

СООБЩЕНИЯ
ОБЪЕДИНЕННОГО
ИНСТИТУТА
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ
ДУБНА

103/83

Э/1-83

P10-82-653

И.Байла, Г.А.Ососков, В.И.Приходько, М.Турзова

К ВОПРОСУ О СЖАТИИ ДАННЫХ
ПРИ БЕСФИЛЬМОВОМ СЪЕМЕ ИНФОРМАЦИИ
СО СТРИМЕРНЫХ КАМЕР

III. Программный имитатор АСД

1982

I. Введение

Вопросы сжатия данных при бесфильмовом съеме информации на базе телевизионных камер рассматривались в ряде работ /1-5/. Основной подход, предложенный в работах /4,5/ по бесфильмовому съему данных со стримерных камер, определяется тремя главными факторами:

1. Требованиям к быстродействию аппаратуры, используемой для сжатия данных, которая должна работать он-лайн, так как процессы съема телеизображения, сжатия данных и их записи на магнитную ленту ЭВМ совмещены во времени.
2. Штриховым характером изображения (рис. 1).
3. Требованиям сохранения физической информации в процессе сжатия и последующей реконструкции изображения.

Первый фактор почти однозначно приводит к необходимости разбиения изображения на фрагменты - слайсы и применения для сжатия специальных быстрых алгоритмов, допускающих параллелизм обработки. Второй фактор позволяет перейти от двумерного представления изображения треков на отдельном слайсе к одномерному векторному представлению /4/. Третий фактор определяется допустимой погрешностью, вносимой при сжатии и реконструкции, а также условием сохранения информации о локальной структуре треков (т.е. о наличии и величине разрывов вдоль треков), необходимой для оценки ионизации /6/.

Эти три фактора определяют и метод сжатия. Как показали исследования /5,8,9/, наиболее эффективным с точки зрения сохранения физической информации и достижения максимального коэффициента сжатия является применение линейных апертурных функций /7/ к векторам, составленным из координат точек изображения, в комбинации с запоминанием признаков неклассифицируемых объектов. Для повышения эффективности сжатия необходимо установление связей между компонентами отдельных векторов (с этой целью в работе /5/ была предложена идея специальной сортировки).

ОБЪЕДИНЕННЫЙ ИНСТИТУТ

КОМПЬЮТЕРНОЙ ТЕХНИКИ И МАТЕМАТИКИ

БИБЛИОТЕКА

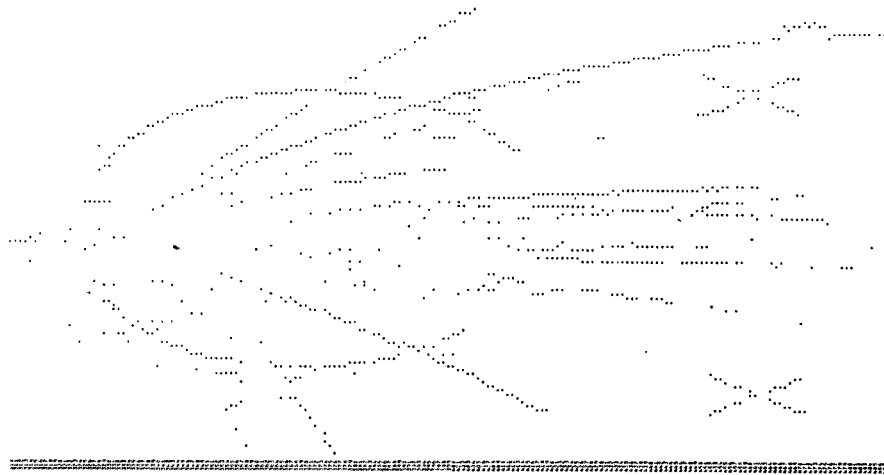


Рис. 1



Рис. 2

В настоящей работе описаны быстрые алгоритмы параллельной и последовательной сортировки и алгоритм сжатия ^{/8/}, реализованные в виде программного имитатора аппаратуры сжатия данных (спецпроцессора АСД), а также результаты исследования имитатора на реальном материале. Анализ введенных ниже количественных характеристик (гл.6) и результатов реконструкции сжатых данных (см. рис. 2) позволяет сделать вывод о приемлемости предложенных в данной работе методов сжатия трековой информации в бесфильмовых системах.

2. Основные понятия

Математическая модель процесса автоматической оцифровки изображения с помощью сканирующего прибора состоит в наложении на него решетки, размер ячеек которой определяется шагом сканирования n и шириной канала p отсчетной системы прибора. При этом каждой точке изображения ставится в соответствие значение двумерной дискретной функции раstra v , характеризующей яркость ячейки, накрывающей эту точку. Область определения функции v — произведение множеств $M \times N = \{(t, j) : t \in M, j \in N\}$, где $M = \{1, 2, \dots, 2^k\}$ определяется числом разрядов k отсчетной системы, а $N = \{1, 2, \dots, n\}$ — полным числом скан-линий. Количество значений, принимаемых функцией $v(t, j)$, обычно является степенью 2 и определяется возможностями прибора.

Однако использование раstra для кодировки изображений с трековых камер в физике высоких энергий крайне неэкономично. Более эффективной для штриховых изображений ($v = \{0, 1\}$) является векторная кодировка, введенная в ^{/4/}, т.к. она использует только то малое подмножество D точек раstra, в которых значения функции v равны 1, $D = \{(t, j) \in M \times N : v(t, j) = 1\}$, причем нумерация компонент векторов определяется естественным порядком их появления при сканировании изображения.

Если пронумеровать индексом i упорядоченные пары $(t, j) \in D$ на j -й скан-линии, следуя порядку их появления при сканировании (т.е. в порядке убывания t), то мы получим для каждого фиксированного j последовательность

$$(t_1, j)_1, (t_2, j)_2, \dots, (t_i, j)_i, \dots, (t_{m_j}, j)_{m_j} \quad (2.1)$$

случайной, но ограниченной длины $m_j \leq m$ (m — максимально допустимое число отсчетов на каждой скан-линии, оно, как правило, ограничивается регистрирующей аппаратурой). Чтобы упростить запись (2.1) и согласовать ее с обозначением для векторного представления, обозначим $x_{ij} = t_i$ (см. рис. 3).

Напомним также принятое в работе /4/ соглашение о пополнении нулевыми элементами последовательностей (2.1), для которых $m_j < m$, что обеспечивает одинаковую размерность m всех векторов-столбцов $\bar{x}_j = (x_{1j}, x_{2j}, \dots, x_{mj})$. Это позволяет сформировать из них прямоугольную матрицу X , строки которой образуют векторы $\bar{z}_i = (x_{i1}, x_{i2}, \dots, x_{in})$.

При отсутствии шумовых точек и пробелов на треках самым рациональным способом автоматического формирования векторов, состоящих из соседних точек отдельных треков (или их сегментов между возможными пересечениями), является выбор в качестве j -й компоненты вектора $\bar{z}_i = (x_{i1}, x_{i2}, \dots, x_{ij}, \dots, x_{in})$ той точки изображения, которая встретится i -й по порядку на j -й скан-линии ($j = \overline{1, n}$) рассматриваемого слайса, состоящего из n скан-линий. Однако неизбежные

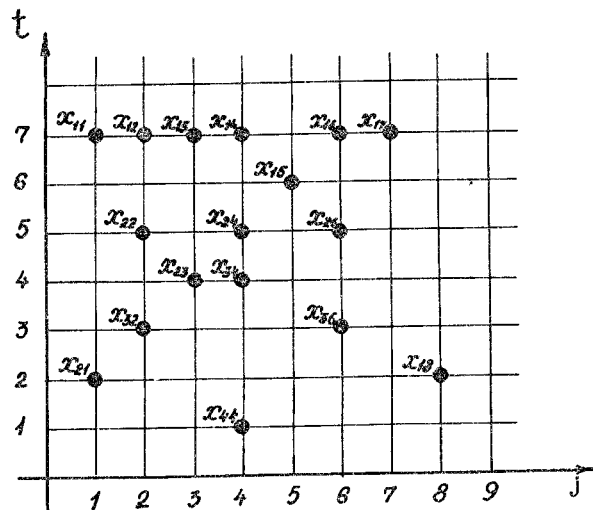


Рис. 3

на реальных изображениях пробелы и шумовые точки приводят к потере однозначности соответствия между векторами и треками, нарушению связности точек векторов, увеличению числа аппроксимирующих отрезков и, как следствие, к резкому уменьшению коэффициента сжатия.

Задачей сортировки является перераспределение точек матрицы X по векторам \bar{z}_i так, чтобы в каждом векторе было максимальное число точек, соответствующих изображению одного из треков. Переходя на язык теории множеств, мы должны осуществить такое разбиение R множества D на непересекающиеся подмножества D_α ($\bigcup_{D_\alpha \in R} D_\alpha = D$), которое минимально по числу связанных подмножеств, т.е. каждое D_α будет объединять как можно большее число точек одного трека.

Решение этой задачи требует строгого определения понятий связности точек и множеств. Определим ϵ -окрестность точки раstra

$x_{ij} \in M \times N$ для произвольного целого $\epsilon > 0$ как такое множество O_{ij}^ϵ точек раstra на следующей по порядку скан-линии, которые отстоят от x_{ij} по координате t не более, чем на ϵ , т.е.

$$O_{ij}^\epsilon = \{ (t, j+1) \in M \times N : |t - x_{ij}| \leq \epsilon \}. \quad (2.2)$$

Упорядоченную пару точек множества D $(x_{ij}, x_{r, j+1})$, для которой $x_{r, j+1} \in O_{ij}^\epsilon$, будем называть связным элементом. Точку x_{ij}^* , для которой не существует ни точки $x_{p, j-1}$, такой, что $x_{ij}^* \in O_{p, j-1}^\epsilon$, $p = \overline{1, m_j}$, ни точки $x_{r, j+1} \in O_{ij}^\epsilon$, $r = \overline{1, m_{j+1}}$, назовем изолированной. Наконец, кортеж $s = (x_{i_1, j}, x_{i_2, j+1}, \dots, x_{i_p, j+p-1})$, где $p \leq n - (j-1)$, будем называть связным сегментом, если любые две его соседние компоненты образуют связный элемент. Таким образом, для всех соседних элементов сегмента s справедлив признак близости

$$x_{i_{k+1}, j+k} \in O_{i_k, j+k-1}^\epsilon, \quad k = \overline{1, p-1}. \quad (2.3)$$

При произвольном выборе параметра ϵ возможны ситуации, когда существуют координаты x_{ij} , являющиеся одновременно компонентами нескольких связанных элементов, что влечет за собой невозможность получения непересекающихся классов разложения R . Чтобы избежать такой неоднозначности, необходимо ограничить ϵ , и договориться о порядке установления связности элементов в случаях возможных альтернатив:

1. Для $\epsilon > 0$ потребуем, чтобы в пересечении ϵ -окрестностей любых двух точек $x_{i_1, j} > x_{i_2, j}$ существовало не более одной точки $x_{i, j+1}$, т.е. $d < r$ (d - диаметр пересечения, равный $x_{i_2, j} - x_{i_1, j} + 2\epsilon$). Используя ограничение $|x_{ij} - x_{i+1, j}| \geq r$, накладываемое разрешающей способностью r , получаем

$$\epsilon < r. \quad (2.4)$$

2. Поскольку выбранное $\epsilon > 0$, удовлетворяющее (2.4), допускает существование точки $x_{i, j+1}$, входящей в ϵ -окрестности двух других точек $x_{i_1, j}$ и $x_{i_2, j}$, то условимся связным элементом в таком случае считать только упорядоченную пару $(x_{i_1, j}, x_{i, j+1})$, если $|x_{i_1, j} - x_{i, j+1}| \leq |x_{i_2, j} - x_{i, j+1}|$. Аналогично, если для точки x_{ij} на следующей скан-линии найдутся две точки, входящие в ее ϵ -окрест-

ность, $x_{i_1, j+1}$ и $x_{i_2, j+1}$, то условимся считать связным элементом пару $(x_{i_j}, x_{i_{j+1}})$ по тому же принципу ближайшей точки.

Признак близости (2.3) позволяет организовать первый этап сортировки, заключающийся в последовательном для $j=1, 2, \dots, n$ перемещении элементов матрицы $X = \{x_{ij}, i=\overline{1, m}, j=\overline{1, n}\}$ по вертикали в пределах каждого вектора-столбца \bar{x}_j ; при этом осуществляется разбиение оцифрованного изображения J на множество $\{s_k\}$ всех связанных сегментов, соответствующих густкам стримеров на треках, и отдельное множество $\{x_{ij}^*\}$ изолированных точек.

Отметим, что на первом этапе не может быть осуществлено разложение R множества D , минимальное по числу связанных подмножеств, т.к. из-за наличия разрывов на треках отдельные вектор-строки \bar{z}_i будут состоять из сегментов, относящихся к разным трекам. Поэтому вводится второй этап сортировки, на котором сегменты и изолированные точки каждого трека объединяются в один вектор. Для решения этой задачи предлагаются следующие два метода:

Метод А, осуществляющий второй этап сортировки с помощью нового критерия близости по направлению, названного признаком продолжения. Этот признак основан на предсказании положения точки i -го вектора на текущей $(j+k+1)$ -й скан-линии. Предсказание $\hat{x}_{i, j+k+1}$ определяется двумя последними сегментами s_p^i, s_ℓ^i этого вектора:

$$\hat{x}_{i, j+k+1} = x_{i, r_p} + \frac{(j+k+1-r_p)(x_{i, r_\ell} - x_{i, r_p})}{(r_\ell - r_p)}, \quad (2.5)$$

где x_{i, r_p} и x_{i, r_ℓ} - последние точки сегментов s_p^i, s_ℓ^i и $j=r_\ell$. Для общности в качестве сегментов рассматриваются и изолированные точки $x_{i, r}^*$.

Вокруг предсказанных значений $\hat{x}_{i, j+k+1}$ строятся окрестности $P_{ij}^\epsilon(k)$ (аналогично (2.2)) для $k=1, 2, \dots$:

$$P_{ij}^\epsilon(k) = \{(t, j+k+1) \in M \times N : |t - \hat{x}_{i, j+k+1}| \leq \epsilon_k\}, \quad (2.6)$$

и "направленный" коридор K_{ij}^ϵ как семейство окрестностей $P_{ij}^\epsilon(k)$:

$$K_{ij}^\epsilon = \bigcup_{k=1, 2, 3, \dots} P_{ij}^\epsilon(k), \quad k=1, 2, 3, \dots \quad (2.7)$$

Признак продолжения для любой точки $x_{q, j+k+1}$ множества D состоит в ее принадлежности к окрестности $P_{ij}^\epsilon(k)$ (см. рис. 4а).

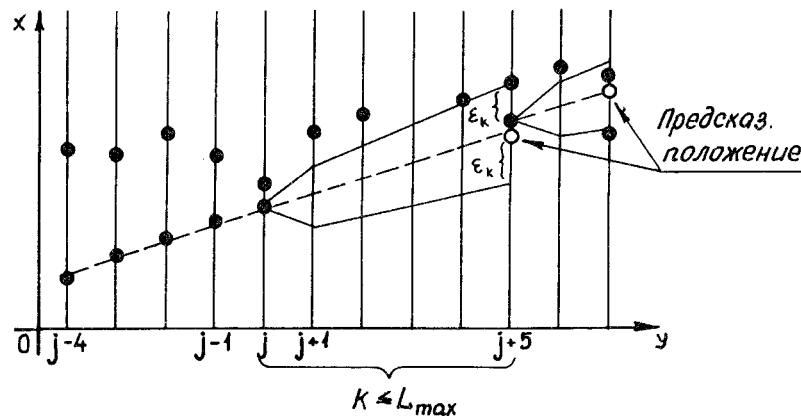


Рис. 4а

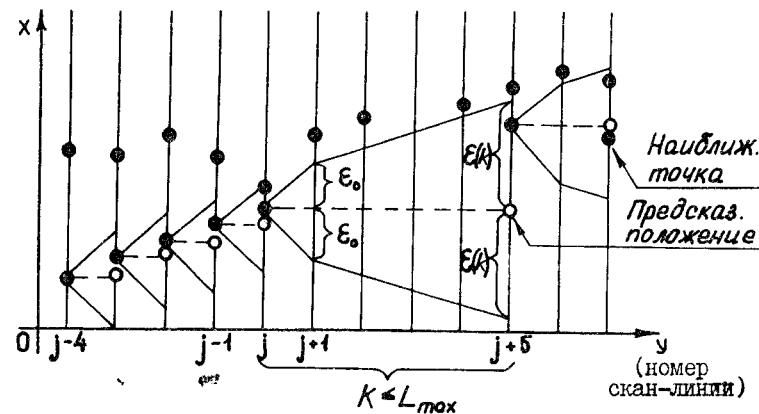


Рис. 4б

Метод Б, объединяющий оба этапа сортировки на основе одного общего признака - расширенного признака близости. Введем понятие k -расширения ϵ -окрестности на $(j+k+1)$ -ю скан-линию, $k=0, 1, 2, \dots$:

$$O_{ij}[\epsilon(k)] = \{(t, j+k+1) \in M \times N : |t - x_{ij}| \leq \epsilon(k)\}. \quad (2.8)$$

Расширенный признак близости i -го вектора для точки $x_{q,j+k+1}$ состоит в ее принадлежности окрестности $O_{ij}[\varepsilon(k)]$.

Заметим, что оба семейства ε -окрестностей (2.6) и (2.8) строятся через пробелы длины k , не превышающей некоторое выбранное значение L_{\max} (см. ниже п. 5). При включении новой точки в вектор строятся новые семейства таких ε -окрестностей (см. рис. 4а, 4б).

Хотя оба признака для методов А и Б построены аналогичным образом на основе семейств некоторого вида ε -окрестностей, отличие между ними хорошо видно из сравнения рис. 4а и 4б и состоит в том, что метод А использует дополнительную информацию о направлении трека и поэтому требует меньшего расширения ε -окрестностей с ростом длины коридора (см. дальше, п. 5). Этим обеспечивается его большая помехоустойчивость, особенно в случае наклонных или пересекающихся треков. Платой за это качество являются затраты времени на вычисление по формуле (2.5), что ведет к снижению быстродействия.

Для обоснования выбора метода сортировки, оптимального с точки зрения быстродействия и достижимого коэффициента сжатия, были проведены исследования по обоим направлениям.

3. Алгоритмы быстрой сортировки

На основе описанных методов А и Б реализованы два варианта алгоритма второго этапа сортировки. Основной чертой обоих алгоритмов является последовательная обработка векторов-столбцов \bar{x}_{j+1} (т.е. обработка скан-линий в порядке их появления), в процессе которой используется информация из предшествующих векторов-столбцов $\bar{x}_j, \bar{x}_{j-1}, \dots, \bar{x}_1$, но перестановки элементов проводятся только для текущего $(j+1)$ -го вектора. С точки зрения сохранения физической информации важным моментом является выделение несвязных, т.е. неидентифицированных в процессе сортировки, точек в массив `NOID`, который сохраняется для последующей офф-лайн обработки. При реализации алгоритмов существенно используется разбиение оцифрованного изображения на слайсы, которые обрабатываются отдельно.

Основным отличием алгоритмов, реализующих методы А и Б, помимо отмеченного выше отличия признака продолжения трека (А) от расширенного признака близости (Б), является различная организация действий (сравнений и перестановок) над элементами матрицы X .

Параллельный алгоритм реализует для каждого вектора-столбца \bar{x}_{j+1} двухэтапную сортировку согласно методу А, используя на первом этапе принцип параллельного выполнения элементарных актов сравнения с помощью набора однотипных модулей.

Последовательный алгоритм, объединяющий оба этапа сортировки согласно методу Б, использует естественную упорядоченность элементов векторов-столбцов по величине для сокращения объема требуемых операций-сравнений и перестановок элементов. Вместе с использованием более простого расширенного признака близости это позволяет применить последовательную обработку данных, значительно более простую для реализации.

Подробное описание обоих алгоритмов сортировки дано в работе /8/.

4. Метод сжатия данных

Разработка новых методов, эффективно производящих сортировку при наличии шумов и пробелов на треках, потребовала создания нового алгоритма сжатия, полностью использующего возможности повышения быстродействия и коэффициента сжатия, заключенные в структуре отсортированных данных, и одновременно менее чувствительного к шумам в сгустках стримеров.

В большинстве случаев точки одной вектор-строки после сортировки с заданной точностью аппроксимируются одним отрезком прямой. Однако необходимо учитывать и самые неблагоприятные ситуации, когда для соблюдения заданной точности требуется аппроксимация точек одного вектора двумя, тремя и более отрезками прямой. Это случается, когда в одном векторе \bar{z}_i находятся сегменты разных, как правило, пересекающихся треков или когда кривизна трека слишком большая.

Алгоритм сжатия построен по следующему иерархическому принципу: на начальном этапе сжатия определяется отрезок прямой линии, соединяющий первую и последнюю ненулевые точки вектора, и проверяется отклонение каждой ненулевой точки вектора от аппроксимирующей прямой. Если отклонения для всех точек меньше заданной величины δ , то аппроксимация принимается и ее сжатое представление запоминается. В противном случае вектор разбивается на два участка и определяются их границы (первая и последняя ненулевые точки). Для обоих участков снова проводится аппроксимация отрезком прямой с контролем точности и возможностью дальнейшего разбиения, если не удовлетворены требования точности.

Участки вектора, аппроксимируемые одним отрезком прямой и содержащие p ненулевых точек (для p больше заданного порога T); определяются как линейные трек-элементы (ЛТЭ). Участки с числом ненулевых точек $p \leq T$ перемещаются в массив неидентифицированных точек `NOID`.

Подробное описание алгоритмов сжатия с анализом условий разбиения на участки и перевода точек в `NOID` дано в /8/.

Кодировка сжатого представления (см. /8/) одного ЛТЭ выбрана в виде последовательности шести величин:

$$\{j_s, x_s, dx, j_F, N_z, N_B\}, \quad (4.1)$$

где j_s, j_F - коорд. у первой и последней ненулевых точек участка,
 x_s - коорд. x первой точки, $x_s = x_{j_s}$,
 dx - разница коорд. x последней и первой точек $dx = x_F - x_s$,
 N_z - число нулевых элементов в участке вектора,
 N_B - число сегментов (сгустков) в участке.

5. Критерии выбора параметров

Эффективность работы алгоритмов сортировки и сжатия определяется главным образом правильным выбором параметров АСД в зависимости от характеристик телевизионной системы (шаг сканирования, разрешающая способность и др.), а также от конкретных свойств обрабатываемых данных, в том числе от величины случайного разброса центров стримеров в направлении, перпендикулярном к оси треков, который характеризуется для оцифрованных данных дисперсией σ_s^2 .

При выборе параметра ε для алгоритмов сортировки должен быть учтен также угол φ наклона трека в пределах слайса, определяемый приращением $\Delta x(\varphi)$ координаты x для соседних точек (на соседних скан-линиях) трека. Конкретно, для величины ε в признаке близости (2.3) помимо ограничения (2.4) должно выполняться неравенство $\varepsilon(\varphi) \geq \varepsilon_\Delta(\varphi) + \varepsilon_s(\varphi)$, где $\varepsilon_\Delta(\varphi) = \Delta x(\varphi)$ учитывает наклон трека φ , а $\varepsilon_s(\varphi) = 2\sigma_s / \cos \varphi$ учитывает шум стримеров (с уровнем достоверности 96%). Чтобы сделать это выражение независимым от φ , вводится максимальный угол наклона трека, при котором его точки еще будут связываться (точки треков с $\varphi > \varphi_{MAX}$ попадают в массив NOID).

В целях сохранения сравнимой точности по обеим осям координат обычно задают $\varphi_{MAX} = 45^\circ$, что дает для ε следующую оценку снизу:

$$\varepsilon \geq \varepsilon_\Delta + \varepsilon_s, \quad (5.1)$$

где $\varepsilon_\Delta = \Delta x(45^\circ)$ и $\varepsilon_s = 2\sqrt{2}\sigma_s$.

Критерий выбора значений ε_k для метода А (2.6) не должен учитывать угол наклона трека, который входит в (2.6) в виде предсказания (2.5), но зато в этом случае необходимо учитывать

ошибку, вызванную разбросом точек трека, входящих в (2.5). Для самого неблагоприятного случая ($\varphi = \varphi_{MAX} = 45^\circ$, $r_1 - r_p = 2$) получаем для ε_k линейную функцию длины пробела $k = 1, 2, \dots$:

$$\varepsilon_k \geq 2\varepsilon_s + k\varepsilon_s. \quad (5.2)$$

При выборе значений $\varepsilon(k)$ для метода Б (2.8) необходимо компенсировать отсутствие информации о наклоне трека φ . При $\varphi = \varphi_{MAX} = 45^\circ$ получаем для $k = 0, 1, 2, \dots$

$$\varepsilon(k) \geq \varepsilon_0 + k \cdot \varepsilon_\Delta, \quad \varepsilon_0 \geq \varepsilon_\Delta + \varepsilon_s. \quad (5.3)$$

Отметим, что если $\varepsilon = \varepsilon_0$, то расширенный признак близости (2.8) для соседних скан-линий ($k=0$) совпадает с признаком близости (2.3) и $\varepsilon(0) = \varepsilon_0 = \varepsilon$.

Величина ε_Δ значительно превышает ε_s , чем объясняется различие в ширине коридоров для методов А и Б (см. рис. 4а, 4б).

Последний параметр алгоритмов сортировки L_{MAX} (максимальную длину пробела, после которой прекращается поиск продолжения трека) следует выбирать на основе статистических оценок средней длины пробела \bar{L} и дисперсии σ_L^2 , $L_{MAX} \leq \bar{L} + 3\sigma_L$ (т.к. более длинные пробелы встречаются очень редко). Кроме того, полезны некоторые эвристические критерии:

1) $L_{MAX} < L_T$, где L_T находится из уравнения $2 \cdot \varepsilon(L_T) = \frac{2^q}{M_T}$ (M_T - максимально допустимое регистрирующей аппаратурой число треков, q - число разрядов счетчика по оси x); 2) $L_{MAX} < \frac{2}{3}n$ (n - длина слайса). Совместив все эти ограничения на L_{MAX} , получаем

$$L_{MAX} \leq \min(\bar{L} + 3\sigma_L, L_T, \frac{2}{3}n). \quad (5.4)$$

Если значения \bar{L} и σ_L^2 неизвестны, то \min следует брать по двум последним аргументам.

Работа алгоритма сжатия существенно зависит от подбора соответствующего значения параметра δ , определяющего точность аппроксимации и в связи с этим степень сохранения геометрических параметров изображения (например, для определения сагитты s трека с наклоном $\varphi < \varphi_{MAX} = 45^\circ$ необходимо, чтобы $2\delta < s / \cos \varphi_{MAX}$). В то же время от выбора δ сильно зависит число аппроксимирующих отрезков и степень

сжатия. Для того чтобы ограничить число разбиений ЛТЭ из-за шума точек σ_s , вводим следующее ограничение снизу:

$$\delta > \varepsilon_s. \quad (5.5)$$

Минимальное число точек Т в одном ЛТЭ следует выбирать из соотношения объемов памяти, которая требовалась бы для хранения Т точек либо в массиве NOID, либо в виде сжатого представления ЛТЭ (4.1):

$$T \cdot MV(\text{NOID}) \approx MV(\text{ЛТЭ}), \quad (5.6)$$

где через mv обозначен объем памяти АСД, требуемый для одного элемента массива NOID или для сжатого представления одного ЛТЭ соответственно.

Эффективность предложенных алгоритмов сжатия с увеличением длины слайса n повышается (это связано с тем, что ЛТЭ на границах слайсов автоматически обрываются); таким образом, n ограничено только возможностями, которые дает объем памяти АСД.

6. Количественные характеристики алгоритмов сжатия

Для количественной оценки работы АСД необходимо использовать те из возможных численных характеристик, которые позволяют оценить степень сжатия, а также степень сохранения физической информации, не проводя при этом слишком громоздких вычислений. Аналогично предыдущему исследованию /9/ предложим следующие четыре характеристики:

1. Коэффициент сжатия
$$\rho_1 = \frac{V_1(N_{\text{SUM}} + N_{\text{SC}})}{V_2 N_L + V_3 N_{\text{NOID}}}$$

где N_{SUM} - число точек исходного изображения, N_{SC} - число скан-линий, N_L - число ЛТЭ после сжатия, V_1, V_2, V_3 - коэффициенты, учитывающие соотношение объемов памяти для исходных данных, сжатого представления и массива NOID соответственно.

2. Коэффициент сжатия
$$\rho_2 = \frac{V_1(N_{\text{SUM}} + N_{\text{SC}})}{V_2 N_L}$$

(коэффициент сжатия ρ_2 полезен в ситуациях, когда массив NOID может быть отброшен без существенной потери физической информации).

3. Относительное число неидентифицированных точек
$$\lambda = N_{\text{NOID}}/N_{\text{SUM}}$$

(N_{SUM} - число точек исходного изображения, N_{NOID} - число точек в массиве NOID).

4. Среднеквадратическое отклонение точек исходного изображения от прямых, аппроксимирующих ЛТЭ /8/, σ .

Коэффициенты сжатия выбраны так, что они определяют, во сколько раз уменьшается объем памяти для записи данных при переходе от их исходного представления к сжатому.

Дополнительные сведения о качестве сжатия дает сравнение исходных данных и данных после их реконструкции по сжатому представлению в виде картинок на широкой печати ЭВМ CDC-6500, образцы которых даны на рис. 1 и 2.

7. Обработка реальных данных

7.1. Подбор параметров

Перед началом обработки данных следует тщательно подобрать параметры настройки, используя при этом известные характеристики телевизионной системы съема и учитывая характер обрабатываемых изображений.

В данном случае оба варианта программного имитатора применены для обработки данных с камеры РИСК. Перечислим нужные нам характеристики телевизионной системы /3/:

отсчетная единица по оси x - 0,122 мм (1000/8192);
 - " - y - 2,66 мм (1300/512);
 шумы точек $\sigma_s = 0,5$ мм, т.е. 4;
 разреш. способность $r = 5$ мм, т.е. 41

(все величины без определения размерности даны в отсчетных единицах по оси x).

Выбрав $\varphi_{\text{max}} = 45^\circ$, с учетом приведенных характеристик ТВ системы получаем величины $\varepsilon_s = 2 \cdot \sqrt{2} \cdot \sigma_s = 11$, $\varepsilon_\Delta = \Delta \times (\varphi_{\text{max}}) = 22$, необходимые для вычисления следующих критериев:

- по формулам (5.1), (2.4)

$$\varepsilon \geq \varepsilon_\Delta + \varepsilon_s, \quad \varepsilon_\Delta + \varepsilon_s = 33;$$

$$\varepsilon < r, \quad r = 41;$$

- для метода А по формуле (5.2)

$$\varepsilon_k = (2 + k) \varepsilon_1, \quad \varepsilon_1 \geq \varepsilon_s;$$

- для метода Б по формуле (5.3)

$$\varepsilon(k) = \varepsilon_0 + k \cdot \varepsilon_2, \quad \varepsilon_\Delta + \varepsilon_s \leq \varepsilon_0 < r, \quad \varepsilon_2 \geq \varepsilon_\Delta.$$

Поскольку в данном случае средняя длина пробелов \bar{L} и дисперсия σ_L^2 неизвестны, при выборе L_{\max} пришлось руководствоваться эвристическими критериями (5.4), т.е. $L_{\max} \leq \min(L_T, \frac{2}{3}n)$. Для данной ТВ-системы $M_T=15, q=13$; отсюда получаем $L_T=21$ для метода А (см. 5.2), $L_T=11$ для метода Б (см. 5.3) и для длины слайса $n \geq 32$ ограничения

$$L_{\max} < 21 \quad \text{для метода А,}$$

$$L_{\max} < 11 \quad \text{для метода Б.}$$

Для величины δ получаем из (5.5) ограничение снизу $\delta > 11$. Величину T согласно (5.6) следует выбирать равной 2 или 3.

7.2. Определение оптимальных значений параметров

С помощью усовершенствованного программного обеспечения были получены экспериментальные зависимости характеристик ρ_1, ρ_2, λ и σ от параметров ε (ε_1 для варианта А, ε_0 и ε_2 для варианта Б),

L_{\max}, δ и n для типичного изображения, показанного на рис.1. Из этих зависимостей были найдены оптимальные для данного типа изображений значения параметров (для последовательного алгоритма использованы результаты исследования /9/).

С увеличением длины слайса работа алгоритма улучшается, поэтому выбираем $n=64$ - наибольшее значение, которое допускает ограниченный объем АСД (при $n=32$ все основные характеристики изменяются незначительно; дальнейшее уменьшение n нежелательно).

Выбранное значение $\delta=20$ соответствует точности восстановления исходного изображения $\pm 2,5$ мм (в пространстве камеры) и обеспечивает $\sigma < 10$. При выборе $t=3$ понижается число ложных ЛТЭ/9/. Для параллельного алгоритма (метод А) отношение (5.2) заменено простым табличным выражением $\varepsilon_k = 2\varepsilon_1$ для $k=1,2,\dots,L_M$, $\varepsilon_k = 3\varepsilon_1$ для $k=L_M+1, L_M+2,\dots,L_{\max}$, где $\varepsilon = \varepsilon_1=32, L_{\max}=15, L_M=7$.

Для последовательного алгоритма (метод Б) выбраны значения $\varepsilon = \varepsilon_0=40, \varepsilon_2=20, L_{\max}=8$.

7.3. Результаты обработки

После подбора оптимальных значений параметров $\varepsilon, L_{\max}, \delta$ и t оба варианта имитатора АСД были применены для обработки реальных событий, записанных в 1981-82 г.г. с помощью телевизионной системы контроля стримерной камеры РИСК /10/ на магнитную ленту ЭВМ

Параллельный алгоритм - метод А



Последовательный алгоритм - метод Б

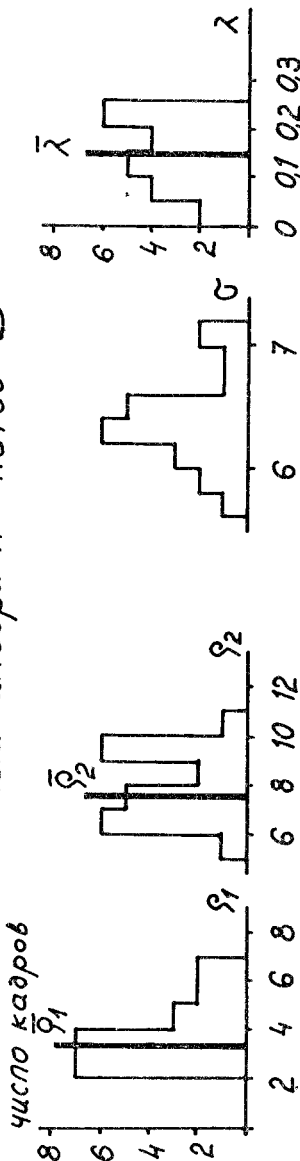


Рис. 5

ЕС-1040. Два десятка кадров после исключения точек реперных крестов были обработаны по обоим вариантам алгоритмов.

На рис. 5 показаны распределения значений ρ_1 , ρ_2 , λ и σ по кадрам для вариантов А и Б, а также общие значения $\bar{\rho}_1$, $\bar{\rho}_2$, $\bar{\lambda}$ для всего обработанного материала (полученные путем подстановки в соответствующие формулы сумм по всем кадрам $\sum N_{SUM}$, $\sum N_{SC}$, $\sum N_L$ и $\sum N_{OID}$). Подтвердилось, что оба варианта алгоритмов сравнимы - полученные по ним значения характеристик (для большинства кадров) лежат в пределах $\rho_1 = 2 \div 5$, $\rho_2 = 6 \div 11$, $\lambda = 0,05 \div 0,3$ при $\sigma = 6 \div 7$. Общие значения для вариантов А и Б соответственно равны $\rho_1 = 2,8$ и $3,3$, $\rho_2 = 8,7$ и $7,6$, $\lambda = 0,21$ и $0,15$. Отметим, что коэффициент ρ_2 , учитывающий только количество найденных ЛТЭ, несколько выше для варианта А, что отчасти объясняется большей надежностью процедуры связывания точек с учетом направления и отчасти увеличением числа неидентифицированных точек. Последнее связано также с уменьшением значения ρ_1 для варианта А по сравнению с вариантом Б.

7.4. Программное обеспечение

Программное обеспечение было написано на языке Фортран-IV для ЭВМ CDC-6500 и включает сам имитатор АСД, состоящий из программ сортировки (LINSOR и ADSOR для варианта А и LINK для варианта Б) и общей для обоих вариантов программы APROX, осуществляющей сжатие. Подробное описание алгоритмов, структуры и параметров этих программ дано в работе /8/. Для проведения массовой обработки реальных данных с магнитной ленты ЭВМ ЕС-1040 установки РИСК система программ была дополнена подпрограммами ввода и распаковки данных, а также средствами регулировки объема выходной печати с помощью ключей при разных режимах обработки.

Заключение

Предлагаемые два варианта программного имитатора спецпроцессора АСД, осуществляющие быструю сортировку и сжатие данных, обеспечили на реальном материале, снятом с помощью телевизионной системы /10/ стримерной камеры РИСК, уменьшение объема памяти примерно в 8 раз, а при сохранении массива неидентифицированных точек - примерно в 3 раза. В обоих случаях сохраняется точность восстановления $\delta = 2,5$ мм (по оси x), а также информация о локальной структуре пробелов в треках.

Оба варианта алгоритмов вполне годятся для аппаратурной реализации в виде спецпроцессора. Однако тщательный учет сложности, аппаратурных затрат и быстродействия /8/ показал преимущество последовательного алгоритма, лучше учитывающего специфику обрабатываемого материала. В то же время можно отметить, что вариант А допускает обработку данных более общего вида, не обязательно упорядоченных по величине, и, кроме того, благодаря использованию процедуры направленного поиска лучше проявил себя при связывании точек треков при их пересечении и при наличии длинных пробелов.

Таким образом, было признано целесообразным использовать для аппаратурной реализации последовательный алгоритм (метод Б). Вопрос о включении в этот алгоритм процедуры направленного поиска должен быть решен после предварительного исследования, выясняющего, окупаются ли временные и аппаратурные затраты на направленный поиск выигрышем эффективности сжатия и устойчивости процесса сортировки в сложных случаях взаимного расположения точек треков.

Литература

1. Garvey J. et al. A Television Camera System for Data Acquisition on the OMEGA Spectrometer, CERN - NP/OM - SN. 140, 1973.
2. Ососков Г.А. и др. ОИЯИ, Р10-11213, Дубна, 1978.
3. Алексеева Н.П. и др. ОИЯИ, Б1, I3-80-I33, Дубна, 1980.
4. Байла И. и др. ОИЯИ, Р10-80-162, Дубна, 1980.
5. Байла И., Ососков Г.А. ОИЯИ, Р10-80-217, Дубна, 1980.
6. Двуреченский А. и др. ОИЯИ, 5-81-362, Дубна, 1981.
7. Свириденко В.А. Анализ систем со сжатием данных. "Связь", М., 1977.
8. Байла И. и др. ОИЯИ, Б1, I0-82-119, Дубна, 1982.
9. Ососков Г.А. и др. ОИЯИ, Р10-82-273, Дубна, 1982.
10. Бечер Ю. и др. Материалы Второго Всесоюзного семинара по обработке физической информации. Изд-во ЕРФИ, Ереван, 1977, с. 428.

Рукопись поступила в издательский отдел
6 сентября 1982 года.

Байла И. и др.

P10-82-653

К вопросу о сжатии данных при бесфильмовом съеме информации со стримерных камер. III. Программный имитатор АСД

Излагаются методы быстрого /он-лайн/ сжатия информации, подходящие для аппаратной реализации в системах бесфильмового съема данных со стримерных камер. Предложены два варианта алгоритмов специальной сортировки /для объединения элементов треков/ и алгоритм линейной аппроксимации участков треков, которые обеспечивают заданную точность восстановления и сохраняют информацию о структуре пробелов в треках, необходимую для последующего определения ионизации. Алгоритмы реализованы в виде программного имитатора спецпроцессора сжатия АСД. На реальном материале со стримерной камеры установки РИСК получено уменьшение объема данных в 8 раз, а при сохранении координат неидентифицированных точек - в 3 раза при точности восстановления 2,5 мм.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна 1982

Bajla I. et al.

P10-82-653

On Data Compression at a Filmless Detection of the
of the Streamer Chamber Information.
III. Software Imitator of the ASD Special Processor

A method of on-line data reduction intended to hardware implementation in the filmless data readout system is considered. The method consists of two parts: /i/ special sorting, /ii/ linear approximation. It provides a required precision and preserves the information about the structure of track gaps needed for the determination of ionization. Two versions of the first part algorithms for track element linking and the second part algorithms of the fast piecewise linear approximation of track segments are developed as a software imitator of the ASD special hardware processor. The real data processing from the RISK spectrometer yields 8:1 compression ratio for the single compression and 3:1 compression ratio when nonidentified print coordinates are stored moreover. The 2.5 mm renewed precision are preserved in both cases.

The investigation has been performed at the Laboratory of Computing Techniques and Automation, JINR.

Communication of the Joint Institute for Nuclear Research. Dubna 1982

Перевод О.С.Виноградовой.