Сеточная кластеризация данных высокой размерности для сегментации мультиспектральных спутниковых изображений

С.А. Рылов[1]

[1]Институт вычислительных технологий СО РАН, пр. Академика Лаврентьева 6, Новосибирск, Россия, 630090

Аннотация. В работе рассматривается проблема выбора структуры данных для сеточных алгоритмов кластеризации в случае пространства признаков высокой размерности. Предлагается специальная структура данных, в которой хранятся только непустые клетки, что позволяет избежать больших затрат памяти. Представлены результаты сравнения предлагаемого подхода с подходом на основе хеширования клеток на мультиспектральных спутниковых изображениях, демонстрирующие преимущество разработанного подхода по затратам оперативной памяти при сопоставимом времени вычислений.

1. Введение

Задача кластеризации возникает при решении многих прикладных задач, например при сегментации спутниковых снимков. Она состоит в том, чтобы разбить множество классифицируемых объектов на сравнительно небольшое число непересекающихся подмножеств, называемых кластерами, так, чтобы каждый кластер состоял из схожих объектов, а объекты разных кластеров существенно отличались.

Сеточные алгоритмы кластеризации (grid-based) [1-2] основываются на введении сеточной структуры в пространстве признаков (разбиение пространства признаков гиперплоскостями). Предполагается, что элементы данных, попавшие в одну ячейку сетки, с высокой вероятностью принадлежат одному кластеру. Таким образом, осуществляется переход от обработки данных к обработке элементов сеточной структуры (клеток). Данный подход позволяет добиться высокой скорости обработки (линейной вычислительной сложности от объема данных) больших массивов данных и выделять кластеры сложной, заранее неизвестной формы.

В связи с этим применение сеточных алгоритмов является целесообразным при обработке спутниковых изображений, которые, как правило, состоят из миллионов пикселей, при этом априорные сведения о вероятностных характеристиках классов неизвестны. В частности, в ИВТ СО РАН было разработано семейство сеточных алгоритмов: CCA, ECCA, HCA, HECA [3-5], позволяющих выделять, в том числе многомодовые и пересекающиеся в пространстве признаков кластеры, и показавших свое преимущество перед стандартно используемыми алгоритмами кластеризации при обработке спутниковых данных.

Однако, сеточные алгоритмы являются вычислительно эффективными только при небольшой размерности пространства признаков (до 4-5 каналов изображения), что связано с экспоненциальным ростом объема сеточной структуры при увеличении размерности данных.

Помимо увеличения времени работы, наиболее существенным ограничением становится требуемый объем памяти [6]. Таким образом, на практике применение этих алгоритмов ограничено пятью каналами.

Для преодоления ограничения по затратам памяти при высокой размерности логичным выходом является переход от хранения всех клеток сеточной структуры к хранению только непустых ячеек. Но в публикациях по сеточным алгоритмам сложно найти конкретные реализации таких подходов или их показатели по затратам памяти [7]. Наиболее очевидным решением является использование хэш-таблицы для хранения элементов сеточной структуры.

В работе предлагается специальная структура данных для хранения многомерной сеточной структуры. Ее использование позволяет сеточным алгоритмам кластеризации обрабатывать данные высокой размерности (от 5 до 8) без больших затрат памяти. Данная структура реализована для сеточных алгоритмов кластеризации HCA и HECA. Представлены результаты сравнения предлагаемого подхода с прямым применением хэш-таблицы на мультиспектральных спутниковых изображениях, демонстрирующие преимущество разработанного подхода по затратам оперативной памяти при сопоставимом времени вычислений.

2. Проблема хранения сеточной структуры при обработке мультиспектральных изображений

Пусть множество классифицируемых объектов X состоит из векторов, лежащих в пространстве признаков R^D: $X = \{x_i = (x_i^1, \ldots, x_i^D) \in R^D, i = 1, \ldots, N\}$. Векторы x_i лежат в прямоугольном гиперпараллелепипеде $\Omega = [l^1, r^1] \times \ldots \times [l^D, r^D]$, где $l^j = min\, x_i^j$, $r^j = max\, x_i^j$, $x_i \in X$. Сеточная структура определяется как разбиение пространства признаков гиперплоскостями: $x^j = (r^j - l^j) \cdot i / m + l^j$, $i = 0, \ldots, m$, где m – число разбиений Ω по каждой размерности. Минимальным элементом этой структуры является клетка (замкнутый прямоугольный гиперпараллелепипед, ограниченный гиперплоскостями).

Каждая клетка характеризуется плотностью: объемом и количеством попавших в нее элементов данных. Сеточные алгоритмы разделяют клетки на плотные и неплотные или же оценивают изменения плотности между соседними клетками [8-9].

Формирование сеточной структуры схоже с построением многомерной гистограммы, но с уменьшенным числом уровней квантования значений векторных компонент элементов данных. Это в частности позволяет лучше оценивать распределение вероятностей «нереализовавшихся» значений векторов-пикселей. Стоит отметить, что проблема построения многомерных гистограмм (с квантованием и без), возникающая при обработке мультиспектральных изображений, рассматривалась в статье Денисовой А.Ю. и Сергеева В.В. [6].

Для мультиспектральных изображений классифицируемыми объектами являются пиксели изображения, а в качестве признаков выступают вектора спектральных яркостей. Таким образом, размерность пространства признаков определяется числом каналов изображения. Многие мультиспектральные съемочные системы ограничиваются набором из 4-х каналов: синий, зеленый, красный и ближний инфракрасный. Но существует и множество спутников, производящих съемку в большем числе каналов. Например, крайне востребованные данные среднего пространственного разрешения со спутников серии Landsat, а также наиболее совершенные данные высокого разрешения WorldView-2 и WorldView-3 содержат 8 каналов.

Общее число клеток сеточной структуры составляет m^D, т.е. экспоненциально зависит от размерности данных D (числа каналов изображения). Как правило, сеточная структура реализуется с помощью целочисленного массива, хранящего плотности всех клеток. Для хранения плотности клетки оптимально использовать 4 байта. Исходя из нашего опыта, при обработке мультиспектральных спутниковых изображений значение параметра m следует задавать приблизительно в рамках от 20 до 40. Таким образом, например, при m=38 для размерности D=4 число клеток составляет всего ~2 млн, в тоже время при размерности D=6 – уже ~3 млрд, что соответствует 11 гигабайтам занимаемой оперативной памяти, а при D=8 речь идет уже о терабайтах (см таблицу 1).

Таблица 1. Число клеток сеточной структуры и занимаемый объем памяти в зависимости от размерности данных (значения приблизительные).

m=38	D=4	D=5	D=6	D=7	D=8
Число клеток	2 млн	79 млн	3 млрд	114 млрд	4 трлн
Объем памяти	8 МБ	300 МБ	11 ГБ	426 ГБ	16 ТБ

В связи с этим большинство сеточных алгоритмов кластеризации работают с данными размерности не более 5. Существуют сеточные алгоритмы, позволяющие обрабатывать данные высокой размерности, такие как CLIQUE [10] и MAFIA [11]. Но они базируются на выделении плотных областей (интервалов) во всех одномерных проекциях данных, которые затем используются при формировании кластеров в многомерном пространстве признаков. Однако использование проекций не обеспечивает выделения всех имеющихся кластеров, что может приводить к ошибкам кластеризации [12]. Для полноценной оценки распределения плотности необходимо использовать равномерную сетку, состоящую из клеток одинакового объема.

Помимо хранения информации о плотностях клеток сеточная структура должна обеспечивать быстрый доступ к этим значениям по многомерным координатам клетки. Кроме того, при реализации алгоритмов кластеризации требуются такие операции, как перебор всех непустых клеток и определение принадлежности клетки к кластеру. Эти моменты необходимо учитывать при создании структуры данных для хранения многомерной сеточной структуры.

3. Предлагаемый подход для хранения многомерной сеточной структуры

Поскольку число непустых клеток не может превосходить числа обрабатываемых элементов (пикселей изображения) N, размер сеточной структуры можно также ограничить числом N, если реализовать хранение только клеток с ненулевой плотностью. Однако время, затрачиваемое на прямое построение такой структуры, иногда называемой «гистограммой-списком», пропорционально квадрату объема данных, что является неприемлемым для обработки изображений [6]. Логичным способом ускорения данной операции является использование хэш-таблиц с хешированием элементов сеточной структуры. Данный подход был также нами реализован и использовался для сравнения.

Для хранения сеточной структуры высокой размерности (от 5 до 8) нами предлагается специальная структура данных, основанная на хранении подпространства сеточной структуры размерности 4.

На первом этапе происходит построение четырехмерной сеточной структуры по первым четырем координатам векторов-признаков (каналам изображения). Вычисляются плотности соответствующих клеток (число попавших в клетку элементов данных). Данная операция происходит с помощью стандартного подхода с использованием массива G4 длины m^4, хранящего плотности всех клеток. Для размерности 4 затраты памяти и времени крайне малы.

Затем выделяются три массива длины N для хранения информации о плотностях (PL), координатах (CO) и номерах кластеров (CL) непустых клеток D-мерной сеточной структуры. Исходя из соображения, что число непустых клеток с одинаковыми значениями первых четырех координат не может превышать значение плотности соответствующей четырехмерной клетки, массив плотностей четырехмерных клеток G4 преобразуется в указатели на индексы этих массивов. Таким образом, его элементы указывают на неупорядоченные списки непустых клеток с соответствующими первыми четырьмя координатами (если такие имеются).

При доступе к произвольной клетке сначала по ее первым четырем координатам вычисляется индекс в массиве G4. Значение соответствующего элемента является индексом первого элемента списка непустых клеток в массиве координат CO. Последовательно проходя по элементам этого списка, находится искомая клетка с совпадающими координатами. По индексу найденной клетки можно узнать или изменить ее плотность или номер кластера в массивах PL и CL. Если в рассматриваемом списке не обнаружилось искомой клетки, значит она пустая. Добавление новых непустых клеток осуществляется записью в конец списка.

В массиве координат CO хранятся номера клеток, полученные по последним координатам клетки, без учета первых четырех. Это позволяет обойтись 32-битовым типом данных. В итоге, в предлагаемой структуре данных координаты клетки разбиваются на две части: первые четыре размерности определяют индекс в массиве G4, а остальные – значение в массиве CO. Данный подход позволяет гарантированно справляться с данными размерности вплоть до 8. Обработка данных более высокой размерности возможна, но в таком случае необходимо жестко ограничивать параметр m или расширять используемый тип данных.

Для хранения предлагаемой структуры данных требуется выделение памяти под 3 целочисленных массива длины N и один массив длины m^4. Для всех массивов используется 4-х байтовый тип данных. При этом в стандартной реализации для хранения той же информации требуется 2 массива длины m^D (для хранения плотностей и номеров кластеров всех клеток). Соответственно, объем требуемой памяти для предлагаемой структуры данных уже не зависит от размерности D и составляет $4\cdot(m^4+3\cdot N)$ байт вместо $4\cdot(2\cdot m^D)$ байт в случае хранения всех клеток. Для обработки данных размерности более 8 потребуется лишь изменить тип данных массива координат CO на 64-битный, таким образом затраты памяти составят $4\cdot(m^4+4\cdot N)$ байт, что не сможет принципиально повлиять на эффективность разработанного подхода. Исходя из этих оценок ясно, что новый подход разумно применять при размерностях данных более 5, если речь идет об обработке изображений размером в десятки миллионов пикселей или меньше.

Предложенный подход позволяет использовать сеточную структуру высокой размерности (от 5 до 8) без больших затрат памяти. Кроме того, можно заранее оценить объем требуемой памяти. Разработанная структура данных была реализована для сеточных алгоритмов кластеризации HCA и HECA и может быть применена для других сеточных алгоритмов, расширяя их область применения на данные высокой размерности.

4. Результаты экспериментальных исследований

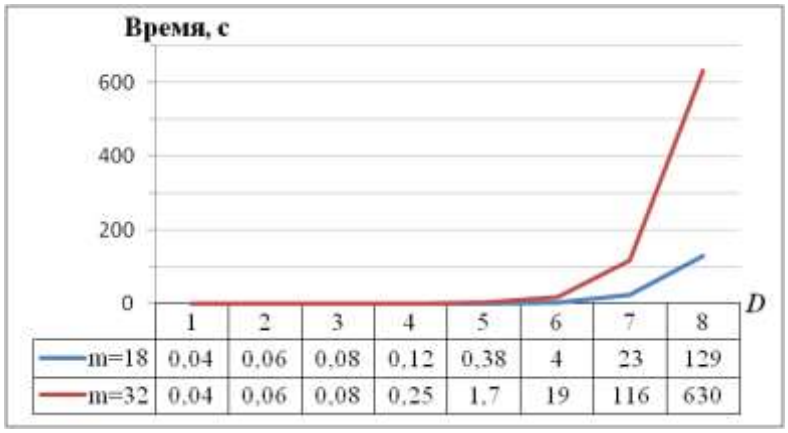
Для проведения экспериментальных исследований был взят сеточный алгоритм HCA [5], для которого была реализована предложенная структура данных. Ансамблевый алгоритм HECA сочетает результаты работы алгоритма HCA, полученные при различных значениях параметра стеки m, поэтому достаточно рассмотреть только базовый алгоритм HCA. Программная реализация выполнена на языке программирования Java. Вычисления осуществлялись на персональном компьютере с центральным процессором Intel Core i7 960, 3.2 ГГц.

На рисунке 1 представлены график и показания времени работы алгоритма кластеризации HCA при обработке различного числа каналов изображения. Выбирались первые k каналов, $k = 1,\ldots,8$. Предложенная структура данных задействовалась при $D>5$, а при небольших размерностях использовался стандартный подход. Обрабатывалось 8-канальное спутниковое изображение WorldView-2 размера 2048×2048 пикселей. Композит данного изображения и результат кластеризации по 8 каналам представлены на рисунке 2.

На графике наблюдается экспоненциальная зависимость времени обработки от числа каналов. Однако, даже на 8 каналах время обработки исчисляется минутами, что является приемлемым для большинства практических задач.

В таблице 2 также приведены показания времени работы алгоритма HCA при обработке различного числа каналов изображения, но при реализации структуры данных на основе хэш-таблицы с хешированием элементов сеточной структуры. Как показывают полученные результаты, время работы при использовании обоих подходов очень схоже.

В таблице 3 приведено сравнение фактических затрат оперативной памяти при выполнении алгоритма HCA при использовании предложенной структуры данных и структуры на основе хэш-таблицы. Объем памяти, занимаемый программой непосредственно перед запуском алгоритма (после загрузки данных и их визуализации), был вычтен. Результаты измерений демонстрируют многократное преимущество предложенного подхода по затратам памяти.



	1	2	3	4	5	6	7	8
m=18	0,04	0,06	0,08	0,12	0,38	4	23	129
m=32	0,04	0,06	0,08	0,25	1,7	19	116	630

Рисунок 1. Время работы алгоритма кластеризации HCA с предложенной многомерной структурой данных на спутниковом изображении WorldView-2 с параметром сетки m=18 и m=32 в зависимости от числа используемых каналов D.

Таблица 2. Время работы алгоритма HCA со структурой данных на основе хэш-таблицы в зависимости от числа используемых каналов (указано в секундах).

Число каналов	4	5	6	7	8
m=18	0.34	0.67	4.2	23	122
m=32	0.58	2.8	22	144	609

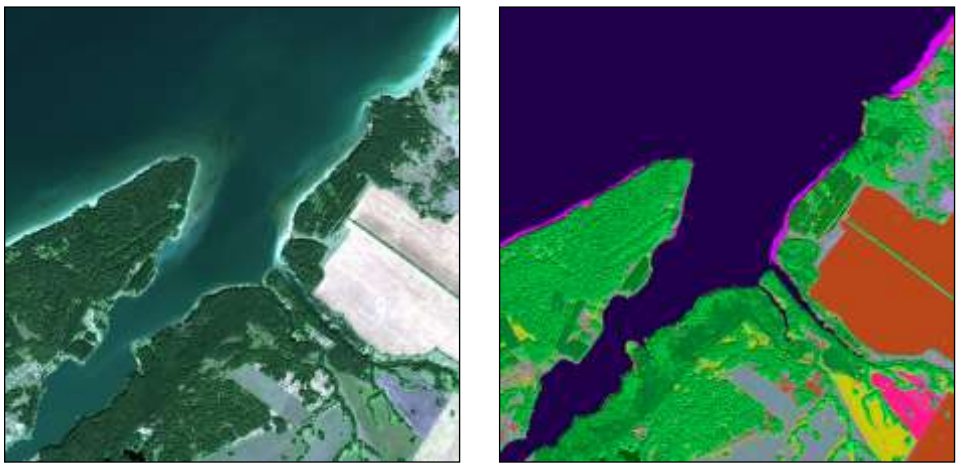


Рисунок 2. Спутниковое изображение WorldView-2 (RGB-композит, каналы 5, 3, 2) и результат кластеризации алгоритмом HCA по 8 каналам.

Таблица 3. Затраты памяти при выполнении алгоритма HCA при использовании предложенной структуры данных и структуры на основе хэш-таблицы (указано в МБ).

	Предложенный подход				Хеширование клеток			
Число каналов	5	6	7	8	5	6	7	8
m=18	64	66	66	69	186	426	1515	2029
m=32	79	115	170	251	436	1353	2024	2125

5. Заключение

В работе предложена специальная структура данных для хранения многомерной сеточной структуры. Ее использование позволяет сеточным алгоритмам кластеризации обрабатывать данные высокой размерности (от 5 до 8), в то время как стандартно используемый подход

требует для этого недопустимо больших затрат памяти. Данная структура была реализована для сеточных алгоритмов кластеризации HCA и HECA.

Было показано, что применение предложенного подхода позволяет сеточным алгоритмам обрабатывать 8-канальные спутниковые изображения. Проведенное экспериментальное сравнение разработанной структуры данных и подхода на основе хеширования показало серьезное преимущество первого по затратам оперативной памяти при сопоставимом времени вычислений. Данная разница может быть критичной при обработке изображений большого размера или, например, в случае применения ансамблевого подхода (алгоритм HECA), когда производится обработка сразу серии сеточных структур различного масштаба.

В дальнейшем планируется провести исследования по качеству кластеризации 8-канальных спутниковых снимков по сравнению с использованием меньшего числа каналов, а также использовать разработанный подход при обработке гиперспекральных изображений.

6. Литература

[1] Ilango, M.R. A survey of grid based clustering algorithms / M.R. Ilango, V. Mohan // Intern. J. Eng. Sci. and Technology. – 2010. – Vol. 2(8). – P. 3441-3446.

[2] Aggarwal, C.C. Data clustering: algorithms and applications. / C.C. Aggarwal, C.K. Reddy – CRC Press, 2014. – 626 p.

[3] Пестунов, И.А. Ансамблевый алгоритм кластеризации больших массивов данных / И.А. Пестунов, В.Б. Бериков, Е.А. Куликова, С.А. Рылов // Автометрия. – 2011. – Т. 47, № 3. – С. 49-58.

[4] Пестунов, И.А. Иерархические алгоритмы кластеризации для сегментации мультиспектральных изображений / И.А. Пестунов, С.А. Рылов, В.Б. Бериков // Автометрия. – 2015. – Т. 51, № 4. – С. 12-22.

[5] Рылов, С.А. Быстрая иерархическая кластеризация мультиспектральных изображений на графических процессорах NVIDIA / С.А. Рылов, И.А. Пестунов // Сборник трудов IV международной конференции и молодежной школы «Информационные технологии и нанотехнологии» (ИТНТ). – Самара: Новая техника, 2018. – С. 865-873.

[6] Денисова, А.Ю. Алгоритмы построения гистограмм многоканальных изображений с использованием иерархических структур данных / А.Ю. Денисова, В.В. Сергеев // Компьютерная оптика. – 2016. – Т. 40, № 4. – С. 535-542. DOI: 10.18287/2412-6179-2016-40-4-535-542.

[7] Zhuo, C. A Fast Incremental Clustering Algorithm Based on Grid and Density / C. Zhuo, L. Xiang-Shuang, Z. Xiao-Dong // Proc. Third International Conference on Natural Computation (ICNC) IEEE. – 2007. – Vol. 5. – P. 207-211.

[8] Dou, W. A half-split grid clustering algorithm by simulating cell division / W. Dou, J. Hu // Proc. Int. Joint Conf. on Neural Networks (IJCNN), 2014. – P. 2183-2189.

[9] Esfandani, G. GDCLU: a new Grid-Density based CLUstring algorithm / G. Esfandani, M. Sayyadi, A. Namadchian // Proc. 13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel & Distributed Computing (SNPD), 2012. – P. 102-107.

[10] Agrawal, R. Automatic subspace clustering of high dimensional data for data mining applications / R. Agrawal, J. Gehrke, D. Gunopulos, P. Raghavan // Proc. ACM SIGMOD International Conference on Management of Data – ACM, 1998. – Vol. 27(2). – P. 94-105.

[11] Nagesh, H.S. Adaptive Grids for Clustering Massive Data Sets / H.S. Nagesh, S. Goil, A.N. Choudhary // Proc. 1st SIAM Int. Conf. on Data Mining (SDM), 2001. – P. 1-17.

[12] Sarmah, S. A grid-density based technique for finding clusters in satellite image / S. Sarmah, D.K. Bhattacharyya // Pattern Recognition Letters. – 2012. – Vol. 33(5). – P. 589-604.

High-dimensional grid-based clustering for multispectral satellite image segmentation

S.A. Rylov[1]

[1]Institute of computational technologies SB RAS, Lavrentiev ave. 6, Novosibirsk, Russia, 630090

Abstract. The paper considers the problem of optimal data structure choice for grid-based clustering algorithms in high dimensional case. A special data structure is proposed that stores only non-empty cells avoiding large memory costs. The results of comparing the proposed approach with the simple use of a hash table on multispectral satellite images are presented. They demonstrate a great advantage of the developed method in memory costs with comparable computation time.