

УДК 004.93:550.8

К. С. Алсынбаев

АЛГОРИТМЫ ОПРЕДЕЛЕНИЯ ТЕЛ ОБЪЕМНЫХ ОБЪЕКТОВ В ТРЕХМЕРНОМ НЕРЕГУЛЯРНОМ ОБЛАКЕ ТОЧЕК

Описана методика распознавания форм тел в трехмерном облаке точек. На первом шаге проводится кластеризация облака с критерием максимального расстояния между точками для идентификации распознаваемых тел. Далее производится переход к воксельному представлению. Критерием принадлежности вокселя телу считается существование тетраэдра с вершинами в точках облака и ограниченными по размеру ребрами, для которого тестируемый воксель – внутренний. Для оптимизации разработан и используется быстрый алгоритм заполнения воксельных тетраэдров. Работа входит в состав программного обеспечения обработки данных микросейсмического мониторинга.

A technique for recognition of shapes of bodies in a 3D point cloud is described. At the first step cloud clustering is conducted using the criterion of maximum distance between the points to identify the bodies being determined.



Then proceeding to voxel presentation is being done. Existence of a tetrahedron with its points in cloud points and size-limited sides for which the voxel being tested is internal is treated as a criterion of belongment of voxel to a body. A fast algorithm for voxel tetrahedron filling is developed and used for optimization. The work is a part of software for micro-seismic monitoring data processing.

Ключевые слова: облако точек, кластеризация, воксел, микросейсмический мониторинг, алгоритмы распознавания, алгоритм Брезенхейма.

Key words: a point cloud, Clustering, Voxel, micro-seismic monitoring, Recognition algorithms, Bresenham's line algorithm.

1. Контекст разработки алгоритмов

В рамках работ по созданию программного обеспечения обработки данных микросейсмического мониторинга [1] появилась потребность выделения объемных объектов в множестве микросейсмических событий, которое по сути представляет собой трехмерное нерегулярное облако параметризованных точек. Эти алгоритмы были необходимы для разработки функций оценки характеристик объемных объектов горной природы, а также в качестве промежуточного шага для сепарации специфичных объектов, например разломов и трещиноватостей, представляющих образования, близкие к двумерным поверхностям.

Решение задачи распознавания объемных объектов предполагает:

- определение локализации и формы пространства, которое соответствует близким в метрическом смысле точкам, включая возможность невыпуклости этой формы и наличия отверстий и полостей;
- порождение идентификаторов объектов, соотнесение вычисленных форм и присвоенных им идентификаторов;
- возможность вычисления объемов объектов и подготовки отчетов в виде списков идентификаторов объектов и их объемов;
- возможность граневой визуализации выделенного объекта.

2. Основной алгоритм формирования фигур распознаваемых тел

Наиболее известное решение задачи определения форм тел, представленных множеством точек, — построение тетраэдризации Делоне (трехмерной триангуляции Делоне) с заданной границей [2]. Существуют также алгоритмы поиска границ, например [4]. Полученные таким образом тетраэдры, составляющие граневую аппроксимацию, позволяют вычислить объем тела, организовать интерактивную 3D-визуализацию. В контексте данной задачи также эффективен подход, когда строится триангуляция Делоне для выпуклой оболочки, затем удаляются тетраэдры, у которых радиусы описанных окружностей превышают заданный [3]. Алгоритмы имеют вычислительную эффективность $O(N \log N)$. Одной из проблем тетраэдризации является чувствительность к точности вычислений.

Другой подход основан на представлении трехмерного пространства в виде дискретной сетки, элемент которой по аналогии с пикселем называется «воксел». Иначе говоря, производится своеобразная про-



странственная растеризация трехмерной области. С каждым вокселем связывается определенное значение или совокупность значений. Искомое трехмерное тело составляется в виде множества вокселей, имеющих необходимые значения.

В случае, когда исходное облако точек является «плотным» и регулярным, например полученным в результате томографического обследования, каждый воксел соответствует одной точке, изначально с точкой соотнесено определенное значение, воксельная матрица образуются автоматически, и выделение отдельного тела (соответствующего, например, какому либо человеческому органу) в виде воксельной модели трудности не представляет. Если же исходные данные представляют разреженное нерегулярное облако точек, как в случае микросейсмических данных, необходимо алгоритмическое решение вопроса о принадлежности или непринадлежности каждого вокселя телу.

В данной работе в некотором смысле объединены оба подхода. Область пространства с точками предварительно дискретизируется трехмерной сеткой, то есть представляется в виде трехмерной матрицы вокселей. Каждому вокселу соответствует область пространства кубической формы. Часть вокселей соответствует точкам облака (соответствующий кубик содержит одну или несколько точек облака) и помечается признаком «кластерный воксел» (выбор термина будет понятен ниже), остальным вокселям присваивается значение «0». Для каждого вокселя со значением «0» определяется, находится ли он внутри объекта. Для определения понятия «внутри» устанавливается значение D максимального возможно диаметра внутренней полости тел. Внутренние воксели тетраэдров, вершинами которых являются точки облака, и длины ребер не превышают значение D , считаются внутренними для тела (объекта).

Чтобы решить проблему предварительной локализации каждого тела и присвоения ему идентификатора, еще до этапа дискретизации проводится кластеризация, при которой точки считаются принадлежащими одному кластеру, если их для любой точки кластера существует точка этого же кластера, расстояние между которыми не превышает заданного. Предварительная кластеризация проводится с заданным максимальным расстоянием, равным D . После кластеризации каждый кластер получает идентификатор, равный порядковому номеру кластера, и считается, что точки кластера представляют отдельное тело, возможно, несвязное. Воксели, соответствующие после дискретизации точкам кластера, будем называть кластерными вокселями, а в программной реализации будем помечать соответствующим значением. Кластерные воксели изначально считаются принадлежащими телу.

Согласно формулированному выше воксел считается принадлежащим телу, если найдется тетраэдр, вершинами которого являются кластерные воксели, содержащий данный воксел. Суть алгоритма заключается в переборе всех вокселей и проверке для каждого из них, существует ли допустимый воксельный тетраэдр с вершинами в кластерных вокселях, содержащий текущий воксел.

Воксельное пространство проверяется маской кубической формы размера $M \cdot M \cdot M$, где $M = 2 \cdot \text{int}(D/P + 1) + 1$, с шагом, равным $\text{int}(D/P + 1) + 1$, то есть примерно половине стороны маски. Для кластерных вокселей внутри маски строится тетраэдризация Делоне [2], удаляются тетра-

эдры с ребрами, превышающими D , и все воксели внутри полученных тетраэдров приобретают специальное значение «принадлежит телу и не кластерный». Возможны случаи, когда воксел содержится внутри вырожденного тетраэдра, внутри треугольника или отрезка. В этом случае заполняется треугольник в трехмерном пространстве или проводится отрезок.

В качестве оптимизации используется следующая операция. Если тестирование определило, что воксел содержится внутри кластерного тетраэдра, то все внутренние точки этого тетраэдра помечаются как «принадлежащие телу, но не кластерные», с использованием быстрого алгоритма заполнения тетраэдра. Алгоритмы заполнения тетраэдра, треугольника и трехмерного отрезка используют принципы растровой графики, восходящие к алгоритму Брезенхейма, и при вычислениях в основных циклах не используется плавающая арифметика, операции умножения и деления. О реализации алгоритмов изложено ниже.

Полученная совокупность вокселей тела — приближенная относительно решения в вещественных значениях с точностью, равной размеру ребра куба, соответствующего вокселу. Поскольку каждый кластер обрабатывается отдельно, то порождаемая при этом воксельная модель является отдельным объектом.

В работе реализованы средства граневой визуализации, простого изображения кубиками (рис. 1), или полученную алгоритмом *marching-cubes* (реализована для данной задачи М. В. Козловым).

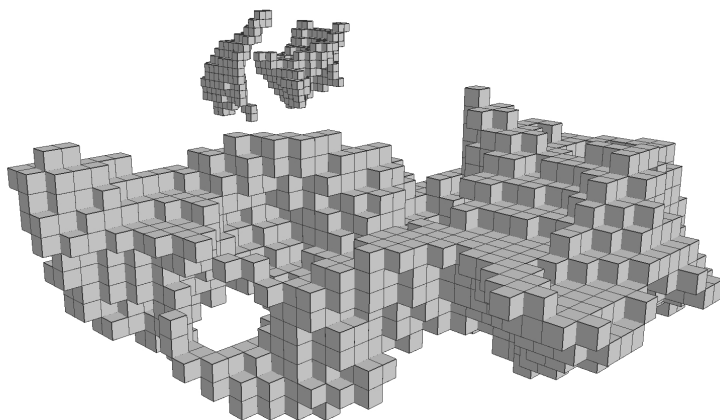


Рис. 1. Изображения оболочек трех распознанных тел.
Показаны в воксельной модели гранями кубиков

Приведем оценку вычислительной эффективности алгоритма. Пусть N — число точек облака, D — максимальный диаметр полости тела, P — размер стороны вокселя, K — количество кластеров, R — среднее число точек облака в кластере, V — среднее количество вокселей в кластерах, L — среднее количество точек облака в области маски, S — средний линейный размер кластера, L — среднее количество точек облака (или кластерных вокселей) в пределах маски.



Пусть значок \sim означает «приблизительно пропорционально». В кластере необходимо $\sim C^3/D^3$ перемещений маски. Для перебора всех тетраэдров в пределах маски нужно $\sim L \log L$ операций для заполнения тетраэдров в пределе маски $\sim M^3$ операций. Таким образом, получаем оценку мощности вычислений $\sim K(C^3/D^3)(L \log L + M^3)$ (не будем использовать префикс «O», чтобы избежать некорректности его использования). Принимая $R \sim N/K$, $M \sim D/P$, $L \sim RD^3/C^3 \sim ND^3/(KC^3)$, получаем оценку мощности вычислений

$$N(\log N + 3\log D - \log K + 3\log C) + KC^3/P^3.$$

Таким образом, при постоянных C , P и K мощность алгоритма соответствует $O(N \log N)$, причем такой рост обуславливается только ростом числа точек в окрестности тестируемой точки. Заметим, что при росте C уменьшается K . Время вычислений растет с увеличением C и резко растет с уменьшением P .

При значениях $K = 300$ и $P = 0,5$ м плотного облака в 480000 точек данных микросейсмического мониторинга длилась 2 часа 39 минут (без этапа кластеризации) на компьютере с процессором Intel® Xeon CPU W3565 3,19 GHz 12 Гб.

Реализация позволяет производить поиск тел только для указанных кластеров, что обеспечивает возможность подбора значений P при повторных запусках и завершения алгоритма для всех кластеров в приемлемое время. Алгоритм легко распараллеливается, в том числе для кластерных суперкомпьютеров.

3. Трехмерные аналоги алгоритмов Брезенхейма

Как известно, алгоритмы Брезенхейма и их производные предназначены для порождения графических объектов в растровых множествах. Особенность их в том, что они не содержат операций деления и умножения, а в основном цикле вычисления производятся исключительно на основе операций целочисленных сложения, вычитания, сравнения, благодаря чему достигается высокая скорость выполнения графических функций.

Воксельная матрица — трехмерный аналог растровых данных, и для них возможно создание аналогичных функций.

В данной статье были разработаны алгоритмы для генерации трехмерных воксельных векторов, трехмерных многоугольников, тетраэдров.

В основе генерации заполненных тетраэдров — быстрый алгоритм порождения плоских многоугольников. Для каждого горизонтального слоя тетраэдра генерируется отдельный выпуклый плоский растровый многоугольник, соответствующий сечению — треугольнику или четырехугольнику, вершины которых вычисляются трехмерными алгоритмами Брезенхейма проведения линий. На рисунке 2, а показаны два горизонтальных слоя, создаваемых быстрыми алгоритмами, а ребра тетраэдра проведены (для демонстрации) трехмерными воксельными векторами. На рисунке 2б — соответствующий заполненный воксельный тетраэдр.

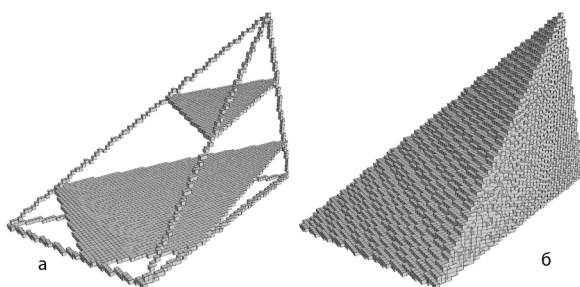


Рис. 2: *a* – элементы алгоритма создания заполненного тетраэдра.
б – заполненный воксельный тетраэдр

4. Алгоритм кластеризации облака точек

Оптимизация алгоритма осуществляется на исключении из сравнения точек, находящихся на достаточно большом расстоянии от текущей проверяемой. Принцип схожего алгоритма для двумерного случая, называемый «кластеризация точек на регулярной сети», приведен, например в [5]. В рассматриваемой задаче он модифицирован для повышения вычислительной эффективности.

Алгоритм заключается в следующем. Пусть задано максимальное расстояние между точками одного кластера D_{max} и минимальный размер каждого кластера. Вычисляется максимальная дистанция для сравнения по каждой из координат, равная $GRIDV = D_{max} * 2$. Каждая точка (x, y, z) снабжается полем, задающим номер кластера типа «целое» и полем для индексации типа «целое». Создается массив $SizeOfCluster$, содержащий размеры каждого кластера. На предварительном этапе вычисляется покоординатные индексы iX, iY, iZ каждой точки по формуле $iX = \text{int}((x - \min(x)) / GRIDV)$ и общий индекс-свертка на основе позиционного способа $ind(x) = iZ * \max(iX + 1) * \max(iY + 1) + iY * \max(iX + 1) + iX$. Очевидно, зная $\max(iX + 1)$ и $\max(iY + 1)$, из $ind(x)$ легко восстановить iX, iY, iZ . Значения номеров кластеров для всех точек «сбрасываются» присвоением условного значения «-1».

В основном блоке производится обход всех точек во внешнем цикле, фиксируется очередная точка, если она не принадлежит никакому кластеру, порождается новый кластер и номер для нее, если точка уже в кластере, ее номер запоминается как текущий; производится обход всех точек во внутреннем цикле 2-го уровня, назовем эти точки тестируемыми. Проверяется, является ли тестируемая точка близкой к текущей на основе проверки условия $abs(iX - iX_{test}) \leq 1$ and $abs(iY - iY_{test}) \leq 1$ and $abs(iZ - iZ_{test}) \leq 1$. Далее, если выполняется условие, что текущая и тестируемая точка расположены по метрике Евклида ближе, чем D_{max} , то тестируемая точка либо присоединяется (если она не принадлежит ни к какому кластеру) к кластеру текущей точки, либо два кластера, текущей и тестируемой точки, объединяются. Для объединения кластеров необходим еще один внутренний цикл (3-го уровня) по просмотру номеров кластеров точек. В заключительном блоке проводится отсеивание мелких кластеров, оптимизация нумерации кластеров и массива их размерности.



Если использовать наиболее простой алгоритм, мы имеем три вложенных цикла по массиву точек (событий), вычислительная эффективность алгоритма равна $O(N^3)$, где N — количество точек (событий). Возможно улучшение эффективности алгоритма до $O(N^2 \log(N))$ при организации кластеров в списочные структуры. Если провести упорядочивание по индексу-свертке $i(x)$ и индексацию точек массива точек, то можно улучшить эффективность до $O(N \log^2(N))$. Однако каждая оптимизация по времени требует дополнительной памяти: для организации списочной структуры — 4 целочисленных массива размером в количество обрабатываемых событий, упорядочивание по индексу-свертке — еще двух таких же массивов.

Эксперименты показали, что время выполнения кластеризации на тестовом массиве 7000 точек около одной секунды на компьютере с процессором Intel® Xeon CPU W3565 3,19 GHz 12 Гб. Время кластеризации на тестовом массиве 480 000 точек составило около 3 часов.

После оптимизации алгоритма — хранение кластеров в списках и объединение их путем слияния списков (при этом при каждом объединении просмотр точек присоединяемого кластера и переназначение номера кластера) — время поиска кластеров для 480 000 точек составило 16 минут. Прогноз поиска кластеров для миллиона точек длится, таким образом, около 2 часов, что является приемлемым.

В нашей задаче целесообразен выбор D_{max} равным D или немного меньшим D .

Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (код проекта 14-07-00699).

Список литературы

1. Алсынбаев К. С., Козлов А. В. Средства распознавания и визуализации разломов и зон техногенной трещиноватости на основе обработки данных микросейсмического мониторинга // Вестник Балтийского федерального университета им. И. Канта. 2014. Вып. 4. С. 127–134.
2. Боровиков С. Н., Иванов И. Э., Крюков И. А. Построение тетраэдризации Делоне с ограничениями для тел с криволинейными границами // Журнал вычислительной математики и математической физики. 2005. Т. 45, № 8. С. 1407–1423.
3. Суков С. А. Методы генерации тетраэдральных сеток и их программные реализации // Препринты ИПМ им. М. В. Келдыша. 2015. № 23.
4. Baidurja Ray, Avi Lin, Jianfu Ma. Unconventional micro-seismicity based enhanced 3D SRV estimator using advanced parameter-free concave methodology // SEG Technical Program Expanded Abstracts. 2014. P. 2304–2308.
5. Кластеризация точек на регулярной сети. URL: <http://habrahabr.ru/post/138185/> (дата обращения: 19.08.2015).

Об авторе

Камил Салихович Алсынбаев — канд. техн. наук, доц., Балтийский федеральный университет им. И. Канта, Калининград.
E-mail: KAlsynbaev@kantiana.ru

About author

Dr Kamil Alsynbaev — Ass. Prof., head of the laboratory, I. Kant Baltic Federal University, Kaliningrad.
E-mail: KAlsynbaev@kantiana.ru