

Применение однородных структур для обнаружения космических частиц на астрономических изображениях

О.М. Смирнов

1. Постановка задачи

Современные астрономические наблюдения используют ПЗС-камеры (матрицы) как основное средство получения изображений. Их широкое распространение связано с развитием компьютерных методов обработки данных. Снимок, полученный компьютером напрямую с матрицы, содержит значительно больше информации и меньше шумов, чем аналогичный снимок, снятый на фотопластинку и проецированный микроденситометром.

Тем не менее, даже ПЗС-изображения могут содержать значительные помехи, которые можно условно разбить на три группы:

- электронный шум неба и шумы считывания;
- эффекты рассеивания, связанные с «неидеальностью» оптики;
- треки космических частиц (КЧ).

Как правило, удаление помех вовсе не является конечной или даже промежуточной целью обработки астрономических изображений. Конечная цель — получение каких-то физических величин (относительная яркость звезд, параметры спектральных линий, и т.п.) и их погрешностей. Поэтому помехи обычно не удаляются, а учитываются в процессе обработки. Помехи двух первых групп однородны в пределах изображения, и с ними более (а иногда и менее) легко справляются методы математической статистики. С треками космических частиц, распределенными по изображению случайно, ситуация иная.

Для их нахождения существует множество алгоритмов, но все они далеки от совершенства. Цель этой работы — разработка эффективного алгоритма обнаружения треков КЧ, основанного на понятии однородной структуры [1].

1.1. Формирование изображений

Прямоугольная ПЗС-матрица регистрирует изображение, то есть непрерывную функцию яркости $g(x, y)$, где $x_0 \leq x \leq x_1$, $y_0 \leq y \leq y_1$. Матрица разбита на $M \times N$ чувствительных элементов, или пикселей, Δ_{ij} (как правило, M и N находятся в пределах 300–1000), каждый в отсутствие КЧ фиксирует интеграл от g по своей поверхности, называемый *значением* пикселя:

$$\bar{g}_{ij} \stackrel{\text{def}}{=} \iint_{\Delta_{ij}} g(x, y) dx dy + \xi_{ij} \quad (i = 1 \dots M, j = 1 \dots N) \quad (1)$$

(здесь i, j отвечают целочисленным координатам пикселя, а ξ_{ij} есть некоторый шум, соответствующий пикселю Δ_{ij}). Следует отметить, что реально каждый элемент считает отдельные фотоны, поэтому \bar{g}_{ij} суть целые неотрицательные числа, в совокупности ограниченные разрядностью датчика.

1.2. Космические частицы

Если в пиксель Δ_{ij} попадает КЧ, то его значение увеличивает-ся на какое-то число δ_{ij} (*добавок* частицы), зависящее от энергии и траектории частицы. Трек КЧ обычно охватывает несколько пикселей и может иметь самые разнообразные формы. Поскольку образы объектов изучения (звезд, спектральных линий) имеют малые — в пределах нескольких пикселей — размеры, наличие необнаруженных КЧ в них может существенно повлиять на результаты обработки. Поэтому поиску задетых пикселей уделяется большое внимание. Обнаруженные пиксели считаются испорченными, и их значения либо не учитываются в дальнейшей обработке, либо заменяются на результат интерполяции незадетых соседей.

Задача состоит в том, чтобы имея набор чисел $\{g_{ij} = \bar{g}_{ij} + \delta_{ij}\}$ (для незадетых пикселей $\delta_{ij} = 0$), обнаружить и пометить задетые пиксели. Она обладает некоторыми характерными особенностями.

- Треки частиц настолько разнообразны, что не существует общего аналитического критерия для их обнаружения.
- Плотность КЧ может быть очень высока (до 1–2% пикселей), особенно в изображениях, снятых с большой выдержкой и в изображениях с Хаббловского космического телескопа.
- Несмотря на отсутствие четкого критерия, человек-эксперт может легко опознать визуально большинство треков. Но объемы современных наблюдений способны полностью загрузить этой работой всех астрономов мира.
- КЧ низкой энергии трудно обнаружимы, но мало влияют на результат.
- КЧ любой энергии, попавшие в не очень гладкий образ объекта, порой с трудом обнаруживаются даже экспертом. Отдельные частицы могут полностью слиться с деталями картинки. Это ставит некий потолок на возможности алгоритма. К сожалению, именно такие частицы скорее влияют на результат.
- Отдельные элементы изображения могут по своей структуре быть очень похожи на трек. Важно, чтобы алгоритм как можно реже реагировал на них.

1.3. Требования к алгоритмам

Эффективность алгоритма поиска КЧ оценивается двумя показателями. Частота обнаружения (detection rate) R_D определяется как среднее отношение числа правильно помеченных КЧ к общему числу КЧ (найденных экспертом). Частота ложных тревог (false alarm rate) R_F есть отношение числа ошибочно помеченных пикселей к общему числу помеченных пикселей. Для идеального алгоритма, очевидно, $R_D = 1$ и $R_F = 0$; у большинства современных алгоритмов эти показатели лежат в пределах, соответственно, 0.5–0.7 и 0.1–0.3 [2].

Другим важным критерием оценки является количество потребляемых вычислительных ресурсов. Поиск КЧ относится к предварительной обработке, и должен вестись по возможности быстро.

Наконец, последним требованием к алгоритму является наличие возможности настройки и/или обучения. Наблюдения, как правило, обрабатываются сериями, в пределах которых мало отличаются условия съемок, а следовательно и особенности КЧ. Настраиваемый алгоритм должен иметь несколько параметров, подбирая значения которых, эксперт может добиться лучших показателей R_D и R_F для изображений данной серии. Обучающийся алгоритм может сам подбирать оптимальные значения параметров, приняв за основу одно–два изображения, где КЧ уже отмечены экспертом.

2. Основные результаты

Для нахождения КЧ был реализован алгоритм, в основе которого лежит однородная структура со входами и выходами (ОСВВ). На пробных данных этот алгоритм показал $R_D \approx 0.8$ и $R_F \leq 0.1$ при сравнительно малом времени работы.

2.1. Предварительные преобразования

На целочисленной решетке введем метрику $\|ij\| \stackrel{\text{def}}{=} |i| + |j|$. Для целочисленного дискретного изображения $\{g_{ij}\}$ можно построить некоторый аналог дифференцирования:

$$g'_{i_0j_0} \stackrel{\text{def}}{=} \max_{\|ij\|=1} (g_{i_0j_0} - g_{i_0+ij_0+j}) \quad (2)$$

(для граничных пикселей максимум берется по меньшему множеству). Полученное изображение можно еще раз дифференцировать: $g''_{ij} = (g'_{ij})'$. Теперь на основе этих изображений строим *изображение критерия*. Для этого вводим уже действительную величину G_{ij} :

$$\bar{G}_{ij} \stackrel{\text{def}}{=} G(g'_{ij}, g''_{ij}), \quad (3)$$

где G есть одна из следующих функций:

$$\begin{aligned}
G_\lambda^s(u, v) &= u + \lambda v && \text{(аддитивная)} \\
G^m(u, v) &= uv && \text{(мультипликативная)} \\
G_\lambda^q(u, v) &= u^2 + \lambda v^2 && \text{(квадратичная)} \\
G_\lambda^r(u, v) &= \sqrt{u^2 + \lambda v^2} && \text{(радиальная)}
\end{aligned} \tag{4}$$

λ является настраиваемым параметром алгоритма, как и собственно выбор функции. На практике лучше всего работает радиальный критерий с $0.5 \leq \lambda \leq 2$. Целочисленное изображение критерия G_{ij} получается из \bar{G}_{ij} путем масштабирования:

$$G_{i_0 j_0} \stackrel{\text{def}}{=} \left\lceil \frac{\bar{G}_{i_0 j_0} L}{\max_{i,j} |\bar{G}_{ij}|} \right\rceil, \tag{5}$$

где константа $L = 2^K - 1$ есть модуль наибольшего целого числа в машинном представлении с двоичной разрядностью K , а $[x]$ обозначает взятие целой части от x .

2.2. Однородная структура

Для обнаружения КЧ изображение критерия погружается в ОСВВ. Введем для этого два семейства ОСВВ. При этом в целях удобства записи и реализации мы несколько отступим от канонического определения ОСВВ. Пусть ОСВВ $\sigma_g(\omega, \theta) = (\mathbb{Z}^2, W^g(\omega, \theta), V)$ и $\sigma_l(\omega, \theta) = (\mathbb{Z}^2, W^l(\omega, \theta), V)$. Здесь ω есть вектор неотрицательных действительных параметров $(\omega_0, \omega_x, \omega_y)$, а θ — положительное пороговое значение. Индексы g и l отличают структуры глобального и локального поиска. Шаблон соседства V крестообразен, то есть $V = \{(\pm 1, 0), (0, \pm 1)\}$. Пусть $E_0 = \{0, 1\}$, а $E = \{-L, -L+1, \dots, L-1\}$ — множество машинно представимых K -разрядных чисел; $E = E_0^K$. Тогда автоматы W^g и W^l , порождающие наши структуры, имеют следующий вид.

- Множество состояний $S = E_0$.
- Один K -разрядный основной вход (значения в E); один основной выход со значениями в E_0 .
- По четыре боковых входа и выхода со значениями в E .

- Уравнения автомата имеют вид:

$$\begin{cases} s(t+1) = s(t) = \dots = s(0), \\ y(t) = \varphi_{1,\theta}(x(t), b(t-1), s(t-1)), \\ a(t) = \varphi_{2,\omega}(x(t), b(t-1), s(t-1)) \end{cases} \quad (6)$$

$$\begin{aligned} \varphi_{1,\theta} &: E \times E^4 \times E_0 \rightarrow E_0, \\ \varphi_{2,\omega} &: E \times E^4 \times E_0 \rightarrow E_4 \end{aligned}$$

Здесь x и $b = (b_\alpha)$ отвечают основным и боковым входам, y и $a = (a_\alpha)$, $\alpha \in V$ основным и боковым выходам, а s — состоянию автомата. Через $X_t(\beta)$, $B_t(\beta)$, $Y_t(\beta)$, $A_t(\beta)$, $\Sigma_t(\beta)$ обозначаются значения этих функций для ячейки β в момент времени t .

Погружение изображения критерия в ОСВВ осуществим следующим образом. Определим прямоугольник $\Delta \subset \mathbb{Z}^2$, отвечающий изображению, как прямое произведение целочисленных отрезков $[1, M] \times [1, N]$, а его внутренность как $\text{int } \Delta = [2, M-1] \times [2, N-1]$. Зададим начальные состояния ячеек и входы структуры в момент t :

$$\Sigma_0(i, j) = \begin{cases} 1, & (i, j) \in \text{int } \Delta, \\ 0, & (i, j) \notin \text{int } \Delta \end{cases} \quad X_t(i, j) = \begin{cases} G_{ij}, & (i, j) \in \Delta, \\ 0, & (i, j) \notin \Delta \end{cases} \quad (7)$$

(входы и состояния постоянны во времени). Для определенности уравнений (6) будем считать $(B_0(i, j))_\alpha \equiv G_{ij}$. В каждый момент времени t наша ОСВВ осуществляет отображение $\{G_{ij}\} \mapsto \{Y_t(i, j)\}$ изображения критерия на двоичное изображение *флагов*. Будем считать пиксель помеченным (задетым КЧ), если его флаг равен 1.

Заметим, что так определенные ОСВВ не отвечают каноническому определению, так как их автоматы имеет недвоичные входы и выходы, а их функция взаимодействия $\varphi_{2,\omega}$ зависит не только от состояния. Это будет рассмотрено подробнее в третьей части.

2.2.1. Автоматные функции

Зададим для каждой клетки *среднее значение* и *средний прирост* градиента:

$$v(t) \stackrel{\text{def}}{=} \frac{1}{4} \sum_{\alpha \in V} b_{\alpha}(t-1) \quad (8)$$

$$d(t) \stackrel{\text{def}}{=} x(t) - v(t) \quad (9)$$

а также $V_t(\beta)$ и $D_t(\beta)$ — их значения для ячейки β . Теперь рассмотрим *глобальное* или *локальное* среднеквадратичное отклонение $\chi(t)$, а также *нормализованную невязку* $\eta(t)$:

$$\chi(t) \stackrel{\text{def}}{=} \begin{cases} \sqrt{\frac{1}{(M-1)(N-1)} \sum_{\beta \in \text{int } \Delta} D_t(\beta)^2}, & \text{для автомата } W^g \\ \sqrt{\frac{1}{4} \sum_{\alpha \in V} [b_{\alpha}(t-1) - v(t)]^2}, & \text{для автомата } W^l \end{cases} \quad (10)$$

$$\eta(t) \stackrel{\text{def}}{=} \frac{d(t)}{\chi(t)} \quad (11)$$

(в случае W^g для определенности полагаем $\eta(1) = 0$). Функция выхода тогда определяется так:

$$y(t) = \varphi_{1,\theta}(\dots) \stackrel{\text{def}}{=} \begin{cases} 1, & \eta(t)s(t) \geq \theta \\ 0, & \eta(t)s(t) < \theta \end{cases} \quad (12)$$

(множитель $s(t)$ обеспечивает нулевую функцию выхода для всех ячеек вне $\text{int } \Delta$). Теперь определим *локальное значение* ячейки:

$$l(t) \stackrel{\text{def}}{=} \begin{cases} x(t), & y(t) = 0 \\ \min_{\alpha \in V} b_{\alpha}(t), & y(t) = 1 \end{cases} \quad (13)$$

Так же, как и раньше, вводится и функция $L_t(\beta)$. Наконец, определим *веса* $\{w_{\alpha}, \tilde{w}_{\alpha}\}$:

$$\omega_{\alpha} = \begin{cases} \omega_x / (\omega_x + \omega_0), & \alpha = (\pm 1, 0), \\ \omega_y / (\omega_y + \omega_0), & \alpha = (0, \pm 1); \end{cases} \quad \tilde{\omega}_{\alpha} = \begin{cases} \omega_0 / (\omega_x + \omega_0), & \alpha = (\pm 1, 0), \\ \omega_0 / (\omega_y + \omega_0), & \alpha = (0, \pm 1). \end{cases} \quad (14)$$

Очевидно, что $\omega_\alpha + \tilde{\omega}_\alpha \equiv 1$. Теперь можно выписать функцию взаимодействия:

$$a_\alpha(t) = (\varphi_{2,\omega}(\dots))_\alpha \stackrel{\text{def}}{=} \begin{cases} x(t), & s(t-1) = 0 \\ [\tilde{\omega}_\alpha l(t) + \omega_\alpha b_{-\alpha}(t-1)], & s(t-1) = 1 \end{cases} \quad (15)$$

(квадратные скобки означают взятие целой части). Легко проверить, что так определенная $a_\alpha(t)$ будет действительно лежать в E , а также то, что так определенные автоматные функции будут действительно иметь вид (6).

На этом формальное описание данных ОСВВ закончено. Некоторые эмпирические соображения по их структуре будут высказаны в третьей части.

2.3. Программная реализация

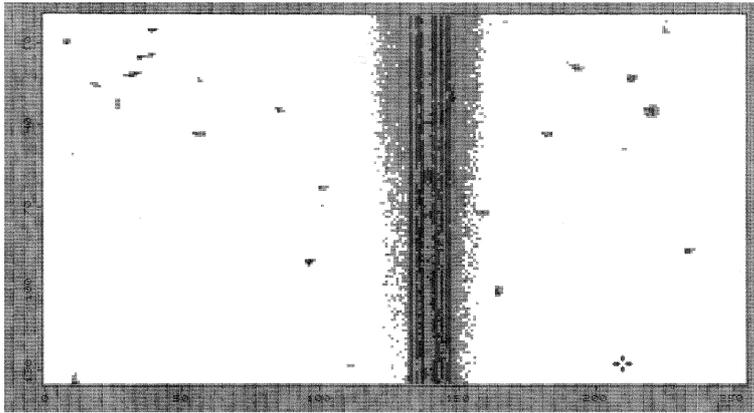
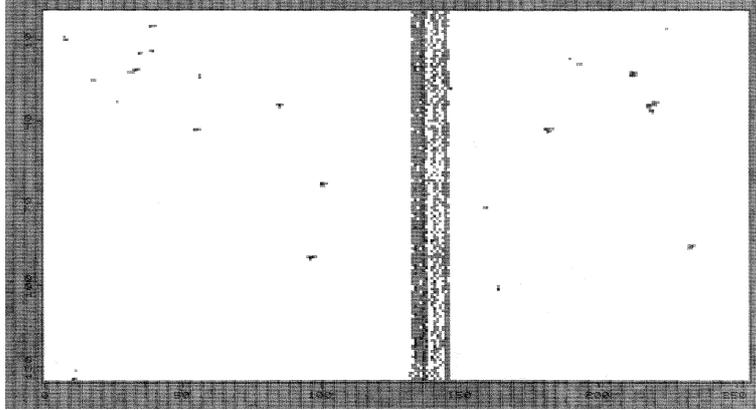
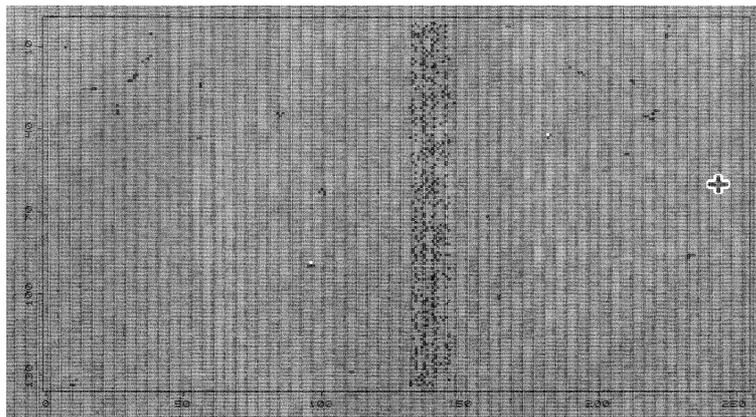


Рис. 1. Исходное изображение (фрагмент эшельного спектра).

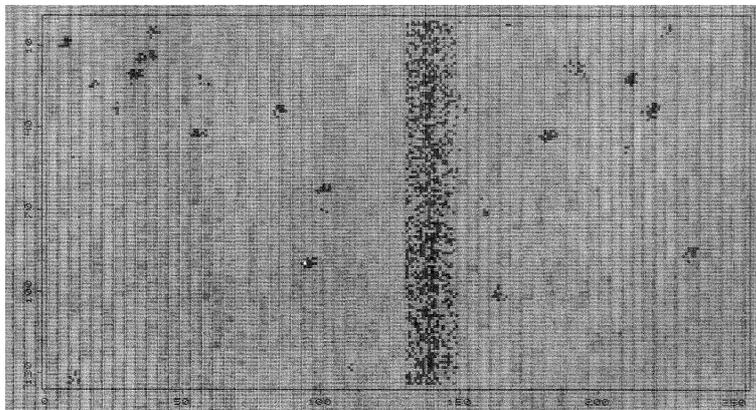
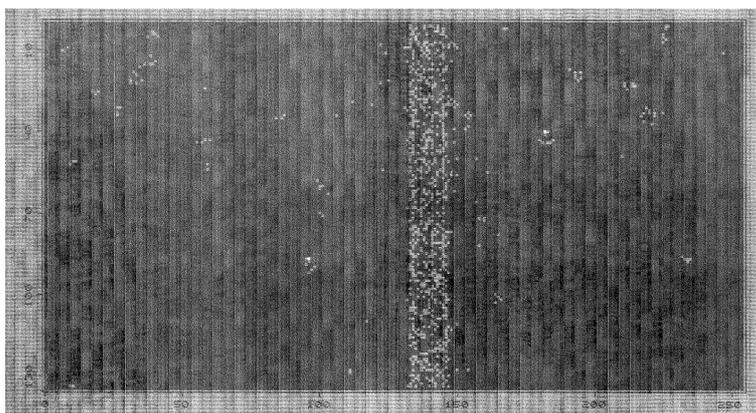
Программная реализация ОСВВ σ_g и σ_l существенно облегчается одним очевидным, но важным фактом, который будет доказан позже:

Теорема 1. (ограниченность) Для σ_g и σ_l с начальными состояниями, заданными (7), выходы произвольной ячейки $\beta_0 \in \text{int } \Delta$ в произвольный момент времени t_0 не зависят от боковых выходов любой

Рис. 2. Изображение критерия G_{ij} (радиальный; $\lambda = \frac{1}{2}$).Рис. 3. $\eta(1)$, $\omega_0 = 1$, $\omega_x = 0,2$, $\omega_y = 2$.

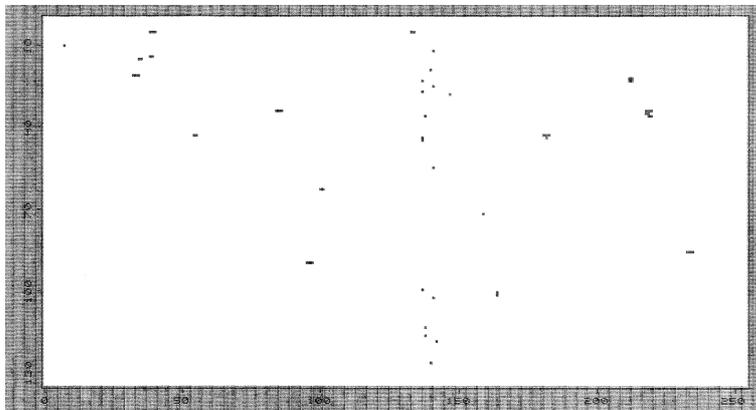
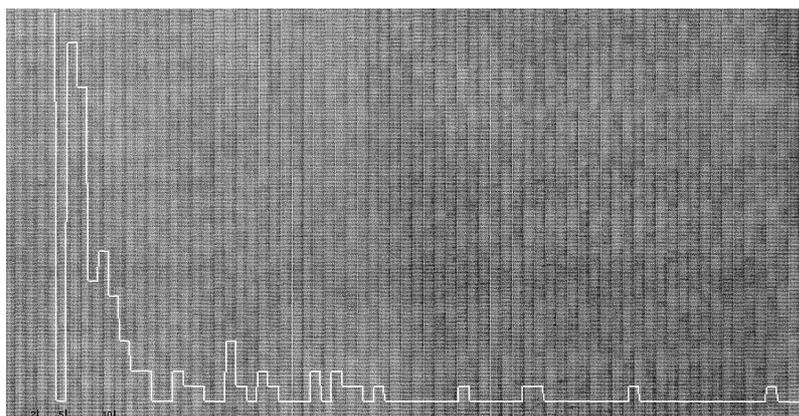
ячейки $\beta \notin \Delta$ в любой момент времени $t \leq t_0$ (иначе говоря, не изменятся при произвольном перераспределении последних).

Так как очевидно, что от других параметров ячеек $\beta \notin \Delta$ выходы β_0 не могут зависеть по определению, то эта теорема позволяет вести расчеты только в пределах прямоугольника Δ . В ее отсутствие реализация была бы просто невозможна.

Рис. 4. $\eta(2)$.Рис. 5. $\eta(3)$.

Что касается времени работы, то практика показывает, что в случае реальных данных уже при $t \geq 5$ заметных изменений в Y_t не наступает. К сожалению, нельзя утверждать что такая ОСВВ сходится, в определенных ситуациях возможны и колебательные процессы, хотя в случае реальных данных и приемлемых значений параметров это почти исключено.

ОСВВ σ_g и σ_l были реализованы единой программой HSCR на языке Си++ в среде системы обработки изображений PCIPS [3]. По-

Рис. 6. $y(5)$, $\theta = 6$.Рис. 7. Гистограмма распределения для $\eta(5)$. По вертикали — число попаданий, по горизонтали — диапазоны значений.

сле запроса параметров и типа ОСВВ, она каждый такт отображала на экран вид функции $\eta(t)$ и $y(t)$ по всему изображению. Параллельно строилась гистограмма распределения значений $\eta(t)$ (рис 1–7). Испытания с разными типами изображений (снимки звездного поля, эшельные спектры) показали высокие значения R_D и R_F при правильном подборе параметров λ , ω , θ .

2.3.1. Возможности самообучения

Из определения видно, что поведение ОСВВ достаточно простым образом зависит от ее параметров; более того, если считать все числа действительными, то легко оценить производные основных функций по параметрам. Это позволяет применить к ним градиентные методы минимизации, и таким образом, имея достаточное количество «примеров» (а именно 1–2 изображения, где КЧ уже отмечены экспертом), автоматически подобрать оптимальные значения параметров. Это уже выходит за пределы данной работы (но изыскания в этом направлении тем не менее ведутся).

3. Обоснования

3.1. Эмпирические соображения

Трек КЧ представляет собой резкий всплеск яркости, поэтому можно рассчитывать обнаружить его, изучая первую производную. Опыт показывает, что ее, как правило, недостаточно для безошибочного обнаружения, поэтому вводится составной критерий G , содержащий в себе и вторую производную яркости.

Следует учесть, что G может сильно варьироваться в пределах изображения, поэтому не может существовать порогового значения, эффективно разделяющего задетые и нетронутые пиксели. В ОСВВ σ_g и σ_l фактически происходит распространение статистики о распределении G в некоторой крестообразной окрестности. Действительно, из (15) следует, что

$$(A_t(\beta))_\alpha \approx \tilde{\omega}(L_t(\beta) + \omega_\alpha L_t(\beta - \alpha) + \dots + \omega_\alpha^n L_t(\beta - n\alpha)) \quad (16)$$

(здесь n зависит от t и размеров изображения; точного равенства нет, так как в (15) берется целая часть). Вес ω_α определяет то, как быстро затухает информация о значении L_t в данном пикселе при распространении по направлению α . $L_t(\beta)$ будет по определению (13) либо значением G в пикселе β , либо, если пиксель помечен как заде-тый, минимальной входной статистикой. Это позволяет постепенно

исключить из сбора статистики влияние обнаруженных КЧ с их локально высоким значением G , то есть каждая ячейка получает на вход информацию о взвешенном среднем значении G в своей крестообразной окрестности с *исключенными КЧ*, причем значения в более удаленных точках окрестности имеют меньший вес.

Осталось понять, насколько значение $G = x(t)$ должно превосходить среднее значение $v(t)$ в окрестности, чтобы пиксель считался задетым. Данные ОСВВ нормализуют эту невязку по среднеквадратичному отклонению (СКО) $\chi(t)$, определенному соотношением (10): если она превышает его более чем в θ раз, то пиксель объявляется задетым. Это позволяет задавать порог обнаружения в терминах некой неформальной *степени невероятности* всплеска. ОСВВ σ_g и σ_l отличаются лишь методом подсчета $\chi(t)$, первая (глобального поиска) берет в качестве него СКО невязок по всему изображению, подсчитанное на предыдущем такте, а вторая (локального поиска) — СКО четырех входов от их среднего $v(t)$. Локальный подход хорошо работает в случае изображения с разнообразным сигналом (например, густозаселенное звездное поле) и сильным уровнем шума, в других случаях он дает неприемлемо большой процент ложных тревог. Глобальный подход дает адекватные результаты почти всегда, но уступает локальному в вышеперечисленных случаях.

Граничные ячейки изображения имеют нулевое состояние, и вследствие (15) всегда распространяют статистику только о значении G в них. Это позволяет исключить влияние ячеек, лежащих за пределами изображения, в которых значение G просто не определено.

3.2. Сведение к каноническим ОСВВ

Определение ОСВВ σ_g и σ_l не является каноническим по трем причинам.

- Значения входов и выходов не двоичны, а K -разрядны.
- Уравнения автоматов (6) не отвечают каноническому виду, а именно:

$$\begin{cases} y(t) = \varphi_1(x(t), b(t), s(t)), \\ a(t) = \varphi_2(s(t)), \\ s(t+1) = \varphi(x(t), b(t), s(t)) \end{cases} \quad (17)$$

- ОСВВ глобального поиска использует дополнительно величину $\chi(t)$, зависящую от всех ячеек $\beta \in \text{int } \Delta$.

Первая проблема устраняется простым переобозначением. Любая величина $x \in E = e_0^K$ раскладывается по двоичным разрядам $(x_0, \dots, x_{K-1}) \in E_0^K$ следующим стандартным образом:

$$x = \begin{cases} x_0 + 2x_1 + \dots + 2^{K-2}x_{K-2}, & x_{K-1} = 0 \\ -2^K + x_0 + 2x_1 + \dots + 2^{K-2}x_{K-2}, & x_{K-1} = 1 \end{cases} \quad (18)$$

Воспользовавшись этим, можно каждый K -разрядный вход и выход заменить на K двоичных каналов. Фактически, $x(t)$, например, можно рассматривать как сокращенную запись $(x_0, \dots, x_{K-1})(t)$.

Вторая проблема тоже легко устранима. Расширим множество состояний до $S' = E \times E^4 \times S$ с элементами $s' = (s'_1, s'_2, s'_3)$, и тогда, учитывая, что $x(t) \equiv \text{const}$, и соответствующим образом задав исходные состояния, уравнения (6) можно записать в каноническом виде:

$$\begin{cases} s'(t+1) = (x(t), b(t), s'_3(t)), \\ y(t) = \varphi_{1,\theta}(s'_1(t), s'_2(t), s'_3(t)), \\ a(t) = \varphi_{2,\omega}(s'_1(t), s'_2(t), s'_3(t)) \end{cases} \quad (19)$$

Последняя проблема не мешает реализации алгоритма (так как на однопроцессорной машине ячейки обрабатываются последовательно, и легко вычислить $\chi(t)$ в процессе обработки структуры). Формально устранить ее можно двумя способами:

- Добавить к каждой ячейке по K' -разрядному входу $\hat{\chi}(t)$ и $K+1$ -разрядному выходу $\hat{d}(t)$. На выход подавать значение $d(t)$, а на вход — представленное с конечной точностью значение $\chi(t)$, которое получается из значений выходов $\hat{d}(t-1)$ по формулам (10). Так фактически и сделано в алгоритме.
- Решить проблему, формально оставаясь в пределах ОСВВ (но существенно усложнив ее), можно путем добавления к шаблону

соседства параллелепипеда V' со сторонами $2M + 1$ и $2N + 1$ и введения еще большого числа соответствующих боковых входов $\hat{\chi}_\gamma(t)$ и выходов $\hat{d}_\gamma(t)$, где $\gamma \in V'$. На каждый выход подается значение $d(t)s(t)$, тогда из (7) и (10) следует, что в любой ячейке $\beta \in \text{int } \Delta$ единое глобальное значение $\chi(t)$ можно найти как функцию от входов:

$$\chi(t) = \sqrt{\frac{1}{(M-1)(N-1)} \left(d(t)^2 + \sum_{\gamma \in V'} \hat{\chi}_\gamma(t)^2 \right)} \quad (20)$$

Все вышесказанное позволяет утверждать, что наши ОСВВ выразимы в терминах канонически определенных ОСВВ.

3.3. Ограниченность

В заключение вспомним теорему об ограниченности. Она утверждает, что для σ_g , и σ_l выходы и состояние произвольной ячейки $\beta \in \text{int } \Delta$ в произвольный момент времени t_0 не зависят от боковых выходов любой ячейки $\beta \notin \Delta$ в любой момент времени $t \leq t_0$. Докажем это.

Обозначим через $\mathcal{Y}(\beta_0, t_0)$ множество всех таких ячеек β , что при каком-то $\alpha \in V$ и $t \leq t_0$ $Y_t(\beta_0)$ зависит от $(A_t(\beta))_\alpha$. Аналогичным образом введем множество $\mathcal{A}_\alpha(\beta_0, t_0)$. Надо доказать, что при $\beta_0 \in \text{int } \Delta$ эти множества целиком лежат в Δ . Из определений (7–15) легко выводятся следующие свойства:

$$\mathcal{Y}(\beta_0, t_0), \mathcal{A}_{\alpha_0}(\beta_0, t_0) \subseteq \bigcup_{\alpha \in V} \mathcal{A}_\alpha(\beta_0 + \alpha, t_0 - 1) \quad (21)$$

$$\beta \notin \text{int } \Delta \implies \forall t, \alpha \mathcal{A}_\alpha(\beta, t) = \{\emptyset\} \quad (22)$$

Теперь, рекурсивно расписывая (21), с учетом (22) получаем, что $\mathcal{Y}(\beta_0, t_0), \mathcal{A}_{\alpha_0}(\beta_0, t_0) \subseteq \Delta$. Теорема доказана.

Список литературы

- [1] Кудрявцев В.Б., Подколзин А.С., Болотов А.А. Основы теории однородных структур. М.: Наука, 1990.

- [2] Murtagh P., Adorf H.-M. // 3rd ESO/ST-ECF Data Analysis Workshop. Munich: ESO, 1991. P. 51.
- [3] Smirnov O.M., Piskunov N.E. PCIPS 2.0: Low-cost multiprofile image processing // Astronomical Data Analysis Software & Systems II. San Francisco: PASPC.