

АЛГОРИТМЫ ПОСТРОЕНИЯ ГИСТОГРАММ МНОГОКАНАЛЬНЫХ ИЗОБРАЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ ИЕРАРХИЧЕСКИХ СТРУКТУР ДАННЫХ

А.Ю. Денисова^{1,2}, В.В. Сергеев^{1,2}

¹ Самарский национальный исследовательский университет имени академика С.П. Королева, Самара, Россия,

² Институт систем обработки изображений РАН – филиал ФНИЦ «Кристаллография и фотоника» РАН, Самара, Россия

Аннотация

В статье представлены новые алгоритмы построения и хранения гистограмм многоканальных изображений на основе иерархических структур данных. Разработанные методы позволяют оперировать гистограммами многоканальных изображений с меньшими затратами памяти и большей скоростью, чем непосредственное построение гистограммы в виде таблицы всех возможных значений пикселей с их частотами. Даны теоретические и экспериментальные оценки объемов памяти и времени построения гистограммы. Рассмотрено практическое применение предложенных алгоритмов для оценки с различной точностью многомерной плотности вероятностей многоканальных изображений дистанционного зондирования Земли.

Ключевые слова: многоканальные изображения, гистограмма, иерархическая структура данных, дерево, список.

Цитирование: 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.

Введение

Построение гистограмм является одним из этапов многих методов обработки различного рода данных [1]. В круге задач обработки многоканальных (содержащих несколько спектральных компонент) изображений наиболее востребованными приложениями гистограмм являются кластеризация и сегментация [2 – 4], распознавание образов и классификация [5].

Для монохромного (одноканального) изображения построение гистограммы является простейшей процедурой. Так, для стандартного байтового формата данных достаточно организовать и заполнить массив из 256 элементов, соответствующих возможным значениям пикселей. Однако для цветных (RGB) изображений подобный массив превращается уже в трехмерную таблицу из $256^3 = 16\,777\,216$ ячеек. А для еще большего числа каналов (4, 6, 8 и т.д.), характерного для космических данных дистанционного зондирования Земли (ДЗЗ), формирование многомерной таблицы становится практически невозможным.

Для экономии памяти при построении гистограмм многоканальных изображений используются различные подходы. Во-первых, это дополнительное квантование (загрубление) каждой компоненты многомерных (векторных) пикселей с целью уменьшения множества их значений [5]. Однако данный подход применим для изображений со сравнительно малым числом каналов, поскольку сохраняется экспоненциальная зависимость объема памяти, требуемого для хранения гистограммы, от размерности векторов-пикселей. Кроме того, гистограмма квантованных пикселей может оказаться слишком грубой аппроксимацией исходной гистограммы, т.е. непригодной для решения некоторых задач обработки изображений.

Другим методом является представление многомерной гистограммы в виде списка уникальных значений векторов-пикселей и соответствующих им частот [2 – 4, 6]. В этом случае длина списка заведомо

мо не может быть больше размера (числа пикселей) изображения, а зависимость объема памяти от числа каналов становится линейной. Однако принципиальным недостатком многомерной гистограммы, представленной списком уникальных значений векторов-пикселей, является отсутствие у нее аппроксимирующих свойств, т.е. невозможность оценки вероятности «нереализовавшихся» значений векторов-пикселей.

В настоящей работе предлагаются алгоритмы построения гистограммы многоканальных изображений, позволяющие за счёт использования иерархических структур данных сократить объём памяти и вычислительные затраты, необходимые для построения гистограммы и ее использования в прикладных задачах. Многие алгоритмы кластеризации и сегментации изображений применяют иерархические структуры данных уже после построения гистограмм, на этапе их обработки для поиска «соседних» значений (см., например, [2, 3, 6]). Использование же иерархических структур для построения и хранения гистограмм многоканальных изображений является новым подходом.

Разработанные алгоритмы позволяют как производить построение гистограммы с единичным интервалом по каждой компоненте изображения, так и получать аппроксимацию гистограммы с адаптивным по частоте размером ячейки (ячейки с меньшими частотами имеют больший размер и наоборот). Кроме того, предложенная иерархическая структура имеет известный размер ячейки на каждом уровне, что позволяет легко выполнять оценку вероятности нереализовавшихся значений в отличие от представления гистограммы в виде списка уникальных значений.

1. Структуры данных для хранения гистограмм изображений

Рассмотрим подробнее некоторые известные структуры данных, используемые для представления гистограмм многоканальных изображений.

Для некоторой выборки $\hat{X} = \{x_0, x_2, \dots, x_{N-1}\}$ значений многомерной случайной величины $X \in R^L$ гистограмма $H_X(i) = v_i$ определяется как значение частоты v_i (или, что эквивалентно, числа) попаданий элементов выборки в некоторый интервал Δ_i , принадлежащий разбиению $\{\Delta_i\}_{i=1, \dots, l}$ области значений D случайной величины: $\bigcup_{i=1, \dots, l} \Delta_i = D$; $\Delta_i \cap \Delta_j = \emptyset, i \neq j$, где i – номер интервала [7].

Для многоканальных изображений в качестве случайной величины рассматривается вектор яркостей пикселя в каждой точке $x_n, 0 \leq n \leq N-1$, где N – размер (число пикселей) изображения. Для цифровых изображений область значений D определяется числом каналов изображения L и числом B , отводимых на целочисленное кодирование яркости в каждом канале.

Наиболее подробная гистограмма изображения соответствует единичному размеру ячейки гистограммы по каждой компоненте изображения. В таком случае область определения гистограммы, представляющая собой гиперкуб, содержит $K = 2^{BL}$ возможных значений. Простейшим способом представления гистограммы является многомерная таблица с K ячейками, хранящими целочисленное значение частоты яркости пикселя. Ниже для краткости будем называть такую таблицу «гистограмма-гиперкуб».

Значение частоты в каждой ячейке, очевидно, не может превышать числа пикселей изображения – N . Таким образом, объём памяти V_1 , отводимый для хранения гистограммы-гиперкуба, можно оценить следующей величиной (здесь и везде далее будем измерять ее в байтах)

$$V_1 = (2^{BL} \lceil \log_2 N \rceil) / 8, \tag{1}$$

где $\lceil \cdot \rceil$ – операция округления до ближайшего целого, большего или равного заданному числу.

Очевидно, что размер таблицы K , используемой для хранения гистограммы, увеличивается экспоненциально с ростом числа каналов L и разрядности B , что делает невозможным использование полной таблицы частот для гистограммы изображения с большим числом каналов.

В качестве примера рассмотрим стандартный режим работы процесса в 32-разрядной операционной системе Windows. Известно, что в этом случае на один процесс выделяется всего 2 Гб (2,147E+09 байт) памяти [8]. Если размер изображения составляет 4096×4096 точки по вертикали и по горизонтали (большинство практически значимых случаев для изображений ДЗЗ), то для хранения значений частоты достаточно 3 байта. Умножая это значение на размер полной таблицы частот яркостей, получим объём данных, требуемый в данном случае для хранения гистограммы. Ограничение на объём памяти, выделяемый для одного процесса, делает невозможным хранение гистограммы-гиперкуба изображения при значениях B и L , соответствующих ячейкам, отмеченным в табл. 1 жирным шрифтом.

Простейшими способами преодоления ограничений, связанных с объёмом данных, являются аппроксимация гистограммы путём укрупнения ее интервалов (применение квантования компонент изображения, т.е. уменьшения B) и использование сокращённой гистограммы, содержащей только частоты реализовавшихся значений.

Табл. 1. Объём памяти гистограммы-гиперкуба (байт) при различных параметрах изображений

B	L			
	1	2	3	4
8	768	196608	50331648	1,29E+10
10	3072	3145728	3,22E+09	3,30E+12
12	12288	50331648	2,06E+11	8,44E+14
16	196608	1,29E+10	8,44E+14	5,53E+19

Недостатком первого способа является низкая точность получаемой гистограммы с точки зрения оценки распределения вероятностей яркостей исходного (неквантованного) изображения. Кроме того, квантование, осуществляемое в разумных пределах, не может принципиально отодвинуть ограничение по числу каналов изображения. Так, для рассмотренного выше примера зависимость максимально возможного числа каналов L от разрядности B иллюстрируется табл. 2.

Табл. 2. Зависимость максимально возможного числа каналов L от разрядности B

B	16	12	10	8	6	5	4	3	2
L	1	2	2	3	4	5	7	8	14

Из таблицы видно, что построение гистограммы для изображения с числом каналов $L \geq 6$ возможно только для практически неинтересных случаев $B \geq 4$.

Второй способ выглядит более перспективным, поэтому рассмотрим его подробнее. Поскольку число ненулевых элементов гистограммы-гиперкуба заведомо не превышает размера изображения N , то гистограмма может быть вычислена и сохранена в сокращённом виде без нулевых (пустых) ячеек. В этом случае требуется хранить только список уникальных значений векторов яркости и соответствующие этим значениям частоты. Ниже для краткости будем называть такую структуру данных «гистограмма-список».

Очевидно, что гистограмма-список соответствует наименьшему объёму данных, которым можно представить гистограмму без применения специальных способов дополнительного сжатия. Затраты памяти V_2 можно в этом случае оценить сверху как

$$V_2 = (N \lceil (\log_2 N + BL) \rceil) / 8 \tag{2}$$

байт.

Время, необходимое для построения гистограммы-списка, можно оценить через количество выполняемых операций целочисленного сравнения. Очевидно, что в наихудшем случае, когда все пиксели изображения уникальны, количество элементарных операций сравнения пар целых чисел составит:

$$U_1(N) = L(1 + 2 + 3 + \dots + (N-1)) = LN/2 \cdot (N-1) \tag{3}$$

(в этой формуле учтено, что операция сравнения двух векторов-пикселей представляет собой L сравнений каждой компоненты).

Из формулы (3) видно, что время, затрачиваемое на построение гистограммы-списка, пропорционально квадрату объёма данных, при значительных размерах изображения оно может оказаться неприемлемо большим. Существуют различные способы ускорения построения этого списка, например, хэширование компонент вектора яркости и построение таблиц поиска с полученным хеш-значением в качестве ключа [2, 4]. Однако в любом случае принципиальным недостатком гистограммы-списка является невозможность оценки вероятности векторов яркости, не реализовавшихся на конкретном анализируемом изображении, каковых с увеличением числа каналов изображения становится подавляющее большинство (например, для 4-компонентного изображения размером 4096×4096 в байтовом формате – не менее 99,6 %).

Ниже в качестве базы для сравнения с предлагаемой структурой данных мы будем использовать гистограмму-список.

2. Иерархическая структура данных для многомерной гистограммы

В настоящей работе для построения гистограммы предлагается использовать структуру данных – дерево, которое строится в соответствии с битовым представлением пикселя в каждом канале изображения. Назовем такую структуру «гистограмма-дерево».

Пусть каждый пиксель представляется в виде $x_n = (x_{n0}, \dots, x_{nL-1})$, где $x_{ns} \in [0, 2^B - 1]$, $s = 0, L - 1$. Нулевой уровень $r = 0$ дерева содержит один корневой узел. Каждый уровень $1 \leq r \leq B$ дерева будем ставить в соответствие определённой битовой плоскости вектора x_n . Число $q^{(r)}$, сформированное из r -х бит в каждом канале изображения, является номером ветви, к которой относится значение x_n на уровне r , и назовем *маской узла*. Последовательность масок узлов $q^{(1)}, q^{(2)}, \dots, q^{(B)}$ представляет собой путь к терминальному узлу, соответствующему заданному, однозначно определяемому значению вектора яркости. Узел дерева на произвольном r -м уровне хранит два целых числа: маску узла $q^{(r)}$ и «частоту узла», которая равна числу пикселей изображения, первые r бит которых определяются масками $q^{(1)}, q^{(2)}, \dots, q^{(r)}$ узлов на пути к рассматриваемому узлу от корневого.

Описанная структура представляет собой 2^L -е несбалансированное дерево глубины B . Например, для случая, когда изображение представлено в цветовой схеме RGB с 8 битами в каждом канале ($L = 3$, $x_{ns} \in [0, 255]$, $s = 0, L - 1$), структура данных представляет собой октодерево глубины 8 с номерами ветвей от 0 до 7. На первом уровне в нулевой ветви будут сосредоточены значения пикселей, имеющие в каждом канале значения от 0 до 127, в первой ветви значения от 0 до 127 в красном и зелёном каналах и от 128 до 255 в синем канале и т.д. Схематически дерево для данного примера представлено на рис. 1.

3. Алгоритмы построения гистограммы в виде иерархической структуры данных

В данной работе предлагаются два алгоритма построения гистограммы-дерева.

Первый из них – алгоритм «в глубину» выполняет достраивание дерева до максимального уровня для каждого рассматриваемого пикселя. Пиксели изображения просматриваются последовательно, и для каждого из них выполняется следующее:

1. Инициализируется счётчик текущего уровня дерева $r = 0$.
2. Для текущего узла увеличивается значение частоты на единицу и извлекается из вектора яркости битовая плоскость $r + 1$, которой соответствует маска узла $q^{(r+1)}$.
3. Если среди потомков текущего узла, расположенных на уровне $r + 1$, есть узел с маской $q^{(r+1)}$, то осуществляется переход к этому потомку.
4. Если среди потомков текущего узла, расположенных на уровне $r + 1$, отсутствует узел с маской $q^{(r+1)}$, то такой узел создаётся и производится переход к этому узлу.
5. Если максимальное значение номера уровня не достигнуто, то выполняются шаги 2 – 4, иначе осуществляется переход к обработке следующего пикселя изображения.

Данный алгоритм реализуется рекурсивно. Каждый пиксель изображения просматривается только один раз.

Второй алгоритм – алгоритм «в ширину». Он предполагает, что изображение просматривается B раз и каждый раз заполняется только один уровень дерева. Заполнение текущего уровня происходит аналогично предыдущему алгоритму с той лишь разницей, что новые потомки могут быть созданы только в рамках текущего заполняемого уровня дерева.

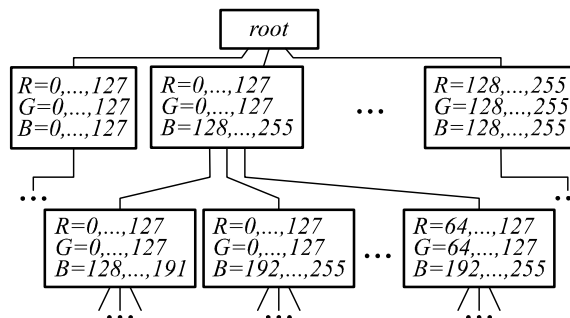


Рис. 1. Схема предлагаемой структуры данных для трехканального байтового изображения

4. Теоретическая оценка скорости построения гистограммы-дерева

Для добавления нового значения в дерево с помощью алгоритма «в глубину» необходимо выполнить не более $2^L \cdot B$ операций целочисленного сравнения, где 2^L – максимальное число масок узлов, с которыми производится сравнение на каждом уровне. В общей сложности для всего изображения число сравнений для алгоритма «в глубину» не будет превышать:

$$U_2(N) = 2^L \cdot B \cdot N. \tag{4}$$

Для алгоритма «в ширину» выполняемое количество операций сравнения можно оценить сверху следующей формулой:

$$U_3(N) = 2^L \cdot N(1+2+\dots+B) = 2^L \cdot B \cdot N \cdot (B+1)/2. \tag{5}$$

Оценка (5) вытекает из того соображения, что заполнение очередного уровня дерева r эквивалентно с точки зрения количества операций сравнения построению дерева глубины r .

Из (4) и (5) видно, что алгоритм «в глубину» теоретически работает в $((B+1)/2)$ раз быстрее алгоритма «в ширину».

Построение гистограммы-дерева алгоритмом «в глубину» является более эффективным по времени, чем построение гистограммы-списка, если выполняется следующее условие, вытекающее из выражений (3) и (4)

$$N > (2^{L+1} B)/L+1. \tag{6}$$

Для алгоритма «в ширину» аналогичное условие, получающееся из сопоставления (3) и (5), имеет вид:

$$N > (2^L B(B+1))/L+1. \tag{7}$$

Неравенства (6) и (7) показывают, что предложенные алгоритмы ориентированы на работу с большими изображениями. Они позволяют теоретически оценить преимущество построения гистограммы-списка из расчёта на самый худший случай представления данных, когда все пиксели изображения уникальны.

Например, при $B = 12$ и $L = 16$ минимальное число пикселей изображения, при котором гистограмма-дерево строится быстрее, чем гистограмма-список, для алгоритма «в ширину» составляет $N = 638977$ (т.е. изображение размерами 800×800 точек), а для алгоритма «в глубину» – $N = 98305$ (изображение 314×314 точек). При $B = 12$ и $L = 12$ для обоих алгоритмов времени дерево строится быстрее, чем список, если размеры изображений превышают 256×256 точек, т.е. во всех практически значимых случаях.

5. Теоретическая оценка объёма гистограммы-дерева

Общий объём предлагаемой структуры данных в худшем случае, когда все пиксели изображения уникальны, можно оценить величиной:

$$V = \left\lceil \frac{1}{8} (\log_2 N + L) \left[\left(\frac{2^{L(r_{\max}+1)} - 1}{2^L - 1} \right) + (B - r_{\max}) N \right] \right\rceil \tag{8}$$

байт, где $r_{\max} = \max_{1 \leq r \leq B} (2^r < N)$ – минимальный номер уровня дерева, число ветвей которого больше либо равно количеству пикселей изображения, а $r_{\max} = \lceil (\log_2 N)/L \rceil$.

Выражение (8) означает, что, начиная с некоторого уровня дерева, потенциальное число ветвей дерева превосходит количество пикселей изображения. Поскольку при построении дерева создаются узлы только с ненулевой частотой, то, начиная с этого уровня, число узлов будет ограничено размером изображе-

ния. Член $\log_2 N$ отвечает за количество бит, отводимых на хранение частоты пикселя, L бит – объём памяти, необходимый для хранения маски узла.

На рис. 2 приведен пример оценки сверху для объёмов памяти, необходимых для представления гистограммы-гиперкуба (1), гистограммы-списка (2) и в виде предложенной гистограммы-дерева (8).

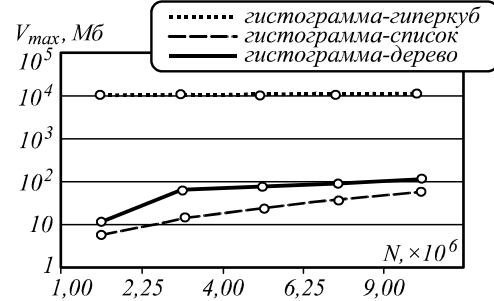


Рис. 2. Оценка сверху объёма данных при $L = 4, B = 8$ для гистограммы, представленной с использованием различных структур данных

Гистограмма-дерево требует значительно меньшего объёма памяти, чем гистограмма-гиперкуб, но, естественно, большего, чем гистограмма-список.

Заметим, что при необходимости гистограмма-список всегда может быть получена из построенной гистограммы-дерева. Для этого строится путь от корневого до терминального узла дерева, с помощью масок узлов восстанавливается значение компонент яркости в каждом канале, а соответствующее значение частоты хранится непосредственно в самом узле. Таким образом, гистограмма-дерево может рассматриваться и просто как промежуточная структура данных для ускоренного построения гистограммы-списка.

6. Аппроксимация гистограммы с помощью иерархической структуры данных

Терминальные узлы рассматриваемого дерева глубины B однозначно соответствуют отсчётам гистограммы изображения с единичным интервалом по каждой компоненте вектора-пикселя. При ограничении максимальной глубины построения дерева некоторым значением $r < B$ список терминальных узлов будет соответствовать гистограмме с интервалом 2^{B-r} . Таким образом, останавливая построение дерева на некотором уровне, можно получить аппроксимацию гистограммы, эквивалентную дополнительному квантованию (загрублению) компонент пикселей. Недостаток такого квантования был обсужден выше (см. табл. 2).

Другой, более эффективный способ получения аппроксимированной гистограммы заключается в отсеивании узлов с частотой, меньшей некоторого заданного значения v_{\min} . В таком случае узлы, соответствующие статистически наименее значимым векторам яркости, будут отброшены, что эквивалентно представлению гистограммы с неравномерным размером ячейки. При этом ячейки большего размера соответствуют менее вероятным значениям вектора яркости.

Достоинством аппроксимированной гистограммы является меньший объём результирующей структуры данных при сохранении приемлемой точности описания истинного многомерного распределения вероятности векторной случайной величины.

При использовании аппроксимации с заданным пороговым значением частоты значений яркости более эффективен алгоритм построения гистограммы «в ширину», так как он не требует создания потомков у узлов, для которых число пикселей, попавших в узел, меньше, чем v_{\min} . Алгоритм «в глубину» требует построения полного дерева с последующим отсечением узлов с частотой, меньшей v_{\min} , т.е. выполнения лишних вычислений и получения завышенного объёма данных на промежуточном этапе.

Выбор значения v_{\min} может быть сделан из соображений баланса между требованиями экономии памяти и точности описания многомерного распределения вероятности векторов-пикселей. В любом случае в практических задачах рекомендовано брать $v_{\min} > 5 \div 10$ [10].

7. Экспериментальные исследования

Целью исследований, результаты которых излагаются ниже, являлась оценка времени работы предложенных алгоритмов и объёмов памяти, занимаемых

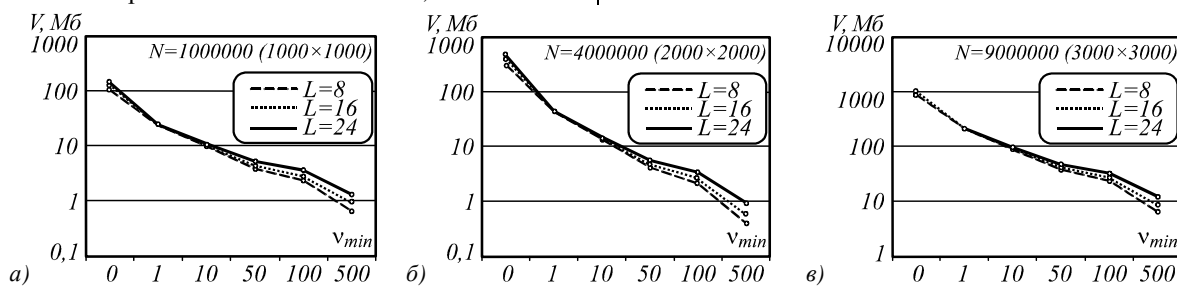


Рис. 3. Зависимость объёма памяти, необходимого для хранения гистограммы-дерева, от параметра v_{\min}

Результирующая гистограмма, создаваемая алгоритмами «в глубину» и «в ширину», имеет одинаковую структуру и, следовательно, объём данных для одних и тех же значений v_{\min} . На рис. 3 приведены графики для алгоритма «в ширину». Он позволяет построить аппроксимированную гистограмму при большем размере исходного изображения, чем алгоритм «в глубину», за счёт поуровневого заполнения структуры данных. Алгоритм «в глубину» выполняет удаление узлов, частота которых меньше либо равна v_{\min} , только после построения полного дерева, что приводит к использованию максимально возможного объёма данных даже при построении аппроксимированной гистограммы. Это ограничивает использование этого алгоритма 16 каналами (для рассматриваемых размеров изображений и разрядности).

На рис. 4 и 5 приведены результаты экспериментов по оценке времени работы обоих алгоритмов в зависимости от значения параметра v_{\min} .

Из графиков видно, что алгоритм «в глубину» работает значительно быстрее алгоритма «в ширину», причём время, необходимое на усечения дерева по

предлагаемой структурой данных. Экспериментальная реализация алгоритмов производилась на языке C++ в среде Microsoft Visual Studio 2013 в виде 32-разрядного приложения, что в стандартном режиме соответствует максимально выделяемой памяти в 2 Гб. Для исследований было использовано оборудование со следующей конфигурацией: 64-разрядная операционная система Windows 8, объём оперативной памяти – 8 Гб, процессор Intel Core i5-3470.

В качестве исходных данных использовались многоканальные изображения, полученные в виде композитов снимков сенсора MODIS космических аппаратов Terra/Aqua (продукт MOD021KM) [9]. Тестовые изображения имели размеры 1000×1000 , 2000×2000 , 3000×3000 . Число каналов L составляло 8, 16, 24. Разрядность представления каждой компоненты пикселя – $B = 12$ бит.

Приведённые ниже результаты демонстрируют работу предложенных алгоритмов построения гистограммы-дерева при использовании аппроксимации с различным ограничением минимальной частоты v_{\min} . На рис. 3 приведены результаты экспериментов по оценке объёма данных, занимаемого гистограммой-деревом в зависимости от v_{\min} .

частоте, мало и практически не влияет на скорость построения дерева. Однако при количестве каналов $L = 24$ и размере изображения $N = 9000000$ (3000×3000) из-за ограничений на объём доступной памяти построить гистограмму не удаётся.

Для алгоритма «в ширину» уже при $v_{\min} = 1$ обработка такого изображения становится возможной, при этом из рис. 5 видно, что с увеличением параметра v_{\min} время выполнения алгоритма «в ширину» постепенно снижается.

Следует отметить, что время обработки изображений размера $N = 4000000$ (2000×2000) при $L = 8$ для построения гистограммы-списка составило 27960 с, что в 300 и 2050 раз дольше, чем время построения гистограммы-дерева алгоритмами «в ширину» и «в глубину» при $v_{\min} = 0$ соответственно.

Заключение

В статье предложен подход к построению гистограммы многоканальных изображений с использованием иерархической структуры данных. Получаемая в результате применения такого подхода гистограмма-дерево обладает многими преимуществами перед

известными структурами данных, традиционно используемыми для многомерных гистограмм. Так, по сравнению с гистограммой, представленной в виде многомерной таблицы частот возможных значений яркости (гистограммой-гиперкубом), предлагаемая гистограмма-дерево требует существенно меньшего объема памяти ЭВМ для своего хранения. По сравнению с линейным списком уникальных значений век-

торов-пикселей и соответствующих им частот (гистограммой-списком) гистограмма-дерево обеспечивает радикальный выигрыш по скорости построения и, кроме того, обладает аппроксимирующими свойствами, т.е. позволяет оценивать вероятность векторов-пикселей, значения которых не встретились на конкретном анализируемом изображении.

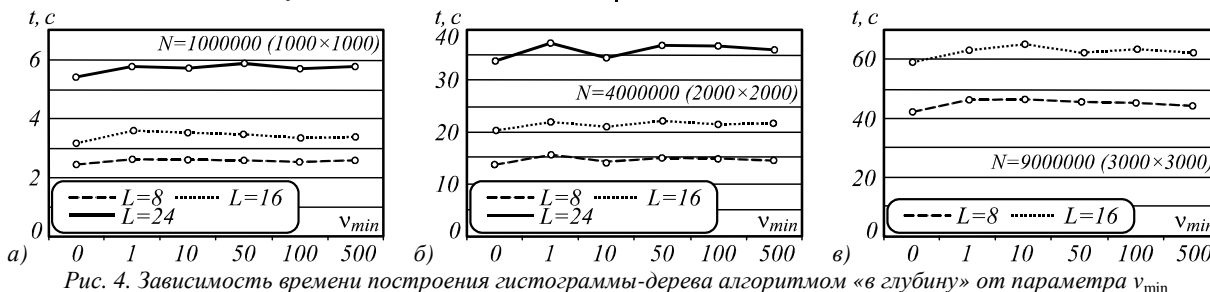


Рис. 4. Зависимость времени построения гистограммы-дерева алгоритмом «в глубину» от параметра v_{min}

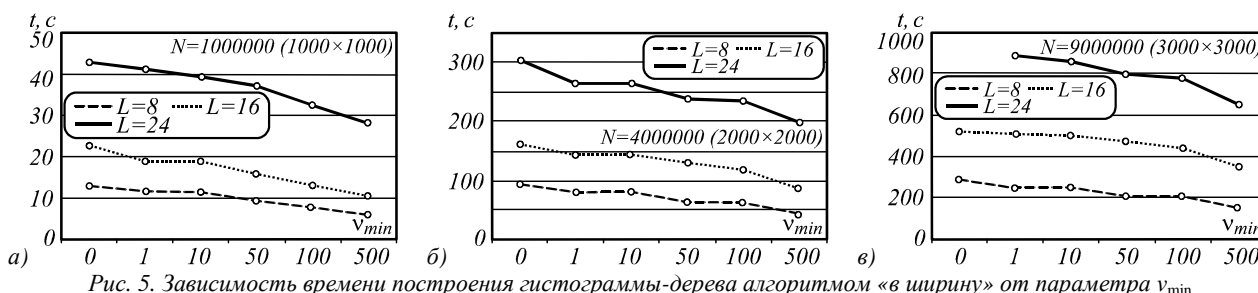


Рис. 5. Зависимость времени построения гистограммы-дерева алгоритмом «в ширину» от параметра v_{min}

Важной для практических применений является возможность построения аппроксимированной гистограммы-дерева, в которой иерархическая структура данных обрывается на узлах с частотой попаданий векторов-пикселей меньшей некоторого заданного порога. Достоинством такой гистограммы является уменьшенный объем результирующей структуры данных при сохранении приемлемой точности описания истинного многомерного распределения вероятности векторов-пикселей.

Предложены два алгоритма построения гистограммы-дерева. Алгоритм «в глубину» выполняет достраивание дерева до максимального уровня для каждого вектора-пикселя, при этом пиксели изображения просматриваются последовательно. Этот алгоритм более эффективен по скорости работы.

Алгоритм «в ширину» предполагает, что изображение просматривается многократно и каждый раз заполняется только один, следующий уровень дерева. Он имеет преимущество при построении аппроксимированной гистограммы-дерева, поскольку экономнее использует память и, как следствие, позволяет обрабатывать изображения с большим числом каналов.

Оба алгоритма работают во много раз быстрее, чем алгоритм построения гистограммы-списка.

Приведенные в статье теоретические оценки и результаты вычислительных экспериментов демонстрируют, что построение гистограммы многоканальных изображений в виде гистограммы-дерева имеет несомненные достоинства по сравнению с гистограммой-списком и тем более гистограммой-гиперкубом.

По мнению авторов, предлагаемый подход не имеет альтернатив при гистограммном анализе изображений, имеющих большое число каналов, в частности мульти- и гиперспектральных данных ДЗЗ.

Благодарности

Работа выполнена при частичной финансовой поддержке стипендии Президента РФ молодым учёным и аспирантам № СП-3322.2016.5 и Российского фонда фундаментальных исследований, грант РФФИ №16-29-09494-офи_м.

Литература

1. **Ioannidis, Y.** The history of histograms (abridged) / Y. Ioannidis // Proceedings of the 29th international conference on Very large data bases-Volume 29. – VLDB Endowment, 2003. – P. 19-30.
2. **Narendra, P.M.** A non-parametric clustering scheme for LANDSAT / P.M. Narendra, M. Goldberg // Pattern Recognition. – 1977. – Vol. 9(1). – P. 207-215. – DOI: 10.16/0031-3203(77)90005-X.
3. **Асмус, В.В.** Кластерный анализ и классификация с обучением многоспектральных данных дистанционного зондирования Земли / В.В. Асмус, А.А. Буничев, В.П. Пяткин // Техника и технологии. Engineering & Technologies. – 2009. – Т. 2, № 1. – С. 23-31.
4. **Сидорова, В.С.** Многомерная гистограмма и разделение векторного пространства признаков по унимодальным кластерам / В.С. Сидорова // Труды конференции GraphiCon. – 2005. – Т. 2005. – С. 267-274.
5. **Mason, M.** Using histograms to detect and track objects in color video / M. Mason, Z. Duric // IEEE Xplore: Applied Imagery Pattern Recognition Workshop, AIPR 2001 30th. – 2001. – P. 154-159. – DOI: 10.1109/AIPR.2001.991219.

6. **Wharton, S.W.** A generalized histogram clustering scheme for multidimensional image data / S.W. Wharton // Pattern Recognition. – 1983. – Vol. 16(2). – P. 193-199. – DOI: 10.1016/0031-3203(83)90022-5.
7. **Вентцель, Е.С.** Теория вероятностей / Е.С. Вентцель // Учеб. для вузов. – 6-е изд., стер. – М.: Высшая школа, 1999. – 576 с.
8. **Руссинович, М.** Внутреннее устройство Microsoft Windows: Windows Server 2003, Windows XP и Windows / М. Руссинович, Д. Соломон – 4-е изд. – СПб.: Питер, 2005.
9. MODIS Level 1B Product User's Guide / Members of the MODIS Characterization Support Team. – NASA/Goddard Space Flight Center, Greenbelt, MD 20771. – July 20, 2012.
10. **Денисов, В.И.** Прикладная статистика. Правила проверки согласия опытного распределения с теоретическим. Методические рекомендации. Часть I. Критерии типа χ^2 / В.И. Денисов, Б.Ю. Лемешко, С.Н. Постовалов – Новосибирск: Изд-во НГТУ, 1998. –126 с.

Сведения об авторах

Сведения об авторе **Денисова Анна Юрьевна** см. стр. 532 этого номера.

Сергеев Владислав Викторович родился в 1951 году. В 1974 году окончил Куйбышевский авиационный институт. В 1993 году защитил диссертацию на соискание степени доктора технических наук. Основное место работы – заведующий кафедрой геоинформатики и информационной безопасности Самарского национального исследовательского университета имени академика С.П. Королёва (Самарский университет). По совместительству работает заведующим лабораторией математических методов обработки изображений в Институте систем обработки изображений РАН (ИСОИ РАН) – филиале ФНИЦ «Кристаллография и фотоника» РАН. Круг научных интересов: цифровая обработка сигналов, анализ изображений, распознавание образов, геоинформатика. E-mail: vsereg@geosamara.ru.

ГРНТИ: 28.21.15, 28.17.19, 89.57.35, 89.57.45

Поступила в редакцию 18 августа 2016 г. Окончательный вариант – 29 августа 2016 г.

ALGORITHMS FOR CALCULATING MULTICHANNEL IMAGE HISTOGRAMS USING HIERARCHICAL DATA STRUCTURES

A.Y. Denisova^{1,2}, V.V. Sergeyev^{1,2}

¹ Samara National Research University, Samara, Russia,

² Image Processing Systems Institute of RAS – Branch of the FSRC “Crystallography and Photonics” RAS, Samara, Russia

Abstract

In the article we offer a novel approach to calculating multichannel image histograms using hierarchical data structures. The proposed methods result in a histogram-tree which has many advantages over existing data structures traditionally used for multidimensional histograms. The suggested data structure requires a significantly smaller memory space for storage in comparison with a histogram that is presented as a frequency table for all possible brightness values (histogram-hypercube). The histogram-tree provides a dramatic decrease in speed in comparison with the list of unique brightness values and their frequencies (histogram-list). Moreover, the histogram-tree allows one to approximate the probabilities of brightness values which are not presented in the analyzed image in contrast with the histogram-list.

The possibility of histogram approximation is very important for practical applications. Such approximation for histogram-tree is constructed by means of tree reduction. Nodes with frequencies lower than a given threshold are removed from the histogram-tree. The reduced histogram-tree benefits from the decrease of the required memory space while maintaining the accuracy of the original probability density distribution representation.

We propose two algorithms for constructing a histogram-tree. The “depth” algorithm operates with image pixels sequentially. For each pixel it constructs appropriate branches and nodes until the maximum depth of the tree is achieved. This algorithm has a recursive realization and is more time efficient than the other one. The “across” algorithm performs a multiple scan of the image, each time filling just one tree level. This algorithm allows one to operate with images with a larger number of channels than the “depth” algorithm because it is more memory efficient in the runtime when approximating a histogram. Both algorithms build the histogram-tree significantly faster when compared with the construction of the histogram-list.

The theoretical estimates and experimental results described in the article demonstrate that computing histogram of multichannel images using histogram-tree has many advantages over the histogram-hypercube and histogram-list.

Keywords: multichannel images, histogram, linked list, hierarchical data structure, non-balanced tree.

Citation: Denisova AY, Sergeyev VV. Algorithms for calculating multichannel image histogram using hierarchical data structures. Computer Optics 2016; 40(4): 535-542. DOI: 10.18287/2412-6179-2016-40-4-535-542.

Acknowledgements: The work was partially funded by the scholarship of the President of Russian Federation for postgraduate students and young researches No. SP-3322.2016.5 and RFBR project 16-29-09494 ofi_m.

References

- [1] Ioannidis Y. The history of histograms (abridged). Proceedings of the 29th international conference on Very large data bases-Volume 29. VLDB Endowment; 2003: 19-30.
- [2] Narendra PM, Goldberg M. A non-parametric clustering scheme for LANDSAT. Pattern Recognition 1977; 9(4): 207-215. DOI: 10.1016/0031-3203(77)90005-X.
- [3] Asmus VV, Bunichev AA, Pyatkin VP. The Cluster Analysis and Classification with Training of Multispectral Data of Earth Remote Sensing [in Russian]. Journal of Siberian Federal University: Engineering & Technologies 2009; 2(1): 23-31.
- [4] Sidorova VS. Multidimensional histogram and separating of the vector feature space on the unimodal clusters [in Russian]. International conference Graphicon, 2005; 2005: 267-274.
- [5] Mason M, Duric Z. Using histograms to detect and track objects in color video. Applied Imagery Pattern Recognition Workshop, AIPR 2001 30th. IEEE, 2001: 154-159. DOI: 10.1109/AIPR.2001.991219.
- [6] Wharton SW. A generalized histogram clustering scheme for multidimensional image data. Pattern Recognition 1983; 16(2): 193-199. DOI: 10.1016/0031-3203(83)90022-5.
- [7] Ventzel ES. The Theory of possibility [In Russian]. Moscow: "Vysshaya shkola" Publisher; 1999.
- [8] Russinovich M, Solomon D. Internal structure of Microsoft Windows: Windows Server 2003, Windows XP and Windows [In Russian]. Saint-Petersburg: "Piter" Publisher; 2005.
- [9] MODIS Level 1B Product User's Guide. Members of the MODIS Characterization Support Team. NASA/Goddard Space Flight Center, Greenbelt, MD 20771. July 20, 2012.
- [10] Denisov VI, Lemeshko BU, Postovalov SN. Applied statistics. The rules of coherence of experimental distribution with theoretical distribution. Guidelines. Part 1. χ^2 tests [In Russian]. Novosibirsk: "NGTU" Publisher, 1998.

Authors' information

The information about author **Anna Yurievna Denisova** you can find on page 533 of this issue.

Vladislav Victorovich Sergeev (1951 b.), graduated (1974) from S. P. Korolyov Kuibyshev Aviation Institute (presently, Samara National Research University (or shortly, Samara University)). He received his Candidate's degree in Technical Sciences in 1978 and DrSc degree in Mathematics and Physics in 1993. At present, he is the head of Geoinformation Science and Information Security sub-department at Samara University, also holding a part-time position as the head of a laboratory at the Image Processing Systems Institute of the RAS – Branch of the FSRC "Crystallography and Photonics" RAS. The areas of research interests include digital signal and image processing, geoinformatics and pattern recognition. E-mail: vserg@geosamara.ru.

Received August 18, 2016. The final version – August 29, 2016.