога: если предыдущий алгоритм подразумевал хранение информации в виде матриц, то здесь предложена идея просчитать хеш-таблицу триплетов после их округления. Методика позволяет реализовать почти мгновенный поиск в звёздном каталоге, так как обращение к нему больше похоже на обращение к массиву по индексу элемента, а не на поиск, как в предыдущем примере. Таким образом, основная вычислительная сложность здесь перекладывается на этап подготовки звёздного каталога. Полный же алгоритм будет выглядеть следующим образом:

1. Подготовка звёздного каталога в виде хеш-таблицы (вычисление триплетов и расчет хешей от них);

2. Получение и обработка кадра из камеры (коррекция дисторсий, отбрасывание звёзд с яркостью ниже пороговой величины);

3. Вычисление триплетов для полученного кадра;

4. Подсчет хеша полученного кадра;

5. Выборка триплета из хеш-таблицы звёздного каталога;

6. Проверка полученного результата

(из-за грубого округления данных на этапе подготовки хеш-таблицы возможны ложные срабатывания);

В другой работе [5] авторы применяют так называемый алгоритм пирамиды (алгоритм TRIAD):

1. Находятся центроиды звёзд, превышающих пороговый уровень яркости;

2. Для каждой из звёзд строится вектор из центра линзы;

3. Для четырех звёзд строится пирамида, для которой вычисляется внутреннее произведение 6 пар векторов;

4. Выполняется поиск в каталоге, который содержит предварительно вычисленные аналогично п.3 произведения, отсортированные в убывающем порядке;

5. Наиболее близкая величина является искомой, для нее запускается процесс вычисления точных углов разворота.

Алгоритм показывает достаточно высокую точность и относительно высокую скорость исполнения.

### Заключение

В данном обзоре рассмотрены современные подходы к решению задачи ориентации космического аппарата по изображению звёздного неба. Как видно из приведенных описаний алгоритмов, в настоящее время существует тенденция на использование итеративных алгоритмов, в которых поиск точного соответствия происходит в несколько этапов. На первом этапе подготавливается каталог звёзд, обеспечивающий алгоритмическую возможность наиболее быстрого поиска, здесь наиболее перспективными являются алгоритмы с хешированием или сортировкой. Далее происходит поиск центроидов звёзд на изображении и обработка в соответствии с алгоритмом подготовки каталога для грубого поиска. Затем производится непосредственно поиск и, после нахождения подходящих звёзд, оценка ошибки, либо запуск следующего этапа с более ресурсоемким, но более точным алгоритмом поиска соответствия, как, например, алгоритм TRIAD. В целом, такая методика с разделением на этапы позволяет комбинировать удачные подходы и получать высокоточные алгоритмы с высокой скоростью исполнения.

### Литература

1. Система определения положения и ориентации макета спутника на основе блока инерциальных датчиков и звездного датчика / Д.Биндель [и др.] // Препринты ИПМ им. М.В.Келдыша. 2011. № 24. 30 с.
2. Кружилов И. С. Методы и программные средства повышения эффективности распознавания групп звезд в автономной астронавигации: автореф. дис. ... канд. Тех.наук. Москва, 2010, 20 с.
3. Development of Star Pattern Recognition Algorithm for Star Sensor Using  $C^3_n$  Combination of Triangles Pattern / N S Ardi, R E Poetro [et al.] // IOP Conf. Series: Materials Science and Engineering 645 (2019) 012001
4. Revital Marbel, Boaz Ben-Moshe, Roi Yozevitch Star-Tracker Algorithm for Smartphones and Commercial Micro-Drones // Received: 11 December 2019; Accepted: 11 February 2020; Published: 18 February 2020.
5. Development and Ground Evaluation of Fast Tracking Algorithm for Star Trackers / Yuji Sato, Toshinori Kuwahara [et al.] // Trans. JSASS Aerospace Tech. Japan. 2018. Vol. 16. № 3. P. 202–209.

## ANALYSIS OF THE MAIN CHARACTERISTICS OF IMAGE PROCESSING ALGORITHMS FOR THE STAR SENSOR

S. A. Volkov

The present article provides a general overview in regards to the fundamentals of a star tracker – mathematical algorithms of image processing and star identification. The underlying principle is based on the procedure of image processing, images taken by the photosensitive matrix are recorded into RAM of a processor which is assigned to perform calculations. The next step is accomplished by the processor which commences performing predefined algorithms and afterwards sends a request to star catalogue databases meanwhile searching for the right match through comparison. There are a lot of techniques and methods to solve the existing key issue in star identification and setting coordinates of a spacecraft. The research gives a detailed description of the general algorithms classification, one of the existing algorithms, and the overall mechanism.

**Key words:** star tracker, algorithm, performance, star identification, overview.

*Статья поступила в редакцию 08.07.2020 г.*