

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ

3/2011

НАУЧНО - ТЕХНИЧЕСКИЙ ЖУРНАЛ

ЦИФРОВАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ

тематический выпуск:

- КОДИРОВАНИЕ И СЖАТИЕ
- НОРМАЛИЗАЦИЯ И СЕГМЕНТАЦИЯ
- ОБНАРУЖЕНИЕ АНОМАЛИЙ
- КОМПЕНСАЦИЯ ДВИЖЕНИЯ
- ДИСТАНЦИОННОЕ ЗОНДИРОВАНИЕ
- БЫСТРЫЕ АЛГОРИТМЫ

ISSN 1684-2634



9 771684 263005 >

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ

Научно-технический журнал
№ 3/2011

Издаётся с 1999 года
Выходит четыре раза в год

ГЛАВНЫЙ РЕДАКТОР
Ю.Б. ЗУБАРЕВ

ЗАМЕСТИТЕЛИ ГЛАВНОГО РЕДАКТОРА:
В.В. ВИТЯЗЕВ, В.П. ДВОРКОВИЧ

РЕДАКЦИОННАЯ КОЛЛЕГИЯ:
Б.А. Алпатов, Б.А. Бабаян, Ю.И. Борисов,
С.А. Грибачев, Г.В. Зайцев, Р.В. Зубарев,
А.П. Кирпичников, М.И. Кривошеев,
Н.А. Кузнецов, В.Г. Мистюков, С.Л. Мишенков,
А.А.Петровский, Ю.Н. Прохоров,
Ю.Г. Сосулин, В.В. Шахильдян, Ю.С. Шинаков

Адрес редакции:
107031 Москва, Рождественка, 6/9/20, стр. 1
Тел.: +7 (495) 621-06-10, +7 (495) 621-16-39
Факс: +7 (495) 621-16-39
E-mail: nto.popov@mtu-net.ru
vityazev.v.v@rsreu.ru
http://www.dspsa.ru

Для писем:
129090, Москва, а/я 48.

Издатель:
Российское научно-техническое общество
радиотехники, электроники и связи им. А.С. Попова
Компьютерная вёрстка: Р.Ю. Шуриков
Дизайн: Н.С. Осипова

Подписной индекс по каталогу
ОАО «Роспечать» – **82185**

Подписано в печать 20.10.2011г.
Формат 60х90/8.
Гарнитура «Arial». Печать офсетная.
Бумага офсетная. Печ.л. 6,5. Тираж 700 экз.

Заказ № 5019. Отпечатано в
ООО НПЦ «Информационные технологии»
Рязань, ул. Островского, д. 21/1
тел.: (4912) 98-69-84

Издание зарегистрировано в Министерстве
Российской Федерации по делам печати,
телерадиовещания и средств
массовых коммуникаций.
Свидетельство о регистрации ПИ № 77-1488
от 14.01.2000

УЧРЕДИТЕЛИ:

**ЗАО «Инструментальные системы»
ФГУП «НИИ радио»**

В НОМЕРЕ:

Вашкевич М.И., Петровский А.А.

Применение полиномиальных алгебр и теории Галуа для
синтеза быстрых алгоритмов дискретных косинусных
преобразований.....2

Умняшкин С.В., Табориский А.Е.

Метод обработки Р-кадров для видеокодека на основе
дискретного вейвлет-преобразования.....11

Алпатов Б.А., Бабаян П.В., Шубин Н.Ю.

Алгоритм оценки координат объектов на основе преобразо-
вания Радона.....17

Корепанов С.Е., Стротов В.В.

Алгоритм слежения за движущимися объектами в бортовых
видеоинформационных системах.....21

Васильев К.К., Лучков Н.В.

Эффективность цифрового обнаружения протяженных аномалий
с неизвестными уровнями на фоне мешающих по-
мех..... 24

**Гектин Ю.М., Еремеев В.В., Егоскин Н.А., Зенин В.А.,
Москатынцев И.В.**

Нормализация изображений от геостационарной космической
системы наблюдения земли.....28

Ветров А.А., Кузнецов А.Е.

Сегментация облачных объектов на панхроматических
изображениях земной поверхности..... 32

Ветров А.А., Кочергин А.М., Кузнецов А.Е.

Принципы построения геопортала спутниковых изображений,
разрабатываемого в рамках проекта MEDEO37

Пашенцев Д.Ю., Акимов А.А., Скимунт В.К., Тишкин Р.В.

Высокоскоростной алгоритм сегментации изображений
звёздного неба, полученных от датчиков сканерного типа.....42

Кузнецов А.Е., Пресняков О.А.

Организация процесса формирования мозаичных изображений
земной поверхности.....47

Хрящев В.В., Бекренев В.А., Соловьев В.Е., Никитин А.Е.

Улучшение качества JPEG2000-изображений на основе
модифицированного билатерального фильтра.....53

Бабаян П.В., Суходольская Н.В.

Комбинированный алгоритм сегментации изображений
полуфабрикатов кожи..... 58

Вычужанин В.В.

Реализация цифровой обработки сигналов переменной
точности на ПЛИС.....62

Бибииков С.А., Никоноров А.В., Фурсов В.А., Якимов П.П.

Программный комплекс для обработки изображений в массив-
но-многопоточной CUDA-среде.....67

Подписной индекс по каталогу
ОАО «Роспечать» – **82185**

ПРИМЕНЕНИЕ ПОЛИНОМИАЛЬНЫХ АЛГЕБР И ТЕОРИИ ГАЛУА ДЛЯ СИНТЕЗА БЫСТРЫХ АЛГОРИТМОВ ДИСКРЕТНЫХ КОСИНУСНЫХ ПРЕОБРАЗОВАНИЙ

Вашкевич М.И., аспирант кафедры «Электронных вычислительных средств» Белорусского государственного университета информатики и радиоэлектроники, г. Минск, e-mail: vashkevich@bsuir.by.

Петровский А.А., зав. кафедрой «Электронных вычислительных средств» Белорусского государственного университета информатики и радиоэлектроники, г. Минск, д.т.н., профессор, e-mail: palex@bsuir.by.

Ключевые слова: дискретные преобразования, полиномиальная алгебра, теория Галуа, быстрые алгоритмы, модели сигнала.

Введение

Для описания и синтеза быстрых алгоритмов дискретных преобразований, таких как дискретное косинусное преобразование (ДКП) и дискретное преобразование Фурье (ДПФ) в настоящее время используются различные математические системы обозначений [1]:

- алгебраическая система обозначений, предложенная Кули и Тьюки, используется для получения коротких рекурсивных уравнений, которые оказываются удобными при составлении программ и исследовании ошибок округления;
- матричная система обозначений, в которой быстрый алгоритм преобразования представляется в виде произведения структурированных матриц;
- полиномиальные алгебры. Позволяют перейти от матрицы преобразования к рассмотрению полиномиальных алгебр и использовать для синтеза быстрых алгоритмов математический аппарат теории групп и колец [2-4];
- модель сигнала. Понятие модели сигнала обобщает использование полиномиальных алгебр для синтеза быстрых алгоритмов преобразований и позволяет с единых позиций рассматривать различные дискретные преобразования и их быстрые алгоритмы [5].

В данной работе используется подход на основе понятия модели сигнала [5]. Однако, в отличие от [5], в качестве основного выбирается поле рациональных, а не комплексных чисел. Это изменение приводит к тому, что для синтеза быстрого алгоритма приходится вводить в рассмотрение расширения поля рациональных чисел, что в свою очередь ведёт к новой структуре быстрых алгоритмов. Упомянутая особенность также позволяет найти применение в ЦОС изящного математического аппарата теории Галуа.

В качестве практического применения предлагаемого алгебраического подхода разработан быстрый алгоритм 8-точечного ДКП-2, содержащий в ядре своей структуры только 5 операций умножения и 29 операций сложения.

Статья имеет следующую структуру. В первом разделе, для удобства читателя, приводятся основные сведения из АТОС. Во втором разделе описываются модели сигнала соответствующие ДКП-2 и ДКП-4. Третий раздел содержит описание предлагаемого метода синтеза быстрых алгоритмов ДКП-2 и ДКП-4 с использованием теории Галуа, а также практические примеры. Ниже приведены некоторые пояснения по поводу используемых обозначений.

Предлагается систематический подход к синтезу быстрых алгоритмов дискретных косинусных преобразований второго и четвертого типов (ДКП-2/ДКП-4), основанный на алгебраической теории обработки сигналов (АТОС). В рамках АТОС, быстрый алгоритм преобразования получается не путем действий с коэффициентами матрицы преобразования, а как пошаговая декомпозиция полиномиальной алгебры, отвечающей данному преобразованию. Декомпозиция предполагает пошаговую факторизацию полинома, для чего предлагается использовать подполя поля разложения полинома, получаемые с использованием основной теоремы теории Галуа.

Представление алгоритма. Традиционно в ЦОС линейное преобразование записывается в виде

$$y_k = \sum_{0 \leq \ell < n} t_{k,\ell} s_\ell,$$

где $s = (s_0, \dots, s_{n-1})^T$ – входной сигнал, $y = (y_0, \dots, y_{n-1})^T$ – выходной сигнал, а $t_{k,\ell}$ – коэффициенты преобразования. Тем не менее, часто более удобной является векторно-матричная форма записи преобразования: $y = Ts$, где $T = [t_{k,\ell}]_{0 \leq k, \ell < n}$

В этом случае быстрый алгоритм для преобразования представляется в виде факторизации матрицы T в произведение слабозаполненных, структурированных матриц [6].

Обозначения. Ниже приведены используемые в статье типы матриц

$$I_n = \begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{bmatrix}, J_n = \begin{bmatrix} & & 1 \\ & \ddots & \\ 1 & & \end{bmatrix}.$$

Все матрицы, как правило, обозначаются заглавными латинскими буквами (например A_n – квадратная матрица порядка n), а вектора – строчными латинскими буквами (полуужирное начертание шрифта). Диагональные матрицы записываются как $\text{diag}(\alpha_0, \dots, \alpha_{n-1})$. Также используется оператор прямой суммы матриц

$$A \oplus B = \begin{bmatrix} A & \\ & B \end{bmatrix}.$$

Полиномы обозначаются строчными латинскими буквами, например $p(x), q(x)$. Часто для удобства и экономии пространства аргумент x опускается.

1. Алгебраическая теория обработки сигналов: обзор Полиномиальная алгебра

В [5] показано, что любое дискретное тригонометрическое преобразование (в том числе ДПФ и ДКП) связано с определённой полиномиальной алгеброй. Под алгеброй здесь понимается векторное пространство $\mathcal{A}$ над числовым полем $\mathbb{F}$, в котором установлена операция умножения, приводящая в соответствие каждой паре элементов p, q из $\mathcal{A}$ элемент $r \in \mathcal{A}$ [7]. Примером алгебр могут служить система комплексных чисел, кватернио-

нов или $\mathbb{F}[x]$ (множество всех полиномов с коэффициентами из поля $\mathbb{F}$).

Практическую значимость для ЦОС представляет полиномиальная алгебра, определяемая как

$\mathcal{A} = \frac{\mathbb{F}[x]}{p(x)} = \{q(x) \mid \deg q < \deg p\}$, множество всех полиномов со степенью меньше $\deg p$ и операциями сложения и умножения, выполняемыми по модулю $p(x)$.

Концепция модели сигнала

Развитие понятия полиномиальной алгебры привело к появлению алгебраической теории обработки сигналов (АТОС) (от англ. «algebraic signal processing theory»), в которой нашли применение многие понятия современной алгебры [8].

АТОС – это общий аксиоматический подход к ЦОС, который строится на концепции *модели сигнала*, определяемой тройкой $(\mathcal{A}, \mathcal{M}, \mathcal{Z})$, где $\mathcal{A}$ – это пространство фильтров (алгебра), $\mathcal{M}$ – пространство сигналов ($\mathcal{A}$ -модуль) и $\mathcal{Z}$ – обобщенная концепция z-преобразования. Модель сигнала $(\mathcal{A}, \mathcal{M}, \mathcal{Z})$ устанавливает связь векторно-матричных операций, в которых выражаются дискретные линейные преобразования, с их алгебраической структурой (можно сказать, что существуют две изоморфные области – область алгебраических структур и область их векторно-матричного представления).

Множество сигналов $\mathcal{M}$ представляет собой векторное пространство. Таким образом, сигналы можно складывать и умножать на константы из основного поля. В ЦОС сигналы обрабатываются линейными системами, которые, как правило, называют фильтрами. В АТОС фильтрацию представляют, как умножение

$$s' = h \cdot s, \quad (1)$$

где $s, s' \in \mathcal{M}$, а h принадлежит пространству фильтров $\mathcal{A}$. Пространство фильтров имеет более сложную структуру, чем пространство сигналов, в нём определены следующие операции: $h + h' \in \mathcal{A}$; $\alpha h \in \mathcal{A}$, где α константа из основного поля; $h \cdot h' \in \mathcal{A}$.

Первые две операции определяют $\mathcal{A}$, как векторное пространство, а третья делает $\mathcal{A}$ алгеброй.

Таким образом, пространство фильтров является алгеброй $\mathcal{A}$, которая действует в векторном пространстве $\mathcal{M}$, образуя в нем структуру левого $\mathcal{A}$ -модуля. По определению левым $\mathcal{A}$ -модулем называется векторное пространство $\mathcal{M}$ с операцией умножения слева на элементы алгебры $\mathcal{A}$, обладающей следующими свойствами:

$$\begin{aligned} h \cdot (s + s') &= h \cdot s + h \cdot s', \\ (h + h') \cdot s &= h \cdot s + h' \cdot s, \\ (h \cdot h') \cdot s &= h \cdot (h' \cdot s), \end{aligned} \quad (2)$$

для любых $h, h' \in \mathcal{A}$ и $s, s' \in \mathcal{M}$.

В [8] показано, что если модель сигнала строится для конечномерных сигналов $s = (s_0, \dots, s_{n-1}) \in \mathbb{F}^n$ и ей свойственна инвариантность к сдвигу, то $\mathcal{A}$ обязана быть полиномиальной алгеброй $\mathbb{F}[x]/p(x)$. Далее, если в $\mathbb{F}[x]/p(x)$ задан базис $b = (p_0, \dots, p_{n-1})$, то $\mathcal{A} = \mathcal{M} = \mathbb{F}[x]/p(x)$ с отображением

$$\mathcal{Z}: \mathbb{F}^n \rightarrow \mathcal{M}, s \mapsto s = s(x) = \sum_{0 \leq i < n} s_i p_i,$$

определяют модель сигнала, где $\mathcal{Z}$ это «z-преобразование» для данной модели. Можно сказать, что $\mathcal{Z}$ выполняет отображения конечномерного сигнала $s \in \mathbb{F}^n$ в пространство $\mathcal{M}$. Заметим, что $\mathcal{Z}$ зависит от выбора базиса b в $\mathcal{M}$. Выражение $\mathcal{A} = \mathcal{M}$ показывает, что базисное множество у алгебры $\mathcal{A}$ и векторного пространства $\mathcal{M}$ совпадают, в этом случае $\mathcal{M}$ называют *регулярным* $\mathcal{A}$ -модулем. Тем не менее, даже, если множества $\mathcal{A}$ и $\mathcal{M}$ равны, их алгебраическая структура различна. Так, например, для элементов из $\mathcal{M}$ не определена операция умножения.

Матричное представление алгебр

Свойства (2) показывают, что каждый фильтр $h \in \mathcal{A}$ определяет линейное преобразование $\mathcal{M} \rightarrow \mathcal{M}$ (см. выражение (1)). При любом выборе базиса $b = (p_0, \dots, p_{n-1})$ в $\mathcal{M}$ линейному преобразованию h будет отвечать матрица M_h размера $n \times n$, которая называется (матричным) представлением фильтра h . Применяя h к каждому базисному вектору p_j , можно найти матрицу

$$M_h = [m_{ij}]_{0 \leq i, j < n} \quad h \cdot p_j = \sum_{i=0}^{n-1} m_{ij} p_i, m_{ij}, m_{ij} \in \mathbb{C}^n. \quad (3)$$

Определяя, таким образом, M_h для каждого $h \in \mathcal{A}$, получаем отображение алгебры $\mathcal{A}$ в алгебру матриц $\mathbb{F}^{n \times n}$: $\phi: \mathcal{A} \rightarrow \mathbb{F}^{n \times n}, h \mapsto \phi(h) = M_h$.

Отображение ϕ является гомоморфизмом алгебр, т.е. таким отображением, которое сохраняет структуру алгебры: $\phi(h + h') = \phi(h) + \phi(h')$, и $\phi(hh') = \phi(h)\phi(h')$.

ϕ называют (матричным) представлением $\mathcal{A}$, отвечающим $\mathcal{A}$ -модулю $\mathcal{M}$ с базисом b .

Если каждому элементу s пространства сигналов $\mathcal{M}$ соответствует сигнал s из $\mathbb{F}^n$, то каждому элементу h пространства фильтров $\mathcal{A}$ соответствует матрица некоторого линейного оператора из $\mathbb{F}^{n \times n}$. В случае, когда $\mathcal{A} = \mathbb{F}[x]/p(x)$ для установления связи между $\mathcal{A}$ и $\mathbb{F}^{n \times n}$ достаточно определить линейный оператор, который будет играть роль умножения на x т.е.

$$\phi: \mathcal{A} \rightarrow \mathbb{F}^{n \times n}, x \mapsto \phi(x) = A.$$

Тогда любому (фильтру) полиному $h(x) \in \mathcal{A} = \mathbb{F}[x]/p(x)$ будет соответствовать операторный полином в $\mathbb{F}^{n \times n}$: $\phi: h(x) = h_{n-1}x^{n-1} + \dots + h_1x + h_0 \mapsto h(A) = h_{n-1}A^{n-1} + \dots + h_1A + h_0I$.

В [8] доказывается, что $x \in \mathcal{A}$ является оператором сдвига и порождающим элементом алгебры $\mathcal{A} = \mathbb{F}[x]/p(x)$, а $\phi(x)$ – матричным представлением оператора сдвига.

Инвариантность к сдвигу. В ЦОС важным является понятие инвариантности линейной системы к сдвигу. В АТОС это свойство принимает весьма простую форму. А именно, если x оператор сдвига, h любой фильтр, то инвариантность к сдвигу имеет место, если для любого $s \in \mathcal{M}$ выполняется тождество $h(xs) = (hx)s$, что эквивалентно:

$$x \cdot h = h \cdot x, \forall h \in \mathcal{A}. \quad (4)$$

В случае если x – порождающий элемент $\mathcal{A}$, то алгебра $\mathcal{A}$ является коммутативной и, следовательно, выполнение (4) гарантировано. Тем самым доказывается, что если модель сигнала $(\mathcal{A}, \mathcal{M}, \mathcal{Z})$ поддерживает свойство инвариантности к сдвигу, то соответствующая алгебра $\mathcal{A}$ является коммутативной.

Пример: модель сигнала дискретного времени. В качестве примера рассмотрим модель сигнала $\mathcal{A} = \mathcal{M} = \mathbb{F}[x]/(x^3 - 1)$ с базисом $b = (x^0, x^1, x^2)$ в $\mathcal{M}$, тогда для сигнала $s = (s_0, s_1, s_2) \in \mathbb{F}^3$ получим:

$$\mathcal{Z}: s \mapsto s = s(x) = s_0 + s_1x + s_2x^2 \in \mathbb{F}[x]/(x^3 - 1).$$

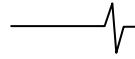
Операция умножения, которая соответствует фильтрации, в данной модели для $h(x) \in \mathcal{A}$ и $s(x) \in \mathcal{M}$ определяется как

$$h(x)s(x) \bmod (x^3 - 1), \quad (5)$$

что соответствует вычислению круговой свёртке коэффициентов h и s . Далее, используя (3), построим матричное представление простейшего фильтра $h(x) = x \in \mathcal{A}$:

$$\begin{aligned} x \cdot x^0 \bmod (x^3 - 1) &= x^1, \\ x \cdot x^1 \bmod (x^3 - 1) &= x^2, \\ x \cdot x^2 \bmod (x^3 - 1) &= x^0, \end{aligned}$$

откуда



$$\phi(x) = A = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}. \quad (6)$$

В (6) легко узнать матрицу оператора циклического сдвига. Применяя (6) к фильтру $h(x) = h_0 + h_1x + h_2x^2 \in \mathcal{A}$, получим его матричное представление:

$$\begin{aligned} h(x) &= h_0 + h_1x + h_2x^2 \xrightarrow{\phi} h(A) = h_2A^2 + h_1A + h_0I = \\ &= h_2 \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} + h_1 \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} + \\ &+ h_0 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} h_0 & h_2 & h_1 \\ h_1 & h_0 & h_2 \\ h_2 & h_1 & h_0 \end{bmatrix}. \end{aligned}$$

Таким образом, операции умножения (5), производимой в $\mathcal{A}$ -модуле $\mathcal{M}$, соответствует умножение вектора на матрицу, результатом которого является циклическая свёртка векторов h и s :

$$\begin{bmatrix} s'_0 \\ s'_1 \\ s'_2 \end{bmatrix} = \begin{bmatrix} h_0 & h_2 & h_1 \\ h_1 & h_0 & h_2 \\ h_2 & h_1 & h_0 \end{bmatrix} \begin{bmatrix} s_0 \\ s_1 \\ s_2 \end{bmatrix}.$$

Полиномиальное преобразование

Основные определения. Пусть задана полиномиальная алгебра $\mathcal{A} = \mathbb{C}[x]/p(x)$, где $\mathbb{C}$ – поле комплексных чисел. Предположим, что все корни $\alpha = (\alpha_0, \dots, \alpha_{n-1})$ полинома $p(x)$ попарно различны. Тогда любой полином из $\mathcal{A}$ можно разложить согласно Китайской теореме об остатках (КТО) следующим образом

$$\mathcal{F}: \mathbb{C}[x]/p(x) \rightarrow \bigoplus_{0 \leq k < n} \mathbb{C}[x]/(x - \alpha_k), \quad (7)$$

$s(x) \mapsto (s(\alpha_0), \dots, s(\alpha_{n-1}))$. Преобразование (7) линейно и имеет простую интерпретацию [9]: полином степени $n - 1$ полностью определяется либо своими коэффициентами, либо списком своих значений в n различных точках. Преобразование (выполняет переход от коэффициентов полинома к его значениям в точках $\alpha = (\alpha_0, \dots, \alpha_{n-1})$. Как векторное пространство $\mathcal{A} = \mathbb{C}[x]/p(x)$ под действием $\mathcal{F}$ раскладывается в прямую сумму одномерных подпространств $\mathbb{C}[x]/(x - \alpha_k)$. Следовательно, если зафиксировать базис $b = (p_0, \dots, p_{n-1})$ в $\mathcal{A}$ и выбрать базис (с единичной нормой) $\|x^0\| = 1$ в каждом подпространстве $\mathbb{C}[x]/(x - \alpha_k)$, то преобразование $\mathcal{F}$ приобретёт матричную форму:

$$\mathcal{F} = \mathcal{P}_{b,\alpha} = [p_\ell(\alpha_k)]_{0 \leq k, \ell < n} \quad (8)$$

$\mathcal{P}_{b,\alpha}$ называют полиномиальным преобразованием для $\mathcal{A} = \mathbb{C}[x]/p(x)$ с базисом b . Если в подпространствах $\mathbb{C}[x]/(x - \alpha_k)$ выбрать базисы с нормой отличной от единицы $\|x^0\| = \beta_k$, то получаемое преобразование $\mathcal{F} = \text{diag}(1/\beta_0, \dots, 1/\beta_{n-1}) \cdot \mathcal{P}_{b,\alpha}$ называют *масштабированным полиномиальным преобразованием*. Часто (8) также называют преобразованием Фурье для модуля $\mathcal{M}$.

Пример: полиномиальное преобразование для модели сигнала дискретного времени.

Рассмотрим модель сигнала $\mathcal{A} = \mathcal{M} = \mathbb{C}[x]/(x^n - 1)$ с базисом $b = (x^0, \dots, x^{n-1})$ в $\mathcal{M}$. Корнями полинома $(x^n - 1)$ являются $\alpha = (\omega_n^0, \dots, \omega_n^{n-1})$, где $\omega_n = e^{-j2\pi/n}$. Таким образом, преобразование Фурье для данной модели задается как $\mathcal{F}: \mathbb{C}[x]/(x^n - 1) \rightarrow \bigoplus_{0 \leq k < n} \mathbb{C}[x]/(x - \omega_n^k)$, что в точности совпадает с матрицей ДПФ и объясняет, почему модель названа моделью дискретного времени.

Синтез быстрых алгоритмов с использованием концепции модели сигнала

Важнейшим применением понятия модели сигнала является изучение, вывод и классификация быстрых алгоритмов преобразований. Имеется большое число различных преобразований, широко используемых в ЦОС, таких как ДПФ и ДКП, для которых разработаны быстрые алгоритмы. Получение большинства этих алгоритмов заключается в

искусном обращении с коэффициентами преобразования. Тем не менее, подобные способы не помогают уяснить ни структуру, ни суть быстрых алгоритмов.

В АТОС основная идея заключается в получении быстрого алгоритма по модели сигнала, описывающего преобразование, а не из самого преобразования. Рассмотрим модель сигнала $\mathcal{A} = \mathcal{M} = \mathbb{C}[x]/p(x)$, для которого $\mathcal{F}$ является преобразованием Фурье. Преобразование $\mathcal{F}$ раскладывает модуль $\mathcal{M}$ на неприводимые компоненты, называемые спектром. С точки зрения линейной алгебры $\mathcal{M}$ (как векторное пространство) раскладывается на инвариантные подпространства относительно линейного оператора, играющего роль умножения на x . В случае полиномиальной алгебры преобразование $\mathcal{F}$ является частным случаем КТО

$$\mathcal{F}: \mathbb{C}[x]/p(x) \rightarrow \bigoplus_{0 \leq k < n} \mathbb{C}[x]/(x - \alpha_k), \quad (9)$$

где $\deg(p) = n$ и α_k корни полинома p . Заметим, что каждое слагаемое в правой части имеет размерность равную единице. Суть быстрого алгоритма заключается в поэтапном выполнении (9). Как правило, рассматриваются два основных способа поэтапной декомпозиции (9):

- факторизация $p: p(x) = q(x) \cdot r(x)$;
- декомпозиция $p: p(x) = q(r(x))$.

Получение быстрого алгоритма путём факторизации состоит в рекурсивном разбиении полинома p . Если $p = q \cdot r$, то

$$\mathbb{C}[x]/p(x) \rightarrow \mathbb{C}[x]/q(x) \oplus \mathbb{C}[x]/r(x) \rightarrow \quad (10)$$

$$\rightarrow \bigoplus_{0 \leq i < k} \mathbb{C}[x]/(x - \beta_i) \oplus \bigoplus_{0 \leq j < m} \mathbb{C}[x]/(x - \gamma_j) \rightarrow \quad (11)$$

$$\rightarrow \bigoplus_{0 \leq i < n} \mathbb{C}[x]/(x - \alpha_i). \quad (12)$$

В приведённых выражениях $\deg(q) = k$, $\deg(r) = m$, β_i – корни полинома q , γ_j – корни полинома r , очевидно, что β_i и γ_j подмножества корней α_i полинома p . Шаги (10) и (11) используют КТО, в то время как (12) является простым переупорядочиванием компонент спектра. В [5] доказывается следующая

Теорема 1. Пусть $p(x) = q(x) \cdot r(x)$, выберем c и d в качестве базисов для модулей $\frac{\mathbb{C}[x]}{q(x)}$ и $\frac{\mathbb{C}[x]}{r(x)}$, соответственно, тогда, обозначив через β и γ множества корней q и r соответственно, получаем

$$\mathcal{P}_{b,\alpha} = P(\mathcal{P}_{c,\beta} \oplus \mathcal{P}_{d,\gamma})B. \quad (13)$$

Матрица B отвечает переходу (10), т.е. отображению базиса b в конкатенированный базис (c, d) , а P является матрицей перестановки, которая отображает конкатенацию (β, γ) в список корней α в выражении (12).

Пример: 4-точечное БПФ. Рассмотрим модель сигнала $\mathcal{A} = \mathcal{M} = \mathbb{C}[x]/(x^4 - 1)$ с базисом $b = (1, x^1, x^2, x^3)$ в $\mathcal{M}$. Полиномиальным преобразованием данной модели служит ДПФ. Заметим, что $x^4 - 1 = (x^2 - 1)(x^2 + 1)$, тогда, используя теорему 1, получаем следующую декомпозицию

$$\mathbb{C}[x]/(x^4 - 1) \rightarrow \mathbb{C}[x]/(x^2 - 1) \oplus \mathbb{C}[x]/(x^2 + 1) \rightarrow \quad (14)$$

$$\rightarrow \bigoplus_{0 \leq i < 2} \mathbb{C}[x]/(x - \beta_i) \oplus \bigoplus_{0 \leq j < 2} \mathbb{C}[x]/(x - \gamma_j) \rightarrow \quad (15)$$

$$\rightarrow \bigoplus_{0 \leq i < 4} \mathbb{C}[x]/(x - \alpha_i). \quad (16)$$

Выберем $c = d = (1, x)$ в качестве базисов меньших модулей $\mathbb{C}[x]/(x^2 - 1)$ и $\mathbb{C}[x]/(x^2 + 1)$. Вначале получим матрицу смены базиса B . Для этого необходимо выразить элементы $x^\ell \in b$ в базисе (c, d) :

$$\begin{aligned} 1 &\equiv 1 \bmod (x^2 - 1), 1 \equiv 1 \bmod (x^2 + 1), \\ x &\equiv x \bmod (x^2 - 1), x \equiv x \bmod (x^2 + 1), \end{aligned}$$

$$x^2 \equiv 1 \pmod{x^2 - 1}, x^2 \equiv -1 \pmod{x^2 + 1}, \\ x^3 \equiv x \pmod{x^2 - 1}, x^3 \equiv -x \pmod{x^2 + 1}.$$

Таким образом, получаем отображение:
 $1 \mapsto (1, 1), x \mapsto (x, x), x^2 \mapsto (1, -1), x^3 \mapsto (x, -x)$,
 которое определяет матрицу B :

$$B = \begin{bmatrix} I_2 & I_2 \\ I_2 & -I_2 \end{bmatrix},$$

Полиномиальные преобразования $\mathcal{P}_{c,\beta}$ и $\mathcal{P}_{d,\gamma}$ можно найти исходя из (8) и учитывая, что $\beta = \{1, -1\}$, а $\gamma = \{j, -j\}$,
 $\mathcal{P}_{c,\beta} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \mathcal{P}_{d,\gamma} = \begin{bmatrix} 1 & j \\ 1 & -j \end{bmatrix}.$

Поскольку корни $(x^4 - 1)$ образуют упорядоченное множество $\alpha = (1, j, -1, -j)$, то переходу (16) соответствует матрица

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

которая отображает конкатенацию (β, γ) в множество корней α . Используя (13), запишем полиномиальное преобразование для данной модели сигнала:

$$\mathcal{P}_{b,\alpha} = \text{DFT}_4 = P \begin{bmatrix} \mathcal{P}_{c,\beta} & \\ & \mathcal{P}_{d,\gamma} \end{bmatrix} B = \\ = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & j \\ 1 & -j \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}. \quad (17)$$

Выражение (17) представляет собой факторизацию матрицы 4-точечного ДПФ. Каждая матрица-сомножитель является слабозаполненной, из чего следует, что (17) определяет быстрый алгоритм для DFT_4 .

2. Модели сигналов дискретных тригонометрических преобразований

Дискретные тригонометрические преобразования.

Шестнадцать типов дискретных тригонометрических преобразований (ДТП) (восемь косинусных и восемь синусных преобразований) являются преобразованиями Фурье для соответствующих моделей одномерного конечно-пространства [5]. Наиболее широкое распространение получило ДКП-2, применяемое в стандарте кодирования изображений JPEG [10], а также ДКП-4, применяемое в стандарте кодирования аудио МРЗ, а также при построении косинусно-модулированных банков фильтров.

В отличие от дискретной модели времени, которая определяет ДПФ, для моделей конечно-пространства базисными полиномами являются полиномы Чебышева первого (T_ℓ), второго (U_ℓ), третьего (V_ℓ) и четвертого (W_ℓ) рода.

Полиномы Чебышева

Полиномы Чебышева [11] образуют специальный класс ортогональных многочленов и играют важную роль во многих областях математики. В данном подразделе рассматриваются только те свойства полиномов Чебышева, которые будут использованы в дальнейшем изучении.

Обозначим через $C_0(x) = 1$ и $C_1(x)$ полиномы нулевой и первой степени, тогда $C_n(x)$ для $n > 1$ определяется рекуррентной формулой $C_n(x) = 2xC_{n-1}(x) - C_{n-2}(x)$.

Данный ряд полностью определяется условием $C_0 = 1$ и выбором C_1 . Для ЦОС практическую значимость имеют четыре частных случая полиномов Чебышева [5]. Они обозначаются, как $C \in \{T, U, V, W\}$ и называются полиномами Чебышева первого, второго, третьего и четвертого рода (Таблица 1).

В дальнейшем рассмотрении понадобятся сведения факторизации полиномов Чебышева над полем рациональных чисел [4]. В [12] доказываются теоремы, которые будут нами использованы в дальнейшем.

Таблица 1
Полиномы Чебышева 1-4 рода.

	Первые члены ряда	Аналитический вид $(\cos \theta = x)$	Вид симметрии	Корни, $0 \leq k < n$
T_n	$1, x$	$\cos(n\theta)$	$T_{-n} = T_n$	$\cos \frac{(k+\frac{1}{2})\pi}{n}$
U_n	$1, 2x$	$\frac{\sin(n+1)\theta}{\sin \theta}$	$U_{-n} = -U_{n-2}$	$\cos \frac{(k+1)\pi}{n+1}$
V_n	$1, 2x - 1$	$\frac{\cos(n+\frac{1}{2})\theta}{\cos \frac{1}{2}\theta}$	$V_{-n} = V_{n-1}$	$\cos \frac{(k+\frac{1}{2})\pi}{n+\frac{1}{2}}$
W_n	$1, 2x + 1$	$\frac{\sin(n+\frac{1}{2})\theta}{\sin \frac{1}{2}\theta}$	$W_{-n} = -W_{n-1}$	$\cos \frac{(k+1)\pi}{n+\frac{1}{2}}$

Теорема 2. Пусть $n > 1$ целое число, тогда

$$T_n(x) = 2^{n-1} \prod_h D_h(x), \quad (18)$$

$$D_h(x) = \prod_{\substack{k=1 \\ (2k-1)n=h}}^n \left(x - \cos \frac{(k+\frac{1}{2})\pi}{n} \right),$$

где $h \leq n$ пробегает через все положительные делители n и $D_h(x)$ – неприводимый полином над полем рациональных чисел.

Теорема 3. Пусть $n \geq 2$ целое число, тогда

$$U_n(x) = \prod_h E_h(x), \quad (19)$$

$$E_h(x) = 2^{l_h} \prod_{\substack{k=1 \\ (k, 2n+2)=h}}^n \left(x - \cos \frac{(k+1)\pi}{n+1} \right),$$

где $h \leq n$ пробегает все положительные делители числа $(2n+2)$ и $E_h(x)$ – неприводимые полиномы над полем рациональных чисел, через (k, n) обозначается наибольший общий делитель чисел k и n . В последнем выражении $l_h = \phi((2n+2)/h)/2$ (ϕ – функция Эйлера).

Модели сигнала для ДКП-2 и ДКП-4

В [5] показано, что ДКП-2 отвечает модель сигнала $\mathcal{M} = \mathcal{A} = \mathbb{C}[x]/2(x-1)U_{n-1}$ с обобщенным z-преобразованием, которое в данном случае носит название конечно- V -преобразования

$$\mathcal{Z}: \mathbb{C}^n \rightarrow \mathcal{M}, \quad s \mapsto \sum_{0 \leq \ell < n} s_\ell V_\ell \in \mathcal{M}, \quad \text{где } s \in \mathbb{C}^n.$$

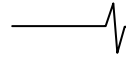
После того, как определена модель сигнала $(\mathcal{A}, \mathcal{M}, \mathcal{Z})$ остальные концепции, такие как преобразование Фурье и свертка выводятся автоматически. Покажем, что для рассматриваемой модели сигнала преобразование Фурье совпадает с ДКП-2. Согласно таблице 1, корни полинома $2(x-1)U_{n-1}$ задаются выражением $\alpha_k = \cos k\pi/n, 0 \leq k < n$. Таким образом, преобразование Фурье для $\mathcal{M}$ в соответствии с (8)

$$\mathcal{P}_{b,\alpha} = [V_\ell(\alpha_k)]_{0 \leq \ell, k < n} = \left[\frac{1}{\cos k\pi/2n} \cdot \cos \frac{k(\ell + \frac{1}{2})\pi}{n} \right]_{0 \leq \ell, k < n}.$$

Для получения ДКП-2 необходимо выполнить масштабирование полученного преобразования

$$\text{DCT}_{2n} = \text{diag}_{0 \leq k < n} (\cos k\pi/(2n)) \cdot [V_\ell(\alpha_k)]_{0 \leq \ell, k < n}. \quad (20)$$

Тем самым показывается, что DCT_{2n} является преобразованием Фурье для регулярного модуля $\mathcal{M} = \mathbb{C}[x]/(x-1)U_{n-1}$.



ДКП-4 соответствует модель сигнала $\mathcal{M} = \mathcal{A} = \mathbb{C}[x]/2T_n$ с базисом $\mathbf{b} = (V_0, \dots, V_{n-1})$ и обобщённым z -преобразованием таким же, как и у ДКП-2. Корнями полинома $2T_n(x)$ являются числа $\alpha_k = \cos((k+\frac{1}{2})\pi/n)$. Следовательно, полиномиальное преобразование данной модели сигнала задаётся как

$$\mathcal{P}_{b,a} = [V_\ell(\alpha_k)]_{0 \leq \ell, k < n} = \left[\frac{1}{\cos(k + \frac{1}{2})\pi/(2n)} \cdot \cos \frac{(k + \frac{1}{2})(\ell + \frac{1}{2})\pi}{n} \right]_{0 \leq \ell, k < n}. \quad (21)$$

ДКП-4 получается из (21) умножением на диагональную матрицу

$$\text{DCT}4_n = \text{diag}_{0 \leq k < n} \left(\cos \frac{(k + \frac{1}{2})\pi}{2n} \right) \cdot [V_\ell(\alpha_k)]_{0 \leq \ell, k < n}. \quad (22)$$

Каждому дискретному тригонометрическому преобразованию ДТТ отвечает полиномиальное преобразование, которое обозначается как $\overline{\text{DCT}}$. Например $\overline{\text{DCT}}2_n$ соответствует матрица (20).

3. Синтез быстрых алгоритмов ДКП-2 и ДКП-4

Основная идея

В разделе 1 было показано, что синтез быстрого алгоритма преобразования, связанного с алгеброй $\mathcal{A} = \mathbb{C}[x]/p(x)$, строится на основе факторизации полинома $p(x)$. Для ДКП-2 и ДКП-4 в роли $p(x)$ выступают полиномы $2(x-1)U_{n-1}(x)$ и $2T_n(x)$, соответственно. Основная идея состоит в использовании для синтеза быстрых алгоритмов факторизации $U_{n-1}(x)$ и $T_n(x)$ соотношений (19) и (18) соответственно. При этом в качестве основного поля изначально выбирается поле рациональных чисел $\mathbb{Q}$:

$$\mathbb{Q}[x]/2(x-1)U_{n-1}(x) \rightarrow \mathbb{Q}[x]/2(x-1) \oplus \bigoplus_h \mathbb{Q}[x], \quad (23)$$

$$\mathbb{Q}[x]/2T_n(x) \rightarrow \bigoplus_h \mathbb{Q}[x]/D_h(x). \quad (24)$$

Разложения (23) и (24) требуют операции умножения на элементы из поля рациональных чисел, которые, как правило, имеют простую аппаратную реализацию. Отсутствие нетривиальных умножений обусловлено тем, что полиномы $U_{n-1}(x)$ и $T_n(x)$ имеют разложение над $\mathbb{Q}$.

Поскольку полиномы E_h и D_h неприводимы над $\mathbb{Q}$, то для выполнения декомпозиции подмодулей $\mathbb{Q}[x]/E_h(x)$ и $\mathbb{Q}[x]/D_h(x)$ необходимо расширить основное поле $\mathbb{Q}$ до поля разложения полинома E_h и D_h соответственно. Такое расширение выполняется последовательно, используя башню полей, которая строится с использованием основной теоремы теории Галуа [13].

В следующем разделе приводится описание основных понятий из теории групп и теории Галуа, которые используются при описании процедуры синтеза быстрых алгоритмов.

Основные понятия теории групп и теории Галуа

Определение группы. Группой G называется совокупность элементов, на которой задана групповая операция « $\cdot$ », сопоставляющая любой паре элементов $g_1, g_2 \in G$ некоторый элемент g_3 из той же совокупности G . При этом групповая операция должна удовлетворять трём условиям:

- ассоциативность: $g_1 \cdot (g_2 \cdot g_3) = (g_1 \cdot g_2) \cdot g_3$;
- существование единицы: в группе есть такой элемент e (иногда обозначается «1»), что $e \cdot g = g$, $e = g$ для всех $g \in G$;
- существование обратного элемента: для любого $g \in G$ существует такой $g^{-1} \in G$, что $g \cdot g^{-1} = g^{-1} \cdot g = e$.

Поле разложение полинома. Для полинома $p(x) \in \mathbb{F}[x]$ полем разложения называют наименьшее расширение $\mathbb{F}$, которое содержит все корни $p(x)$. Например, для $p(x) =$

$x^2 - 2$ полем разложения является $\mathbb{Q}(\sqrt{2})$. Поле $\mathbb{Q}(\sqrt{2})$ есть расширение поля $\mathbb{Q}$, которое образуется из $\mathbb{Q}$ присоединением числа $\sqrt{2}$. Все элементы поля $\mathbb{Q}(\sqrt{2})$ имеют вид: $a + b\sqrt{2}$, где $a, b \in \mathbb{Q}$.

Аutomорфизмы полей. Взаимнооднозначное отображение $\mathcal{B}: G \rightarrow G$ группы на себя, сохраняющее групповую операцию, называют автоморфизмом. Группа автоморфизмов обозначается $\text{Aut } G$. Автоморфизмы полей $\mathcal{B}: \mathbb{F} \rightarrow \mathbb{F}$ определяются аналогично, с тем уточнением, что взаимнооднозначное отображение $\mathcal{B}$ поля на себя обязано сохранять обе операции [12],

$$\mathcal{B}(x + y) = \mathcal{B}(x) + \mathcal{B}(y), \mathcal{B}(xy) = \mathcal{B}(x)\mathcal{B}(y), x, y \in \mathbb{F}.$$

Пример: определим функцию $f: \mathbb{Q}(\sqrt{2}) \rightarrow \mathbb{Q}(\sqrt{2})$

следующим образом

$$f(a + b\sqrt{2}) = a - b\sqrt{2}, \quad (25)$$

тогда f – автоморфизм поля $\mathbb{Q}(\sqrt{2})$. Таким образом, суть идеи автоморфизма полей состоит в перенумерации элементов поля, без изменения его структуры в целом.

Пусть $\mathbb{E}$ нормальное расширение поля $\mathbb{F}$ (т.е. $\mathbb{E} \supset \mathbb{F}$). В группе автоморфизмов $\text{Aut } \mathbb{E}$ выделим подгруппу $\text{Aut } \mathbb{E}/\mathbb{F}$ тех автоморфизмов $\mathcal{B}: \mathbb{E} \rightarrow \mathbb{E}$, которые поле $\mathbb{F}$ оставляют на месте, т.е. $\mathcal{B}(x) = x$, если $x \in \mathbb{F}$. Элементы группы $\text{Aut } \mathbb{E}/\mathbb{F}$ называют автоморфизмами поля $\mathbb{E}$. Например, автоморфизм (25) определяет $\mathbb{Q}$ -автоморфизм поля $\mathbb{Q}(\sqrt{2})$.

Рассмотрим полином $p(x) \in \mathbb{F}[x]$, не имеющий кратных корней, полем разложения которого является расширение $\mathbb{E}$, тогда группу $\text{Aut } \mathbb{E}/\mathbb{F}$ называют группой Галуа полинома $p(x)$ (или соответствующего расширения $\mathbb{E}$) и обозначают $\text{Gal}(\mathbb{E}/\mathbb{F})$. Возьмём полином $p(x) = x^2 - 2$, полем разложения которого является $\mathbb{Q}(\sqrt{2})$. Группа Галуа полинома $p(x)$ состоит из двух элементов $\text{Gal}(\mathbb{Q}(\sqrt{2})/\mathbb{Q}) = \{f, g\}$, где f определена в (25), а $g(a + b\sqrt{2}) = a + b\sqrt{2}$. Очевидно, что g – единичный элемент группы (оставляет все на своих местах), также справедливо тождество $f \cdot f = g$. Полученная группа является циклической группой второго порядка.

Соответствие Галуа. Стержнем теории Галуа является соответствие между структурой расширения полей и структурой подгрупп автоморфизмов. Каждой подгруппе $H \subset \text{Gal}(\mathbb{E}/\mathbb{F})$ отвечает подполе $\mathbb{L} \subset \mathbb{E}$, состоящее из элементов $\mathbb{F}$, неподвижных под действием автоморфизмов из H . И наоборот, каждому подполю $\mathbb{L} \subset \mathbb{E}$ отвечает подгруппа H автоморфизмов, оставляющих элементы $\mathbb{L}$ на месте. В результате изучение всех подполей поля $\mathbb{E}$ сводится к изучению всех подгрупп группы $\text{Gal}(\mathbb{E}/\mathbb{F})$. При этом каждой башне (цепочке вложенных) полей

$$\mathbb{F} = \mathbb{L}_0 \subset \mathbb{L}_1 \subset \dots \subset \mathbb{L}_r = \mathbb{E}, \quad (26)$$

отвечает нормальный ряд вложенных (в противоположном направлении) групп

$$\text{Gal}(\mathbb{E}/\mathbb{F}) = G_0 \supset G_1 \supset \dots \supset G_r = \{1\}, \quad (27)$$

и наоборот (соответствие Галуа).

Если имеется неприводимый полином $p(x)$ с коэффициентами из поля $\mathbb{F}$, полем разложения которого является $\mathbb{E}$, то используя соответствие Галуа для него, можно построить башню полей (26). Причём в каждом поле $\mathbb{L}_i$ $p(x)$ будет распадаться в произведение неприводимых полиномов, а в поле $\mathbb{E}$ разложиться в произведение линейных сомножителей. Таким образом, использование (26) позволяет выполнить поэтапную факторизацию полинома $p(x)$, что и требуется при синтезе быстрого алгоритма дискретного тригонометрического преобразования.

Быстрый алгоритм 4-точечного ДКП-2

В данном разделе подробно рассматривается пример синтеза быстрого алгоритма 4-точечного ДКП-2 на базе теоретического материала изложенного в предыдущих разделах. Исходным пунктом является модель

сигнала для ДКП-2:

$$\mathcal{A} = \mathcal{M} = \frac{\mathbb{Q}[x]}{V_4 - V_3} = \frac{\mathbb{Q}[x]}{2(x-1)U_3}, \quad (3)$$

$$b = (V_0, \dots, V_3)$$

Рассмотрим полином $p(x) = 2(x-1)U_3(x)$, используя выражение (19) для факторизации $U_3(x)$ над полем $\mathbb{Q}$, получаем

$$p(x) = (2x-2)(2x)(4x^2-2), \quad (29)$$

откуда, объединив на время сомножители, взятые в фигурную скобку, находим

$$p(x) = V_4(x) - V_3(x) = (4x^2 - 4x)(4x^2 - 2) = (V_2(x) - V_1(x))(V_2(x) + V_1(x)). \quad (30)$$

Согласно таблице 1, корни полинома $p(x)$ равны $\alpha_k = \cos k\pi/4$, $k = 0, \dots, 3$. Поскольку для исходного модуля $\mathcal{M}$ задан базис $b = (V_0, \dots, V_3)$, нам важно факторизовать полином p в том же базисе. Теперь, используя (30) и КТО, получаем:

$$\mathbb{Q}[x]/(V_4 - V_3) \rightarrow \mathbb{Q}[x]/(V_2 - V_1) \oplus \mathbb{Q}[x]/(V_2 + V_1). \quad (31)$$

Преобразованию (31) соответствует матрица смены базиса B_4 . Для определения B_4 необходимо выразить базисные элементы $V_\ell \in b$ в базисе (c, d) , где c и d – это базис подмодуля $\mathbb{Q}[x]/(V_2 - V_1)$ и $\mathbb{Q}[x]/(V_2 + V_1)$, соответственно. В таблице 2 производятся необходимые вычисления.

Используя последний столбец таблицы 2, получаем матрицу B_4 смены базиса

$$b \mapsto (c, d): B_4 = \begin{bmatrix} I_2 & J_2 \\ I_2 & -J_2 \end{bmatrix}.$$

Важно отметить тот факт что, разложение полинома $p = V_4 - V_3 = (V_2 - V_1)(V_2 + V_1)$ над полем $\mathbb{Q}$ приводит к тому, что переход (31), реализуемый в виде матрицы B_4 , не требует умножения на числа не из поля $\mathbb{Q}$.

Таблица 2

Переход от базиса b к базису (c, d) , где $c = d = (V_0, V_1)$

Номер базисного элемента ℓ	$V_\ell \bmod (V_2 - V_1)$	$V_\ell \bmod (V_2 + V_1)$	$b \mapsto (c, d)$
0	$V_0 \bmod (V_2 - V_1) = V_0,$	$V_0 \bmod (V_2 + V_1) = V_0,$	$V_0 \mapsto (V_0, V_0)$
1	$V_1 \bmod (V_2 - V_1) = V_1,$	$V_1 \bmod (V_2 + V_1) = V_1,$	$V_1 \mapsto (V_1, V_1)$
2	$V_2 \bmod (V_2 - V_1) = V_1,$	$V_2 \bmod (V_2 + V_1) = -V_1,$	$V_2 \mapsto (V_1, -V_1)$
3	$V_3 \bmod (V_2 - V_1) = V_0,$	$V_3 \bmod (V_2 + V_1) = -V_0,$	$V_3 \mapsto (V_0, -V_0)$

Отметим, что слагаемые в правой части (31) соответствуют алгебрам для 2-точечных ДКП-2 и ДКП-4, соответственно. Тем самым задача синтеза быстрого алгоритма 4-точечного ДКП-2 распадается на две подзадачи синтеза быстрых алгоритмов для 2-точечных ДКП-2 и ДКП-4.

Продолжим рассмотрение модуля $\mathbb{Q}[x]/(V_2 - V_1)$, который раскладывается на два неприводимых модуля (размерности один), используя соотношение $V_2(x) - V_1(x) = (2x-2)(2x) = (V_1(x)-1)(V_1(x)+1)$ следующим образом

$$\mathbb{Q}[x]/(V_2 - V_1) \rightarrow \mathbb{Q}[x]/(V_1 - 1) \oplus \mathbb{Q}[x]/(V_1 + 1). \quad (32)$$

Преобразованию (32) отвечает матрица

$$B_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

Шаги (31)-(32) позволяют выполнить частичную декомпозицию исходного модуля $\mathcal{M}$:

$$\mathbb{Q}[x]/(V_4 - V_3) \rightarrow \mathbb{Q}[x]/(V_1 - 1) \oplus \mathbb{Q}[x]/(V_1 + 1) \oplus \mathbb{Q}[x]/(V_2 + V_1), \quad (33)$$

которой соответствует факторизация (29). Наша конечная цель состоит в полном разложении $\mathcal{M}$ на неприводимые подмодули размерности один. Чтобы продолжить разложение (33), необходимо расширить основное поле $\mathbb{Q}$.

Рассмотрим полином $V_2(x) + V_1(x) = 4x^2 - 2$, очевидно, что полем его разложения является $\mathbb{Q}(\sqrt{2})$, которое для краткости обозначим $\mathbb{Q}_{\sqrt{2}}$:

$$V_2(x) + V_1(x) = (2x - \sqrt{2})(2x + \sqrt{2}) = (V_1(x) + (1 - \sqrt{2}))(V_1(x) + (1 + \sqrt{2})).$$

Воспользуемся данной факторизацией для разложения модуля $\mathbb{Q}[x]/(V_2 + V_1)$:

$$\mathbb{Q}[x]/(V_2 + V_1) \rightarrow \mathbb{Q}_{\sqrt{2}}[x]/(V_1 - (1 - \sqrt{2})) \oplus \mathbb{Q}_{\sqrt{2}}[x]/(V_1 + (1 + \sqrt{2})), \quad (34)$$

учитывая, что

$$V_0 \bmod (V_1 + (1 - \sqrt{2})) = V_0 \equiv 1,$$

$$V_1 \bmod (V_1 + (1 - \sqrt{2})) = \sqrt{2} - 1,$$

$$V_0 \bmod (V_1 + (1 + \sqrt{2})) = V_0 \equiv 1,$$

$$V_1 \bmod (V_1 + (1 + \sqrt{2})) = -\sqrt{2} - 1,$$

матрица перехода, отвечающая (34), запишется как

$$T_2 = \begin{bmatrix} 1 & \sqrt{2} - 1 \\ 1 & -\sqrt{2} - 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -1 \\ \sqrt{2} & \end{bmatrix}. \quad (35)$$

Из (35) легко заметить, что умножение на матрицу T_2 требует только одной операции умножения.

Выполнение (34) завершает полное (позатпное) разложение модуля $\mathcal{M} = \mathbb{Q}[x]/(V_4 - V_3)$, описывающего 4-точечное ДКП-2, на неприводимые подмодули размерности один (Рис. 1).

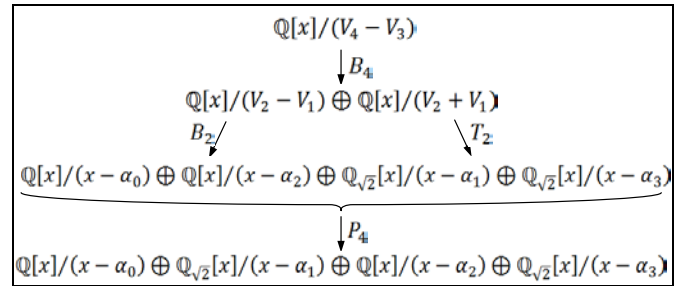


Рис. 1 – Общая схема разложения модуля $\mathcal{M} = \mathbb{Q}[x]/(V_4 - V_3)$

На рис. 1 учитывается, что

$$\mathbb{Q}[x]/(V_1 - 1) \cong \mathbb{Q}[x]/(x - 1) = \mathbb{Q}[x]/(x - \alpha_0),$$

$$\mathbb{Q}[x]/(V_1 + 1) \cong \mathbb{Q}[x]/(x - 0) = \mathbb{Q}[x]/(x - \alpha_2),$$

$$\mathbb{Q}_{\sqrt{2}}[x]/(V_1 - (1 - \sqrt{2})) \cong$$

$$\cong \mathbb{Q}_{\sqrt{2}}[x]/(x - \sqrt{2}/2) = \mathbb{Q}_{\sqrt{2}}[x]/(x - \alpha_1),$$

$$\mathbb{Q}_{\sqrt{2}}[x]/(V_1 + (1 + \sqrt{2})) \cong$$

$$\cong \mathbb{Q}_{\sqrt{2}}[x]/(x + \sqrt{2}/2) = \mathbb{Q}_{\sqrt{2}}[x]/(x - \alpha_3).$$

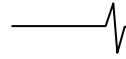
Матрица P_4 отвечает за перестановку неприводимых подмодулей в нужном порядке

$$P_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Полиномиальное преобразование для модели сигнала (28), описывающей 4-точечное ДКП-2, имеет вид $\text{DCT}2_4 = \mathcal{P}_{b,\alpha} = P_4(B_2 \oplus T_2)B_4$.

В соответствии с (20), для получения матрицы ДКП-2 необходимо $\text{DCT}2_4$ умножить на масштабирующую диагональную матрицу: $\text{DCT}2_4 = \text{diag}_{0 \leq k < 4}(\cos k\pi/8) \cdot \text{DCT}2_4$.

В итоге получен быстрый алгоритм 4-точечного ДКП-2,



требующий четырех операций умножения (одно умножение для DCT_4 и три умножения для $\text{diag}_{0 \leq k < 4}(\cos k\pi/8)$).

Быстрый алгоритм 4-точечного ДКП-4

В качестве ещё одного примера рассмотрим синтез быстрого алгоритма 4-точечного ДКП-4, модель сигнала которого приведена ниже

$$\mathcal{A} = \mathcal{M} = \frac{\mathbb{Q}[x]}{V_4 + V_3} = \frac{\mathbb{Q}[x]}{2T_4}, b = (V_0, \dots, V_3).$$

Анализ полинома $p(x) = 2T_4(x)$ с использованием выражения (18) показывает, что $T_4(x)$ неприводим над полем $\mathbb{Q}$. Поэтому для его поэтапной факторизации необходимо расширить основное поле $\mathbb{Q}$. Корнями полинома $T_4(x)$ являются $\alpha_k = \cos(k\frac{1}{2}\pi/4)$, $k = 0, \dots, 3$, поэтому $\mathbb{Q}(\alpha_0, \alpha_1, \alpha_2, \alpha_3)$ – поле разложения полинома $p(x)$. Если заметить, что

$$\alpha_0 = \cos \frac{\pi}{8} = -\alpha_3 = -\cos \frac{7\pi}{8} = \frac{1}{2}\sqrt{2+\sqrt{2}},$$

$$\alpha_1 = \cos \frac{3\pi}{8} = -\alpha_2 = -\cos \frac{5\pi}{8} = \frac{1}{2}\sqrt{2-\sqrt{2}},$$

то поле разложения полинома $p(x)$ запишется как $\mathbb{Q}(\alpha_0, \alpha_1) \cong \mathbb{Q}(\sqrt{2+\sqrt{2}})$. Поле $\mathbb{Q}(\alpha_0, \alpha_1)$ можно рассматривать, как четырехмерное векторное пространство, поскольку любой элемент $\theta \in \mathbb{Q}(\alpha_0, \alpha_1)$ представим в виде $\theta = a + b\alpha_0 + c\alpha_1 + d\alpha_0\alpha_1$, где $a, b, c, d \in \mathbb{Q}$. Группа $\text{Aut } \mathbb{Q}(\alpha_0, \alpha_1)/\mathbb{Q}$ является группой Галуа полинома $p(x)$. Элементами группы Галуа являются следующие автоморфизмы:

$$g_0(\theta) = \theta, g_1(\theta) = a + c\alpha_0 + b\alpha_1 + d\alpha_0\alpha_1,$$

$$g_2(\theta) = a + b\alpha_0 + c\alpha_1 - d\alpha_0\alpha_1,$$

$$g_3(\theta) = a + c\alpha_0 + b\alpha_1 - d\alpha_0\alpha_1.$$

Приведём таблицу Кэли для данной группы автоморфизмов (таблица 3).

Таблица 3

Таблица Кэли группы $\text{Gal}\left(\frac{\mathbb{Q}(\alpha_0, \alpha_1)}{\mathbb{Q}}\right) = \text{Aut}\frac{\mathbb{Q}(\alpha_0, \alpha_1)}{\mathbb{Q}}$

$\circ$	g_0	g_1	g_2	g_3
g_0	g_0	g_1	g_2	g_3
g_1	g_1	g_0	g_3	g_2
g_2	g_2	g_3	g_0	g_1
g_3	g_3	g_2	g_1	g_0

Из таблицы 3 видно, что $H \in \{g_0, g_1\}$ образует подгруппу группы $\text{Gal}(\mathbb{Q}(\alpha_0, \alpha_1)/\mathbb{Q})$, которая определяет подполе $\mathbb{Q}_H$ такое, что

$$\mathbb{Q}_H = \{\theta \in \mathbb{Q}(\alpha_0, \alpha_1) | g(\theta) = \theta \forall g \in H\}.$$

В рассматриваемом случае $\mathbb{Q}_H = \mathbb{Q}(\alpha_0\alpha_1) \cong \mathbb{Q}(\sqrt{2})$. Таким образом, согласно теории Галуа (26-(27), получаем нормальный ряд вложенных групп $\text{Gal}(\mathbb{Q}(\alpha_0, \alpha_1)/\mathbb{Q}) \supset H \supset \{1\}$, которому отвечает башня полей $\mathbb{Q} \subset \mathbb{Q}(\alpha_0\alpha_1) \subset \mathbb{Q}(\alpha_0, \alpha_1)$. (36)

На основании (36) можно предложить следующую схему факторизации полинома $p(x)$,

$$p = \frac{V_4 + V_3}{\mathbb{Q}} = \frac{(V_2 + V_1 - \sqrt{2})(V_2 + V_1 + \sqrt{2})}{\mathbb{Q}(\alpha_0\alpha_1)} =$$

$$= \underbrace{\left(V_1 + \left(1 - \sqrt{2 + \sqrt{2}}\right)\right) \left(V_1 + \left(1 + \sqrt{2 + \sqrt{2}}\right)\right)}_{\mathbb{Q}(\alpha_0, \alpha_1)} \left(V_1 + \left(1 - \sqrt{2 - \sqrt{2}}\right)\right) \left(V_1 + \left(1 + \sqrt{2 - \sqrt{2}}\right)\right), \quad (37)$$

где над фигурными скобками указаны соответствующие поля разложения.

Руководствуясь (37), предлагается следующая схема разложения модуля $\mathcal{M} = \mathbb{Q}[x]/p$

$$\begin{aligned} \mathbb{Q}[x]/(V_4 + V_3) &\xrightarrow{R_4} \mathbb{Q}_{\alpha_0\alpha_1}[x]/(V_2 + V_1 - \sqrt{2}) \oplus \\ &\oplus \mathbb{Q}_{\alpha_0\alpha_1}[x]/(V_2 + V_1 + \sqrt{2}) \\ &\xrightarrow{R_2 \oplus S_2} \mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(V_1 + (1 - \sqrt{2 + \sqrt{2}})) \oplus \\ &\oplus \mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(V_1 + (1 + \sqrt{2 + \sqrt{2}})) \oplus \\ &\oplus \mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(V_1 + (1 - \sqrt{2 - \sqrt{2}})) \oplus \\ &\oplus \mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(V_1 + (1 + \sqrt{2 - \sqrt{2}})) \\ &\xrightarrow{P_4} \bigoplus_{0 \leq i < 4} \mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(x - \alpha_i). \end{aligned} \quad (38)$$

Над стрелками в (38) указаны матрицы, отвечающие за соответствующий переход, также во всех подмодулях предполагается, что базис состоит из полиномов Чебышева третьего рода V_ℓ . В таблице 4 приведены вычисления, необходимые для определения матрицы R_4 .

Таблица 4.

Вычисление матрицы R_4

Номер базисного элемента ℓ	$V_\ell \bmod (V_2 + V_1 - \sqrt{2})$	$V_\ell \bmod (V_2 + V_1 + \sqrt{2})$
0	V_0	V_0
1	V_1	V_1
2	$-V_1 + \sqrt{2}V_0$	$-V_1 - \sqrt{2}V_0$
3	$\sqrt{2}V_1 - V_0$	$-\sqrt{2}V_1 - V_0$

Используя таблицу 4, можно записать матрицу

$$\begin{aligned} R_4 &= \begin{bmatrix} 1 & \sqrt{2} & -1 \\ & 1 & -1 & \sqrt{2} \\ 1 & -\sqrt{2} & -1 \\ & 1 & -1 & -\sqrt{2} \end{bmatrix} = \\ &= \begin{bmatrix} 1 & & 1 & \\ & 1 & & 1 \\ 1 & -1 & & \\ & 1 & -1 & \end{bmatrix} \cdot \begin{bmatrix} 1 & & -1 \\ & 1 & -1 & \sqrt{2} \\ & & \sqrt{2} \\ & & & \sqrt{2} \end{bmatrix}. \end{aligned}$$

Исходя из соотношений:

$$V_0 \bmod (V_1 + (1 - \sqrt{2 + \sqrt{2}})) = V_0 \equiv 1,$$

$$V_1 \bmod (V_1 + (1 - \sqrt{2 + \sqrt{2}})) = \sqrt{2 + \sqrt{2}} - 1,$$

$$V_0 \bmod (V_1 + (1 + \sqrt{2 + \sqrt{2}})) = V_0 \equiv 1,$$

$$V_1 \bmod (V_1 + (1 + \sqrt{2 + \sqrt{2}})) = -\sqrt{2 + \sqrt{2}} - 1,$$

определяется матрица R_2

$$R_2 = \begin{bmatrix} 1 & \sqrt{2 + \sqrt{2}} - 1 \\ 1 & -\sqrt{2 + \sqrt{2}} - 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -1 \\ \sqrt{2 + \sqrt{2}} \end{bmatrix}.$$

Аналогично из

$$V_0 \bmod (V_1 + (1 - \sqrt{2 - \sqrt{2}})) = V_0 \equiv 1,$$

$$V_1 \bmod (V_1 + (1 - \sqrt{2 - \sqrt{2}})) = \sqrt{2 - \sqrt{2}} - 1,$$

$$V_0 \bmod (V_1 + (1 + \sqrt{2 - \sqrt{2}})) = V_0 \equiv 1,$$

$$V_1 \bmod (V_1 + (1 + \sqrt{2 - \sqrt{2}})) = -\sqrt{2 - \sqrt{2}} - 1,$$

находится матрица

$$S_2 = \begin{bmatrix} 1 & \sqrt{2-\sqrt{2}}-1 \\ 1 & -\sqrt{2-\sqrt{2}}-1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -1 \\ \sqrt{2-\sqrt{2}} \end{bmatrix}.$$

Поскольку

$$\mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(v_1+(1-\sqrt{2+\sqrt{2}})) \cong \mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(x-\alpha_0),$$

$$\mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(v_1+(1+\sqrt{2+\sqrt{2}})) \cong \mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(x-\alpha_3),$$

$$\mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(v_1+(1-\sqrt{2-\sqrt{2}})) \cong \mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(x-\alpha_1),$$

$$\mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(v_1+(1+\sqrt{2-\sqrt{2}})) \cong \mathbb{Q}_{(\alpha_0, \alpha_1)}[x]/(x-\alpha_2),$$

то матрица перестановки в (38) определяется как

$$P_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

Полиномиальное преобразование 4-точечного ДКП-4, согласно пошаговой декомпозиции (38), имеет вид

$$\overline{\text{DCT}}_4 = \mathcal{P}_{b,\alpha} = P_4(R_2 \oplus S_2)R_4.$$

Матрица ДКП-4 получается из $\overline{\text{DCT}}_4$, согласно (22), умножением на масштабирующую диагональную матрицу:

$$\text{DCT}_4 = \text{diag}_{0 \leq k < 4}(\cos(k + \frac{1}{2})\pi/8) \cdot \overline{\text{DCT}}_4.$$

В итоге получен быстрый алгоритм 4-точечного ДКП-4, требующий восемь операций умножения (четыре умножения для $\overline{\text{DCT}}_4$ и четыре умножения для $\text{diag}_{0 \leq k < 4}(\cos k\pi/8)$).

Быстрый алгоритм 8-точечного ДКП-2

Самое широкое распространение в последнее время получил алгоритм 8-точечного ДКП-2, что связано с его применением в стандарте кодирования изображений JPEG [10]. Применим предлагаемый в статье подход к синтезу быстрого алгоритма 8-точечного ДКП-2. Как и в предыдущих примерах, начальным пунктом является модель сигнала, описывающего ДКП-2:

$$\mathcal{A} = \mathcal{M} = \frac{\mathbb{Q}[x]}{V_8 - V_7} = \frac{\mathbb{Q}[x]}{2(x-1)U_7}, \mathbf{b} = (V_0, \dots, V_7)$$

Рассмотрим полином $p(x) = 2(x-1)U_7(x)$, используя выражение (19) для факторизации $U_7(x)$ над полем $\mathbb{Q}$, получаем $p(x) = (2x-2)(2x)(4x^2-2)(16x^4-16x+2)$,

откуда, объединив на время сомножители, взятые в фигурную скобку, находим

$$p(x) = (16x^4 - 16x^3 - 8x^2 + 8x)(16x^4 - 16x + 2) = (V_4(x) - V_3(x))(V_4(x) + V_3(x)). \quad (39)$$

Согласно таблице 1, корни полинома $p(x)$ равны $\alpha_k = \cos k\pi/8, k = 0, \dots, 7$. Теперь, используя (39) и КТО, можно перейти от исходного модуля $\mathcal{M}$ размерности восемь к прямой сумме двух подмодулей размерности четыре:

$$\mathbb{Q}[x]/(V_8 - V_7) \xrightarrow{B_8} \mathbb{Q}[x]/(V_4 - V_3) \oplus \mathbb{Q}[x]/(V_4 + V_3) \quad (40)$$

Преобразованию (40) соответствует матрица смены базиса B_8 . Для определения B_8 необходимо выразить базисные элементы $\mathbf{b}$ в базисе (c, d) , где $c = d = (V_0, \dots, V_3)$ это базис подмодуля $\mathbb{Q}[x]/(V_4 - V_3)$ и $\mathbb{Q}[x]/(V_4 + V_3)$, соответственно. Выполняя действия, аналогичные выполненным в таблице 3, получаем

$$B_8 = \begin{bmatrix} I_4 & J_4 \\ I_4 & -J_4 \end{bmatrix}.$$

Алгебры в правой части (40) соответствуют 4-точечным ДКП-2 и ДКП-4, соответственно. Тем самым задача синтеза быстрого алгоритма 8-точечного ДКП-2 распадается на две подзадачи синтеза быстрых алгоритмов 4-точечных ДКП-2 и ДКП-4, которые были решены в предыдущих разделах. Таким образом, общая схема разложения модуля $\mathbb{Q}[x]/(V_8 - V_7)$ запишется как

$$\mathbb{Q}[x]/(V_8 - V_7) \xrightarrow{B_8} \mathbb{Q}[x]/(V_4 - V_3) \oplus \mathbb{Q}[x]/(V_4 + V_3)$$

$$\xrightarrow{\overline{\text{DCT}}_4 \oplus \overline{\text{DCT}}_4} \bigoplus_{i=0,2,4,6} \frac{\mathbb{Q}_{(\alpha_2)}[x]}{(x-\alpha_i)} \oplus \quad (41)$$

$$\oplus \bigoplus_{i=1,3,5,7} \mathbb{Q}_{(\alpha_1, \alpha_3)}[x]/(x-\alpha_i)$$

$$\xrightarrow{P_8} \bigoplus_{0 \leq i < 8} \mathbb{Q}_{(\alpha_1, \alpha_3)}[x]/(x-\alpha_i).$$

В (41) P_8 – матрица перестановки неприводимых подмодулей в соответствии с индексами корней полинома $p = V_8 - V_7$. Отметим также, что $\mathbb{Q}_{(\alpha_2)} \cong \mathbb{Q}_{\sqrt{2}}$ является подполем $\mathbb{Q}_{(\alpha_1, \alpha_3)} \cong \mathbb{Q}_{(\sqrt{2+\sqrt{2}})}$. Используя (41) получаем быстрый алгоритм полиномиального 8-точечного ДКП-2 $\overline{\text{DCT}}_8 = \mathcal{P}_{b,\alpha} = P_8(\overline{\text{DCT}}_4 \oplus \overline{\text{DCT}}_4)B_8$.

Для получения матрицы ДКП-2 умножим $\overline{\text{DCT}}_8$ на масштабирующую диагональную матрицу:

$$\text{DCT}_8 = \text{diag}_{0 \leq k < 8}(\cos k\pi/16) \cdot \overline{\text{DCT}}_8. \quad (42)$$

В итоге получен быстрый алгоритм для 8-точечного ДКП-2. Схема алгоритма, соответствующего (42), приводится на рис. 2. Алгоритм требует 29 сложений и 12 умножений (или 5 умножений, если не выполнять масштабирования).

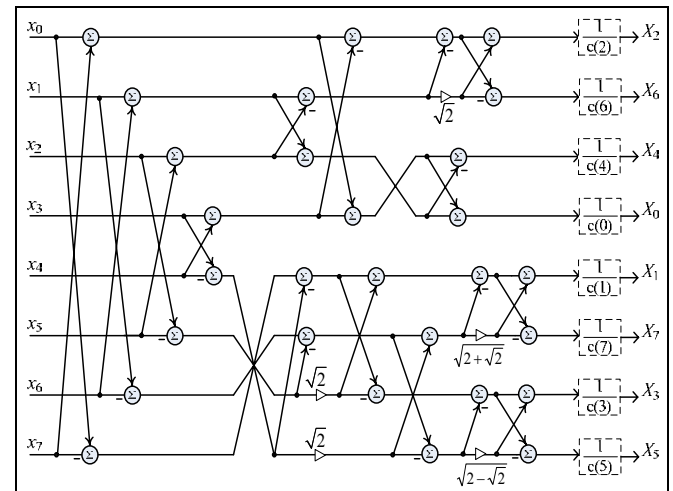


Рис. 2 – Граф-схема быстрого алгоритма 8-точечного ДКП-2, $c(k) = \cos \frac{\pi k}{16}, k = 0, \dots, 7$

Заключение

Представлен систематический подход к синтезу быстрых алгоритмов дискретных тригонометрических преобразований на основе алгебраической теории обработки сигналов. Основное внимание уделено дискретным косинусным преобразованием второго и четвертого типов. Используемый в статье подход упрощает синтез быстрых алгоритмов и даёт лучшее представление об их алгебраической структуре. Ключевым является соответствие между дискретным тригонометрическим преобразованием и полиномиальной алгеброй $\mathbb{Q}[x]/p(x)$. Быстрый алгоритм получается в результате действий над алгеброй, а не над матрицей преобразования. По сути, быстрый алгоритм сводится к нахождению пошаговой факторизации полинома $p(x)$. Для этой цели великолепно подходит теория Галуа, позволяющая определить башню полей, в которых $p(x)$ раскладывается на сомножители меньшей степени.

В качестве практического результата получен быстрый алгоритм 8-точечного ДКП-2, содержащий в ядре своей структуры только 5 операций умножения и 29 операций сложения.

Литература

1. Белый А.А., Бовбель Е.И., Микулович В.И. Алгоритмы быстрого преобразования Фурье и их свойства // Зарубежная радиоэлектроника. – 1979. – №2. – С. 3-29.
2. Вайрадян А.С., Пчелинцев И.П., Челышев М.М. Алгоритмы вычисления цифровых сверток // Зарубежная радиоэлектроника. – 1982. – № 3. – С. 3-34.
3. Нуссбаумер, Г. Быстрое преобразование Фурье и алгоритмы вычисления сверток – М.: Радио и связь, 1985. – С. 248.
4. Крот А.М. Дискретные модели динамических систем на основе полиномиальной алгебры – Мн.: Наука и техника, 1990. – С. 312.
5. Püschel M., Moura J.M.F. Algebraic Signal Processing Theory: Cooley-Tukey Type Algorithms for DCTs and DSTs // IEEE Transactions on Signal Processing. – 2008. – Vol. 56, № 4. – 1502-1521 pp.
6. Трахтман А.М., Трахтман В.А. Основы теории дискретных сигналов на конечных интервалах – М.: Советское радио, 1975. – С. 208.
7. Шиллов Г.Е. Математический анализ. Конечномерные линейные пространства – М.: Наука, 1969 – С. 421.
8. Püschel M., Moura J.M.F. Algebraic Signal Processing Theory: foundation and 1-D time // IEEE Transactions on Signal Processing. – 2008. – Vol. 56, № 8. – 3572-3585 pp.
9. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов – М.: Мир, 1979. – С. 536.
10. Миано Дж., Форматы и алгоритмы сжатия изображений в действии – М.: Триумф, 2003. – С. 336.
11. Данилов Ю.А., Многочлены Чебышева – Мн.: Вышшая школа, 1984. – С. 160.
12. Rayes M., Trevisan V., Wang P.S. Factorization properties of Chebyshev polynomials // Computers and mathematics with applications. – 2005. – vol. 50. – 1231-1240 pp.
13. Босс В., Теория Групп – М: «Либроком», 2009. – С. 216.

Application of polynomial algebras and Galois theory for synthesis of fast algorithm of discrete cosine transform

M.I. Vashkevich, A.A. Petrovsky

The systematic approach to synthesis of fast algorithm of discrete cosine transform of type 2 and 4 (DCT-2/DCT-4) based on algebraic signal processing theory (ASP) is proposed. In the ASP framework fast algorithm are not derived by manipulating with the entries of transform matrix, but by stepwise decomposition of polynomial algebra associated with given transform. This decomposition assumes stepwise factorization of polynomial. For that purpose we suggest to use intermediate fields of splitting field of polynomial that obtained by using fundamental theorem of Galois theory.

14-я Международная Конференция и Выставка ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ЕЁ ПРИМЕНЕНИЕ

28 марта – 30 марта 2012 г., Москва, Россия

The 14th International Conference

DIGITAL SIGNAL PROCESSING AND ITS APPLICATIONS

ИНФОРМАЦИОННОЕ ПИСЬМО

Уважаемый коллега!

Приглашаем Вас принять участие в работе 14-й Международной конференции «Цифровая обработка сигналов и её применение- DSPA'2012», которая состоится в ИПУ РАН 28 марта – 30 марта 2012 года

ОРГАНИЗАТОРЫ:

- Российское научно-техническое общество радиотехники, электроники и связи им. А.С. Попова
- IEEE Signal Processing Society
- Российская секция IEEE
- Институт радиотехники и электроники РАН
- Институт проблем управления РАН
- Институт проблем передачи информации РАН
- Московский научно-исследовательский телевизионный институт (ЗАО МНИТИ)
- Компания AUTEX Ltd. (ЗАО «АВТЭКС»)

ПРИ УЧАСТИИ:

- Федеральное агентство по промышленности РФ
- Министерство образования и науки РФ
- Международный союз приборостроителей и специалистов по информационным и телекоммуникационным технологиям
- ФГУП ГКРЧ
- ГСКБ «АЛМАЗ-АНТЕЙ»
- ЗАО «Инструментальные системы»
- НТЦ «МОДУЛЬ»
- ЗАО «СКАН Инжиниринг Телеком»
- ГУП НПЦ «Элвис»
- Владимирский государственный университет
- Московский авиационный институт
- Московский государственный технический университет им. Н.Э. Баумана
- Московский институт радиотехники, электроники и автоматики
- Московский технический университет связи и информатики
- Московский энергетический институт
- Рязанский государственный радиотехнический университет
- Санкт-Петербургский государственный университет телекоммуникаций
- Санкт-Петербургский государственный электротехнический университет – ЛЭТИ
- Ульяновский государственный технический университет
- Ярославский Государственный Университет
- Московский Физико-технический институт (университет)

УДК 004.932

МЕТОД ОБРАБОТКИ Р-КАДРОВ ДЛЯ ВИДЕОКОДЕКА НА ОСНОВЕ ДИСКРЕТНОГО ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЯ

Умняшкин С.В., д.ф.-м.н., профессор Национального исследовательского университета РФ «Московский государственный институт электронной техники (технический университет)», e-mail: vrintf@miie.ru
Табориский А.Е., аспирант Национального исследовательского университета РФ «Московский государственный институт электронной техники (технический университет)»

Ключевые слова: видеокодек, вейвлет-преобразование, компенсация движения, алгоритм, маска перекрытия, разностный кадр.

Введение

Одна из важных проблем разработки алгоритмов видеокомпрессии заключается в том, чтобы эффективно соединить этапы устранения *пространственной* (внутрикадровой) и *временной* (межкадровой) избыточности данных. В качестве преобразования, снижающего пространственную избыточность данных, часто выступает блочное дискретное косинусное преобразование (ДКП). Однако при использовании ДКП при высоких уровнях сжатия на декодируемом изображении начинает проявляться соответствующая блочная структура, ухудшающая субъективную оценку его качества. В отличие от ДКП, дискретное вейвлет-преобразование (ДВП) применяется ко всему изображению целиком, что позволяет избежать «разваливания» декодированного изображения на отдельные блочные фрагменты.

Снижение временной избыточности данных достигается, как правило, за счет использования некоторой процедуры предсказания кодируемого кадра на основе одного или нескольких ранее закодированных и восстановленных кадров. При этом необходимо построение прогнозного кадра, для чего чаще всего используется блочная компенсация движения в пространственной области. Далее полученная разница между кодируемым и прогнозным кадрами (т.н. разностный кадр), обрабатывается как статическое изображение с использованием для снижения внутрикадровой избыточности данных декоррелирующих преобразований (ДКП, ДВП и т.п.). Кадры, к которым применяется описанная выше процедура обработки с прогнозом, называются предсказанными кадрами, или *Р-кадрами*. Кадры, к которым применяется только декоррелирующее преобразование и которым, таким образом, при декодировании не требуется дополнительной информации о других кадрах, называются ключевыми, или *І-кадрами*.

В данной работе рассматривается видеокодек на основе ДВП [1], эффективность которого в ряде случаев превышает эффективность MPEG-4 Layer 2 [2]. Нами предлагается модификация кодера [1], основанная на изменении способа обработки Р-кадров и включающая:

- использование дробнопиксельной компенсации движения;
- использование оптимизированной маски перекрытия блоков при построении прогнозного кадра;
- учёт маски перекрытия блоков при определении наилучшего вектора перемещения в алгоритме компенсации движения;
- изменение схемы нормализации разностного кадра.

Предложен метод блочной компенсации движения для видеокодеков и рассмотрены варианты его реализации. Соответствующие алгоритмы ориентированы на совместное применение с дискретным вейвлет-преобразованием, но могут быть использованы и для кодеков на основе других преобразований.

Эффективность кодера оценивается в терминах пикового соотношения сигнал-шум (PSNR), производительность кодера оценивается по времени сжатия тестовых видеоизображений. Эксперименты были проведены для различных битовых затрат из диапазона 0,2-0,4 бита на пиксел, на примере двух известных тестовых видеопоследовательностей «Теннис» (Tennis) и «Цветочный сад» (Flower Garden).

Основные принципы работы видеокодека

Будем рассматривать вариант кодера, предназначенный для обработки полутоновых (монохроматических) видеоизображений. На рис. 1 показаны основные этапы алгоритма кодирования, реализованного в видеокодеке [1]. Кодек состоит из модуля компенсации движения, осуществляющего поиск векторов перемещений блоков кадра, модуля вычисления разностного изображения, модуля арифметического кодирования векторов перемещения и модуля сжатия статического изображения. В проводившихся нами экспериментах для сжатия статического изображения в кодеке использовался алгоритм SPIHT [3] с биортогональным ДВП CDF 9/7, принятом в стандарте JPEG 2000.

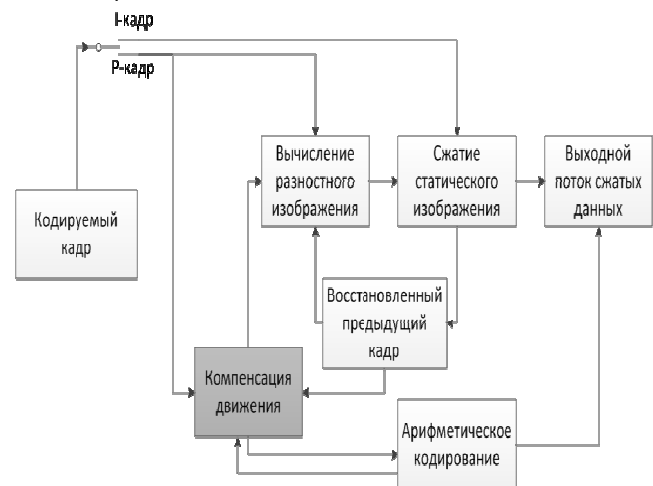


Рис. 1. Схема сжатия кодера на основе дискретного вейвлет-преобразования

І-кадр при кодировании подается непосредственно на модуль сжатия статического изображения. При обработке Р-кадра используется восстановленный предыдущий кадр, см. рис. 2. Рассмотрим метод обработки Р-кадра подробнее.

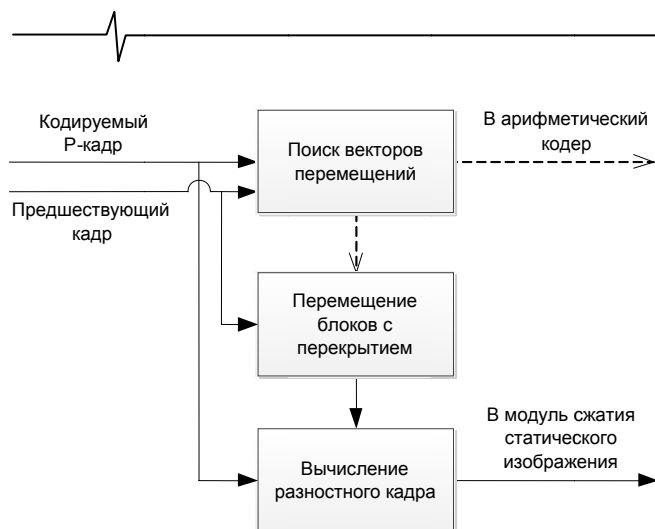


Рис. 2. Общая схема обработки Р-кадров

Сначала кодируемый Р-кадр и восстановленный предыдущий кадр вместе подаются в модуль компенсации движения. Данный модуль разбивает кодируемый кадр на блоки и для каждого из них находит вектор перемещения, соответствующий наиболее близкому приближению блока в восстановленном предыдущем кадре. Найденные векторы перемещений подаются в арифметический кодер, а затем в выходной поток. Результатом применения найденных векторов перемещений к восстановленному предыдущему кадру является прогнозный кадр (приближение кодируемого кадра), который строится из блоков предыдущего восстановленного кадра. Далее на основании кодируемого и прогнозного кадров строится разностный кадр, который подается в модуль сжатия статического изображения.

Рассмотрим базовый метод поиска векторов перемещений блоков кадра, следуя работе [1]. Каждый Р-кадр разбивается на блоки 8x8 пикселей, блоки далее распределяются по трем слоям, см. рис. 3. Слои блоков обрабатываются последовательно, от 1-го к 3-му.

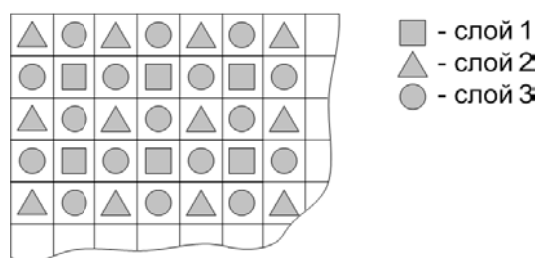


Рис. 3. Распределение блоков по слоям

Из первого слоя прореживанием выбираются 10...25% блоков, по которым при помощи медианной фильтрации координат векторов перемещений отыскивается вектор глобального перемещения всего кадра. Далее, для каждого блока изображения, последовательно по слоям, проводится поиск векторов перемещения по схеме, определяемой номером слоя. При поиске вектора перемещения очередного блока учитывается предсказание этого вектора, которое используется в качестве центра области поиска. Для блоков слоя 1 предсказанием является глобальное перемещение кадра. Для блоков слоя 2 предсказанием является результат медианной фильтрации векторов перемещений четырех соседних диагональных блоков слоя 1. Для блоков слоя 3 при построении прогноза для медианной фильтрации берутся найденные ранее векторы перемещений двух блоков слоя 1, двух блоков слоя 2 и двух блоков слоя 3,

которые окружают текущий обрабатываемый блок третьего слоя. В зависимости от номера слоя используются различные схемы поиска: чем больше номер слоя, тем более точными являются построенные прогнозы перемещений, и тем меньше выбирается область для поиска вектора перемещения обрабатываемого блока.

Исследуем возможность повышения характеристик описанного метода за счёт дополнительных модификаций.

Дробнопиксельная компенсация движения

Эффективность рассматриваемого метода компенсации движения можно повысить, если учесть тот факт, что блок за время перехода к следующему кадру может сместиться не на целое количество пикселей, а на дробное [4]. В настоящее время в системах видеосжатия использование дробнопиксельной компенсации является известным методом повышения эффективности видеосжатия.

Нами был реализован алгоритм дробнопиксельной компенсации, в котором после применения приведенного выше метода поиска векторов перемещений производится их уточнение до полупикселей, а затем и до четвертьпикселей. Для подобного уточнения восстановленное изображение предыдущего кадра увеличивается с использованием интерполяции, принятой в стандарте H.264 [5].

Уточнение вектора перемещений до полупикселей производится только для смещений, не превышающих h пикселей. Зона полупиксельного уточнения имеет размер $2h$ на $2h$. Аналогично, уточнение до четвертьпикселей производится только для смещений, не превышающих q пикселей, и зона четвертьпиксельного уточнения имеет размер $2q$ на $2q$. Таким образом, в разработанном алгоритме варьируются два параметра: h и q .

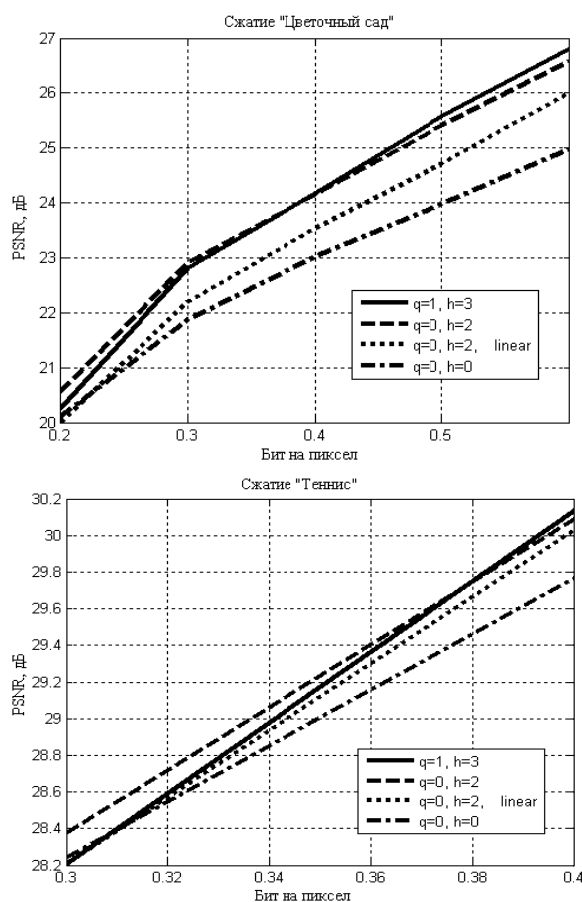


Рис. 4. Зависимость PSNR от битовых затрат при различных методах компенсации движения для последовательностей «Цветочный сад» и «Теннис»

Эффективность дробнопиксельной компенсации движения была проанализирована в виде зависимости от размера зоны перемещений, для которой производится дальнейшее дробнопиксельное уточнение. На рис. 4 приведены зависимости PSNR от битовых затрат, полученные для тестовых последовательностей для четырёх алгоритмов: четвертьпиксельная компенсация ($q=1, h=3$), полупиксельная компенсация ($q=0, h=2$), полупиксельная компенсация с использованием менее ресурсоемкой линейной (linear) интерполяции ($q=0, h=2$), отсутствие дробнопиксельной компенсации ($q=0, h=0$).

В табл. 1 приведены данные для оценки производительности приведённых методов. Измерялось суммарное время, затраченное на поиск векторов перемещений.

Таблица 1.

Оценка производительности методов

Алгоритм	Среднее время обработки последовательности, с	Дополнительные вычисл. затраты, %
Отсутствие дробнопиксельной компенсации ($q=0, h=0$);	2,28	0
Полупиксельная компенсация ($q=0, h=2$, линейная интерполяция)	2,71	19
Полупиксельная компенсация ($q=0, h=2$)	2,89	27
Четвертьпиксельная компенсация ($q=1, h=3$)	4,33	90

Приведённые данные показывают, что для битовых затрат 0,3 бита на пиксел оптимальным является применение только полупиксельной компенсации при перемещении блока изображения относительно предсказанного вектора перемещения на расстояние, не превышающее 2 пиксела, т.е. $q=0, h=2$. Для битовых затрат 0,4 бита на пиксел оптимальной будет большая зона дробнопиксельной компенсации: $q=1, h=3$. Отметим, что в случае ограниченных вычислительных ресурсов, даже на более высоких битрейтах, следует отказаться от использования ресурсоемкой четвертьпиксельной компенсации ($q>0$) в пользу полупиксельной ($q=0, h=2$). Кроме того, в случае ограниченных вычислительных ресурсов вместо относительно сложной интерполяции, принятой в стандарте H.264, можно использовать более простую линейную интерполяцию, что позволяет существенно сократить время вычислений ценой некоторого уменьшения эффективности сжатия.

Оптимизация маски перекрытия

В кодеке [1] для блоков изображения используется весовая маска с перекрытием размера 16×16 пикселей, взятая из [6]. То есть прогнозный кадр строится из перекрывающихся блоков 16×16 пикселей, взятых из предыдущего декодированного кадра и покомпонентно умноженных на рис. 5. Однако для практических целей использование столь большой маски перекрытия может оказаться нецелесообразным в силу высоких вычислительных затрат. Так, при использовании маски 12×12 вместо 16×16 количество умножений пикселей на

весовые коэффициенты уменьшается в $256/144 \approx 1,78$ раза, что существенно увеличивает быстродействие кодека. Перекрытие с использованием маски 12×12 показано на рис. 6.

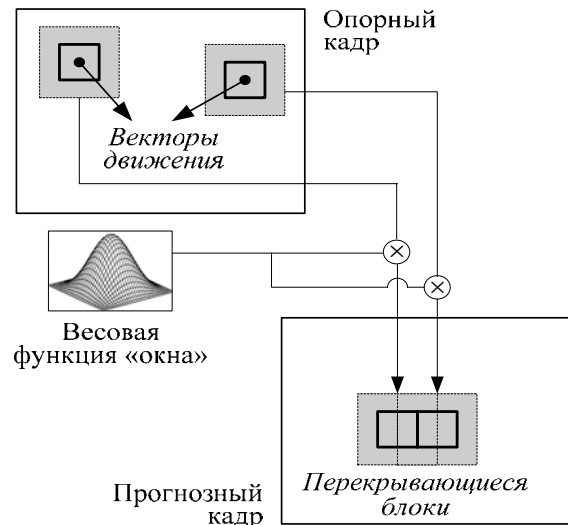


Рис. 5. Блочная компенсация движения с перекрытием с маской 16×16 пикселей

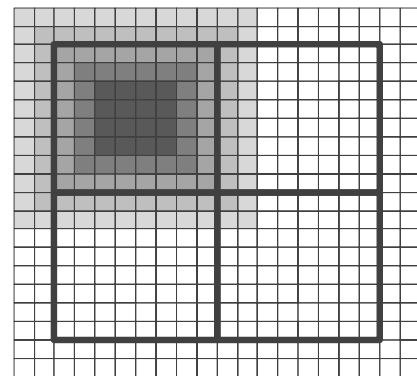


Рис. 6. Перекрытие соседних блоков при использовании маски 12×12 пикселей

Весовая маска должна удовлетворять следующим условиям. Во-первых, для каждого пиксела сумма соответствующих коэффициентов всех перекрывающихся в данном пикселе масок должна быть равна 1. Во-вторых, матрица должна быть симметрична относительно обеих осей. Такую маску 12×12 можно задать двумя параметрами – числами A, B из диапазона $[0; 1]$ – соответствующая матрица весов W будет иметь следующий общий вид:

$$W(A,B) = \begin{pmatrix} A \cdot A & A \cdot B & A \cdot B & A \cdot A & A & A & A & A & A & A \cdot B & A \cdot B & A \cdot A \\ B \cdot A & B \cdot B & B \cdot B & B \cdot A & B & B & B & B & B & B \cdot A & B \cdot B & B \cdot A \\ B \cdot A & B \cdot B & B \cdot B & B \cdot A & B & B & B & B & B & B \cdot A & B \cdot B & B \cdot A \\ A \cdot A & A \cdot B & A \cdot B & A \cdot A & A & A & A & A & A & A \cdot B & A \cdot B & A \cdot A \\ A & B & B & A & 1 & 1 & 1 & 1 & A & B & B & A \\ A & B & B & A & 1 & 1 & 1 & 1 & A & B & B & A \\ A & B & B & A & 1 & 1 & 1 & 1 & A & B & B & A \\ A & B & B & A & 1 & 1 & 1 & 1 & A & B & B & A \\ A \cdot A & A \cdot B & A \cdot B & A \cdot A & A & A & A & A & A & A \cdot B & A \cdot B & A \cdot A \\ B \cdot A & B \cdot B & B \cdot B & B \cdot A & B & B & B & B & B & B \cdot A & B \cdot B & B \cdot A \\ B \cdot A & B \cdot B & B \cdot B & B \cdot A & B & B & B & B & B & B \cdot A & B \cdot B & B \cdot A \\ A \cdot A & A \cdot B & A \cdot B & A \cdot A & A & A & A & A & A & A \cdot B & A \cdot B & A \cdot A \end{pmatrix}$$

где $A' = (1 - A)$, $B' = (1 - B)$.

Оптимальные значения коэффициентов A и B , дающие максимальную эффективность кодирования в терминах PSNR–битрейт, были определены экспериментально. Всего было проведено около 1400 различных измерений для двух тестовых последовательностей «Теннис» и «Цветочный сад», для ряда битрейтов [0,2, 0,25, 0,3, 0,35, 0,4]. В результате было получено, что оптимальными параметрами являются: $A=0,8$, $B=0,6$, что соответствует линейному спаду весов на границе маски.

На рис. 7 показана зависимость PSNR от используемой весовой маски блока. Замена оригинальной маски 16x16 на меньшую оптимальную маску 12x12 не привела к существенному падению PSNR и даже, напротив, дала некоторый выигрыш на малых битрейтах. За счёт того, что в маске 12x12 16 весовых коэффициентов являются единичными, умножение блока на такую маску требует 144-16=128 умножений чисел – это в 2 раза меньше, чем при использовании маски 16x16. Поэтому предлагаемую модификацию весовой маски следует признать эффективной.

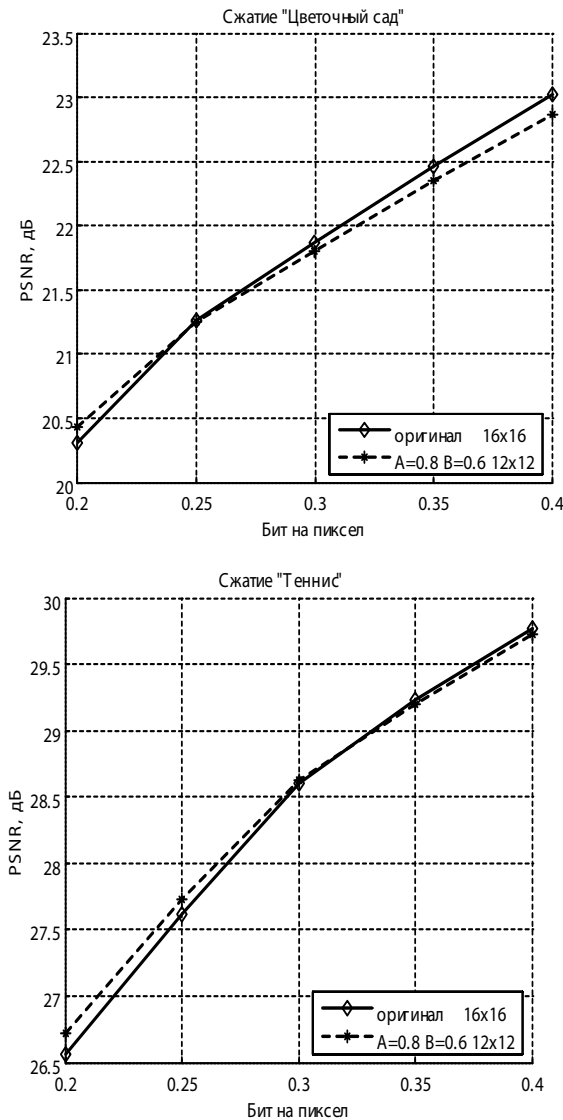


Рис. 7. Зависимость PSNR от используемой маски перемещения для последовательностей «Цветочный сад» и «Теннис»

Учёт маски перекрытия при поиске наилучшего вектора

Для поиска наилучшего вектора перемещений блока в [1] используется RD-критерий. Мера отклонения прогнозного блока $B' = B'(r)$ из предыдущего декодированного кадра от блока B текущего кодируемого кадра определяется как

$$J(r) = D(r) + \lambda R(r), \quad (1)$$

где $D = \sum_{y=1}^N \sum_{x=1}^N (p_{x,y} - p'_{x,y})^2$ – суммарная попиксельная

квадратичная разность блоков, $R = R(r)$ – количество бит, затрачиваемых на кодирование вектора перемещения r , λ – варьируемый параметр, $p_{x,y}$ – значение яркости

пиксела (x, y) блока B кодируемого кадра, $p'_{x,y}$ – значение яркости пиксела (x, y) блока B' предыдущего восстановленного кадра, N – размер блока (в наших экспериментах $N=8$). Наилучшим вектором признается тот, у которого величина J (1) минимальна.

Такой способ оценки векторов перемещений (1) позволяет учесть как отклонение блока восстановленного кадра от рассматриваемого блока, так и эффективность сжатия векторов перемещений при дальнейшем арифметическом кодировании. Однако, в (1) не учитывается тот факт, что при дальнейшем перемещении пикселей выбранного блока, необходимо для построения прогнозного кадра, имеет место перекрытие соседних блоков. То есть суммарная попиксельная квадратичная разность D неточно характеризует ошибку, реально вносимую при построении прогнозного кадра.

Предлагается заменить разность D в (1) на взвешенную разность

$$D' = \sum_{y=1}^M \sum_{x=1}^M ((p_{x,y} - p'_{x,y}) \cdot \mu_{x,y})^2, \quad (2)$$

где $\mu_{x,y}$ – весовой коэффициент маски перекрытия W для пиксела (x, y) , M – размер маски перекрытия в пикселах (в наших экспериментах $M=12$). Таким образом, модифицированная мера отклонения прогнозного блока может быть представлена как

$$J' = D' + \lambda R. \quad (3)$$

Модифицированный критерий J' , в отличие от J (1), учитывает влияние перекрытия соседних блоков (рис. 6) при дальнейшем вычислении прогнозного кадра.

Так как размер маски перекрытия M может быть больше размера блока N , и оценка вектора перемещения проводится чрезвычайно часто, применение модифицированного критерия J' может заметно замедлять работу алгоритма поиска перемещенного блока. Поэтому для выполнения видеокомпрессии на системах, чувствительных к быстродействию, также может иметь смысл использование и другого модифицированного (ускоренного) критерия

$$J'' = D'' + \lambda R,$$

$$D'' = \sum_{y=1}^N \sum_{x=1}^N ((p_{x,y} - p'_{x,y}) \cdot \mu_{x,y})^2. \quad (4)$$

Использование ускоренного критерия J'' , как показали результаты экспериментов, не позволяет достичь столь же высоких результатов, как использование J' (3), однако замедляет работу кодера незначительно.

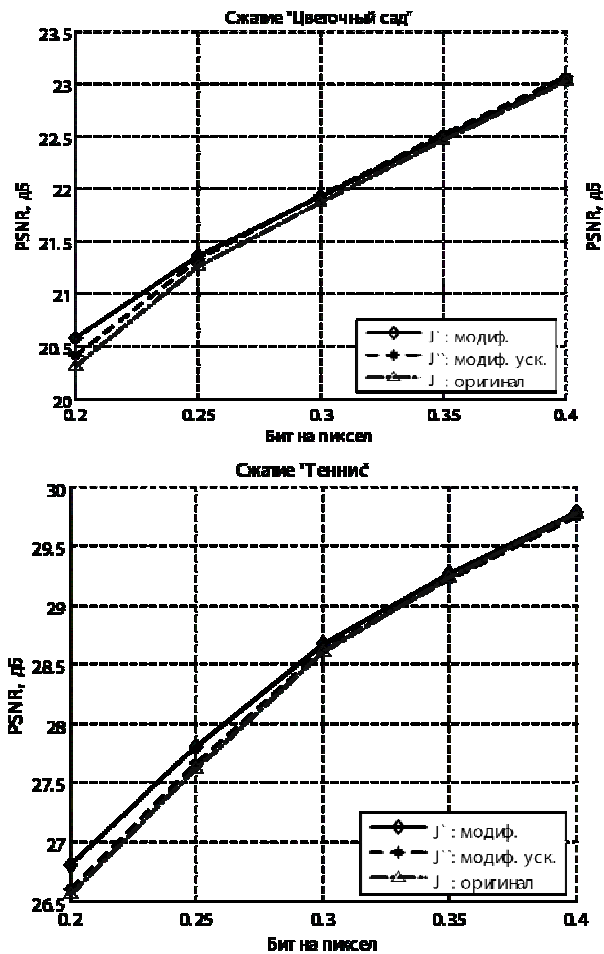


Рис. 8. Зависимость PSNR от используемого критерия оценки вектора перемещения

На рис. 8 приведены характеристики работы кодера при использовании различных критериев выбора наилучших векторов перемещений. Заметим, что по сравнению с использованием критерия (1) предложенные методы поиска по критериям (3) и (4) повышают эффективность кодирования во всем рассмотренном диапазоне битовых затрат. Ускоренный модифицированный критерий (4) целесообразно применять в системах, чувствительных к быстродействию, а критерий (3) – во всех остальных случаях.

Нормализация разностного кадра

При кодировании пиксела изображения целым числом из диапазона $[0...255]$ (одним байтом) по-пиксельная разность прогнозного и кодируемого кадра в общем случае будет лежать в промежутке $[-255...255]$. Однако для обеспечения высокой производительности кодера и эффективного использования памяти вычислительных устройств для разностного изображения также удобнее использовать формат данных, использующий 1 байт на пиксел. Поэтому используемый нами модуль сжатия статического изображения был реализован для обработки кадров, представленных в таком формате. По этой причине перед кодированием разностного изображения попиксельная разность должна быть прежде отображена в диапазон $[0...255]$, т.е. нормализована.

Простейший способ такой нормализации заключается в выполнении операции целочисленного деления на 2 и последующего прибавления числа 128.

Заметим, что операция нормализации, т.е. отображения $[-255...255] \rightarrow [0...255]$, может быть задана табличной функцией. Мы исследовали некоторые варианты такого отображения: на рис. 9а показаны оригинальная (соответствующая делению значений пикселей на 2), линейная и сглаженная нормализации. Сглаженная функция построена с учетом того факта, что пиксели в разностном кадре редко принимают большие по модулю значения.

На рис. 9б приведены соответствующие различным вариантам нормализации графики зависимостей ошибки PSNR от битовых затрат для видеопоследовательности «Цветочный сад». Лучшие результаты в среднем достигаются при выборе сглаженной таблицы перехода, что особенно заметно на низких битрейтах – аналогичные выводы дает и анализ результатов обработки последовательности «Теннис». Таким образом, в отсутствие дополнительных вычислительных затрат использование таблицы сглаженной нормализации позволяет повысить характеристики сжатия.

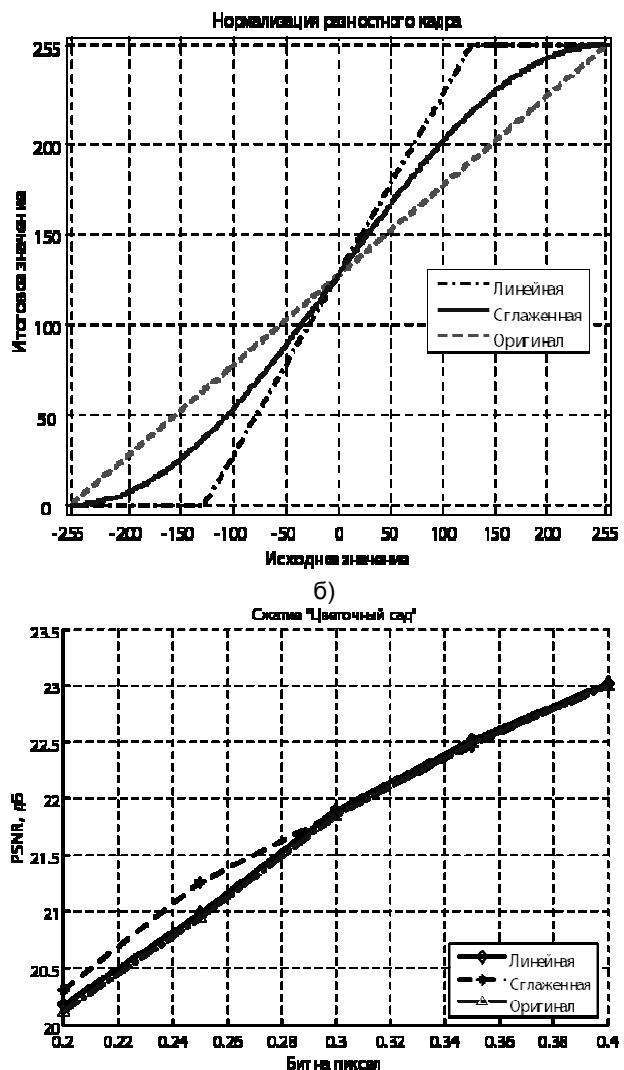


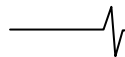
Рис. 9. Схемы нормализации:

а) – функции нормализации;

б) – зависимость PSNR от используемой схемы

Заключение

Все четыре предложенные способа повышения характеристик кодера могут быть использованы совместно. На рис. 10 приводится сравнение характеристик работы оригинальной [1] и итоговой версии кодера, использующей все описанные модификации. Получено



значительное увеличение итоговой величины PSNR (в диапазоне от 0,5 до 1,1 дБ) для всех представляющих интерес значений битовых затрат.

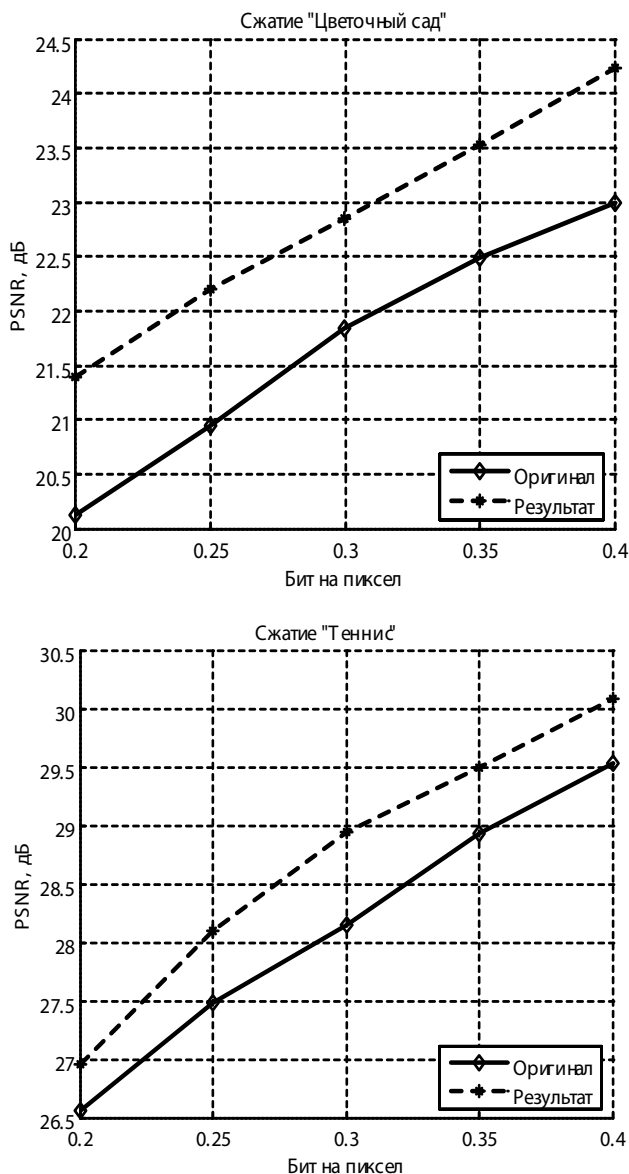


Рис. 10. Итоговые характеристики кодека

Из предложенных модификаций вычислительно сложным является использование дробнопиксельной компенсации, поэтому в системах видеосжатия реального времени, где быстродействие играет ключевую роль, использование данной опции кодека может оказаться нецелесообразным, либо следует использовать её ускоренный вариант. Кроме того, в подобных системах при поиске вектора перемещения также следует использовать ускоренный вариант критерия поиска (4), который практически не приводит к потере производительности.

Тем не менее, даже при использовании опций, дающих наименьшую производительность кодека (использование меры (2) для вычисления ошибки при поиске перемещения блока и включенная четвертьпиксельная компенсация), на компьютере с процессором Intel Core 2 Duo T8300 2,40 ГГц скорость кодирования полутоновой видеопоследовательности Flower

Garden с размером кадра 352×224 пикселя составила 22 кадра в секунду. При использовании опций кодека, обеспечивающих его максимальную производительность, скорость сжатия данной видеопоследовательности на компьютере с указанной конфигурацией возросла до 36 кадров в секунду. Таким образом, разработанная нами программная реализация описанного кодека обеспечивает обработку видео в реальном масштабе времени на вычислительных системах (персональных компьютерах) общего назначения.

Предложенный метод обработки Р-кадров и его составные части могут быть применены не только к видеокодеку [1], и не только к кодекам на основе вейвлет-преобразований, но и для построения других алгоритмов видеокомпрессии, которые используют блочную компенсацию движения.

Литература

1. Александров А.А., Коплович Е.А., Умняшкин С.В. Алгоритм видеокомпрессии на основе дискретного вейвлет-преобразования с трехслойной схемой кодирования векторов движения // Известия вузов. Электроника – №5. – 2008. – С. 69-73.
2. Umnyashkin S., Aleksandrov A., Koplovich E. Wavelet based video codec for telemedicine purposes // Proc. of the 5th Russian-Bavarian Conference on Biomedical Engineering (July 1-4, 2009, Munich, Germany). – 177-180 pp.
3. A. Said, W.A. Pearlman. A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees // IEEE Transactions on Circuits and Systems for Video Technology, vol. 6, pp. 243-250, June 1996.
4. Табориский А.Е. Анализ эффективности дробнопиксельной компенсации перемещения для видеокодека на основе дискретного вейвлет-преобразования // Труды Российского научно-технического общества радиотехники, электроники и связи имени А.С. Попова. – Серия: Цифровая обработка сигналов и её применение. Выпуск XII-2. – С. 148-151.
5. Advanced video coding for generic audiovisual services. Recommendation ITU-T H.264, 03/2009.
6. Orchard M.T., Sullivan G.J. Overlapped block motion compensation: an estimation-theoretic approach // IEEE Transactions on Image Processing, Volume 3 Issue 5, Sep. 1994.

A method for P-frames wavelet-based video codecs

S. Umnyashkin, A. Taborisky

A method of block motion compensation for digital video compression is proposed, some of algorithmic implementations are described. The developed algorithms aim to wavelet based codecs applications. These algorithms can be applied to other transforms as well.

УДК 004.932

АЛГОРИТМ ОЦЕНКИ КООРДИНАТ ОБЪЕКТОВ НА ОСНОВЕ ПРЕОБРАЗОВАНИЯ РАДОНА

Алпатов Б.А., д.т.н., профессор, заведующий кафедрой автоматики и информационных технологий в управлении (АИТУ) Рязанского государственного радиотехнического университета, e-mail: aitu@rsreu.ru
Бабаян П.В., к.т.н., начальник научной лаборатории кафедры АИТУ, e-mail: aitu@rsreu.ru
Шубин Н.Ю., аспирант РГРТУ, e-mail: aitu@rsreu.ru

Ключевые слова: оценка координат, преобразование Радона, модифицированное преобразование Радона, обнаружение прямых.

Введение

Оценка координат объектов – одна из задач компьютерного зрения, которая решается в настоящее время с использованием большого количества подходов [1]. Одним из перспективных направлений в этой области является использование алгоритмов на основе структурного анализа изображений. В соответствии с ними изображение объекта рассматривается как комбинация геометрических примитивов, таких как точки, прямые, дуги эллипсов и т.д. Отслеживание их перемещений за кадр позволяет оценить изменения координат самого объекта.

Для обнаружения прямолинейных границ объектов часто используют преобразование Радона. В данной работе под прямолинейной границей объекта на изображении будет пониматься область сильного перепада яркости изображения, состоящая из пикселей, лежащих вдоль одной прямой. Однако в случае большого количества прямолинейных границ либо значительного уровня шума вероятность ложного обнаружения сильно возрастает. Известны различные пути увеличения устойчивости преобразования Радона к шумам на исходном изображении, такие как подбор подходящего фильтра выделения границ на исходном изображении, фильтрация результата преобразования Радона [2], уменьшение области интегрирования добавлением третьего измерения в параметрическое пространство [3].

Цель работы – разработка алгоритма оценки координат объектов на основе выделения прямолинейных границ с помощью преобразования Радона и модификация преобразования Радона для более эффективного его использования в задаче обнаружения прямых.

Преобразование Радона для непрерывной двумерной функции $f(x, y)$ определяется следующим выражением:

$$R(s, \alpha) = \int_{-\infty}^{+\infty} f(x, y) \Big|_{\substack{x=s \cos \alpha - z \sin \alpha \\ y=s \sin \alpha + z \cos \alpha}} dz, \quad (1)$$

Рассмотрена задача оценки координат объекта с прямолинейными границами с помощью преобразования Радона. Предложено модифицированное преобразование Радона, позволяющее более качественно выделять прямые в условиях значительного аддитивного некоррелированного шума на исходном изображении. Разработан алгоритм оценки координат объектов на основе обычного и модифицированного преобразования Радона. Представлены результаты экспериментальных исследований разработанного алгоритма.

где $R(s, \alpha)$ – результат преобразования; (s, α) – параметры прямой, вдоль которой проводится интегрирование [4]. Наглядно геометрический смысл параметров s и α продемонстрирован на рис. 1.

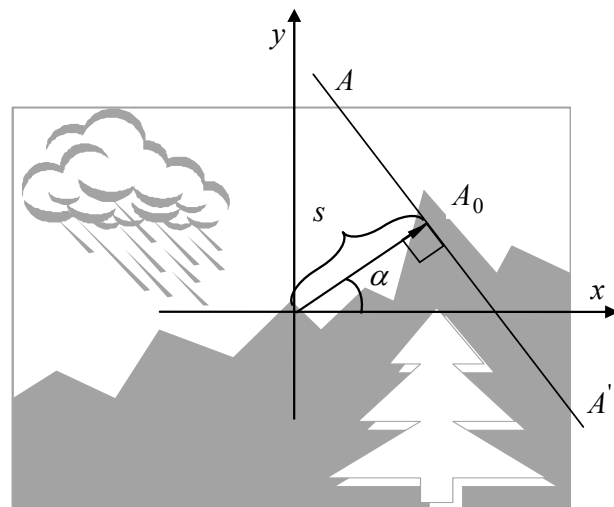
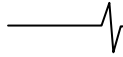


Рис. 1. Геометрический смысл преобразования Радона

На рис. 1 схематично изображено преобразуемое полутоновое изображение $f(x, y)$. Начало координат совпадает с центром изображения. Прямая AA' задаётся полярными координатами (s, α) ближайшей к началу координат точки A_0 прямой AA' . Величина $R(s, \alpha)$ вычисляется путём интегрирования $f(x, y)$ вдоль прямой AA' .

Модифицированное преобразование Радона

В большинстве известных алгоритмов, описанных в литературе, для обнаружения прямолинейных границ объектов используется преобразование Радона [5] от скалярного двумерного поля, являющегося результатом обработки исходного изображения каким-либо фильтром выделения границ. При этом информация о направлении яркостного перепада теряется, что неизбежно сказывается на качестве



обнаружения. Очевидно, что направление вектора градиента в области границы двух объектов различной яркости будет близко к направлению перпендикуляра к самой границе. Таким образом, учёт направления градиента может позволить улучшить качественные характеристики обнаружения прямолинейных границ объектов. Предлагается видоизменить выражение (1) путём замены подынтегрального выражения квадратом проекции вектора градиента исходного изображения на вектор нормали к прямой интегрирования. Результатом является выражение:

$$R'(s, \alpha) = \int_{-\infty}^{+\infty} \langle \nabla f(x, y), \vec{n} \rangle^2 \Big|_{\substack{x=s \cdot \cos \alpha - z \cdot \sin \alpha \\ y=s \cdot \sin \alpha - z \cdot \cos \alpha}} \cdot dz, \quad (2)$$

где $\vec{n}$ – нормаль к прямой AA' ($\vec{n} = (\cos \alpha, \sin \alpha)$). В данном случае в качестве $f(x, y)$ выступает изображение оригинальное, а не обработанное алгоритмом обнаружения границ. Раскрыв скобки, можно привести выражение (2) к следующему виду:

$$\begin{aligned} R'(s, \alpha) = & \cos^2 \alpha \cdot \int_{-\infty}^{+\infty} \left(\frac{\partial f(x, y)}{\partial x} \right)^2 \Big|_{\substack{x=s \cdot \cos \alpha - z \cdot \sin \alpha \\ y=s \cdot \sin \alpha - z \cdot \cos \alpha}} \cdot dz + \\ & + \sin 2\alpha \cdot \int_{-\infty}^{+\infty} \frac{\partial f(x, y)}{\partial x} \cdot \frac{\partial f(x, y)}{\partial y} \Big|_{\substack{x=s \cdot \cos \alpha - z \cdot \sin \alpha \\ y=s \cdot \sin \alpha - z \cdot \cos \alpha}} \cdot dz + \\ & + \sin^2 \alpha \cdot \int_{-\infty}^{+\infty} \left(\frac{\partial f(x, y)}{\partial y} \right)^2 \Big|_{\substack{x=s \cdot \cos \alpha - z \cdot \sin \alpha \\ y=s \cdot \sin \alpha - z \cdot \cos \alpha}} \cdot dz \end{aligned} \quad (3)$$

Выражение (3) представляет собой взвешенную сумму трёх обычных преобразований Радона от квадратов производных исходного изображения по горизонтали и вертикали, а также от произведения этих производных.

Алгоритм измерения координат объекта на основе преобразования Радона

В рамках данной работы разработан алгоритм, использующий результат обычного либо модифицированного преобразования Радона для оценки координат объекта на видеоизображении. Положение объекта представлено в виде описанного вокруг объекта прямоугольника и на первом кадре задаётся вручную. На всех последующих кадрах алгоритм оценивает координаты объекта, основываясь на оценках перемещений прямолинейных границ внутри области интереса кадра. Область интереса имеет форму прямоугольника с центром в точке, где был обнаружен объект на предыдущем кадре. Алгоритм обработки очередного кадра состоит из следующих этапов:

- 1) обычное преобразование Радона от модуля градиента области интереса исходного изображения либо модифицированное преобразование Радона от области интереса исходного изображения;
- 2) поиск локальных максимумов результата обычного либо модифицированного преобразования Радона;
- 3) поиск взаимного соответствия полученных максимумов на текущем и предыдущем кадрах;
- 4) оценка сдвига изображения в области интереса;
- 5) сдвиг области интереса и оцениваемого положения объекта.

Рассмотрим некоторые этапы алгоритма более подробно.

Поиск локальных максимумов преобразования Радона выполняется с учётом особенностей преобразования Радона. Предполагается, что количество

искомых границ на изображении известно заранее. Составляется предварительный список всех локальных максимумов преобразования Радона в порядке убывания их значений. В окончательный список локальных максимумов поочередно добавляются максимумы из предварительного списка, начиная с самого первого элемента. При этом для снижения вероятности обнаружения двух близких максимумов, соответствующих одной прямой, для каждого нового максимума, претендующего на добавление в окончательный список, проверяется условие (4):

$$\left(\left(|s_i - s_j| \geq \Delta s_{\min} \right) \vee \left(|\alpha_i - \alpha_j| \geq \Delta \alpha_{\min} \right) \right) \wedge \left(\left(|s_i + s_j| \geq \Delta s_{\min} \right) \vee \left(|\alpha_i - \alpha_j| \leq 180^\circ - \Delta \alpha_{\min} \right) \right), \quad (4)$$

где: (s_i, α_i) – координаты проверяемого максимума; (s_j, α_j) – координаты максимума из окончательного списка; $\Delta s_{\min}$ и $\Delta \alpha_{\min}$ – минимально допустимые расстояния между максимумами по осям s и α соответственно;

$$\alpha_i, \alpha_j \in [-90^\circ, 90^\circ], s_i, s_j \in \left[-\frac{d}{2}, \frac{d}{2} \right],$$

d – длина диагонали исходного изображения в пикселях. Очередной максимум (s_i, α_i) проверяется на близость по выражению (4) с каждым максимумом (s_j, α_j) , уже принадлежащим окончательному списку.

Если выражение (4) выполняется для каждого (s_j, α_j) ,

то (s_i, α_i) добавляется в окончательный список максимумов. Алгоритм выполняется до тех пор, пока количество максимумов в окончательном списке не достигнет заданного числа, либо предварительный список не подойдёт к концу.

Вторая пара неравенств (4) вытекает из следующего свойства преобразования Радона:

$$R(s, \alpha + 180^\circ) = R(-s, \alpha) \quad (5)$$

Из (5) следует, что точки $(s, -90^\circ)$ и $(-s, 90^\circ)$, находящиеся в двух противоположных границах области определения функции $R(s, \alpha)$, соответствуют одной прямой на исходном изображении.

Поиск взаимного соответствия максимумов на соседних кадрах производится в два этапа. Сначала составляется матрица, номера столбцов которой соответствуют максимумам, найденным в текущем кадре, а номера строк – максимумам в предыдущем кадре. Каждый элемент матрицы численно выражает степень отличия координат соответствующих максимумов, вычисляемую по следующей формуле:

$dif = k_\alpha \cdot \Delta \alpha + k_s \cdot \Delta s$, где: $\Delta \alpha$ и Δs – расстояния между максимумами по осям α и s , k_α и k_s – весовые коэффициенты. При этом, исходя из (5), берутся два различных значения $\Delta \alpha$ и Δs , а значение dif выбирается минимальным из этих двух вариантов.

Далее ведётся поиск минимального элемента заполненной матрицы. Соответствующие этому элементу максимумы преобразования Радона считаются принадлежащими одной прямой в разных кадрах. Из матрицы удаляется

строка и столбец, на пересечении которых лежал найденный элемент, а затем ищется следующий минимальный элемент. Операция повторяется до тех пор, пока в матрице есть элементы, а найденные минимумы не превышают некоторого порогового значения.

Оценка сдвига объекта в области интереса выполняется на основе полученных пар максимумов на соседних кадрах в предположении, что каждая найденная пара максимумов соответствует своей прямой, перемещающейся в зоне интереса от кадра к кадру. Если прямая с параметрами (s, α) будет параллельно перенесена на вектор $(\Delta x, \Delta y)$, то её новыми параметрами будут:

$$(s', \alpha') = (s - \Delta x \cdot \cos \alpha + \Delta y \cdot \sin \alpha, \alpha) \quad (6)$$

Для оценки $(\Delta x, \Delta y)$ составляется система уравнений вида:

$$\begin{pmatrix} -\cos \alpha_1 & \sin \alpha_1 \\ -\cos \alpha_2 & \sin \alpha_2 \\ \vdots & \vdots \\ -\cos \alpha_m & \sin \alpha_m \end{pmatrix} \cdot \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} s_1' - s_1 \\ s_2' - s_2 \\ \vdots \\ s_m' - s_m \end{pmatrix} \quad (7)$$

где: (s_i', α_i') и (s_i, α_i) – координаты i -ой прямой в текущем и предыдущем кадрах соответственно, m – количество прямых. Система (7) в общем случае является несовместной, и ищется её псевдорешение.

На последнем этапе работы алгоритма вычисляются абсолютные координаты объекта на изображении путём суммирования координат объекта на предыдущем кадре с оценками сдвига объекта, полученными из (7).

Для компенсации смещения объекта относительно области интереса все найденные на текущем кадре прямые смещаются на $(-\Delta x, -\Delta y)$ по выражению (6). Полученные координаты прямых используются для обработки следующего кадра на этапе поиска соответствия максимумов.

Экспериментальные исследования

Для сравнения эффективности обычного и модифицированного преобразований Радона в задаче оценки координат объекта были проведены экспериментальные исследования. В качестве тестовой видеопоследовательности выбран видеосюжет, снятый с вертолѐта,двигающегося над городом. Ряд наземных объектов остаются в кадре на протяжении всего сюжета. Один из таких объектов с контрастом 0,5 (при изменении яркостей в диапазоне $[0,1]$) выбран в качестве объекта интереса. Для оценки качества работы алгоритмов измерения координат объектов используется критерий, подробно описанный в [6]. Он сравнивает рамку вокруг объекта, оцененную алгоритмом (рис. 2 а, б), с эталонной заданной вручную рамкой объекта в каждом кадре (рис. 2 в). Показатель качества определяется выражением:

$$K = \frac{2S_{Po \cap Pa}}{S_{Po} + S_{Pa}}, \quad (8)$$

где K – показатель качества оценки координат объекта в текущем кадре ($K \in [0,1]$); S_{Po} , S_{Pa} и $S_{Po \cap Pa}$ – площади эталонной рамки объекта, рамки объекта, оцененной алгоритмом, и их пересечения соответственно. В общем случае, как эталонная рамка, так и рамка, оцененная алгоритмом, способны менять не только положение, но и размер. Однако разработанный алгоритм учитывает лишь преобразования сдвига и не меняет размер рамки в процессе работы. При проведении экспериментальных исследований значения критерия (8) суммировались, начиная с первого кадра и до первого «срыва» алгоритма,

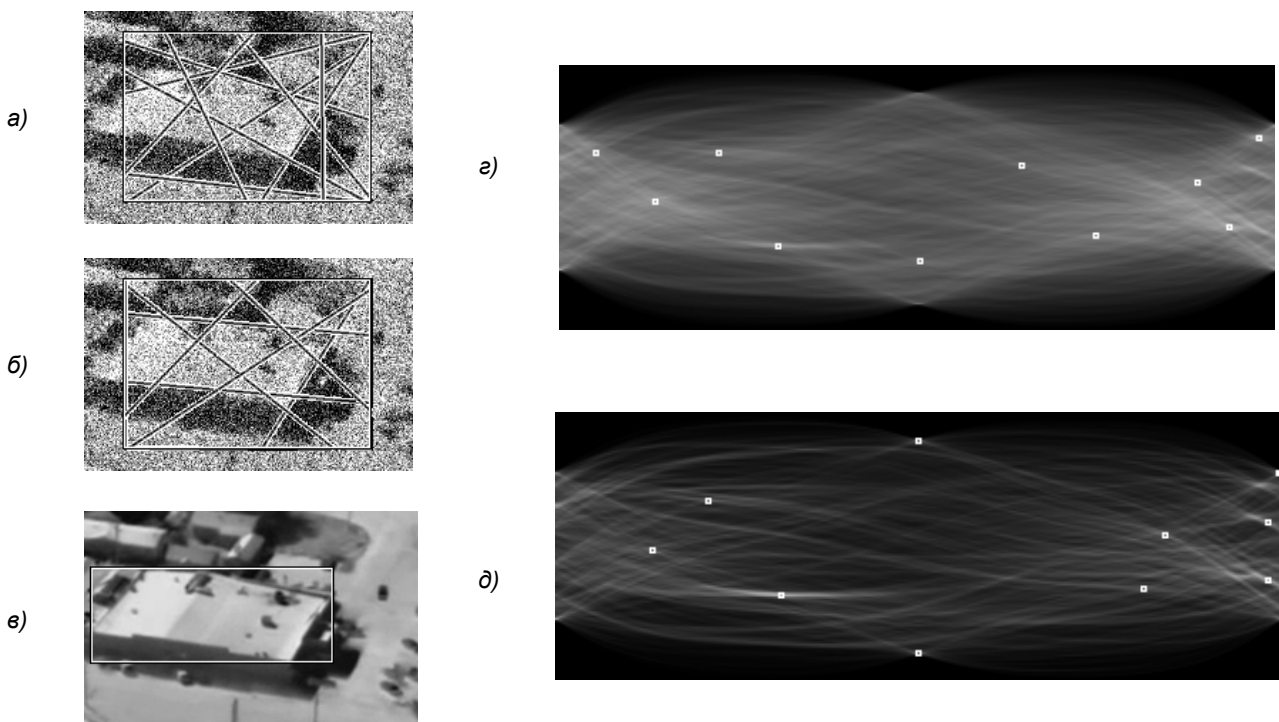
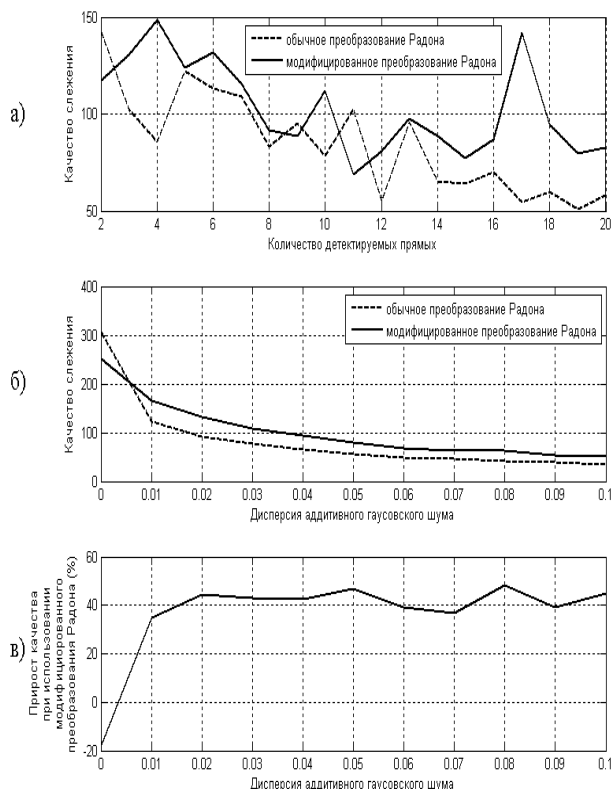


Рис. 2. Промежуточные результаты работы алгоритмов оценки координат объектов на примере видеопоследовательности, используемой в эксперименте (дисперсия аддитивного шума составляет 0,06; количество детектируемых прямых – 10); (а), (б) – прямолинейные границы объектов, обнаруженные с помощью алгоритмов на основе обычного и модифицированного преобразований Радона соответственно; (в) – исходное изображение без зашумления; (г), (д) – найденные максимумы результатов обычного и модифицированного преобразований Радона соответственно

т.е. до первого кадра, в котором $K = 0$.

Эксперименты проводились при различных значениях дисперсии аддитивного шума в диапазоне $[0; 0,1]$ с шагом 0,01. Количество прямых, детектируемых с помощью преобразования Радона, менялось в диапазоне $[2; 20]$. При каждом конкретном значении этих параметров моделирование проводилось более 10 раз для различных реализаций шума, а качество работы алгоритмов по результатам моделирования усреднялось по всем реализациям. Результаты экспериментов приведены на рис. 3.

Рис. 3. Основные результаты экспериментальных исследований; (а) – усреднённая для разных уровней



шума зависимость качества работы алгоритмов от количества детектируемых прямых; (б) – усреднённая для различного количества детектируемых прямых зависимость качества работы алгоритмов от дисперсии аддитивного некоррелированного шума; (в) – прирост качества алгоритма оценки координат объекта при использовании модифицированного преобразования Радона в процентах по сравнению с алгоритмом на основе обычного преобразования Радона

Заключение

Алгоритм оценки координат объекта на основе преобразования Радона, предлагаемый в данной работе, имеет значительно большую устойчивость к аддитивному некоррелированному

шуму по сравнению с алгоритмом, основанным на обычном преобразовании Радона. Из рис. 3в очевидно, что в задаче оценки координат объектов с прямолинейными границами при наличии шума на изображении прирост качества составляет около 40%.

Алгоритм может быть реализован для использования в системах обнаружения и сопровождения объектов в реальном времени, в том числе в бортовых системах навигации роботов, пилотируемых и беспилотных аппаратов.

Работа проводилась при финансовой поддержке Министерства образования и науки РФ (госконтракт № 07.514.11.4034).

Литература

1. Алпатов Б.А., Бабаян П.В., Балашов О.Е., Степашкин А.И. Методы автоматического обнаружения и сопровождения объектов. Обработка изображений и управление. – М.: Радиотехника, 2008. – С. 176: ил.
2. D.B. Volegov, V.V. Gusev and D.V. Yurin. Straight line detection on images via Hartley transform fast Hough transform // GraphiCon'2006, 16-th International Conference, Novosibirsk Akademgorodok, Russia, 2006.
3. A.C. Copeland, G. Ravichandran, and M.M. Trivedi. Localized Radon transform-based detection of ship wakes in SAR images // IEEE Transactions on Geoscience and Remote Sensing, Vol. 33, Issue 1, 1995. – 35-45 pp.
4. Toft P.A., «The Radon Transform: Theory and Implementation», PhD Thesis, Technical University of Denmark, 1996.
5. Бабаян П.В., Шубин Н.Ю. Выделение прямолинейных границ на основе модифицированного преобразования Радона // Проблемы передачи и обработки информации в сетях и системах телекоммуникаций: Материалы 16-й международной научнотехнической конференции. – Рязань, 2010. – С. 21-23.
6. Алпатов Б.А., Балашов О.Е., Шубин Н.Ю. Повышение точности измерения параметров подвижных объектов в оптико-механических системах // Тезисы докладов XII международной конференции «Цифровая обработка сигналов и её применение». – М., 2010. Т.2 – С. 120-123.

OBJECT COORDINATES ESTIMATION ALGORITHM BASED ON RADON TRANSFORM

Alpatov B.A., Babayan P.V., Shubin N.J.

The problem of linear edge object coordinates estimation is described in this paper. The algorithm based on Radon transform is suggested. Line detection quality is increased by Radon transform modifying. Object coordinates estimation algorithm based on common and modified Radon transform is also suggested. The results of experimental research of the proposed algorithm are presented.

АЛГОРИТМ СЛЕЖЕНИЯ ЗА ДВИЖУЩИМИСЯ ОБЪЕКТАМИ В БОРТОВЫХ ВИДЕОИНФОРМАЦИОННЫХ СИСТЕМАХ

Корепанов С.Е., научный сотрудник кафедры автоматики и информационных технологий в управлении (АИТУ) Рязанского государственного радиотехнического университета (РГРТУ),
e-mail: aitu@rsreu.ru.

Строгов В.В., к.т.н., доцент кафедры АИТУ РГРТУ, e-mail: aitu@rsreu.ru

Ключевые слова: обработка видеопоследовательностей, движущийся объект, слежение, геометрические преобразования.

Введение

Слежение за выбранным объектом является одной из наиболее важных задач, решаемых бортовыми видеоинформационными системами. Известно достаточно большое количество методов решения задачи слежения за объектами при подвижном фоне [1, 2], однако они весьма требовательны к вычислительным ресурсам системы. Актуальной является задача создания вычислительно эффективных методов слежения для работы в условиях подвижного поля зрения.

Если рассмотреть иерархию задач, решаемых бортовыми видеосистемами, представленную в [1], то можно видеть, что для решения задачи слежения необходимо предварительно решить целый ряд задач более низкого уровня. При этом для решения задач выделения объектов нередко выполняют оценку параметров геометрических преобразований изображений [3, 4]. Использование данной информации может помочь в адаптации известных алгоритмов для случая подвижных датчиков изображений.

В данной работе предложен алгоритм слежения за движущимися объектами в последовательности изображений с неоднородным сдвигающимся фоном с использованием оценки сдвига фонового изображения.

Постановка задачи

В качестве основной системы координат выберем неподвижную относительно земли систему, начало координат которой совпадает с левым нижним углом первого кадра наблюдаемой видеопоследовательности, а оси параллельны его сторонам. Предположим, что поле зрения видеодатчика изменяется таким образом, что наблюдаемое изображение сдвигается. Как отмечено выше, характер сдвига может быть различным, поэтому, обозначив положение центра в кадре n фонового изображения в основной системе координат вектором $\Lambda_n^g = [\lambda_{x,n}^g, \lambda_{y,n}^g]^T$, положим:

$$\Lambda_n^g = \Lambda_{n-1}^g + [\Delta\lambda_{x,n}^g, \Delta\lambda_{y,n}^g]^T, \quad (1)$$

$$|\Delta\lambda_{x,n}^g| \leq C_g, |\Delta\lambda_{y,n}^g| \leq C_g$$

где C_g – максимально допустимый сдвиг фона за время формирования очередного кадра.

Движение объекта представим в той же системе координат, что и движение фона. Считая движения по координатам x и y независимыми, изменение положения объекта относительно первого кадра наблюдаемой видеопоследовательности опишем следующей моделью:

$$\Lambda_{n+1}^h = F \cdot \Lambda_n^h + \omega_n^h, n = 1, N-1,$$

Рассмотрен алгоритм слежения за движущимися объектами при подвижном поле зрения видеодатчика. В алгоритме используются результаты оценки параметров геометрических преобразований изображений для повышения точности оценки координат объектов. Представлены результаты экспериментальных исследований разработанного алгоритма.

$$\Lambda_n^h = \begin{bmatrix} \lambda_{x,n}^h \\ \dot{\lambda}_{x,n}^h \\ \lambda_{y,n}^h \\ \dot{\lambda}_{y,n}^h \end{bmatrix}, \omega_n^h = \begin{bmatrix} \omega_{x,n} \\ 0 \\ \omega_{y,n} \\ 0 \end{bmatrix}, F = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

где $\lambda_{x,n}^h, \lambda_{y,n}^h$ – координаты центра объекта на n -м кадре наблюдаемой видеопоследовательности; $\dot{\lambda}_{x,n}^h, \dot{\lambda}_{y,n}^h$ – мгновенная скорость объекта; ω_n^h – вектор возмущений, которые описываются моделью белого шума с нулевым средним и диагональной ковариационной матрицей Q^h ; Δt – период следования кадров видеопоследовательности; N – число кадров видеопоследовательности.

Модель движения объекта на наблюдаемом изображении при этом будет иметь вид:

$$\Lambda_n^+ = A \Lambda_n^h - \Lambda_n^g, A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (3)$$

Ошибки при измерении координат объекта $\delta \Lambda_n^+ = [\delta \lambda_{x,n}^+, \delta \lambda_{y,n}^+]^T$ и фона $\delta \Lambda_n^g = [\delta \lambda_{x,n}^g, \delta \lambda_{y,n}^g]^T$ будем считать гауссовыми некоррелированными величинами с нулевыми средними и диагональными ковариационными матрицами $K_{\delta}^+, K_{\delta}^g$ на кадре n .

Наблюдаемое изображение $I_n(i, j)$ для кадра n видеопоследовательности можно описать известной моделью заслона [1]. Предполагается, что все введенные возмущения, начальные значения вектора состояния объекта, шумы и ошибки измерения взаимно некоррелированы.

Поставим задачу слежения за объектами следующим образом. Пусть известно эталонное изображение объекта $I_1^*(i, j)$, которое задаётся целеуказанием оператора на первом кадре видеопоследовательности. Требуется на основе наблюдаемых изображений $I_n(i, j)$, $n = \overline{1, N}$ вычислять оценки координат объекта $\hat{\theta}_{x,n}, \hat{\theta}_{y,n}$ на каждом n кадре.

Оценка параметров объектов при подвижном фоне

Вопросы синтеза алгоритмов оптимального измерения координат при неоднородном подвижном фоне в изображении и экспериментальное подтверждение эффективности данного подхода для некоторых видов фонов приведены в работах [1, 2, 5]. Недостатки аналитического алгоритма связаны с его вычислительной неэффективностью. Так, на каждом кадре необходимо получать оценки изображений объекта h_n , фона g_n и маски объекта r_n , а затем оптимизировать многопараметрический критерий для совместного определения сдвига объекта и фона.

В связи с указанными недостатками на практике предлагается использовать эвристический алгоритм. Он построен на основе подхода, использующего для уточнения координат объекта априорные гауссовы плотности вероятности распределения координат объекта, заданные соответствующими средним значением $\bar{\Lambda}_n^h$ и ковариационной матрицей $\mathbf{M}_n^h$ вектора состояния объекта Λ_n^h . Данные характеристики можно получить, применяя аппарат Калмановской фильтрации.

На первом шаге предлагаемого алгоритма производится оценка параметров геометрических преобразований фонового изображения $\hat{\lambda}_{x,n}^g, \hat{\lambda}_{y,n}^g$ при помощи алгоритма, описанного в работах [3, 4]. На основе значений $\bar{\Lambda}_{x,n}^g$ и $\bar{\Lambda}_n^h$ по формуле (3) вычисляются прогнозируемые координаты объекта на текущем изображении $\bar{\Lambda}_n^+$.

$$\bar{\Lambda}_n^+ = A \bar{\Lambda}_n^h - \bar{\Lambda}_n^g \quad (4)$$

На втором шаге алгоритма определяются измерения $\tilde{\lambda}_{x,n}^+, \tilde{\lambda}_{y,n}^+$ координат искомого объекта на изображении. Для этого в заданной окрестности прогнозируемого положения объекта находится минимум критериальной функции, предложенной в [5]:

$$q(\lambda_{x,n}^+, \lambda_{y,n}^+) = S(\lambda_{x,n}^+, \lambda_{y,n}^+) \sum_{i=1}^{W_x} \sum_{j=1}^{W_y} |h_n(i, j) - h_{n-1}(i - \lambda_{x,n}^+, j - \lambda_{y,n}^+)|, \quad (5)$$

где W_x, W_y – размеры объекта, $S(\theta_x, \theta_y)$ – мультипликативная штрафная функция, вычисляемая по формуле:

$$S(\lambda_x, \lambda_y) = \left(\sqrt{(\lambda_x - \bar{\lambda}_x^+)^2 + (\lambda_y - \bar{\lambda}_y^+)^2} + 1 \right)^\alpha, \quad (6)$$

где α – параметр, характеризующий степень доверия к прогнозируемым координатам. Для измерения координат объекта необходимо минимизировать выражение (5). Штрафная функция (6) позволяет учитывать величину отклонения измеренных координат объекта от прогнозируемых для уточнения координат объекта, подавляя ложные экстремумы критериальной функции, вызываемые появлением в непосредственной близости от наблюдаемого объекта похожих объектов.

На третьем шаге алгоритма на основе данных, полученных на первых двух шагах, находятся измерения координат объекта $\tilde{\lambda}_{x,n}^h, \tilde{\lambda}_{y,n}^h$ в системе координат, связанной с первым кадром объекта:

$$\tilde{\lambda}_{x,n}^h = \tilde{\lambda}_{x,n}^+ + \hat{\lambda}_{x,n}^g, \quad \tilde{\lambda}_{y,n}^h = \tilde{\lambda}_{y,n}^+ + \hat{\lambda}_{y,n}^g. \quad (7)$$

Это позволяет производить уточнение оценок и выполнять прогнозирование в неподвижной системе

координат. При этом ошибка измерения складывается из ошибок измерения $\delta \Lambda_n^+$ и $\delta \Lambda_n^g$. Её математическое ожидание будем считать близким к нулю, а ковариационную матрицу определим как:

$$\mathbf{K}_\delta^h = \mathbf{K}_\delta^+ + \mathbf{K}_\delta^g. \quad (8)$$

На следующем шаге алгоритма определяется оценка вектора состояния объекта. При введенных моделях наблюдения изображений и измерения координат, с точки зрения теории оптимальной фильтрации, оценка координат может быть получена из следующих выражений:

$$\bar{\Lambda}_n^h = \bar{\Lambda}_n^h + \mathbf{K}_n^h (A \bar{\Lambda}_n^h - A \bar{\Lambda}_n^h), \quad (9)$$

где $\mathbf{K}_n^h = \mathbf{P}_n^h A^T (\mathbf{K}_\delta^h)^{-1}$ – матрица коэффициентов;

$\mathbf{P}_n^h = \left((\mathbf{M}_n^h)^{-1} + A^T (\mathbf{K}_\delta^h)^{-1} A \right)^{-1}$ – апостериорная ковариационная матрица оценки.

Для получения средних значений $\bar{\Lambda}_{n+1}^h$ и ковариационных матриц $\mathbf{M}_{n+1}^h$ требуется осуществить прогнозирование на следующий $n+1$ кадр по формулам:

$$\bar{\Lambda}_{n+1}^h = F \bar{\Lambda}_n^h, \quad \mathbf{M}_{n+1}^h = F \mathbf{P}_{x,n}^h F^T + Q^h. \quad (10)$$

Экспериментальные исследования

Для проведения экспериментальных исследований были использованы натурные видеопоследовательности с неподвижным фоном, содержащие несколько похожих движущихся объектов. Координаты объектов на каждом кадре изображения были заранее определены человеком-экспертом. Дополнительно изображения кадров подвергались преобразованию сдвига. Величины сдвигов по каждой из осей являлись некоррелированными целочисленными случайными гауссовскими величинами с нулевыми математическими ожиданиями и известными дисперсиями $\sigma_{dx}^2 = \sigma_{dy}^2 = \sigma_d^2$. Отношение сигнал/шум на всех рассмотренных видеосюжетах было приблизительно равно 8.

Характер движения объектов на использованных видеопоследовательностях, в целом, соответствует введенным моделям. Время от времени два и более объекта сближаются на расстояние меньшее, чем размер зоны поиска объекта. Это может приводить к тому, что координаты объекта, за которым производится слежение, определяются со значительной ошибкой (более 10 пикселей) на протяжении 25 кадров (1 секунды). Такую ситуацию будем называть срывом слежения.

Были проведены исследования четырех модификаций алгоритма слежения за движущимся объектом:

1. Алгоритм, построенный без учёта сдвига фона и без применения мультипликативной штрафной функции (значение параметра $\alpha = 0$).
2. Алгоритм, построенный без учёта сдвига фона, но с применением мультипликативной штрафной функции (значение параметра $\alpha = 0,1$).
3. Алгоритм, построенный с учётом сдвига фона, но без применения мультипликативной штрафной функции (значение параметра $\alpha = 0$).
4. Алгоритм, построенный с учётом сдвига фона и с применением мультипликативной штрафной функции (значение параметра $\alpha = 0,1$).

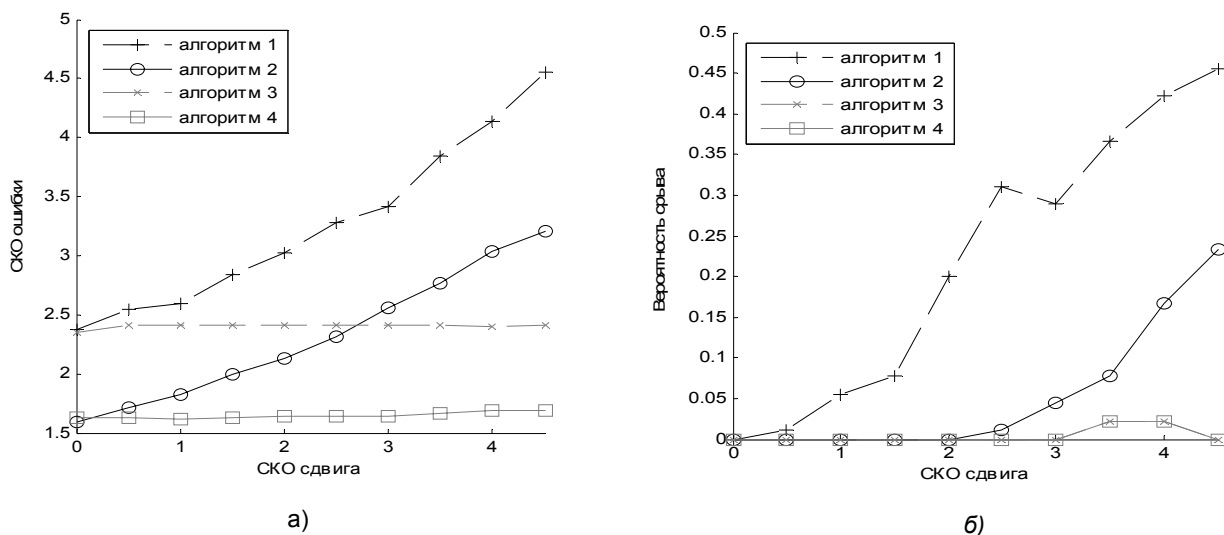
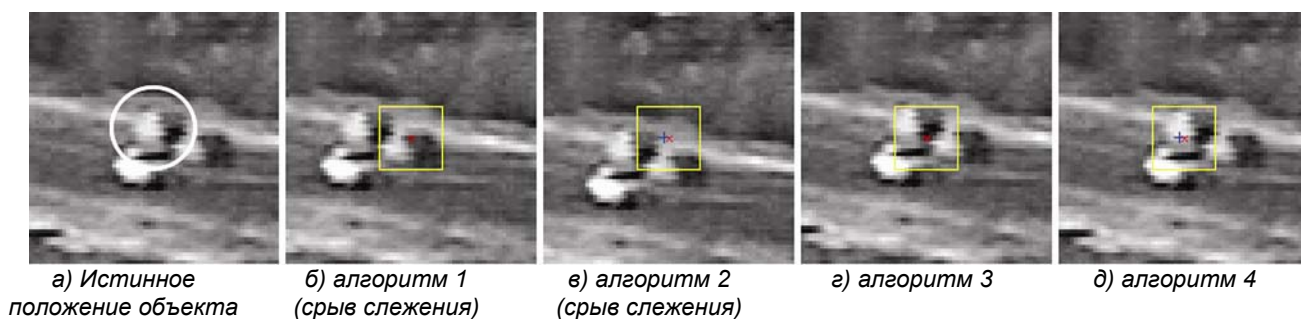


Рис. 1. Результаты экспериментальных исследований алгоритмов

Рис. 2. Результаты оценки координат при наличии в окрестности объекта похожих объектов при $\sigma_d = 4$ пикселя

Результаты исследований приведены на рис. 1а и 1б. На рис. 1а представлены графики зависимости СКО ошибки определения координат и вероятности срыва слежения от величины σ_d . Необходимо отметить, что результаты были усреднены по нескольким видеопоследовательностям и нескольким реализациям случайного смещения с одним и тем же значением дисперсии.

На рис. 2 показаны результаты оценки координат для случая, когда СКО сдвига поля зрения видеодатчика велико, а в ближайшей окрестности объекта оказываются несколько похожих объектов. Из рис. 1а и 1б видно, что точность работы алгоритмов 3 и 4, использующих информацию о сдвиге изображения, практически не зависит от величины смещения, вызванного изменением поля зрения датчика, а использование штрафной функции (б) значительно уменьшает вероятность срыва слежения.

Заключение

В данной статье рассмотрены различные подходы к слежению за движущимися объектами в бортовых видеоинформационных системах при изменении ориентации поля видеодатчика. Предложен алгоритм слежения за объектами, использующий информацию о параметрах геометрических преобразований изображения текущего кадра. Экспериментально показана эффективность данного алгоритма в условиях случайных сдвигов поля зрения видеодатчика.

Исследования проводились при финансовой поддержке Министерства образования и науки РФ (госконтракт №16.740.11.0223).

Литература

1. Алпатов Б.А., Бабаян П.В., Балашов О.Е., Степаш-

кин А.И. Методы автоматического обнаружения и сопровождения объектов. Обработка изображений и управление. – М.: Радиотехника, 2008. – С. 176.

2. Алпатов Б.А. Оптимальное оценивание параметров движущегося объекта в последовательности изображений // Автометрия. – 1994. – №2. – С. 32-37.
3. Стротов В.В. Оценивание параметров смещения изображения в задачах выделения движущихся объектов // Вестник Рязанского государственного радиотехнического университета. – Рязань, 2008 – №23. – С. 30-37.
4. Стротов В.В. Выбор опорных участков в многоэталонном алгоритме определения параметров геометрических преобразований изображений // Вестник Рязанского государственного радиотехнического университета. – Рязань, 2009. – №2 (28). – С. 93-96.
5. Алпатов Б.А., Бабаян П.В., Корепанов С.Е. Корреляционный алгоритм слежения за движущимися объектами при наличии близкорасположенных похожих объектов // Вестник РГРТУ. – Рязань, 2010. – №2 (32). – С. 3-8.

TRACKING MOVING OBJECTS ALGORITHM IN AIRBORNE VIDEO INFORMATION SYSTEMS

Korepanov S.E., Strotov V.V.

The article considers a moving objects tracking algorithm for the case of moving video sensor field of view. The algorithm uses the images geometric transformations parameters estimate to improve the accuracy of objects coordinates estimates. Results of algorithm experimental examinations are given.

ЭФФЕКТИВНОСТЬ ЦИФРОВОГО ОБНАРУЖЕНИЯ ПРОТЯЖЕННЫХ АНОМАЛИЙ С НЕИЗВЕСТНЫМИ УРОВНЯМИ НА ФОНЕ МЕШАЮЩИХ ПОМЕХ

Васильев К.К., зав. кафедрой Ульяновского государственного технического университета, e-mail: vkk@ulstu.ru
Лучков Н.В., e-mail: nik-lnv@mail.ru

Ключевые слова: алгоритм обнаружения, протяженные аномалии, многозональные изображения, случайное поле, квазиоптимальный приём.

Введение

Задачи обнаружения и оценивания параметров аномалий на многомерных изображениях представляют интерес для целого ряда приложений. Среди них можно выделить радио- и гидролокационные комплексы с пространственными антенными решетками, аэрокосмические системы глобального мониторинга Земли, системы технического зрения и др. [1-12]. В последнее время задачи обработки многомерных данных приобретают особую актуальность в связи с широким распространением методов мультиспектральной (до 10 спектральных диапазонов) и гиперспектральной (до 300 диапазонов) регистрации участков земной поверхности. Это обуславливает необходимость разработки новых методов качественного и количественного анализа аэрокосмических наблюдений как единой многомерной совокупности. Для ряда приложений особый интерес представляют задачи обнаружения и оценивания параметров аномалий на многозональных изображениях. Общее решение задачи обнаружения основано на построении модифицированного отношения правдоподобия и сравнении с пороговым значением. По результатам этого сравнения принимается решение в пользу гипотезы о наличии полезного сигнала или гипотезы о его отсутствии. Несмотря на большое число публикаций [1-12] по проблемам обнаружения сигналов, в настоящее время отсутствуют удовлетворительные решения задач синтеза, анализа и сравнения соответствующих алгоритмов, особенно для протяженных сигналов с неизвестными уровнями на отдельных кадрах изображения.

Анализ большого числа экспериментальных данных показывает, что полезный сигнал на реальных многозональных изображениях, как правило, сохраняет свою форму и структуру, но меняет интенсивность в зависимости от спектрального диапазона. В подобных задачах важную роль играют алгоритмы, работающие в режиме реального времени и обеспечивающие быструю реакцию на предъявляемое изображение наблюдаемой сцены при минимальных вычислительных затратах. В данной работе предложена новая структура квазиоптимального правила обнаружения протяженных сигналов с неизвестными уровнями на фоне коррелированных помех, а также дана сравнительная оценка эффективности процедур обнаружения протяженных сигналов с известными и неизвестными уровнями.

Выполнен синтез оптимальных алгоритмов обнаружения протяженных сигналов с неизвестными уровнями на произвольном числе кадров многозонального изображения. Дана новая структура квазиоптимального правила обнаружения протяженных сигналов с неизвестными уровнями на фоне коррелированных помех. Определена область решений параметров СП, для которых возможно применение алгоритма обнаружения с предварительной оценкой уровня полезного сигнала с заданным выигрышем в сравнении с алгоритмом обнаружения с известными значениями уровней полезного сигнала.

Синтез алгоритмов обнаружения сигналов

Представим многозональное изображение как совокупность массивов данных, состоящих из k кадров изображения, $\{z_{ijk}\}$, $k = 1 \dots N$, $i = 1 \dots M_1$, $j = 1 \dots M_2$, которые получены в результате пространственной дискретизации сигналов, поступивших от различных систем датчиков, рис. 1. При этом элементами СП являются яркости изображения в заданных точках ij,k .

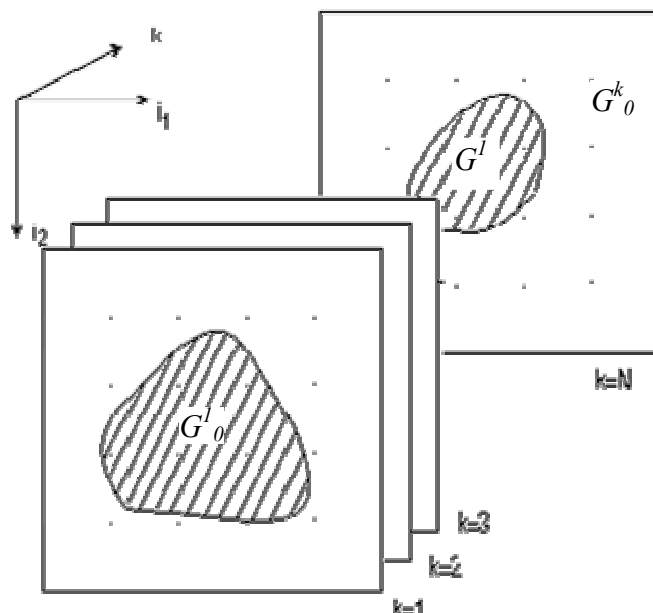


Рис. 1 Кадр многозонального изображения

При отсутствии полезного сигнала (гипотеза H_0) модель наблюдений можно представить аддитивной смесью:

$$z_{ijk} = x_{ijk} + \theta_{ijk}, \quad (i, j) \in G^k, \quad k = 1 \dots N,$$

однородного случайного поля x_{ijk} с нулевым средним и заданной корреляционной функцией (КФ)

$B_{(ml)}^{kt} = M \{x_{ij}^k, x_{i+m, j+l}^t\}$ и пространственного белого шума θ_{ijk} с нулевым математическим ожиданием и дисперсией σ_θ^2 .

При наличии полезного сигнала (гипотеза H_1) модель наблюдений запишется в виде:

$$z_{ijk} = x_{ijk} + s_{ijk} + \theta_{ijk}, \quad (i, j) \in G_0^k, \quad k = 1 \dots N,$$

$$z_{ijk} = x_{ijk} + \theta_{ijk}, \quad (i, j) \notin G_0^k, \quad k = 1 \dots N,$$

где G_0^k – область на k -м кадре, для которого возможно появление полезного сигнала с известными уровнями s_{ijk} , $(i, j) \in G_0^k$. Для упрощения выкладок будем считать, что на каждом из кадров эта область одинакова $G_0^k = G_0$. Если это не так, то можно расширить каждую из областей до размеров наибольшей, а полезный сигнал в добавленных точках считать нулевым.

Протяженный в пространстве полезный сигнал можно записать в виде $s_{ijk} = s_k f_{ijk}$, где s_k – неизвестные уровни сигнала на отдельных кадрах изображения; f_{ijk} – известные значения размеров сигнала, расположенного на k -м кадре [2].

Оптимальный алгоритм обнаружения аномалий получен на основе модифицированного отношения правдоподобия [3-5]:

$$L = \frac{w(\{z_{ijk}\} / H_1, \tilde{s}_1 \dots \tilde{s}_N)}{w(\{z_{ijk}\} / H_0)}, \quad \text{где } \tilde{s}_k, \quad k = 1 \dots N, \quad \text{оценки}$$

неизвестных уровней полезного сигнала, полученные с помощью метода максимального правдоподобия.

Аппроксимируя условные вероятности $w(\{z_{ij}^k\} / H_1, s^1 \dots s^N)$ и $w(\{z_{ij}^k\} / H_0)$ гауссовскими, решающее правило оптимального обнаружения протяженных сигналов по совокупности наблюдений на N кадрах может быть получено в виде [3]:

$$L = \tilde{s}_k f_{ijk} V_{ijk}^{-1} (z_{ijk} - \tilde{x}_{ijk}) \begin{cases} > L_0 - \text{сигнал есть,} \\ < L_0 - \text{сигнала нет,} \end{cases} \quad (1)$$

где $\tilde{x}_{ijk}$ – оптимальный (в смысле минимума дисперсии ошибки) прогноз случайного поля, сделанный на основе всех наблюдений Z_0 , в которых полезный сигнал заведомо отсутствует; $V = P_\theta + V_\theta$; P_θ – ковариационная матрица ошибок при оптимальном прогнозировании. Отметим, что в соответствии с правилами тензорного исчисления запись вида $a_k b_k$ эквивалентна $\sum_{k=1}^N a_k b_k$, т.е. предполагается суммирование по одинаковым нижним индексам.

Неизвестные оценки $\tilde{s}_k$, $k = 1 \dots N$, можно определить из условия максимума

$w(\{z_{ijk}, (i, j, k) \in \Omega_0\} / Z_0, H_1)$, или минимума выражения $(z_{lvt} - m_{lvt}) V_{lvt}^{-1} (z_{ijk} - m_{ijk})$. После дифференцирования квадратичной формы может быть получена следующая система линейных уравнений:

$$\sum_{k=1}^N (f_{lv}^t V_{lv}^{-1} f_{ij}^k) s^k = \sum_{k=1}^N f_{lv}^t V_{lv}^{-1} f_{ij}^k (z_{ij}^k - \tilde{x}_{ij}^k), \quad t = 1 \dots N \quad (2)$$

Существенным недостатком всех рассмотренных алгоритмов является достаточно сложный анализ эффективности обнаружения сигналов. В процессе поиска путей упрощения такого анализа удалось получить еще одну процедуру обнаружения протяженных сигналов. Для этого достаточно в (1) и (2) подставить известную связь между тензорами оценок

$$\tilde{x}_{ijk} = \tilde{x}_{ijk} + P_{lvijk} V_\theta^{-1} (z_{lvt} - \tilde{x}_{lvt}), \quad (i, j) \in G_0^k, \quad k = 1 \dots N, \quad \text{где}$$

$$\tilde{x}_{ijk} - \text{оптимальная оценка СП в области } (i, j) \in G_0^k, \quad k = 1 \dots N, \text{ сделанная на основе всех наблюдений } \{z_{lvt}\},$$

$$(l, v) \in G_0, \quad t = 1 \dots N; \quad P - \text{ковариационная матрица ошибок фильтрации. После элементарных преобразований с учетом того, что } P(E + V_\theta^{-1} P_\theta) = P_\theta,$$

$z - \tilde{x}_3 = (E + V_\theta^{-1} P_\theta)(z - \tilde{x})$, находим следующую формулу решающего правила:

$$L = \frac{\tilde{s}_k f_{ijk} (z_{ijk} - \tilde{x}_{ijk})}{\sigma_\theta^2} \begin{cases} > L_0 - \text{сигнал есть,} \\ < L_0 - \text{сигнала нет.} \end{cases} \quad (3)$$

При этом неизвестные оценки полезного сигнала $\tilde{s}_k$, $k = 1 \dots N$ получаем из следующей системы линейных уравнений:

$$\sum_{k=1}^N (f_{lv}^t (V_\theta - P)_{lvij}^k f_{ij}^k) s^k = \sigma_\theta^2 \sum_{t=1}^N f_{ij}^t (z_{ij}^t - \tilde{x}_{ij}^t), \quad t = 1 \dots N.$$

Поскольку алгоритм (3), в отличие от известных обнаружителей, не требует трудоёмкого расчёта ковариационных матриц ошибок прогнозирования, то его удобно использовать не только при реализации систем обработки изображений, но и для анализа эффективности обнаружения.

Сравнительный анализ алгоритмов

Для определения области применения предложенного алгоритма (3) был проведён расчёт зависимости вероятности правильного обнаружения Pd от уровня полезного сигнала S , обеспечивающего заданные значения вероятности ложной тревоги $Pf=0,001$. При этом на рис. 3 моделируемое СП содержит полезный сигнал, имеющий одинаковые уровни значений на всех кадрах изображения. На рис. 4 на моделируемом изображении полезный сигнал имеет различные уровни значений на всех кадрах изображения. Графики «1» на рис. 2 и рис. 3 построены с использованием оптимального алгоритма обнаружения (1), а графики «2» на рис. 2 и рис. 3 – алгоритма с предварительным оцениванием полезного сигнала (3). Использовались следующие статистические параметры моделируемого СП: $\sigma_X^2 = 1; \sigma_\theta^2 = 1; \rho = 0,9; \rho_k = 0,9$ размер полезного сигнала 7×7 . Для определения порогового уровня, обеспечивающего заданное значение вероятности ложной тревоги и вероятности правильного обнаружения, проведено статистическое моделирование объёмом 10 000 опытов.

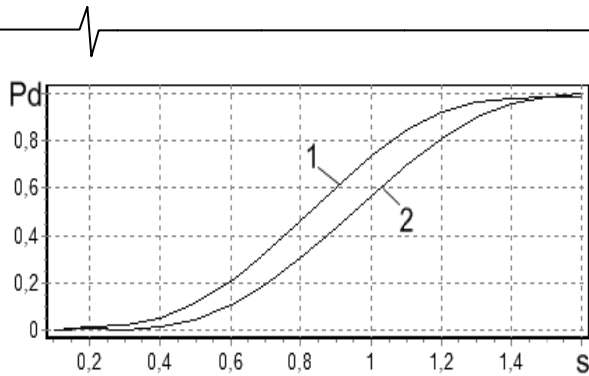


Рис. 2 Зависимость вероятности правильного обнаружения от уровня полезного сигнала

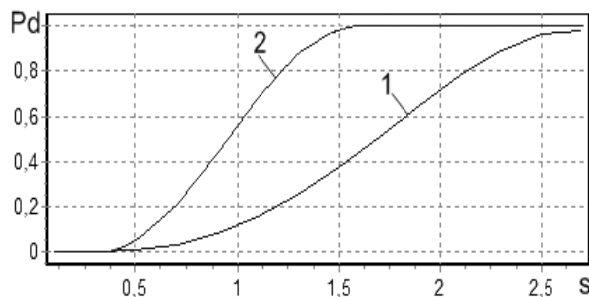


Рис. 3 Зависимость вероятности правильного обнаружения от уровня полезного сигнала

Анализ результатов моделирования алгоритмов обнаружения, представленных на рис. 2 и рис. 3, позволяет сделать вывод, что при одинаковых значениях полезного сигнала на различных кадрах многомерного изображения использование алгоритма (3) приводит к незначительному снижению вероятности правильного обнаружения в сравнении с использованием оптимального алгоритма обнаружения (1). Но при условии, что значения уровней сигнала различны на различных кадрах СП, алгоритм с предварительной оценкой уровней сигнала (3) имеет значительное преимущество по значениям вероятности правильного обнаружения в сравнении с оптимальным алгоритмом (1). С увеличением значений вероятности правильного обнаружения относительный выигрыш по уровню полезного сигнала увеличивается.

Учитывая, что условие различия значений сигнала для разных кадров СП является типичным случаем для реальных многозональных изображений, возникает задача определения области применения, предложенного алгоритма (3), в которой выигрыш обнаружения будет наибольшим в сравнении с оптимальным алгоритмом обнаружения (1). В связи с этим были проведены дополнительные исследования эффективности алгоритма (3) в сравнении с оптимальным алгоритмом обнаружения (1). На рис. 4 – рис. 7 представлены некоторые результаты расчетов при заданных значениях вероятности ложной тревоги $P_f=0,001$ и вероятности правильного обнаружения $P_d=0,9$, иллюстрирующие зависимости относительно-

го выигрыша $Q = 20 \log \frac{S}{\tilde{S}}$ от ряда статистических параметров наблюдаемой сцены обнаружения, где S – уровень полезного сигнала, обеспечивающий заданные вероятностные характеристики оптимального алгоритма (1); $\tilde{S}$ – уровень полезного сигнала, обеспечивающий заданные вероятностные характе-

ристики алгоритма с предварительным оцениванием значения уровня сигнала (3). При этом проведено моделирование СП, состоящего из двух кадров изображения, на одном из которых полезный сигнал присутствует, а на другом отсутствует.

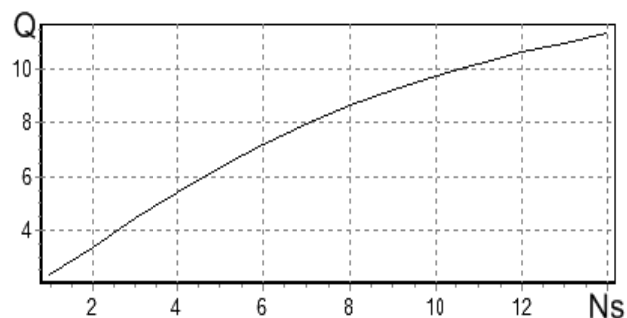


Рис. 4 Зависимость относительного выигрыша от размера полезного сигнала

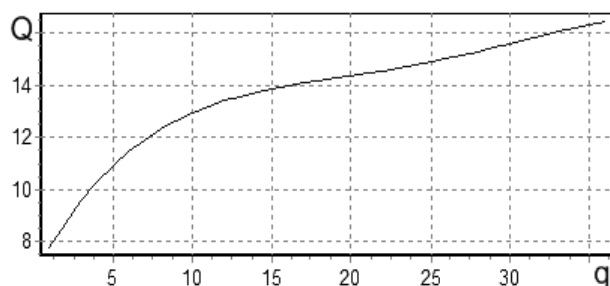


Рис. 5 Зависимость относительного выигрыша от уровня фона

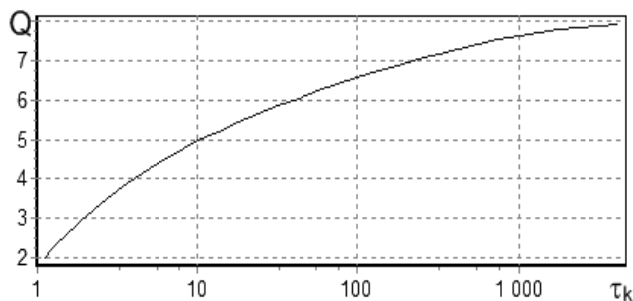


Рис. 6 Зависимость относительного выигрыша от межкадрового интервала корреляции

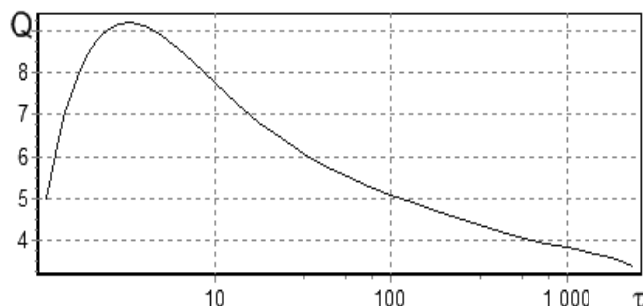


Рис. 7 Зависимость относительного выигрыша от внутрикадрового интервала корреляции, $N_s=7$

Анализ приведенных графиков показывает, что с увеличением размера полезного сигнала выигрыш Q , при использовании алгоритма обнаружения с предварительным оцениванием в сравнении с оптимальным алгоритмом, увеличивается. Так, при уровне полезного сигнала

равного 14, выигрыш составляет 11,5 дБ. Выигрыш Q растёт с увеличением значений уровня фона наблюдае-

мой сцены $q = \frac{\sigma_X^2}{\sigma_\theta^2}$, достигая 16,2 дБ при $q=35$. Увеличе-

ние уровня дисперсии аддитивного шума σ_θ^2 приводит к уменьшению относительного выигрыша Q , что объясняется значительным увеличением дисперсии ошибки предварительного оценивания полезного сигнала. Также выигрыш Q растёт с увеличением межкадрового интервала корреляции $\tau_k = \frac{1}{1 - \rho_k}$ и стремится к уровню 9 дБ,

где ρ_k – межкадровый коэффициент корреляции.

Проанализировав зависимость относительного выигрыша от внутрикадрового интервала корреляции, рис. 7, можно сделать вывод, что величина относительного выигрыша Q стремится к уровню 3дБ и имеет максимум при определенных соотношениях значений внутрикадрового интервала корреляции $\tau = \frac{1}{1 - \rho}$ и протяженности

полезного сигнала N_s , где ρ – внутрикадровый коэффициент корреляции. Дополнительные исследования

показали, что при небольшой протяженности полезного сигнала N_s значения относительного выигрыша Q принимают максимальное значение при $\tau \approx \alpha N_s$, $\alpha \approx 0,5$. При увеличении значения N_s коэффициент α уменьшается и стремится к значению $\alpha \approx 0,2$.

Заключение

Синтез алгоритмов обнаружения протяженных сигналов с неизвестными уровнями на произвольном числе кадров многозонального изображения позволил получить новую структуру квазиоптимального правила обнаружения протяженных сигналов с неизвестными уровнями на фоне коррелированных помех. В её основе лежат оптимальный алгоритм обнаружения аномалий на основе модифицированного отношения правдоподобия и предварительная оценка уровня полезного сигнала методом максимального правдоподобия. Найдены необходимые аналитические соотношения и дана сравнительная оценка эффективности процедур обнаружения протяженных сигналов с известными и неизвестными уровнями. Выигрыш при вероятности правильного обнаружения уровня 0.9 может составлять до 8 дБ. Полученные результаты позволяют определить область решений параметров СП, для которых возможно применение алгоритма обнаружения с предварительной оценкой уровня значений полезного сигнала с заданными значениями вероятности ложной тревоги и вероятности правильного обнаружения, обеспечивая максимально возможный относительный выигрыш в сравнении с применением оптимального алгоритма обнаружения. При увеличении внутрикадрового интервала корреляции выигрыш стремится к 3 дБ, а при увеличении межкадрового интервала корреляции выигрыш стремится к 9 дБ. Разработан пакет программ, позволяющий обрабатывать реальные многозональные изображения, имеющий важное прикладное значение для создания наукоемких технологий обработки многозональных аэрокосмических изображений.

Литература

1. Аникин И.В. Методы нечеткой обработки, распознавания и анализа предметов / И.В. Аникин, М.Р. Шагиахметов // Распознавание образов и анализ сцен: труды 5 межд. конф. – С.-Петербург, 2002. – т. 1. – С. 16-20.
2. Васильев К. К. Прикладная теория случайных процессов и полей / К. К.Васильев, В.А. Омельченко – Ульяновск: УГТУ, 1995.– С. 255.
3. Васильев К.К. Четыре типа оптимальных алгоритмов обнаружения аномалий на многомерных изображениях / К.К. Васильев // Радиолокация, навигация, связь: Труды 13-ой Международной научно-технической конференции – Воронеж, 2007.- том. 1 – С. 171-176.
4. Васильев К.К., Дементьев В.Е. Алгоритмы оптимального обнаружения сигналов с неизвестными уровнями на многозональных изображениях. – Труды VIII МНК «Цифровая обработка сигналов и её применение»: – 2006, т. 2, С. 433-436.
5. Васильев К.К., Крашенинников В.Р. Статистический анализ многомерных изображений / К.К. Васильев, В.Р. Крашенинников. – Ульяновск: УлГТУ, 2007. – С. 170.
6. Грузман И.С., Киричук В.С., Косых В.П., Перетягин Г.И., Спектор А.А. «Цифровая обработка изображений в информационных системах». – Новосибирск, НГТУ, 2002.
7. Прэтт, У. «Цифровая обработка изображений»: пер. с англ. / У. Прэтт; под ред. Д.С. Лебедева. – М., 1982. – Кн. 1. – 312.; Кн. 2 – С. 480.
8. Dikshit S.S. A Recursive Kalman Window Approach to Image Restoration // IEEE Trans., 1984, Vol. com – 32, Jan., 125-139 pp.
9. Luchkov N., Dementiev V. Detection effectiveness analysis for elongated signals on multidimensional grids // PRIA 2010.
10. Soifer V.A., Kotlyar V.V., Khonina S.N., and Khramov A.G. Pattern Recognition and Image Analysis 6(4) 710 (1996).
11. Vasil'ev K.K. Digital Processing of Multizonal Images. – Pattern Recognition and Image Analysis, 2008, vol.18, №3, 376-380 pp.
12. Woods J.W. Two-dimensional Kalman filtering //Topics in Applied Physics, Berlin, 1981, v.42, 155-208 pp.

DETECTION DIGITAL EFFECTIVENESS FOR ELONGATED SIGNALS OF UNKNOWN VALUES AGAINST A NUISANCE BACKGROUND

K. Vasil'ev, N. Luchkov

There are accomplished syntheses of optimal detection algorithms for elongated signals of unknown values on a multizone image for an arbitrary frame count in present article. The quasioptimal decision rule for elongated signals of unknown values on a field of correlated nuisances has been offered. Analytical results enabling to define the conditions of using the detection algorithms for elongated signals of unknown values instead of detection algorithms for elongated signals of the well-known values have been obtained with the given profit.

НОРМАЛИЗАЦИЯ ИЗОБРАЖЕНИЙ ОТ ГЕОСТАЦИОНАРНОЙ КОСМИЧЕСКОЙ СИСТЕМЫ НАБЛЮДЕНИЯ ЗЕМЛИ

Гектин Ю.М., к.т.н., ОАО «Российские космические системы», г. Москва

Еремеев В.В., д.т.н., директор НИИ «Фотон» РГРТУ, г. Рязань, e-mail: foton@rsreu.ru

Егошкин Н.А., к.т.н., с.н.с. НИИ «Фотон», РГРТУ, г. Рязань, e-mail: foton@rsreu.ru

Зенин В.А., м.н.с. НИИ «Фотон», РГРТУ, г. Рязань

Москатинов И.В., начальник центра ФГУП «НПО им. С.А. Лавочкина», г. Москва

Ключевые слова: радиометрическая коррекция, нормализация изображений, геодезическая привязка.

Введение

Геостационарные спутники ориентированы на периодическое (через каждые 5-30 минут) получение изображений всего диска Земли на фоне окружающего космоса, они «зависают» в плоскости экватора над фиксированной точкой Земли и наблюдают один и тот же участок земного диска. В настоящее время требования к детальности изображений (пространственное разрешение порядка 1 км) могут быть достигнуты только с помощью видеодатчиков, которые формируют изображения отдельными частями (сканами). Последние формируются либо одновременно несколькими приборами с зарядовой связью (ПЗС-линейками), либо последовательно одной ПЗС-линейкой. Изображения от «скановых» датчиков в непосредственном виде не пригодны для использования, им свойственны специфические искажения изображений, связанные с различными геометрическими и радиометрическими характеристиками сканов. Необходимо выполнить нормализацию изображений, геодезическую привязку и трансформирование в принятую во всём мире нормализованную геостационарную проекцию. Ниже рассматривается решение этих задач.

Геометрическая нормализация

Геометрическая нормализация состоит в формировании из исходного изображения нового с известными геометрическими характеристиками и представленного в заданной картографической проекции. Основу геометрической нормализации составляют математические модели процесса съёмки. Сканирующие датчики типа МСУ-ГС (разработчик – ОАО «Российские космические системы») включают несколько ПЗС-линеек и подвижное зеркало, осуществляющее развертку изображений.

Рассмотрим произвольную ПЗС-линейку датчика. Введем приборную систему координат (ПСК) так, что-бы её центр совпадал с фокусом O объектива, а ось Z – с оптической осью. Тогда направляющий вектор визирующего луча, падающего на зеркало, для n -го элемента линейки определится как

$$\mathbf{p}_n = -\left(x_0 + n\Delta_x, y_0 + n\Delta_y, f\right)^T, \quad (1)$$

где f – фокусное расстояние объектива, x_0, y_0 – координаты центрального элемента ПЗС-линейки, (Δ_x, Δ_y) – вектор, направленный в фокальной плоскости вдоль ПЗС-линейки.

Рассматриваются модели и алгоритмы геометрической и радиометрической нормализации изображений от геостационарных космических систем. Исследованы алгоритмы уточнения геодезической привязки изображений с использованием контурных точек диска Земли и электронных карт. Выполнена практическая апробация алгоритмов на снимках от космического аппарата «Электро-Л».

Направляющий вектор $\mathbf{r}$ отраженного от зеркала визирующего луча определяется из векторного уравнения как

$$\mathbf{r} = \mathbf{p} - 2(\mathbf{p}, \mathbf{n}) \mathbf{n}, \quad (2)$$

где $\mathbf{n} = (\cos \alpha, -\sin \beta \sin \alpha, \cos \beta \sin \alpha)^T$ – единичный вектор нормали к плоскости сканирующего зеркала в ПСК, определенный парой углов α, β . Углы α, β являются функциями номера строки m , которые либо задаются при проектировании привода развертки, либо непосредственно измеряются для каждой строки специальными датчиками.

Найдём координаты визирующего луча в гринвичской системе координат (ГСК):

$$\mathbf{r}_e = \mathbf{M}_e(t_m) \mathbf{M}_y \mathbf{r}, \quad t_m = t_0 + \Delta t m, \quad (3)$$

где $\mathbf{M}_y$ – матрица установочных углов датчика; $\mathbf{M}_e(t)$ – зависящая от времени матрица ориентации КА относительно ГСК, измеряемая навигационными средствами; t_0, t_m – соответственно время формирования первой и m -й строк; Δt – период опроса ПЗС-линейки.

После этого решается задача пересечения параметрически заданной прямой $\mathbf{P}(l) = \mathbf{R} + l \mathbf{r}_e$ с земным эллипсоидом $X^2/a_3^2 + Y^2/a_3^2 + Z^2/b_3^2 = 1$,

где $\mathbf{R} = (X_0, Y_0, Z_0)$ – вектор положения спутника в момент времени t_m по данным навигационных средств, a_3

и b_3 – экваториальный и полярный радиусы земного эллипсоида. В результате находятся координаты точки земной поверхности $\mathbf{P}_3 = (X_3, Y_3, Z_3)^T$.

Далее для точки с гринвичскими координатами (X_3, Y_3, Z_3) определяются географические координаты (λ, φ) :

$$\lambda = \arctg(Y_3/X_3), \quad \varphi = \arctg(Z_3 \frac{a_3^2}{b_3^2} / \sqrt{X_3^2 + Y_3^2}). \quad (5)$$

После этого по формулам, описывающим нормализованную геостационарную проекцию, находятся соответствующие (λ, φ) координаты (x, y) точки нормализованного изображения.

С использованием (1)-(5) осуществляется синтез из исходного изображения $B(m, n)$, полученного от съёмочной системы, нового изображения $D(x, y)$, представленного в

нормализованной геостационарной проекции с заданным размером пикселя. Обозначим полученные в (1)-(5) математические соотношения между (x, y) и (m, n) как $x = F(m, n)$, $y = G(m, n)$.

(6)

Тогда формирование изображения $D(x, y)$ может осуществляться по обратному закону [1] с использованием скоростных процедур численного обращения функций F, G . Соответствующие алгоритмы универсальны, однако они не учитывают ряд особенностей изображений, имеющих скановую структуру.

Во-первых, изображение $B(m, n)$ состоит из нескольких частей (сканов), имеющих небольшие перекрытия. Математически это выражается в том, что F, G не имеют однозначных обратных функций, т.е. некоторым (x, y) могут соответствовать несколько пар (m_i, n_i) . В этом случае наилучшее качество коррекции обеспечивается при взвешенном суммировании информации от нескольких сканов:

$$D(x, y) = \sum_{i, w_i > 0} w_i B(m_i, n_i) / \sum_{i, w_i > 0} w_i, \quad w_i = 1 - 2|n_i / N|, \quad (7)$$

где веса w_i убывают до нуля по мере приближения n_i к краям ПЗС-линейки, состоящей из N элементов.

Во-вторых, если ПЗС-линейка составная (включает несколько отдельных субпиксельно сдвинутых на μ ПЗС-линеек), то необходимо выполнить объединение информации с тем, чтобы повысить радиометрическое качество и пространственное разрешение. Для этого достаточно при расчёте весов учитывать расстояния пикселей до ближайших узлов дискретизации:

$$D(x, y) = \sum_{i, v_i > 0, w_i > 0} v_i w_i B(m_i, n_i^*) / \sum_{i, v_i > 0, w_i > 0} v_i w_i, \quad n_i^* = \text{ent}(n_i + 0,5), \quad v_i = 1 - |n_i - n_i^*| / \mu. \quad (8)$$

Основная сложность здесь состоит в определении для точки (x, y) нескольких прообразов (m_i, n_i) . Алгоритмы из [1] позволяют это сделать, однако требуют значительного количества памяти. Предложенные в [2] алгоритмы позволяют находить взвешенные суммы (8) более эффективно.

Геодезическая привязка снимков по диску Земли и электронным картам

В работе [3] рассмотрена технология геодезической привязки изображений от геостационарных спутников по контурным точкам диска Земли. Она основана на сегментации земного диска, контрастно отображающегося на фоне космоса. В процессе летных испытаний КА «Электро-Л» ряд положений, положенных в основу этих алгоритмов, пришлось пересмотреть. Так, из-за низкого радиометрического качества изображений пришлось отказаться от обработки исходных изображений. Обработка выполняется после трансформации изображения в специальную плоскость, при этом за счёт комплексирования информации от нескольких ПЗС-линеек радиометрическое качество изображений повышается. Однако даже после этого сегментация изображений на диск и космос оказалась затруднительной. Поэтому для поиска контурных точек диска Земли предложено использовать корреляционно-экстремальный алгоритм с использованием так называемых бинарных масок.

Вначале выполняется трансформация изображения в плоскость, в которой краевые точки диска Земли образуют эллипс. Чтобы найти такую плоскость, запишем условие прохождения визирующего луча датчика по

касательной к поверхности земного эллипсоида (4):

$$(\mathbf{N}, \mathbf{L}) = 0, \quad \mathbf{L} = (X - X_0, Y - Y_0, Z - Z_0)^T,$$

$$\mathbf{N} = (X / a_3^2, Y / a_3^2, Z / b_3^2)^T, \quad (9)$$

где $\mathbf{L}$ – направляющий вектор визирующего луча, $\mathbf{N}$ – вектор нормали к эллипсоиду (4), (X, Y, Z) – координаты точки Земли. После преобразований (9) получим

$$XX_0 / a_3^2 + YY_0 / a_3^2 + ZZ_0 / b_3^2 - 1 = 0. \quad (10)$$

Уравнение (10) определяет плоскость с задающим вектором

$$\mathbf{S} = (X_0, Y_0, Z_0 a_3^2 / b_3^2) / \sqrt{X_0^2 + Y_0^2 + Z_0^2 a_3^4 / b_3^4}.$$

Поскольку эллипсоид и любая плоскость пересекаются по эллипсу, то контурные точки изображения диска Земли находятся на эллипсе в плоскости (10).

Введём в рассмотрение вспомогательную плоскость, проходящую через точку $(0, 0, 0)$ параллельно плоскости (10):

$$XX_0 + YY_0 + ZZ_0 a_3^2 / b_3^2 = 0. \quad (11)$$

Перенесём эллипс из плоскости (10) в плоскость (11) путём проецирования из центра (X_0, Y_0, Z_0) . Анализируя (9)-(11), можно установить, что образ эллипса в плоскости (11) – также эллипс с центром $(0, 0, 0)$ и полуосями

$$a = a_3 \sqrt{\frac{X_0^2 b_3^2 + Y_0^2 b_3^2 + Z_0^2 a_3^2}{X_0^2 b_3^2 + Y_0^2 b_3^2 + Z_0^2 a_3^2 - a_3^2 b_3^2}}, \quad b = \sqrt{\frac{X_0^2 b_3^4 + Y_0^2 b_3^4 + Z_0^2 a_3^4}{X_0^2 b_3^2 + Y_0^2 b_3^2 + Z_0^2 a_3^2 - a_3^2 b_3^2}}. \quad (12)$$

Из соображений симметрии можно показать, что большая полуось этого эллипса направлена по вектору $\mathbf{A} = (-Y_0, X_0, 0) / \sqrt{X_0^2 + Y_0^2}$, а малая полуось перпендикулярна к ней и направлена по вектору $\mathbf{B} = \mathbf{S} \times \mathbf{A}$.

Введём систему координат X', Y', Z' так, чтобы её оси были направлены вдоль векторов $\mathbf{A}$, $\mathbf{B}$, $\mathbf{S}$ соответственно, а центр располагался в плоскости (11) в центре эллипса. В этой системе эллипс описывается уравнением $(X')^2 / a^2 + (Y')^2 / b^2 = 1$.

(13)

Таким образом, в идеальном случае (при отсутствии искажений) краевые точки диска Земли в системе координат X', Y' образуют эллипс (13) с известными параметрами (12). На основе анализа отклонения от эллипса (12) реальной геометрической фигуры контура диска оцениваются параметры геодезической привязки.

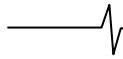
Необходимые для трансформации функции координатного соответствия между (m, n) и (X', Y') определим следующим образом. Вначале для точки (m, n) по (1)-(3) найдём вектор $\mathbf{r}_e$. Затем $\mathbf{r}_e$ и $\mathbf{R}$ пересчитываются в систему координат X', Y', Z' :

$$\mathbf{r}' = \mathbf{M}' \mathbf{r}_e, \quad \mathbf{R}' = \mathbf{M}' \mathbf{R}, \quad \mathbf{M} = [\mathbf{A} | \mathbf{B} | \mathbf{C}]^T. \quad (14)$$

После этого искомые плоскостные координаты (X', Y') образа краевой точки во вспомогательной плоскости определяются как

$$X' = R'_x - R'_z r'_x / r'_z, \quad Y' = R'_y - R'_z r'_y / r'_z \quad (15)$$

За счёт трансформации изображения во вспомогательную плоскость устраняются разрывы изображения из-за его



скановой структуры и повышается качество благодаря комплексированию данных от нескольких ПЗС-линеек. Это приводит к повышению точности и надёжности определения координат контурных точек диска Земли по сравнению с использованием исходных изображений [3].

Далее определяется набор контурных точек диска Земли. Вначале эталонный эллипс (13) разделяется на K секторов, для которых формируются эталонные фрагменты E_i , $i=1, K$, в которых -1 и +1 соответствует частям космоса и Земли, примыкающих к границе эллипса (13):

$$E_i(x, y) = \begin{cases} +1, & (x, y) \in \Theta(a - \Delta, a) \cap S_i, \\ -1, & (x, y) \in \Theta(a, a + \Delta) \cap S_i, \\ 0, & \text{иначе.} \end{cases} \quad (16)$$

$$\text{где } \Theta(\theta_n, \theta_s) = \left\{ (x, y) : \theta_n^2 < x^2 + y^2 \frac{a^2}{b^2} \leq \theta_s^2 \right\},$$

$$S_i = \left\{ (x, y) : \frac{2\pi i}{K} < \arctg \frac{y}{x} < 2\pi \frac{i+1}{K} \right\}.$$

Далее осуществляется корреляционно-экстремальный поиск эталонных фрагментов E_i , при этом центры зон поиска имеют координаты

$$P_i = \left(-a \cos 2\pi \frac{i+0.5}{K}, -b \sin 2\pi \frac{i+0.5}{K} \right), \quad i = \overline{1, K}. \text{ Поскольку}$$

эталонные фрагменты бинарные, то вычисление коэффициента корреляции (КК) требует только операций сложения и вычитания. Кроме того, изменение КК при смещениях E_i по зоне поиска связано только с граничными точками области нулей и единиц на E_i , что также кардинально сокращает объём вычислений. В результате формируется набор фактических координат Q_i , $i = \overline{1, K}$.

По координатам P_i и Q_i , $i = \overline{1, K}$, рассчитывается смещение центра наблюдаемого эллипса по отношению к эталону как взвешенная сумма поправок:

$$\Delta x = \sum_i (P_{ix} - Q_{ix}) P_{ix}^2 / \sum_i P_{ix}^2, \quad \Delta y = \sum_i (P_{iy} - Q_{iy}) P_{iy}^2 / \sum_i P_{iy}^2. \quad (17)$$

При этом веса учитывают, что точность корреляционного поиска наибольшая в направлении, перпендикулярном к границе диска.

Поскольку плоскость (12) перпендикулярна прямой, соединяющей спутник и центр Земли, с использованием простых геометрических преобразований находятся поправки к углам ориентации спутника α и ω :

$$\Delta \alpha = \arctg \Delta x / |R|, \quad \Delta \omega = \arctg \Delta y / |R|. \quad (18)$$

Эти поправки вводятся в матрицу установочных углов M_y при последующей геометрической нормализации.

В ходе лётных испытаний спутника «Электро-Л» показано, что рассмотренный подход обеспечивает точность геопривязки изображений инфракрасного (ИК) диапазона порядка 2 км (при пространственном разрешении съёмки 4 км). В видимом диапазоне (ВД) использование диска Земли нецелесообразно, так как большую часть суток отображается только часть диска. В этом случае геопривязка осуществляется с использованием электронных карт [3], при этом достигается точность привязки 0,15 км. Технологии геопривязки по электронным картам не претерпели принципиальных изменений по сравнению с [3], поскольку они инвариантны к типу датчика.

Радиометрическая нормализация

В работе [4] предложены алгоритмы статистиче-

ской яркостной нормализации космических изображений, получаемых с помощью видеодатчиков, построенных на основе ПЗС. В основе алгоритмов лежит сопоставление статистических характеристик корректируемых видеоданных (корректируемого столбца) от определенного ПЗС-элемента с эталонными характеристиками. Особенностью таких алгоритмов коррекции является то, что параметры искажений в каждом столбце изображения считаются постоянными в течение всего времени съёмки.

Во время лётных испытаний космических систем «Метеор-М» и «Электро-Л» установлено, что параметры ПЗС-элементов могут изменяться непосредственно во время формирования изображения. Это явление аналогично изменению чувствительности человеческого глаза при изменении яркости наблюдаемой сцены. Так как изображение поверхности Земли представляет собой случайный сигнал, то описать изменения искажений во времени аналитически не представляется возможным. Ниже представлено развитие алгоритмов, изложенных в работах [4, 5].

Проведён анализ нескольких полиномиальных моделей времязависимых структурных искажений.

Мультипликативная модель. Определение параметров искажений выполняется по МНК из условия

$$R_n = \sum_{m=1}^M \sum_{i=-S}^S [B_{mn} P_n(m) - B_{m,n+i}]^2 = \min, \quad \text{где } B_{mn} - \text{элементы}$$

изображения, $P_n(m)$ – искомый полином, отражающий изменение чувствительности вдоль строки. В данном случае коррекции времязависимых структурных искажений выполняется по формуле

$$D_{mn} = P_n(m) B_{mn}, \quad m = \overline{1, M}, n = \overline{1, N}.$$

Аддитивная модель.

Оценка параметров искажений осуществляет по МНК:

$$R_n = \sum_{m=1}^M \sum_{i=-S}^S [B_{mn} + P_n(m) - B_{m,n+i}]^2 = \min.$$

Найденные коэффициенты полинома позволяют корректировать изображение по формуле

$$D_{mn} = P_n(m) + B_{mn}, \quad m = \overline{1, M}, n = \overline{1, N}.$$

Полиномиальные модели обладают математической строгостью. Экспериментально подтверждено, что реальные искажения можно описать только полиномами высоких степеней. Это неизбежно приводит к нарастанию ошибок. Более высокую точность имеют кусочно-линейные модели, согласно которым корректируемый элемент изображения сопоставляется с данными от окружающей апертуры из $2S+1$ столбцов и $2V+1$ строк.

Кусочно-линейная мультипликативная модель.

Корректирующий коэффициент для каждого элемента изображения определяется как

$$k_{nm} = \frac{1}{2S+1} \sum_{q=-S}^S \sqrt{\mu_q / \mu_0}, \quad (19)$$

где μ_q – значение автоковариационной функции (АКФ), найденное по столбцу с номером q .

Кусочно-линейная аддитивная модель. Для корректируемого столбца апертуры и для всей апертуры, состоящей из $2S+1$ соседних столбцов, формируются эмпирические функции распределения кодов яркости $F_0(b)$ и $F(b)$. Для них в табличной форме вычисляются обратные функции

$$F_{0p}(b) \rightarrow b_{0p} \quad \text{и} \quad F_p(b) \rightarrow b_p, \quad \text{где } F_{0p}(b) = F_p(b) = p / (P+1),$$

$p = \overline{1, P}$. После чего формируется корректирующая добавка для центрального пикселя апертуры $\Delta_{nm} = \text{med}_{p \in \{1, P\}} (b_p - b_{0p})$. В результате коррекция яркости выполняется как:

$$B_{nm}^* = \text{ent}(B_{nm} + \Delta_{nm} + 0,5), \quad m = \overline{1, M}, \quad n = \overline{1, N}.$$

Анализ изображений со спутника «Электро-Л» показал, что на снимках присутствуют времязависимые изменения характеристик не только отдельных фотоприемников, но и групповые искажения, относящиеся к группе столбцов. Для устранения подобных искажений предлагается адаптивный алгоритм, который корректирует исходное изображение $B(m, n)$ следующим образом:

– с использованием кусочно-линейной аддитивной модели формируется матрица поправок $H(m, n)$ при малом количестве строк в апертуре;

– по кусочно-линейной аддитивной модели находится матрица поправок $L(m, n)$ при числе строк в апертуре, в разы превышающие ширину групповых искажений;

– для каждого пикселя изображения $B(m, n)$ вычисляется значение дисперсии кодов яркости окружающих его пикселей $\sigma^2(m, n)$, а также определяется максимальная дисперсия

$$\sigma_{\max}^2 = \max(\sigma^2(m, n), m = \overline{1, M}, n = \overline{1, N});$$

– окончательная коррекция изображения выполняется согласно формуле:

$$d_{nm} = b_{nm} + \frac{\sigma^2(m, n)}{\sigma_{\max}^2} H(m, n) + \left(1 - \frac{\sigma^2(m, n)}{\sigma_{\max}^2}\right) L(m, n),$$

$$m = \overline{1, M}, \quad n = \overline{1, N}.$$

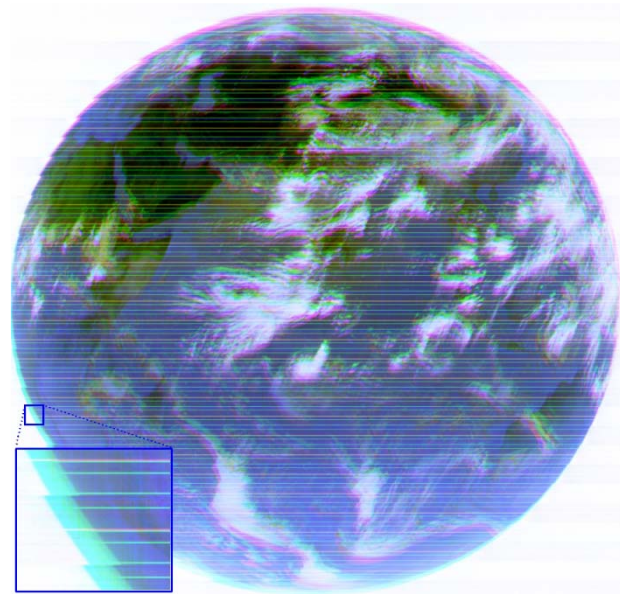
Заключение

Разработаны алгоритмы геометрической и радиометрической нормализации информации от геостационарных систем наблюдения Земли. Они прошли экспериментальную апробацию в ходе лётно-космических испытаний отечественного геостационарного спутника «Электро-Л». В результате удалось добиться выполнения высоких требований к качеству нормализованной информации при существенных искажениях в исходной. Выполненные оценки точности нормализации показали, что геометрическая нормализация выполняется с точностью до долей пикселя, а относительная радиометрическая точность составляет 0,2%.

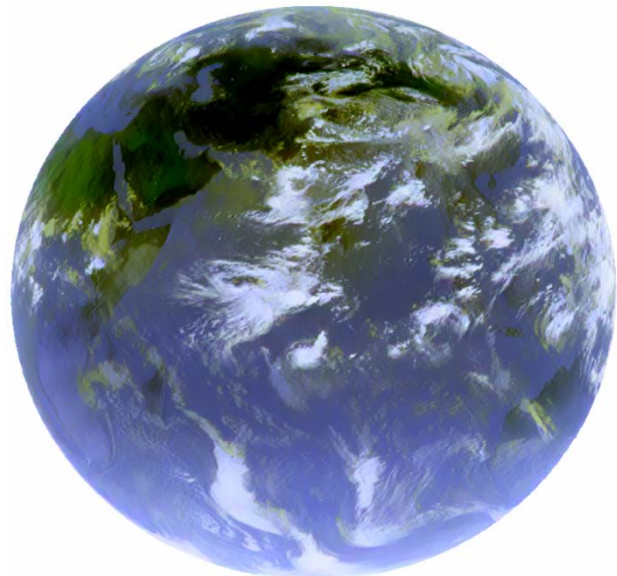
На рис. 1 показан пример радиометрической и геометрической нормализации изображения со спутника «Электро-Л» в трёх инфракрасных каналах. Сверху исходный снимок, на котором можно видеть структурный радиометрический шум в виде горизонтальной «полосатости», цветные муары из-за геометрических рассогласований между спектральными каналами и скановую структуру изображения. Снизу результат нормализации. На обработку данных одного сеанса съёмки поверхности Земли спутником «Электро-Л» в 10 спектральных каналах с общим объёмом видеoinформации 1,5 Гб затрачивается не более 5 минут при использовании сервера IBM x3850 с 4-я двудерными процессорами Intel Xeon 7120N 4x3.0GHz / 667 MHz.

Литература

1. Злобин В.К., Еремеев В.В., Обработка аэрокосмических изображений. М. Физматлит, 2006. С. 288.
2. Воронин А.А., Егоскин Н.А., Еремеев В.В., Москатинев И.В. Геометрическая обработка данных от космических систем глобального наблюдения Земли // Вестник РГРТУ. Вып. 27, Рязань, 2010. №1. 2009. – С. 12-17.
3. Егоскин Н.А., Еремеев В.В., Козлов Е.П. Нормализация космических изображений Земли на основе их сопоставления с электронными картами // Цифровая обработка сигналов №3. 2009. – С. 21-26.



а)



б)

Рис. 1. Изображения со спутника «Электро-Л» до и после нормализации

4. Еремеев В.В., Зенин В.А. Статистические модели коррекции структурных искажений на спутниковых изображениях земной поверхности // Цифровая обработка сигналов. Вып. 3. 2010. – С. 30-36.
5. Еремеев В.В., Зенин В.А. Модели коррекции динамических структурных искажений на космических изображениях // Вестник РГРТУ. Вып. 33, Рязань, 2010. С. 3-7.

NORMALIZATION OF IMAGES FROM GEOSTATIONARY SPACE EARTH OBSERVATION SYSTEM

Gektin U.M., Eremeev V.V., Egoshkin N.A., Zenin V.A., Moskatinyev I.V.

The models and algorithms for geometric and radiometric normalization of images from geostationary space systems. Investigated algorithms refine geodetic reference image with the use of contour points on the disk of the Earth and electronic cards. Carried out practical testing of the algorithms on images from the spacecraft, «Electro-L».

СЕГМЕНТАЦИЯ ОБЛАЧНЫХ ОБЪЕКТОВ НА ПАНХРОМАТИЧЕСКИХ ИЗОБРАЖЕНИЯХ ЗЕМНОЙ ПОВЕРХНОСТИ

Ветров А.А., аспирант, инженер НИИ «Фотон» РГРТУ, e-mail: foton@rsreu.ru

Кузнецов А.Е., д.т.н., заместитель директора НИИ «Фотон» РГРТУ, e-mail: foton@rsreu.ru

Ключевые слова: сегментация облачности, алгоритмы кластерного анализа, коррекция фотометрических искажений.

Введение

В системах дистанционного зондирования Земли (ДЗЗ) облачности отводится двоякая роль. Так, в гидрометеорологии облачные образования, наблюдаемые на спутниковых снимках являются объектом изучения при определении погоды и климата планеты. Что касается систем ДЗЗ, предназначенных для картографирования земной поверхности, то для них облака являются мешающим фактором, поскольку закрывают наблюдаемую территорию, а участки, попадающие в тень от облаков, отображаются на изображениях с пониженной яркостью. Всё это затрудняет выполнение тематической обработки и оперативного дешифрирования материалов съёмки.

Задача выделения облачности возникает на этапе предварительной обработки спутниковых изображений с целью:

- оценки качества видеоматериалов по критерию отношения площади покрытой облачностью к площади снимка;

- исключения участков снимка, содержащих облака, при радиометрической коррекции, классификации объектов, построении ортопланов по множеству разновременных изображений и др.

В случае если выполняется спектрозональная или гиперспектральная съёмка, процедура распознавания и выделения облачности на многоканальных снимках упрощается за счёт использования колориметрических или спектральных характеристик наблюдаемых объектов [1]. При панхроматической (одноканальной) съёмке подобные методы неприменимы, и распознавать облачные образования приходится на основе яркостных критериев. Поскольку известные алгоритмы пороговой сегментации не всегда позволяют получить хорошие результаты, то актуальными становятся исследования, направленные на достижение высокой скорости и надёжности процедуры выделения облачности.

В настоящей работе предлагается комплексное решение задачи предварительной обработки панхроматических изображений, связанное не только с эффективной сегментацией облачных объектов, но и восстановлением яркостей затенённых участков снимка.

Алгоритмы сегментации облачности

Алгоритм выделения облачных объектов на изображении восстановлением яркостей

Рассматриваются пороговые и кластерные алгоритмы сегментации облачных объектов на панхроматических снимках высокого разрешения. Предложен модифицированный алгоритм k-средних, позволяющий достичь высокого качества сегментации облачных образований, и алгоритм улучшения визуальных характеристик изображений. Выполнено апробирование разработанных алгоритмов на снимках от КА «Ресурс-ДК».

$B = \{b_{mn}\}$, $m = \overline{1, M}$, $n = \overline{1, N}$, где b_{mn} – яркость пикселя с координатами (m, n) , $b_{mn} \in [0, b_{\max}]$, сводится к разделению его на два класса объектов B_1 и B_2 , так, что $B = B_1 \cup B_2$, $B_1 \cap B_2 = \emptyset$.

Класс B_1 состоит всего из одного объекта, включающего пиксели b_{mn} , которые не принадлежат облачности. К классу B_2 относятся фрагменты снимка

B_{2i} , $B_2 = \{B_{2i}\}$, $i = 1, 2, \dots$, с изображениями облачных образований.

Поскольку облачные образования на снимке выглядят более ярко (рис. 1), то выполним их сегментацию с использованием порогового критерия: пиксель b_{mn} принадлежит облачному объекту, если

$b_{mn} \geq b_0$, иначе $b_{mn} \in B_1$, где b_0 – пороговое значение яркости пикселей.

Пороговое значение b_0 определим с использованием метода максимизации межклассовой дисперсии [2], в соответствии с которым

$$b_0 = \max \sigma_B^2(b), b = 0, b_{\max}.$$

Значение межклассовой дисперсии $\sigma_B^2(b)$ для кода яркости b определяется выражением

$$\sigma_B^2(b) = \frac{[\bar{b}p(b) - \bar{b}_1]^2}{p(b)[1 - p(b)]},$$

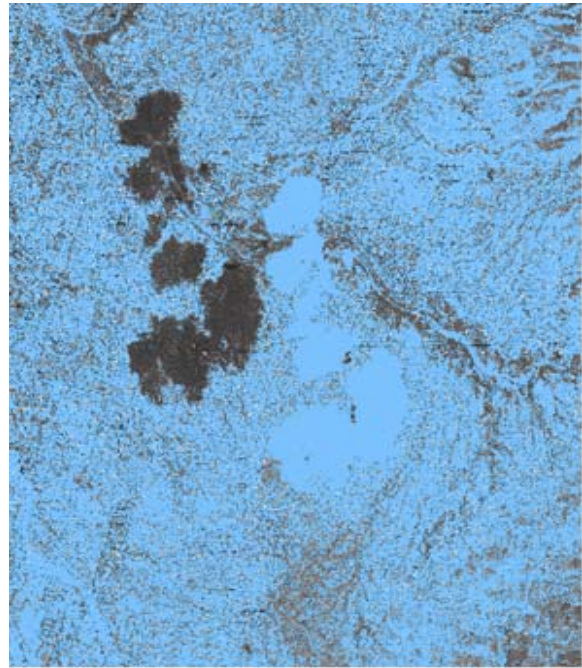
где $\bar{b} = \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N b_{mn}$; $\bar{b}_1$ – средняя яркость пикселей,

для которых $b_{mn} > b$; $p(b)$ – вероятность того, что для случайного пикселя изображения $b_{mn} < b$,

$p(b) = \frac{1}{MN} \sum_{i=1}^{b-1} h(i)$, где $h(i)$ – количество пикселей изображения, для которых $b_{mn} = i$.



а)



б)

Рис. 1. Изображение земной поверхности с облачными образованиями. (а) – до обработки, (б) – после обработки пороговым алгоритмом (облачность подсвечена синим цветом)

Для исследования порогового алгоритма сегментации облачности были использованы 4 снимка земной поверхности от КА «Ресурс-ДК». Первый сюжет, приведенный на рис. 1, характеризуется наличием яркой и темной подстилающей поверхности и небольшим количеством облачных объектов B_{1i} , незначительно отличающихся по яркости от яркой подстилающей поверхности. На остальных изображениях присутствует подстилающая поверхность широкого яркостного диапазона и облачные образования различной яркости.

Поскольку на снимке человек без труда распознаёт облачные объекты, то качество работы алгоритмов сегментации будем сравнивать с ручным вариантом выделения облачности.

Обозначим через $B_2^* = \{B_{2i}^*\}, i = 1, I^*$ множество выделенных вручную фрагментов снимка, содержащих изображения облачных объектов. Множеством $B_2^{\Pi} = \{B_{2i}^{\Pi}\}, i = 1, I^{\Pi}$ опишем облачные объекты, выделенные с помощью порогового алгоритма сегментации. Количество объектов, ложно отнесённых к облачности, обозначим переменной $I_O^{\Pi}, I_O^{\Pi} < I^{\Pi}$, а переменной I_{Π}^{Π} – число нераспознанных облачных образований, $I_{\Pi}^{\Pi} < I^*$. Общую площадь снимка, занимаемую объектами множества B_2^* обозначим через S^* , а площадь, занимаемую объектами множества B_2^{Π} , обозначим через S^{Π} .

Тогда эффективность алгоритма сегментации можно оценить по следующим показателям:

– проценту ложно пропущенных облачных образований,

$$\rho_{\Pi} = \frac{I_O^{\Pi}}{I^*} \cdot 100\%;$$

– проценту ложно обнаруженных облачных объектов,

$$\rho_{\Pi} = \frac{I_O^{\Pi}}{I^*} \cdot 100\%;$$

– проценту общей площади выделенных облаков по отношению к истинному значению, $S = \frac{S^{\Pi}}{S^*} \cdot 100\%$.

В таблицах 1 и 2 приведены значения критериев качества сегментации для четырёх тестовых изображений. Как следует из таблиц, для порогового алгоритма доля ложно обнаруженных облачных объектов составляет значительную величину на каждом из снимков. К основным причинам такого явления относятся:

– неточное определение порога из-за размытости пика функции межклассовой дисперсии $\sigma_B^2(b)$;

– присутствие различных типов облачности, т.е. наличие нескольких классов облачных объектов;

– отнесение к облачности небольших объектов земной поверхности, например, крыш домов и других инженерных сооружений.

Таблица. 1.

Показатели эффективности алгоритмов сегментации по количеству выделенных областей

Номер снимка	Пороговый алгоритм		Алгоритм к-средних		Алгоритм к-средних с обучением	
	$\rho_{\Pi}, \%$	$\rho_{\Pi}, \%$	$\rho_{\Pi}, \%$	$\rho_{\Pi}, \%$	$\rho_{\Pi}, \%$	$\rho_{\Pi}, \%$
1	80,0	1400,0	60,0	720,0	40,0	40,0
2	8,4	16,7	8,3	8,3	4,2	8,4
3	100,0	2900,0	33,3	700,0	33,3	66,7
4	50,0	650,0	50,0	500,0	0,0	0,0

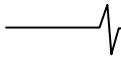


Таблица. 2.
Показатели эффективности алгоритмов сегментации по критерию общей площади выделенной облачности

Номер снимка	Пороговый алгоритм	Алгоритм k-средних	Алгоритм k-средних с обучением
	$S, \%$	$S, \%$	$S, \%$
1	529,38	399,57	110,26
2	42,95	53,56	84,05
3	110,68	80,20	91,08
4	84,77	69,08	119,17

Для устранения перечисленных недостатков будем:

- во-первых, использовать метод k-средних для выделения нескольких классов облачности;
- во-вторых, выполним фильтрацию полученных результатов сегментации по геометрическим размерам для исключения мелких объектов, не характерных для облачности.

В соответствии с методом k-средних, выделим на изображении три класса объектов: O_1 – не облачность, O_2 – вероятно, не облачность и O_3 – облачность. Каждый из классов включает в себя I_k объектов соответствующего типа, $O_k = \{O_{ki}\}, i = \overline{1, I_k}, k = \overline{1, 3}$. Для отнесения пикселя b_{mn} к объекту k-го класса будем использовать функционал, $k = \operatorname{argmin}_k |b_{mn} - \bar{b}_k|, k = \overline{1, 3}$, (1)

Если $\Delta \geq \Delta_0$, где Δ_0 – пороговое значение, то обновим параметры $\bar{b}_k = \bar{b}_k^*, k = \overline{1, 3}$ и вновь выполним кластеризацию изображения B .

После окончания итерационного процесса облачные объекты представим классом $O_3, B_2 = O_3$,

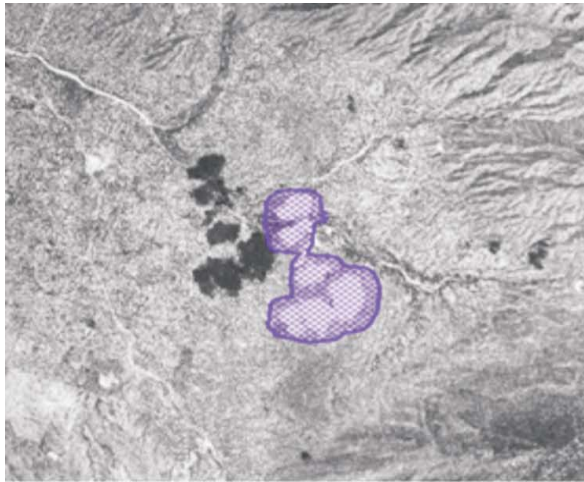
а остальные объекты отнесём к классу $B_1, B_1 = O_1 \cup O_2$.

Для завершения процедуры сегментации отбросим все облачные объекты, площадью меньше порогового значения S_0 , а остальные объекты векторизуем.

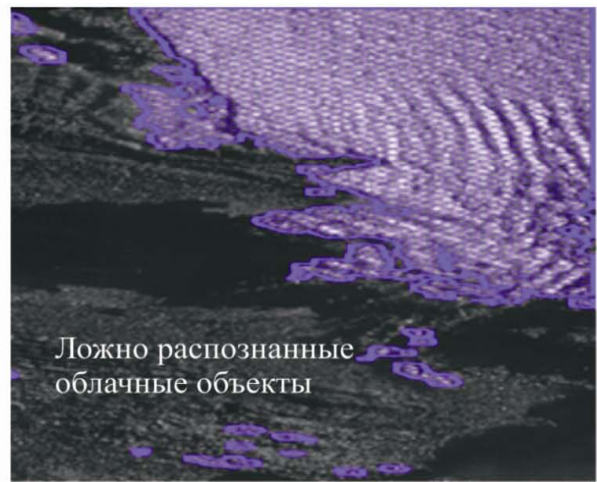
По показателям качества, представленным в таблицах 1 и 2, видно, что метод k-средних более эффективен по сравнению с пороговым алгоритмом выделения облачности. Однако он обладает крайне низким быстродействием и при обработке изображений высокого пространственного разрешения временные затраты становятся недопустимо велики. Кроме того, как показали эксперименты, качество сегментации во многом зависит от успешного задания начальных параметров $\bar{b}_k$. Поэтому предлагается:

- во-первых, для сокращения временных затрат и более точных настроек алгоритма использовать процедуру обучения;
- во-вторых, учитывать статистические характеристики облачных объектов для их более надёжного распознавания.

Под набором обучающих изображений будем пони-



а)



б)

Рис. 2. Первое (а) и третье (б) контрольные изображения, отсегментированные методом k-средних с обучением

где $\bar{b}_k$ – средняя яркость пикселей k-го класса объектов.

Зададим начальные значения $\bar{b}_k$,

$\bar{b}_1 = 0, \bar{b}_2 = 0,5b_{\max}, \bar{b}_3 = b_{\max}$ и выполним кластеризацию изображения B с использованием функционала (1). Затем определим новые значения средних яркостей $\bar{b}_k^*, \bar{b}_k^* = \frac{1}{Q_k} \sum_{b_{mn} \in B_{ki}} b_{mn}$, где Q_k – количество пикселей,

отнесенных к k-му классу объектов, и найдём максималь-

ное изменение параметров кластеров $\Delta, \Delta = \max_k \left| \bar{b}_k^* - \bar{b}_k \right|$.

мать одномерные массивы $D_v = \{d_{vj}\}, j = \overline{1, J_v}$, где v – номер обучающего изображения, состоящего из пикселей изображения B ; J_v – число элементов массива с яркостями d_{vj} . Формирование обучающих изображений происходит интерактивно путём задания отрезка, начало которого принадлежит облачному объекту, а конец – не закрытой облаками части снимка. При этом обучающие изображения будут содержать 4 класса объектов: O_1 – достоверно не облачность, O_2 и O_3 – объекты с пикселями, не принадлежащими к облачности с меньшей достоверностью, O_4 – облачность. Так как облачные

объекты обладают существенно меньшей энтропией, чем подстилающая поверхность, то учтём это свойство в функционале, используемом при кластеризации,

$$k = \arg \min_k \left[\left| \bar{b}_{mn} - \bar{b}_k \right| + c \left| e_{mn} - \bar{e}_k \right| \right], k = \overline{1, 4}, \quad (2)$$

где $\bar{b}_{mn}$ – средняя яркость, определяемая в пределах окна размером 3×3 с координатами центра (m, n) ; e_{mn} – энтропия для квадратного окна со стороной $p, p = 21$, и центром в точке (m, n) , рассчитываемая по формуле

$$e_{mn} = \frac{1}{p^2} \sum_{b=0}^{b_{\max}} h_b (\log_2(h_b) - \log_2(p^2)),$$

где h_b – количество пикселей с кодом яркости b ; $\bar{e}_k$ – средняя энтропия пикселей k -го класса объектов; c – масштабный коэффициент. Использование в выражении (2) $\bar{b}_{mn}$ вместо b_{mn} снижает чувствительность алгоритма к ярким одиночным пикселям.

Зададим начальные параметры для работы алгоритма сегментации следующим образом:

$$\bar{b}_1 = d_{w_v}, \bar{b}_2 = 0, 7\bar{b}_1 + 0, 3\bar{b}_4, \bar{b}_3 = 0, 5\bar{b}_1 + 0, 5\bar{b}_4, \bar{b}_4 = d_{v1},$$

$$\bar{e}_1 = e_{w_v}, \bar{e}_2 = 0, 7\bar{e}_1 + 0, 3\bar{e}_4, \bar{e}_3 = 0, 3\bar{e}_1 + 0, 7\bar{e}_4, \bar{e}_4 = e_{w_v}, c = \frac{b_{\max}}{10},$$

где e_{w_v} – энтропия, рассчитанная по окружению j -го пикселя объектов изображения D_v . Затем, используя метод k -средних, определим $\bar{b}_{kv}$ и $\bar{e}_{kv}$ для всех обучающих изображений, и общие $\bar{b}_k$ и $\bar{e}_k$ в

площади выделенной облачности S (см. рис. 2,б).

Сегментация затенённых участков снимка

Как было отмечено во введении, затенённые облачностью участки снижают не только визуальное качество спутникового изображения, но и оказывают мешающее действие при выполнении последующих обработок видеоданных. Непосредственное обнаружение таких участков весьма затруднительно, поскольку они не обладают уникальными фотометрическими характеристиками. Однако, после распознавания облачности для известных значений высоты и азимута Солнца, появляется возможность решения поставленной задачи.

Будем считать, что изображение B геокодировано, т.е. для каждого пикселя с координатами (m, n) известны его геодезические координаты $\varphi(m, n)$, $\lambda(m, n)$ и время $t(m, n)$. На основе этих значений, при помощи методических указаний, приведенных в [3], вычислим азимутальный угол α_{az} и высоту Солнца – α_o . Азимутальный угол α_{az} позволяет определить относительно планарных координат снимка (m, n) направление, задающее положение отбрасываемой облаком тени. Параметры направляющего вектора зададим координатными смещениями Δm и Δn , а максимальное смещение тени относительно облачности рассчитаем как $r = \frac{\mu H}{\operatorname{tg} \alpha_o}$, где μ – масштаб снимка, H – максимальное значение высоты облачности ($H \leq 15000 \text{ м}$).

Тогда смещение центра облачного объекта B_{2i} относительно его тени $(\Delta m^*, \Delta n^*)$ определим как $\Delta m^* = j \Delta m, \Delta n^* = j \Delta n$,

$$j = \arg \min_j \frac{\frac{1}{Q_k} \sum_{(m, n) \in B_{2i}} b(m + j \Delta m, n + j \Delta n) b(m, n) - \bar{b}(m, n) \bar{b}(m + j \Delta m, n + j \Delta n)}{\sigma(m, n) \sigma(m + j \Delta m, n + j \Delta n)},$$

$$\text{форме: } \bar{b}_k = \frac{1}{V} \sum_{v=1, V} \bar{b}_{kv}, \bar{e}_k = \frac{1}{V} \sum_{v=1, V} \bar{e}_{kv}.$$

Рассчитав $\bar{b}_k$ и $\bar{e}_k$ с использованием функционала (2), разделим изображение B на 4 класса объектов. Облачные объекты будут представлены классом $O_4, B_2 = O_4$, а остальные объекты отнесём к классу $B_1, B_1 = O_1 \cup O_2 \cup O_3$.

Отфильтруем по критерию площади объекты класса B_2 и выполним их векторизацию для получения контурного описания облачных объектов снимка.

По показателям качества, представленным в таблицах 1 и 2, видно, что алгоритм k -средних с обучением значительно более эффективен, чем классический алгоритм. Высокие значения ρ_{Π} и ρ_{Δ} при S , близкой к 100%, на изображениях 1 и 3 объясняются особенностями сюжета. На этих снимках присутствует малое количество крупных и небольших облачных образований, вследствие чего ложное обнаружение небольшого количества облачных образований малой площади резко увеличивает значения ρ_{Π} и ρ_{Δ} , фактически не сказываясь на общей

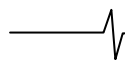
$$j = 1, 2, \dots, J, J = \frac{r}{\sqrt{\Delta m^2 + \Delta n^2}},$$

где $\bar{b}(m, n), \sigma(m, n)$ – средняя яркость и СКО пикселей облачного региона B_{2i} ; $\bar{b}(m + j \Delta m, n + j \Delta n)$,

$\sigma(m + j \Delta m, n + j \Delta n)$ – средняя яркость и СКО пикселей, попадающих в тень от облачного объекта.

Переместив контур облачного объекта B_{2i} вдоль направляющего вектора на величину $\Delta m^*, \Delta n^*$, определим затенённую область изображения B , автоматически выполнив таким образом процедуру сегментации тёмного участка снимка.

Используя результаты сегментации, улучшим визуальные характеристики изображения B путём локального затемнённых участков снимка. Отнесём контрастирования пиксели, принадлежащие к затенённой области к классу B_0 , а пиксели, удалённые не более чем на δ от её внешних границ к классу B_0^* и найдём коэффициенты

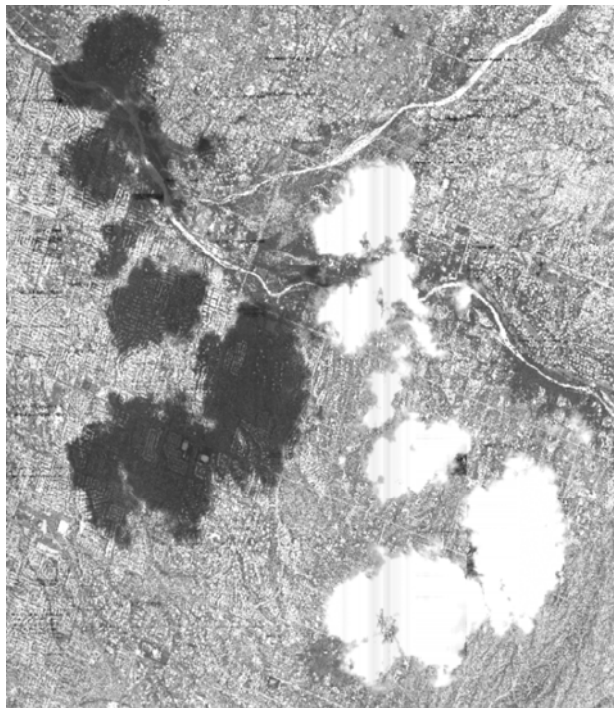


яркостного преобразования вида $b_{mn}^* = a_0 + a_1 b_{mn}$, $b_{mn} \in B_0$, где b_{mn}^* – яркость скорректированного пикселя затенённого участка снимка.

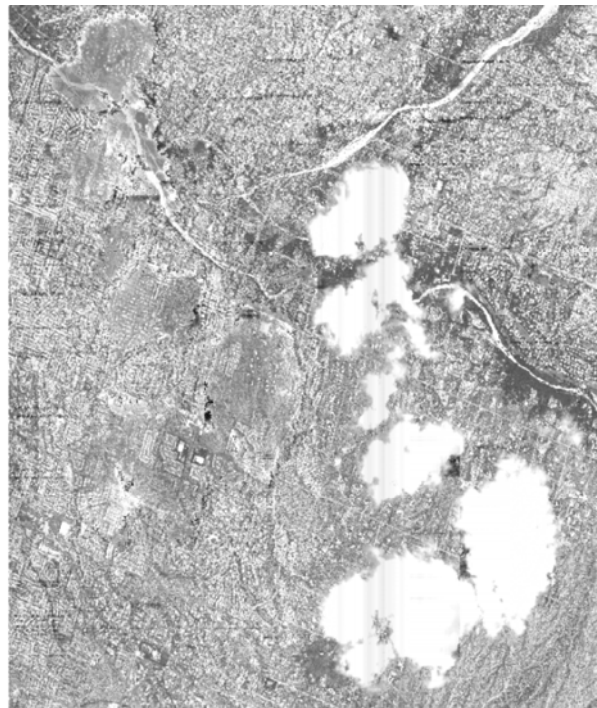
Значение коэффициентов a_1 и a_0 определим по формулам: $a_1 = \frac{\sigma_0^*}{\sigma_0}$, $a_0 = \bar{b}_0^* - a_1 \bar{b}_0$, где $\bar{b}_0, \sigma_0$ – средняя

яркость и СКО пикселей, принадлежащих объектам класса B_0 ; $\bar{b}_0^*, \sigma_0^*$ – средняя яркость и СКО пикселей, принадлежащих объектам класса B_0^* .

Результаты восстановления яркостной однородности изображения показаны на рис. 3.



а)



б)

Рис. 3. Пример участка снимка с затенённой областью до (а) и после яркостной коррекции (б)

Заключение

Практическое использование рассмотренного алгоритма сегментации на большом количестве спутниковых изображений от КА «Ресурс-ДК» и «Метеор-М» показало его высокую эффективность, как с точки зрения высокой надёжности распознавания различных облачных объектов, так и с точки зрения временных затрат. При этом работа с наличием простой процедуры обучения не вызывает затруднений у операторов программных комплексов, что очень важно при выполнении поточной обработки больших массивов видеоинформации от КА серий «Ресурс», «Канопус» и др.

Литература

1. Р. Гонсалес, Р. Вудс. Цифровая обработка изображений. М.: Техносфера, 2005. – С. 1072.

2. Otsu, N., «A Threshold Selection Method from Gray-Level Histograms» IEEE Transactions on Systems, Man, and Cybernetics, Vol. 9, No. 1, 1979, 62-66 pp.
3. Астрономический ежегодник на 2011 год. СПб.: Наука, 2010 – С. 690.

CLOUD SEGMENTATION OF PANCHROMATIC HIGH RESOLUTION SPACE IMAGES OF EARTH SURFACE

A.A. Vetrov, A.E. Kuznetsov

The paper discusses the threshold and clustering algorithms of cloud objects segmentation on high-resolution panchromatic images. Proposed modification of k-means algorithm, allowing to achieve high quality of cloud formations. Proposed algorithm, improving visual characteristics of images. Proposed algorithms tested on images of «Resurs-DK» spacecraft.

ПРИНЦИПЫ ПОСТРОЕНИЯ ГЕОПОРТАЛА СПУТНИКОВЫХ ИЗОБРАЖЕНИЙ, РАЗРАБАТЫВАЕМОГО В РАМКАХ ПРОЕКТА MEDEO

Ветров А.А., аспирант, инженер НИИ «Фотон» РГРТУ, foton@rsreu.ru

Кочергин А.М., к.т.н., с.н.с. НИИ «Фотон» РГРТУ, foton@rsreu.ru

Кузнецов А.Е., д.т.н., заместитель директора НИИ «Фотон» РГРТУ

Ключевые слова: дистанционное зондирование земли, проект MEDEO, электронный каталог, геоportal.

Рассматриваются принципы построения и функционирования геоportала спутниковых изображений, а также применение 3D-технологий при отображении данных ДЗЗ.

Введение

Рынок услуг дистанционного зондирования Земли (ДЗЗ) стремительно развивается и становится одной из движущих сил инновационного развития в России и странах Евросоюза. В тоже время, различия в форматах данных, полученных с космических аппаратов (КА) и ограниченные возможности доступа потребителей разных стран к спутниковым снимкам друг друга делают обмен данными ДЗЗ сложной технической задачей. С целью устранения разрыва между наличием ценных данных ДЗЗ и их практическим использованием для различных экономических и социальных целей, путем устранения технических барьеров на пути циркуляции космических снимков между поставщиками и потребителями спутниковой информации, в России и странах ЕС был создан проект MEDEO. Участниками проекта стали Технический университет Берлина, ECM Office, Aratos Technologies S.A. со стороны ЕС и ЦСКБ «Прогресс», РГРТУ со стороны РФ.

Одной из основных целей проекта является разработка двуязычного (русский и английский языки) Internet-портала доступа российских и европейских потребителей к электронному каталогу спутниковых снимков, полученных с КА «Ресурс-ДК». Портал должен иметь «дружественный» пользовательский интерфейс, а его функциональные возможности обеспечивать поиск информации в базе данных космических изображений по заданным критериям (таким как тип спутника и съёмочного устройства, географический регион интереса, временной интервал съёмки, параметры качества), отображение выбранных данных и формирование заявки на приобретение необходимой видеoinформации.

Архитектура Internet-портала

Проведенные исследования современных методов построения пользовательских интерфейсов, а также анализ преимуществ и недостатков существующих Internet-сайтов и приложений, связанных с интерактивным взаимодействием пользователя с набором данных ДЗЗ (таких как eportal.org, Google Earth и др.), показали, что функционально полным и эффективным решением для пользователей является предоставление им возможности навигации по трехмерной модели земного шара, с возможностью отображения спутниковых снимков на его поверхности. С учётом этого было предложено разрабатывать программу не как отдельное приложение (т.к. подобный подход потребовал бы, во-первых, создания разных версий под все существующие разновидности операционных систем, а, во-вторых, вызвал бы определенные неудобства пользователя, связанные с необходимостью скачивания и установки приложения), а как неотъемлемую часть WEB-страницы. Кроме того, для повышения нагляд-

ности и удобства работы с программой, 3D-модель Земли должна содержать текстурное изображение поверхности всего земного шара, линии географической сетки с подписями широт и долгот, контура различных объектов земной поверхности (таких как береговые линии, гидросеть, дороги и др.). Подобные требования резко увеличивают количество необходимых для построения модели вычислений, и, как следствие, возникает необходимость в использовании аппаратных средств формирования 3D-изображений.

Выполненный анализ современных средств разработки Internet-приложений показал, что оптимальным вариантом для решения поставленной задачи будет использование языка программирования Java. Это позволит реализовать Internet-портал в виде Java-апплета, встроенного в web-страницу сайта, а использование библиотеки JOGL (Java OpenGL) позволит приложению организовать расчёт трехмерных сцен с помощью аппаратного ускорителя 3D-графики компьютера пользователя.

Применение подхода, при котором большая часть необходимых расчётов выполняется на вычислительных средствах пользователя, позволило существенно сократить нагрузку на сервер, оставив в его юрисдикции только вопросы, связанные с передачей данных через Internet, отправку e-mail сообщений и учёт пользователей. Это дало возможность использовать на серверной стороне web-сервера Apache с поддержкой серверного скриптового языка программирования php.

Основываясь на приведенных выше требованиях и ограничениях, был разработан Internet-портал, включающий в себя взаимодействующие между собой web-сайт, встроенный в него Java-апплет, сервер, на котором web-сайт располагается и сервер оператора услуг ДЗЗ – Научного центра оперативного мониторинга Земли ОАО «Российские космические системы» (НЦ ОМЗ). Общая структура портала со схемой взаимодействия его составных частей представлена на рис. 1.

Двуязычный WEB-сайт портала содержит информацию о проекте MEDEO, его участниках, характеристиках спутников и съёмочных устройств, правилах заказа космических снимков (рис. 2). Отдельной частью сайта является страница электронного каталога. При переходе на неё браузер пользователя загружает с сервера MEDEO Java-апплет, предоставляющий сервис по поиску интересующих спутниковых видеоданных в базе данных (БД) электронного каталога. Интерфейс апплета позволяет оператору задавать условия поиска такие как: тип спутника и съёмочного устройства, регион интереса,

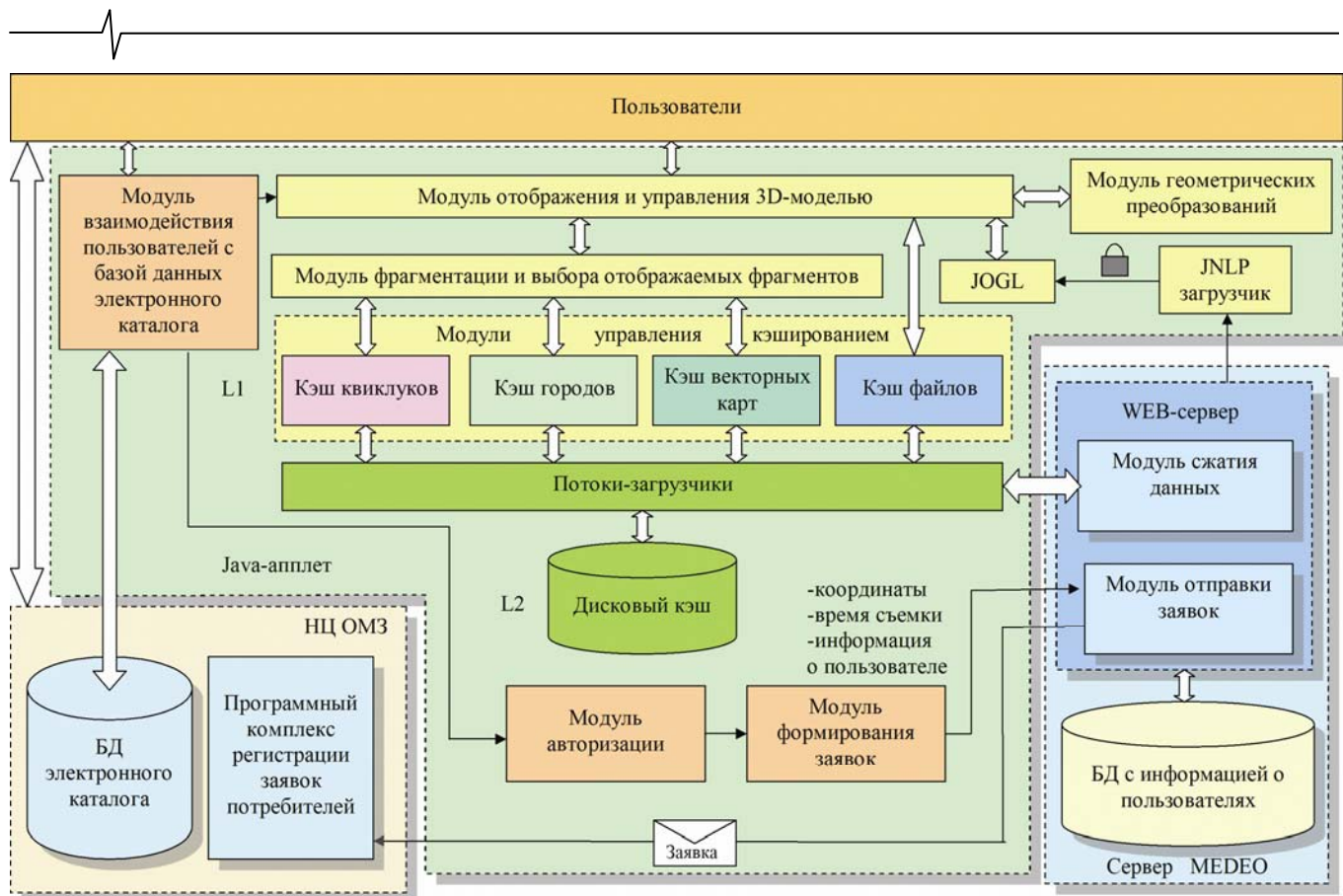


Рис. 1. Структурная схема Internet-портала «MEDEO»



Рис. 2. Окно Internet-портала

временной интервал съёмки, максимальный процент облачного покрытия снимка, визуализировать найденные снимки на поверхности 3D-модели земного шара и формировать заявку на приобретение выбранных данных. Информация о включенных в заявку снимках и реквизитах пользователя передается на сервер MEDEO, где из нее формируется e-mail сообщение заданного формата, которое затем отправляется на адрес системы регистрации заказов НЦ ОМЗ. Получив сообщение, оператор связывается с потребителем по указанным в письме контактными данными и обговаривает условия и способы приобретения запрашиваемых данных ДЗЗ.

Алгоритмическая и программная реализация Java-апплета

При разработке современных центральных процессоров наблюдается тенденция к постоянному увеличению количества вычислительных ядер, что стимулирует разработчиков программного обеспечения производить распараллеливание вычислительных процессов. С учётом этой тенденции в Java-апплете реализован механизм многопоточности, что позволяет не только увеличить эффективность использования ресурсов программных и аппаратных средств, но и сделать программу более дружелюбной к пользователям, позволяя им продолжить работу параллельно с фоновой обработкой изображений, загрузкой файлов данных и другими системными операциями.

Запуск апплета производится при помощи JNLP-загрузчика – специальной программы, задача которой состоит в определении платформы и операционной системы компьютера пользователя и загрузке с сайта компании-разработчика подходящей версии библиотеки JOGL.

Файлы библиотеки являются единственными платформо-зависимыми компонентами приложения. Это вызвано тем, что библиотека написана с использованием нативного кода – кода, использующего инструкции процессора целевой платформы. Подлинность и безопасность файлов библиотеки гарантируется компанией Oracle Inc., подписывающей их электронной подписью.

Апплет условно можно разделить на две взаимодействующие друг с другом части, работающие параллельно: модуль взаимодействия пользователей с базой данных электронного каталога и модуль отображения и управления 3D-моделью земного шара. Первая часть предоставляет пользователю интерфейс задания параметров выборки кадров, формирует и отправляет запросы в базу данных на получение данных по заданным параметрам, отображает полученный результат в табличном виде и посылает запрос второй части приложения на визуализацию выбранных объектов. Модуль отображения и управления 3D-моделью предоставляет следующие функциональные возможности:

1) формирует трёхмерную модель Земли и отображает на ней следующие объекты:

- линии географической сетки с нанесёнными подписями широт и долгот;
- слои векторных карт (гидрография, береговые линии, государственные границы, автомобильные и железные дороги, рельеф);
- названия населённых пунктов;
- контур зоны интереса пользователя;

- контуры и растр выбранных из БД кадров.

2) предоставляет пользователям возможность навигации по модельному изображению земного шара путём его поворота, изменения масштаба и перехода к заданной точке на поверхности.

Объёмное отображение выбранных из БД геокодированных кадров происходит путём их трансформирования в прямоугольную систему координат OXYZ с использованием модуля геометрических преобразований. Этот процесс описывается следующей схемой:

$$D(m, n) \xrightarrow{\phi_B, \phi_L} (B, L) \xrightarrow{F_x, F_y, F_z} D^*(X, Y, Z),$$

где $D(m, n)$ – изображение, хранящееся в БД электронного каталога; ϕ_B, ϕ_L – функции преобразования планарных координат (m, n) в геодезические (B, L) ; F_x, F_y, F_z – функции преобразования геодезических координат в прямоугольные; $D^*(X, Y, Z)$ – спутниковое изображение в прямоугольной системе координат.

Поскольку изображения геокодированны, то функции ϕ_B, ϕ_L – известны, а преобразование геодезических координат в прямоугольные, согласно ГОСТ Р 51794-2008, осуществляется по формулам:

$$X = (N + H) \cos B \cos L$$

$$Y = (N - H) \cos B \sin L,$$

$$Z = [(1 - e^2)W + H] \sin B$$

где X, Y, Z – прямоугольные пространственные координаты точки;

B, L – геодезические широта и долгота точки соответственно;

H – геодезическая высота точки, м;

N – радиус кривизны первого вертикала, м;

e – эксцентриситет эллипсоида.

Значения N и e вычисляются по формулам:

$$N = \frac{a}{\sqrt{1 - e^2 \sin^2 B}},$$

$$e^2 = 2\gamma - \gamma^2,$$

где a – большая полуось эллипсоида, равная 6378245 м, γ – сжатие эллипсоида, равное 1/298,3.

Преобразование пространственных прямоугольных координат в геодезические выполняется по следующему итерационному алгоритму:

1) вычисляется вспомогательная величина D по формуле $D = \sqrt{X^2 + Y^2}$;

2) если $D = 0$, то $B = \frac{\pi Z}{2|Z|}$, $L = 0$,

$H = Z \sin B - a \sqrt{1 - e^2 \sin^2 B}$, и останавливаем вычисления;

3) если $D \neq 0$, при:

$$Y < 0, X > 0, \text{ то } L = 2\pi - L_a;$$

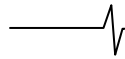
$$Y < 0, X < 0, \text{ то } L = \pi + L_a;$$

$$Y > 0, X < 0, \text{ то } L = \pi - L_a;$$

$$Y > 0, X > 0, \text{ то } L = L_a;$$

$$Y = 0, X > 0, \text{ то } L = 0;$$

$$Y = 0, X < 0, \text{ то } L = \pi,$$



где $L_a = \left| \arcsin \left(\frac{Y}{D} \right) \right|$;

4) если $Z=0$, то $B=0$, $H=D-a$, и останавливаем вычисления;

5) если $Z \neq 0$, то находим вспомогательные величины:

$$r = \sqrt{X^2 + Y^2 + Z^2}, \quad c = \arcsin \left(\frac{Z}{r} \right), \quad \rho = \frac{e^2 a}{2r} \quad \text{и} \quad s_1 = 0, \quad \text{а}$$

затем

а. производим вычисления: $b = c + s_1$,

$$\text{б. } s_2 = \arcsin \left(\frac{\rho \sin(2b)}{\sqrt{1 - e^2 \sin^2 b}} \right), \quad d = |s_2 - s_1|;$$

в. если $d < \varepsilon$, то $B = b$,

$H = D \cos B + Z \sin B - a \sqrt{1 - e^2 \sin^2 B}$, и останавливаем вычисления;

г. если $d \geq \varepsilon$, то $s_1 = s_2$, и повторяем вычисления с пункта а.

Здесь $\varepsilon = 10^{-4}$ – допустимая погрешность вычислений.

Для повышения быстродействия за счет снижения объема передаваемых по сети данных предлагается пирамидальный механизм представления векторных карт и карт городов, согласно которому, эти карты разбиваются на гранулы с различными размерами (в терминах геодезических координат) и уровнями детализации. Данные гранулы хранятся на сервере в виде отдельных файлов, компрессируемых перед передачей их в апплет. Модуль фрагментации и выбора визуализируемых фрагментов определяет оптимальный уровень детализации и форми-

рует списки видимых гранул, после чего запрос на получение файлов передаётся соответствующему модулю управления кэшированием.

Дополнительно для обеспечения высокой скорости отображения разработана подсистема кэширования, состоящая из двух уровней: L1 – кэш в оперативной памяти и L2 – дисковый кэш, размещённый на пользовательском компьютере. Алгоритм функционирования подсистемы выглядит следующим образом:

- модуль отображения и управления 3D-моделью помещает запрос на получение файла в очередь запросов соответствующего модуля управления кэшированием;

- модуль управления кэшированием проверяет наличие файла в кэше L1 и, если файл присутствует, возвращает его модулю отображения;

- если файл отсутствует в L1, модуль управления кэшированием помещает запрос на загрузку этого файла в очередь загрузки;

- очередь загрузки регулярно проверяется специальными потоками-загрузчиками на предмет наличия запросов на загрузку и, обнаружив запрос, поток-загрузчик проверяет наличие требуемого файла в кэше L2. В случае если файл найден, то он копируется в L1 и возвращается модулю отображения;

- в случае если файл отсутствует в L2, поток-загрузчик загружает сжатый файл с сервера, распаковывает его и помещает в L2 и L1.

Каждый модуль кэширования выполняется в отдельном потоке. Так же в отдельных потоках выполняются процедуры очистки кэша, удаляющие невостребованные объекты из L1 и L2.

На рис. 3 представлено главное окно апплета. В левой его части расположены органы управления, позволяющие пользователю задавать условия выборки кадров из базы данных электронного каталога. Атрибутивная информация

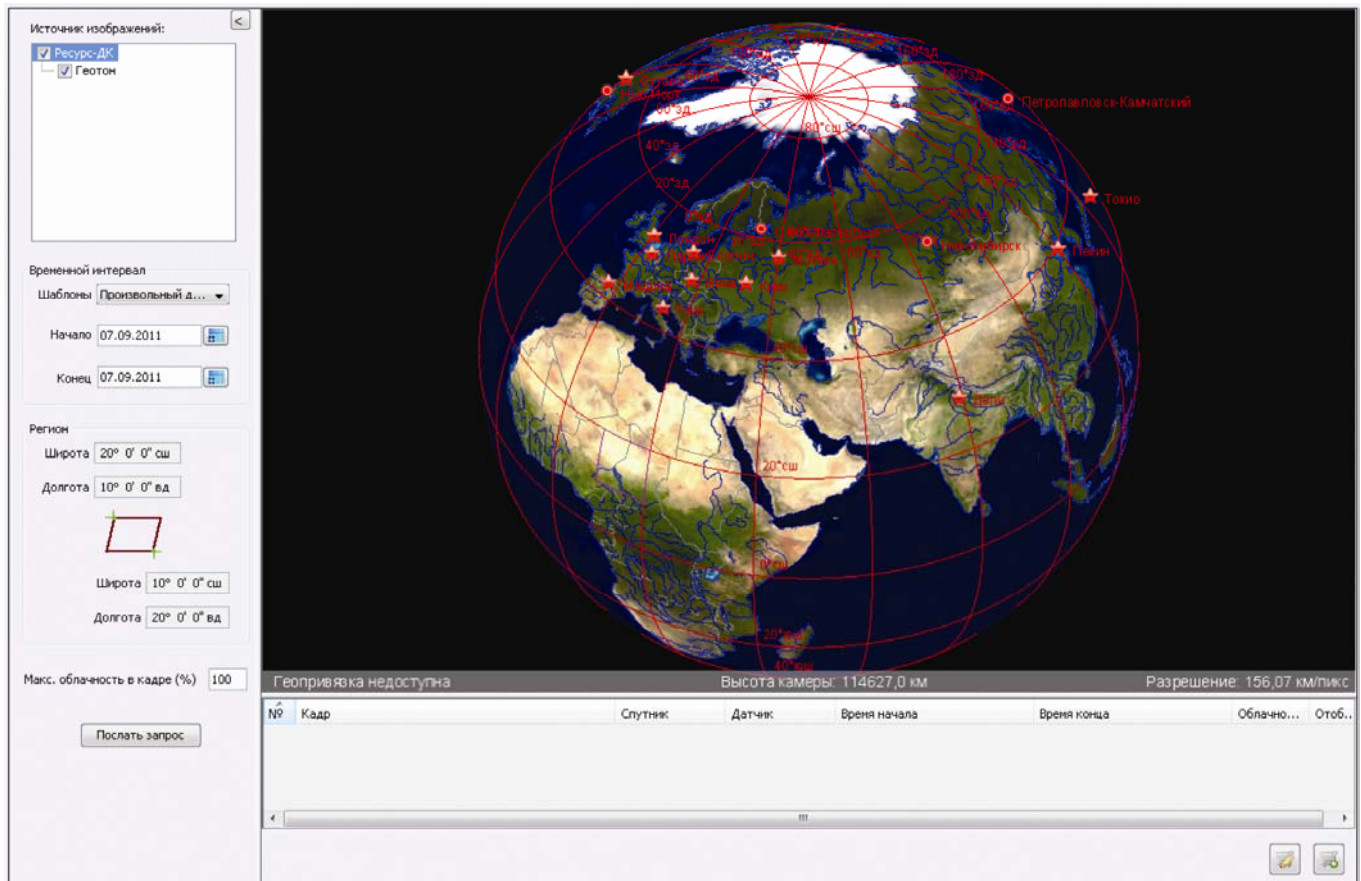


Рис. 3. Главное окно апплета Internet-портала

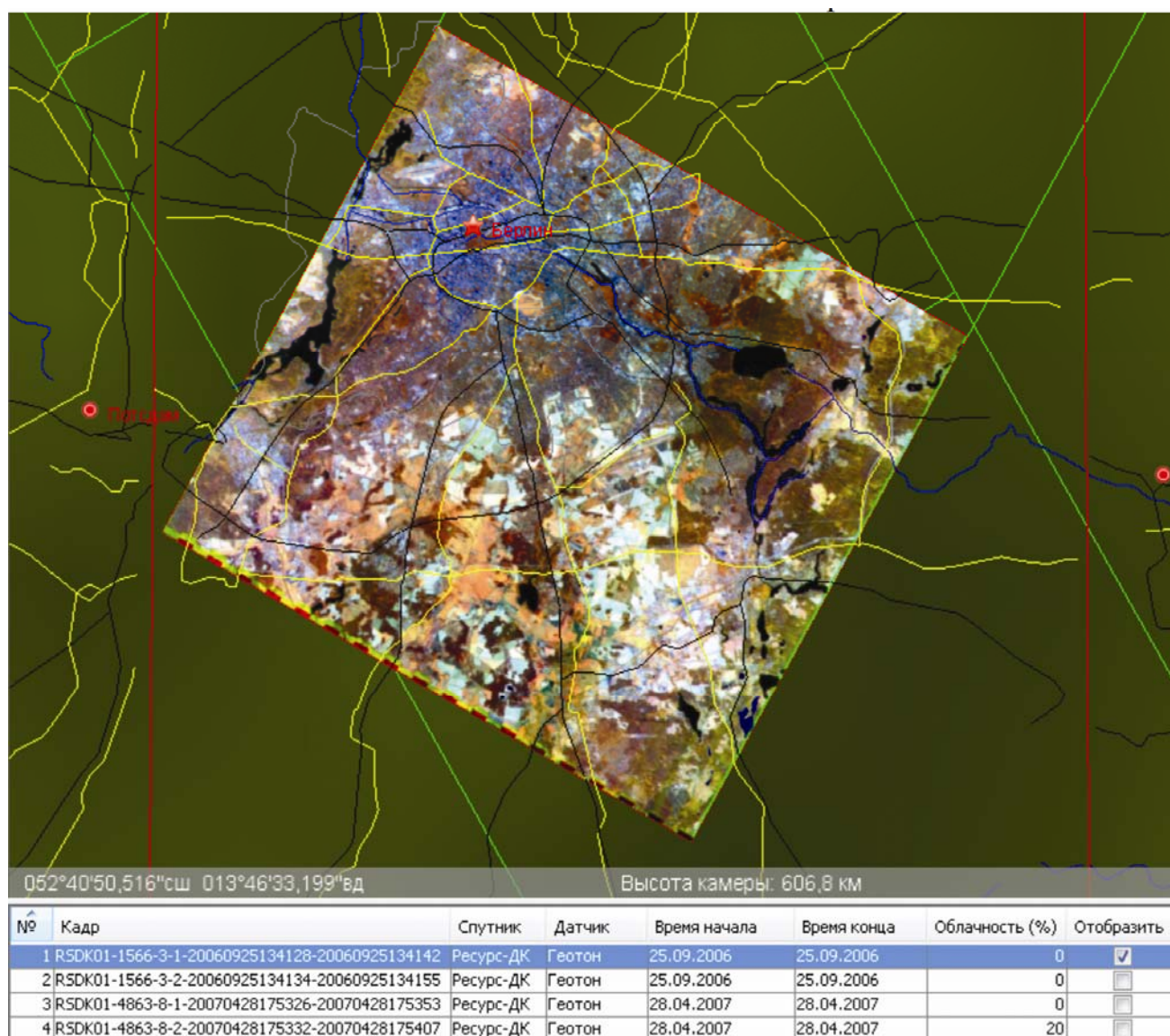


Рис. 4. Пример совместного отображения закаталогизированного кадра и атрибутивных данных

(спутник, прибор, время съёмки, процент покрытия облаками) кадров, удовлетворяющих условиям выборки, помещается в таблицу в нижней части окна, а их контура отображаются на поверхности трехмерной модели земного шара, расположенной в центре. Следует отметить, что регион интереса пользователь может задавать, как непосредственно вводя географические координаты углов области в поля ввода в разделе «Регион», так и визуально при помощи мыши на поверхности 3D-модели.

В целях уменьшения количества передаваемой по сети информации, растровые изображения отобранных кадров визуализируются по запросу пользователя, которому требуется отметить пункт в столбце «Отобразить» строки с интересующим кадром (рис. 4). При помощи кнопки, расположенной в правом нижнем углу окна, интересующие кадры могут быть помещены в корзину для последующего оформления заказа. Оформлять заказы могут только зарегистрированные пользователи, поэтому пользователям предварительно следует пройти процедуру регистрации на соответствующей странице сайта портала. Созданный в модуле формирования заявок список атрибутов заказываемых кадров передается на сервер, где на его основе подготавливается и отправляется в адрес системы

регистрации заявок оператора услуг ДЗЗ e-mail сообщение.

Заключение

Таким образом, разработанный Internet-портал реализует полный цикл взаимодействия потребителей и поставщиков данных ДЗЗ.

Программное обеспечение портала прошло тестирование и в ближайшее время портал начнет функционировать. Следует заметить, что технические решения, заложенные при проектировании портала, позволяют на принципиально новом уровне обеспечить работу всех заинтересованных лиц с информацией ДЗЗ. Это, в свою очередь, очень важно для популяризации спутниковой информации и расширения рынка её практических применений.

PRINCIPLES OF SATELLITE IMAGES GEOPORTAL BUILDING AS PART OF THE MEDEO PROJECT

A.A.Vetrov, A.M.Kochergin, A.E. Kuznetsov

In the article are described principles of satellite images geoportal building, as well as EO data 3D visualization technology.

ВЫСОКОСКОРОСТНОЙ АЛГОРИТМ СЕГМЕНТАЦИИ ИЗОБРАЖЕНИЙ ЗВЕЗДНОГО НЕБА, ПОЛУЧЕННЫХ ОТ ДАТЧИКОВ СКАНЕРНОГО ТИПА

Пашенцев Д.Ю., инженер-программист отдела 4374 филиала ФГУП «ГНПРКЦ «ЦСКБ-Прогресс» – «ОКБ «Спектр»», pashentsev62@mail.ru

Акимов А.А., инженер-программист отдела 4374 филиала ФГУП «ГНПРКЦ «ЦСКБ-Прогресс» – «ОКБ «Спектр»», aaa_akimov@mail.ru

Скирумунт В.К., начальник центра 1130 ФГУП «ГНПРКЦ «ЦСКБ-Прогресс», skir@list.ru

Тишкин Р.В., начальник отдела 4374 филиала ФГУП «ГНПРКЦ «ЦСКБ-Прогресс» – «ОКБ «Спектр»», roman.tishkin@mail.ru

Ключевые слова: дистанционное зондирование Земли, сегментация, бинаризация, высокоскоростной алгоритм.

Введение

Системы дистанционного зондирования Земли (ДЗЗ) находят эффективное применение во многих отраслях народного хозяйства: гидрометеорологии, контроле окружающей среды, геологии, сельском и лесном хозяйствах, военной разведке и др. Однако в исходном виде изображения, формируемые с помощью различных систем ДЗЗ, не могут быть использованы по назначению, поскольку имеют значительные геометрические искажения по отношению к объектам наблюдаемой сцены. Для проведения высокоточной автоматической геометрической коррекции видеоданных, получаемых при ДЗЗ, необходимо знать элементы внутреннего и внешнего ориентирования съёмочного устройства и космического аппарата. К элементам внешнего ориентирования космического аппарата относятся данные, полученные от гироскопических систем, звёздных и солнечных датчиков и систем позиционирования, таких как ГЛОНАСС и GPS. К элементам внутреннего ориентирования съёмочной аппаратуры относятся: дисторсия объектива, координаты главной точки оптики, расположение фотоприёмных матриц (ПЗС-матриц) в фокальной плоскости и фокусное расстояние.

Параметры, относящиеся к внутреннему ориентированию съёмочной аппаратуры [1], измеряются при ее наземной калибровке, и их точность во многом определяет качество дальнейшей обработки полученных изображений. Однако конфигурация ПЗС-матриц меняется под различными воздействиями, возникающими при выводе космического аппарата на орбиту, а также из-за теплового влияния Солнца. Встаёт задача калибровки съёмочной аппаратуры не только в лабораторных условиях, но и в штатном режиме эксплуатации.

Анализ зарубежных публикаций [2–4] показал, что для решения задач полётной калибровки используются опорные геодезические полигоны с большим числом опорных точек (GCP). Однако такой подход требует создания специальных полигонов [5].

Альтернативным способом является калибровка по звёздному небу [6]. Космический аппарат отклоняется по крену на заданный угол и проводит съёмку звёзд (Рис. 1). В данном случае в качестве опорного полигона выступает высокоточный звёздный каталог (ВЗК).

Предложен высокоскоростной алгоритм сегментации изображений звёздного неба. Приведены основные этапы сегментации, а также сравнительные результаты в части быстродействия применительно к обработке изображений звёздного неба.



Рис. 1. Съёмка звёздного неба

Основными задачами режима калибровки по звёздному небу являются:

- определение параметров дисторсии объектива;
- уточнение паспортных значений параметров установок ПЗС-матриц и фотограмметрических параметров;
- оценка фокусного расстояния;
- оценка точности определения параметров внутреннего ориентирования.

В данном режиме, в отличие от калибровки по наземным полигонам, исключается искажающее действие кривизны Земли; эллиптичности орбиты, влияния атмосферы и рельефа местности.

Для идентификации одноименных звёзд как реально наблюдаемых, так и их аналогов из астрокаталога, необходимо выполнить предварительную обработку реального изображения звёздного неба. Основным этапом такой обработки является сегментация изображения (с целью выделения звёзд и определения их координат).

Спутниковые снимки высокого разрешения участков звёздного неба могут превышать объём более 40 Гбайт. Существующие алгоритмы сегментации не обеспечивают необходимую скорость решения этой задачи, что не позволяет производить обработку

данных ДЗЗ в темпе их получения.

В настоящей работе предлагается алгоритм сегментации изображения звёздного неба полученного от датчиков сканерного типа, свободный от указанных недостатков.

Для реализации высокоскоростного алгоритма необходимо построить эталонный кадр (ЭК) по данным ВЗК в системе координат изображения, представляющий собой опорную координатную информацию звёзд в системе координат изображения.

При использовании фотографических снимков построение эталонного кадра на изображении осуществляется стандартными методами фотограмметрии. Однако для сканерных снимков, вследствие немгновенности получения изображения кадра от оптико-электронного преобразователя (ОЭП), центральную проекцию можно считать лишь в строке, которая получается мгновенно.

При этом космическим изображениям звёздного неба, полученным оптико-электронной съёмочной системой, присущи геометрические искажения, обусловленные эллиптичностью орбиты, ошибками изготовления и установки системы управления (СУ) и систем космического аппарата (КА). Эти геометрические искажения носят постоянный, систематический и случайный характер.

Формирование идеального кадра

Для построения идеального кадра необходимо наличие информации о движении визирной системы координат относительно инерциальной и выбора в заданных областях экваториальных координат звёзд (α , δ) из астрокаталога, например tycho2 [7]. Отклонение расположения звёзд на изображении (из-за воздействия ряда факторов) относительно идеального не превышают минимального расстояния между группами звёзд в заданном участке.

Инерциальная система координат (ИСК). Система координат, центр которой совпадает с центром Земли. Ось ОХ направлена в сторону весеннего равноденствия, ось ОZ направлена на северный полюс Земли, ось ОУ дополняет систему до правой тройки [1].

Визирная система координат (ВСК). Система координат, центр которой расположен в главной точке ОЭП, ось ОХ совпадает с визирующим лучом ОЭП, ось ОZ направлена вдоль ПЗС-матриц ОЭП, ось ОУ дополняет систему до правой тройки [1].

Отметим, что из-за бесконечно малого параллакса звёзд центры ВСК и ИСК удобно расположить в одной точке (например, в центре Земли) [8].

Звезда фиксируется ОЭП тогда, когда луч-вектор звезды S попадет на одну из ПЗС-матриц в фокальной плоскости (эта плоскость совпадает с плоскостью YOZ визирной системы координат) ОЭП. Рассмотрим следующий схематичный рисунок (Рис. 2.).

Произвольная звезда S в визирной системе координат характеризуется радиус-вектором $\mathbf{V}_s$, являющимся вектором-направлением звезды S в ВСК.

$$\mathbf{V}_s = \begin{Bmatrix} F \\ -Y \\ -Z \end{Bmatrix}, \quad (1)$$

где F – фокусное расстояние ОЭП, Z и Y – координаты точки проекции звезды S на плоскость YOZ в ВСК.

Этот вектор можно определить, имея экваториальные координаты звезды S прямое восхождение α , склонение δ и кватернион ориентации ВСК относительно ИСК в момент времени t_0 [9]. Тогда:

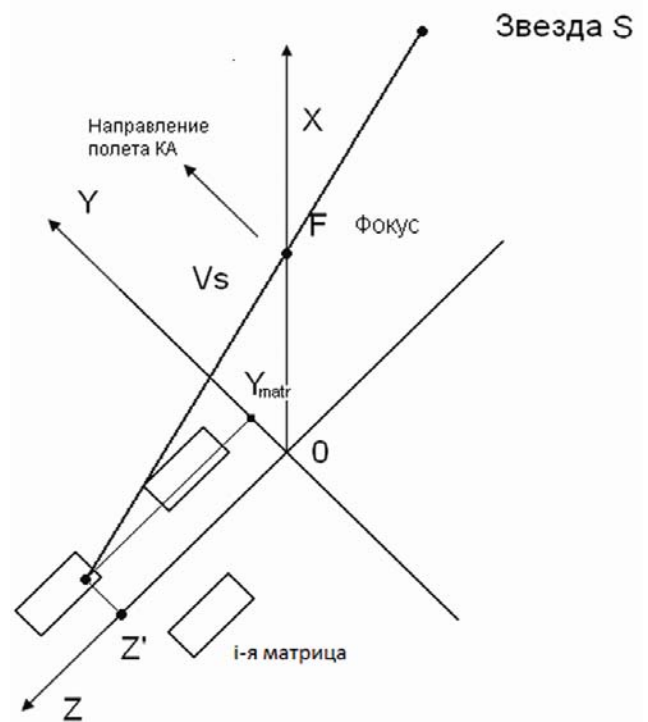


Рис. 2. Расположение ПЗС-матриц в фокальной плоскости ОЭП

$$\mathbf{V}_{sn} = L(t_0)^{-1} \circ \mathbf{R} \circ L(t_0), \quad (2)$$

$$\text{где } \mathbf{R} = \begin{pmatrix} \cos(\alpha) \cdot \cos(\delta) \\ \sin(\alpha) \cdot \cos(\delta) \\ \sin(\delta) \end{pmatrix},$$

$L(t_0)$ – кватернион ориентации визирной системы координат в момент времени t_0 .

Начальный момент времени формирования изображения $t_{нач}$ положим равным нулю.

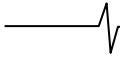
Рассмотрим кватернионную функцию от времени вращения визирной системы координат в инерциальной системе координат:

$$L(t) = \begin{pmatrix} L_0(t) \\ L_1(t) \\ L_2(t) \\ L_3(t) \end{pmatrix} \quad (3)$$

Для дальнейших рассуждений примем, что функция обладает свойством непрерывности, кроме того, для двух произвольных моментов времени t_1 и t_2 и произвольного вектора $\mathbf{R}$, заданного в ИСК, должно выполняться условие: если для t_1 и t_2 проекции вектора $\mathbf{R}$ в визирную систему координат совпадают, то $t_1 = t_2$.

Непрерывность означает, что сканирование звёздного неба происходит без пропусков. Условие равенства двух моментов времени исходит из физического смысла фотографирования (на разных строках изображения не будет одних и тех же звёзд, по крайней мере, для одной конкретной матрицы).

По известным формулам можно выразить функцию матрицы перехода через кватернион [9]:



$$\begin{aligned}
c_{11}(t) &= \lambda_0^2(t) + \lambda_1^2(t) - \lambda_2^2(t) - \lambda_3^2(t) \\
c_{12}(t) &= 2 \cdot (\lambda_1(t) \cdot \lambda_2(t) + \lambda_0(t) \cdot \lambda_3(t)) \\
c_{13}(t) &= 2 \cdot (\lambda_1(t) \cdot \lambda_3(t) + \lambda_0(t) \cdot \lambda_2(t)) \\
c_{21}(t) &= 2 \cdot (\lambda_1(t) \cdot \lambda_2(t) - \lambda_0(t) \cdot \lambda_3(t)) \\
c_{22}(t) &= \lambda_0^2(t) - \lambda_1^2(t) + \lambda_2^2(t) - \lambda_3^2(t) \\
c_{23}(t) &= 2 \cdot (\lambda_2(t) \cdot \lambda_3(t) - \lambda_0(t) \cdot \lambda_1(t)) \\
c_{31}(t) &= 2 \cdot (\lambda_1(t) \cdot \lambda_3(t) - \lambda_0(t) \cdot \lambda_2(t)) \\
c_{32}(t) &= 2 \cdot (\lambda_2(t) \cdot \lambda_3(t) - \lambda_0(t) \cdot \lambda_1(t)) \\
c_{33}(t) &= \lambda_0^2(t) - \lambda_1^2(t) - \lambda_2^2(t) - \lambda_3^2(t)
\end{aligned} \quad (4)$$

Тогда вектор $\mathbf{R}$ звезды в ВСК можно выразить как:
 $\mathbf{V}_{sn} = \mathbf{C} \cdot \mathbf{R}$

Геометрическое расположение ПЗС-матрицы таково, что координата Y в ВСК первой строки ПЗС-матрицы равна постоянной величине Y_{matr} (в идеальном неискаженном случае [1]). Необходимым условием попадания звезды S на ПЗС-матрицу будет условие равенства координаты Y в векторе $\mathbf{V}_s$ и Y_{matr} :

$$-V_s^{(2)} = Y = Y_{matr} \quad (5)$$

Учитывая матрицу $\mathbf{C}$, необходимое условие запишем следующим образом:

$$-F \cdot \frac{c_{21}(t) \cdot R_1 + c_{22}(t) \cdot R_2 + c_{23}(t) \cdot R_3}{c_{11}(t) \cdot R_1 + c_{12}(t) \cdot R_2 + c_{13}(t) \cdot R_3} = -V_s^{(2)} = Y = Y_{matr}, \quad (6)$$

для момента времени t .

Если положить, что $\frac{Y_{matr}}{F} = \tilde{y}_0$, то выражение (6)

можно записать в следующем виде:

$$\begin{aligned}
&\left(c_{21}(t) - \tilde{y}_0 \cdot c_{11}(t) \right) \cdot R_1 + \left(c_{22}(t) - \tilde{y}_0 \cdot c_{12}(t) \right) \cdot R_2 + \\
&\left(c_{23}(t) - \tilde{y}_0 \cdot c_{13}(t) \right) \cdot R_3 = 0
\end{aligned} \quad (7)$$

В силу сделанных предположений относительно функции $L(t)$, можно заключить, что на промежутке построения эталонного кадра функция (7) монотонна, значит, она имеет одно решение или не имеет решения вовсе, что всегда можно проверить, найдя значения функции на крайних точках временного отрезка [10].

Решив уравнение (7), найдем время, для которого выполняется необходимое условие (5). После чего проверим второе необходимое условие попадания звезды на ПЗС-матрицу, а именно – вычислим координату Z и посмотрим, принадлежит ли значение этой координаты множеству значений координаты Z пикселей первой строки ПЗС-матрицы. Координата Z будет рассчитываться как:

$$-F \cdot \frac{c_{31}(t) \cdot R_1 + c_{32}(t) \cdot R_2 + c_{33}(t) \cdot R_3}{c_{11}(t) \cdot R_1 + c_{12}(t) \cdot R_2 + c_{13}(t) \cdot R_3} = Z, \quad (8)$$

для вычисленного значения t .

Таким образом, выполнение этих двух условий является необходимым и достаточным для попадания звезды на ПЗС-матрицу.

При выполнении необходимого и достаточного условий попадания звезды S на матрицу ПЗС можно определить ее координаты на изображении, где

координата Z (номер пикселя) рассчитывается по формуле (8), а координата Y (номер строки) – с использованием найденного времени попадания звезды, а также частоты формирования строк.

После построения идеального кадра, для выделения и вычисления координат звезд в области просмотра, можно использовать один из существующих методов сегментации.

Сегментацию можно разбить на два этапа:

1. Выбор порога сегментации;

В свою очередь, порог сегментации может быть определен двумя методами:

- глобально-оптимальный метод (метод Оцу);
- локально-адаптивный;

2. Задача разметки и параметризации.

Глобальный оптимальный порог (метод Оцу). Метод Оцу (Otsu's Method) использует гистограмму изображения для расчёта порога.

Пример изображения с объектом:

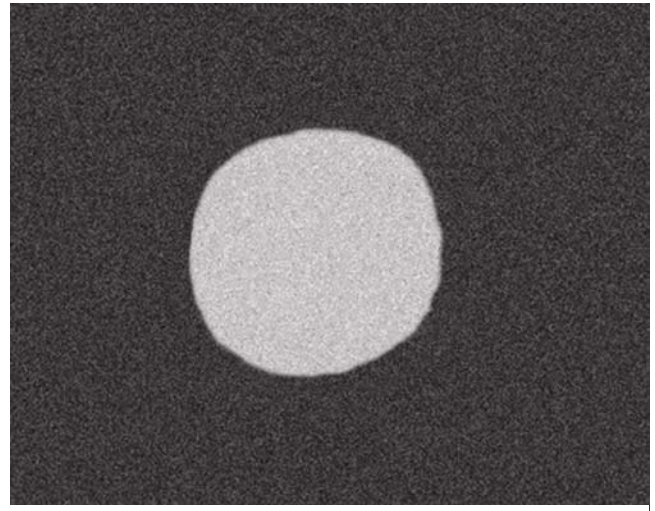


Рис. 3. Тестовое изображение

Гистограмма для такого изображения:

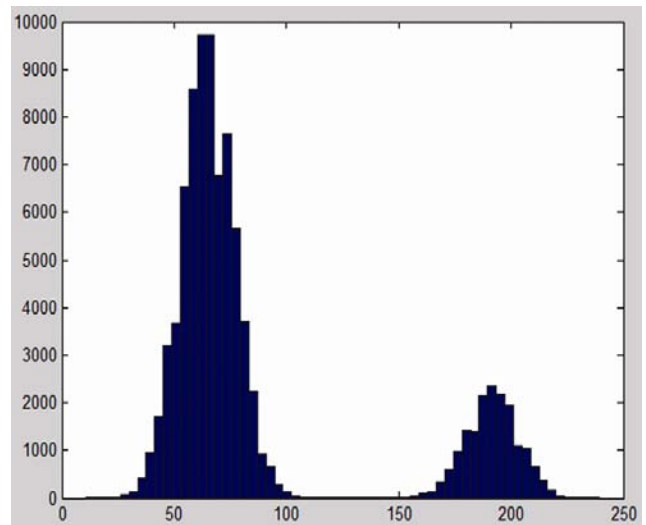


Рис. 4. Распределение гистограммы для тестового изображения

Из гистограммы видно, что имеются два четко разделяющихся класса. Выбор порога в данном методе сводится к минимизации внутриклассовой дисперсии, которая определяется как взвешенная сумма дисперсий двух классов:

$$\sigma_w^2 = w_1 \sigma_1^2 + w_2 \sigma_2^2. \quad (9)$$

Здесь w_1 и w_2 – вероятности первого и второго классов, соответственно. При этом минимизация внутриклассовой дисперсии эквивалента максимизации межклассовой дисперсии, которая равна:

$$\sigma_b^2 = w_1 w_2 (a_1 - a_2)^2. \quad (10)$$

В этой формуле a_1 и a_2 – средние арифметические значения для каждого из классов.

Особенность этой формулы заключается в том, что $w_1(t+1)$, $w_2(t+1)$, $a_1(t+1)$, $a_2(t+1)$ легко выражаются через предыдущие значения $w_1(t)$, $w_2(t)$, $a_1(t)$, $a_2(t)$ (t – текущий порог).

Локальный адаптивный порог. Идея метода состоит в том, что яркость каждого пикселя сравнивается со средним значением яркости пикселей, входящих в окрестность первого. Если яркость обрабатываемого пикселя на t процентов меньше среднего значения окрестности, то оно принимает значение 0, в противном случае 1.

Для ускорения вычисления среднего значения в окрестности (окне) обрабатываемого пикселя перед основным циклом обработки строится матрица (интегральное изображение, Integral Image), позволяющая вычислить среднее значение за постоянное время вне зависимости от размера окна.

Интегральное изображение (матрица сумм) – это инструмент, который можно использовать, когда имеется функция от пикселя, принимающая целочисленные значения (например, яркость) и необходимо вычислить суммарное значение этой функции на части изображения.

Алгоритм высокоскоростной сегментации

Пусть имеется космическое изображение звёздного неба $B(m, n)$, в системе координат m, n , в которой при идеально точной координатной привязке опорные объекты (звёзды) изображения должны совпадать с одноименными объектами эталонного кадра, представленной в этой же системе координат.

На заданный регион съёмки формируется достаточно точный ЭК $E(m, n)$ в системе координат m, n . Предварительно на ЭК имеем множество звёзд с координатами центров в точках (m_{ei}, n_{ei}) , $i = 1, I$.

С учетом сказанного высокоскоростной алгоритм сегментации в задаче локализации центров звезд на изображении $B(m, n)$ включает следующие этапы:

1. Осуществляется выборка звёзд ЭК, имеющих координаты в системе координат m, n .

2. Затем для каждой звезды $S_e \in E$ производится построение фрагмента Ω_e выполнения сегментации. Границу данного фрагмента оператор подбирает на этапе визуального контроля изображения $B(m, n)$.

3. Далее для каждого фрагмента Ω_e выполняется поиск фрагмента Ω_b исходного изображения. На рис. 5 представлено изображение звёзд, для которого найдены фрагменты Ω_b . При этом размеры фрагментов равны $\Omega_{re} = \Omega_{rb}$.

4. Затем каждый фрагмент Ω_b сегментируется по одному из выше приведенных алгоритмов сегмента-

ции, и выполняется локализация найденного фрагмента звезды изображения $B(m, n)$.

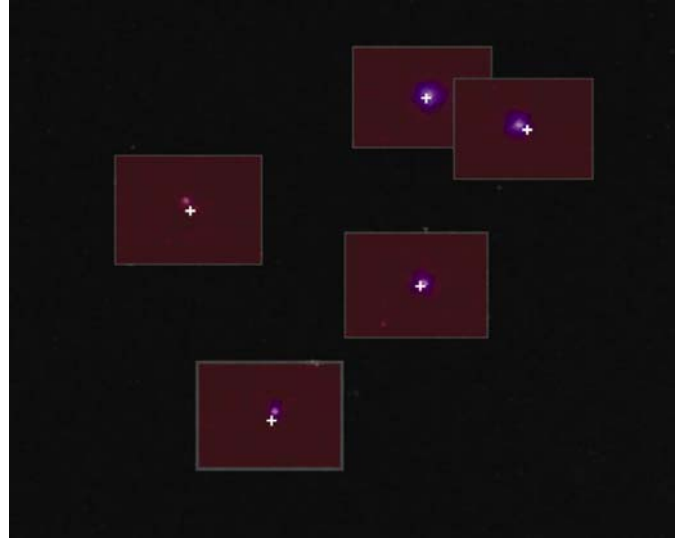


Рис. 5. Изображение звёздного неба с выделенными фрагментами Ω_b



Рис. 6. Обобщенный алгоритм высокоскоростной сегментации снимка звёздного неба

Функциональная блок-схема алгоритма приведена на рис. 6. В таблице 1 приведено сравнение выполнения сегментации методом локально адаптивного порога с использованием высокоскоростного алгоритма и без. Ниже приведены характеристики тестового изображения:

– ширина – 54000 пикселей;

- высота – 16660 пикселей;
- звёзд – 109.

Длительность сегментации изображения без применения высокоскоростного алгоритма составила приблизительно 12 секунд. Количество распознанных звёзд 105.

Необходимо отметить, что реального изображения нет, и оно моделируется [11].

Как видно из таблицы 1, оптимальный размер окна сканирования находится в области 250-350 пикселей. Время на обработку файла сократилось приблизительно в 36 раз. Применение предложенного алгоритма на других изображениях звёздного неба получены схожие результаты в части быстрого действия. Данный алгоритм позволяет уменьшить вычислительные затраты на сегментацию полученных изображений звёздных узоров в случае наличия априорной информации по движению космического аппарата.

При этом следует отметить, что скорость обработки можно увеличить еще больше в случае использования параллельных методов обработки.

Таблица 1

Размер окно сканирования, пиксели	Время, сек	Кол-во найденных точечных источников
400	0,8	105
350	0,406	105
300	0,328	105
250	0,250	105
200	0,203	105
150	0,25	105
100	0,172	102
50	0,078	64
25	0,057	35

Заключение

В работе рассмотрен высокоскоростной алгоритм сегментации изображений звёздного неба, полученных от датчиков сканерного типа. Предложен метод построения идеального кадра для высокоскоростной сегментации изображения звёздного неба. Использование алгоритма, основанного на априорных данных положений звёзд,

позволило сократить время обработки изображений звёздного неба приблизительно в 36 раз.

Литература

1. Бакланов А.И. Системы наблюдения и мониторинга: учебное пособие – М. Бином. Лаборатория знаний, 2009. – С. 234.
2. Jacobsen, K.: Issues and Method for In-Flight and On-Orbit Calibration // Workshop on Radiometric and Geometric Calibration, Gulfport, 2003.
3. Valorge C., Meygret A., Lebegue L., Henry P: Forty years of experience with SPOT in-flight Calibration // Workshop on Radiometric and Geometric Calibration, Gulfport, 2003.
4. Jacobsen K. Calibration of imaging satellite sensors // Institute of Photogrammetry and GeoInformation, University of Hannover.
5. ОКР «Регион В-Валидация», «Создание системы валидационных подспутниковых наблюдений». Проект ТЗ на открытый конкурс. // <http://doc.gostorgi.ru/701/2010-06-02/790730/9.doc>
6. Петрищев В.Ф. Полётная фотограмметрическая калибровка оптико-электронной аппаратуры дистанционного зондирования Земли по звёздному небу // Полёт. – 2005. – N7. – С. 39-42
7. Цветков А.С. Руководство по работе с астрокаталогом Tycho-2. – СПб., 2005.
8. Бакулин П.И., Кононович Э.В., Мороз В.И. Курс общей астрономии. Изд-во «Наука», М., 1970 г., С. 536.
9. Бранец В.Н., Шмыглевский И.П. Применение кватернионов в задачах ориентации твёрдого тела. // Главная редакция физико-математической литературы. Изд-во «Наука», М., 1973, – С. 320.
10. Зорич В.А. Математический анализ. Часть I. Изд. 2-е, испр. И доп. М.: ФАЗИС, 1997. – XIV – С. 554.
11. Акимов А.А., Пашенцев Д.Ю., Пылькин А.Н., Тишкин Р.В. Моделирование точечных изображений звёздного неба, получаемых от датчика сканерной съёмки // Наука и технологии. – Труды XXX Российской школы, посвящённой 65-летию Победы. – М.: РАН. 2010, – С. 386.

A.A. Akimov, V.K. Skirmunt, D.Y. Pashentsev, R.V. Tishkin

We propose a high-speed algorithm for image segmentation sky. The main steps of segmentation, as well as comparative results in terms of performance with respect to imaging the sky.

УДК 004.932.4

ОРГАНИЗАЦИЯ ПРОЦЕССА ФОРМИРОВАНИЯ МОЗАИЧНЫХ ИЗОБРАЖЕНИЙ ЗЕМНОЙ ПОВЕРХНОСТИ

Кузнецов А.Е., д.т.н., зам. директора НИИ «Фотон» РГРТУ, foton@rsreu.ru
Пресняков О.А., к.т.н., с.н.с, НИИ «Фотон» РГРТУ, foton@rsreu.ru

Ключевые слова: дистанционное зондирование земли, обработка изображений, геометрическая коррекция.

Введение

Получение космокарт на обширные территории происходит путём объединения разновременных снимков в непрерывное мозаичное изображение. Этот процесс выполняется в два этапа. На первом этапе решается задача уравнивания блока снимков, в ходе которой за счёт коррекции элементов внешнего ориентирования минимизируются невязки в одноименных и опорных точках изображений. На втором этапе все снимки трансформируются в единую картографическую проекцию с целью получения непрерывного изображения. Специфика данного этапа состоит в том, что для формирования высококачественного снимка необходимо:

- устранить остаточные взаимные геометрические расхождения изображений, чтобы исключить пространственные нестыковки объектов;
- минимизировать яркостные различия в перекрывающихся частях снимков;
- исключить из мозаики участки снимков, закрытые облачностью и попадающие в тень от облаков.

Другая важная особенность формирования мозаичных снимков состоит в необходимости обрабатывать изображения сверхбольшого объёма (десятки и сотни гигабайт) с предоставлением возможности контроля и коррекции параметров геометрического и яркостного совмещения снимков без снижения производительности и исключения повторных преобразований.

В предлагаемой вниманию статье рассматриваются структурные решения по выполнению технологических операций, базирующиеся на принципах виртуальной обработки видеоданных, и математические модели основных этапов преобразования изображений, положенные в основу получивших эффективное внедрение на практике программных комплексов обработки самолётных и космических снимков земной поверхности.

Технология формирования мозаичных изображений

Пусть имеется K геокодированных изображений земной поверхности $B_k(m_k, n_k)$, $k = \overline{1, K}$, $m_k = \overline{1, M_k}$, $n_k = \overline{1, N_k}$, для каждого из которых известны соотношения, задающие связь с координатами пикселей мозаичного снимка $D(x, y)$, $x = \overline{1, X}$, $y = \overline{1, Y}$, $x = F_{kx}(m_k, n_k)$, $y = F_{ky}(m_k, n_k)$.

Тогда формирование мозаичного снимка выполняется путём геометрического трансформирования ис-

Предложена технология формирования мозаичного изображения из снимков поверхности Земли на основе процедуры виртуальной обработки данных с использованием пирамидального представления видеoinформации. Рассмотрены математические модели глобального и локального геометрического и яркостного бесшовного совмещения изображений.

ходных изображений B_k в картографическую систему координат (x, y) ,

$$D(x, y) = \Phi_k \{x, y, B_k[F_{km}(x, y), F_{kn}(x, y)]\}, \quad (2)$$

где Φ_k – оператор яркостного преобразования; F_{km} , F_{kn} – функции обратного геометрического преобразования,

$$m = F_{km}(x, y), \quad n = F_{kn}(x, y). \quad (3)$$

Поскольку для получения мозаичных изображений используются, как правило, разновременные снимки земной поверхности, полученные при различных условиях съёмки, и, кроме того, из-за отсутствия высокодетальных цифровых моделей местности сформировать высококачественные изображения D не всегда удаётся. Это выражается в том, что на мозаичном изображении остаются геометрические и яркостные нестыковки одноименных объектов. Пример подобных различий, существенно снижающих качество выходной продукции, показан на рис. 1.

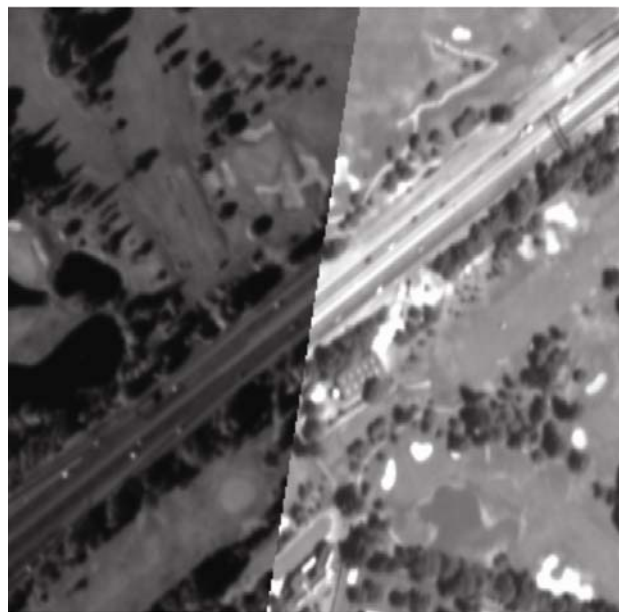


Рис. 1. Пример остаточных геометрических и яркостных искажений совмещаемых изображений

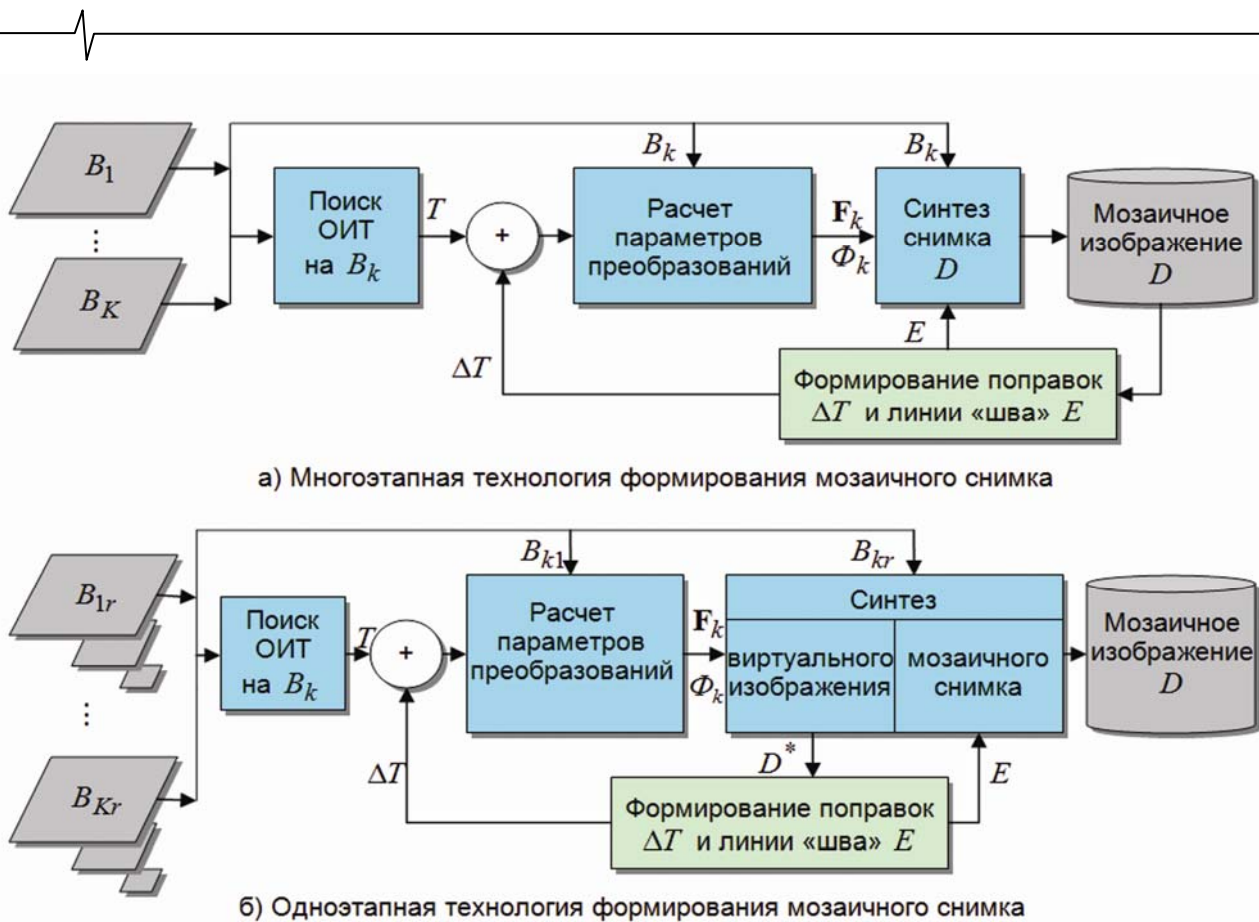


Рис. 2. Технологические схемы формирования мозаичного изображения

В соответствии с опытом мировой практики, для формирования бесшовного мозаичного изображения предлагается:

- осуществлять локальное уточнение функций, F_{kx} , F_{ky} по координатам одноимённых точек (ОИТ), расположенных в перекрывающихся частях изображений B_k ;

- выполнять глобальное и локальное яркостное выравнивание снимков между собой;

- интерактивно задавать линию «шва» в виде многоугольника с вершинами $E = \{m_{ki}, n_{ki}\}$, располагаемыми вдоль качественно совмещённых между собой объектов земной поверхности.

Организовать процесс формирования мозаичного изображения D , с учётом перечисленных требований, можно так, как это показано на рис. 2, а. В соответствии с приведенной схемой, на первом шаге с помощью аппарата корреляционно-экстремального анализа отождествляются одноименные точки (ОИТ) на перекрывающихся частях изображений B_k . На основе набора ОИТ $T = \{m_{kj}, n_{kj}, x_{kj}, y_{kj}\}$ корректируются функции

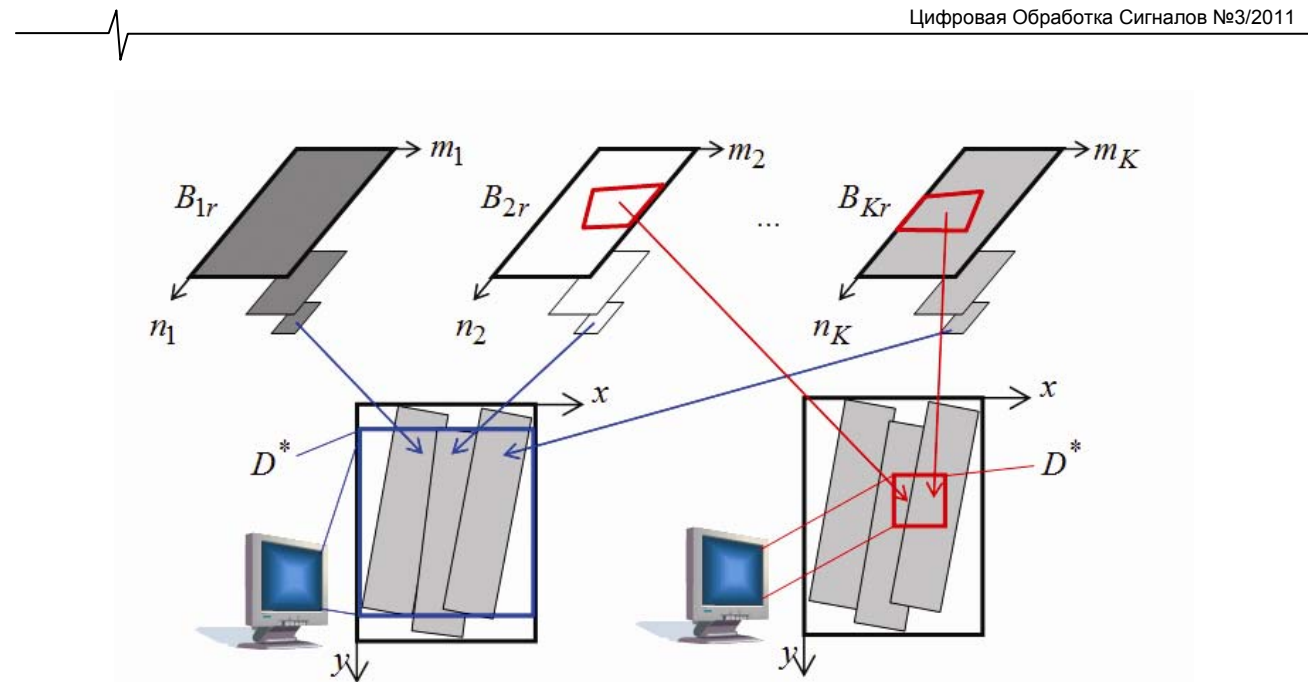
F_{kx} , F_{ky} . Кроме того, для каждого B_k вычисляются параметры глобальной и локальной яркостной нормализации изображений Φ_k . По найденным функциям геометрической $F_k = (F_{kx}, F_{ky})$ и яркостной Φ_k коррекции

из исходных изображений синтезируется мозаичный снимок D . Изображение сохраняется на жестком диске, а затем открывается для интерактивного анализа качества совмещения снимков. Если на непрерывном изображении обнаруживаются дефекты совмещения, то: формируется линия «сшивки» E , а также осуществ-

ляется редактирование ОИТ путём формирования корректирующего множества точек ΔT . По скорректированным параметрам вновь рассчитываются функции геометрического и яркостного преобразования и повторно выполняется синтез мозаичного изображения D с улучшенными характеристиками. При необходимости, процесс может повторяться до получения требуемого результата.

Можно заметить, что рассматриваемая технология отличается крайне низкой производительностью, поскольку требует многоэтапной обработки сверхбольших по объёму массивов видеоданных. Для устранения этого недостатка была разработана одноэтапная технология синтеза мозаичного изображения, схема которой приведена на рис. 2, б. В основе технологии лежит механизм виртуальной обработки данных, в соответствии с которым мозаичное изображение оперативно синтезируется в пределах области, соответствующей размерам экрана монитора. Для оперативного синтеза разномасштабных изображений используется пирамидальное представление снимков B_{kr} , где r – коэффициент прореживания исходного снимка. Пример, поясняющий технологию синтеза виртуального изображения D^* , показан на рис. 3.

В соответствии с рис. 2, б, контроль качества совмещения изображений, задание линии «сшивки» E и формирование корректирующих точек ΔT осуществляются по виртуально синтезированному изображению D^* . При этом основное сокращение временных затрат происходит за счёт того, что, во-первых, визуальный анализ выполняется по обзорным изображениям и, во-вторых, для контроля оперативно синтезируются лишь границы совмещаемых снимков.



а) Формирование полного кадра по прореженным снимкам B_{kr}

б) Формирование фрагмента детального кадра по исходным изображениям B_{k1}

Рис. 3. Иллюстрация принципа виртуального синтеза мозаичного изображения

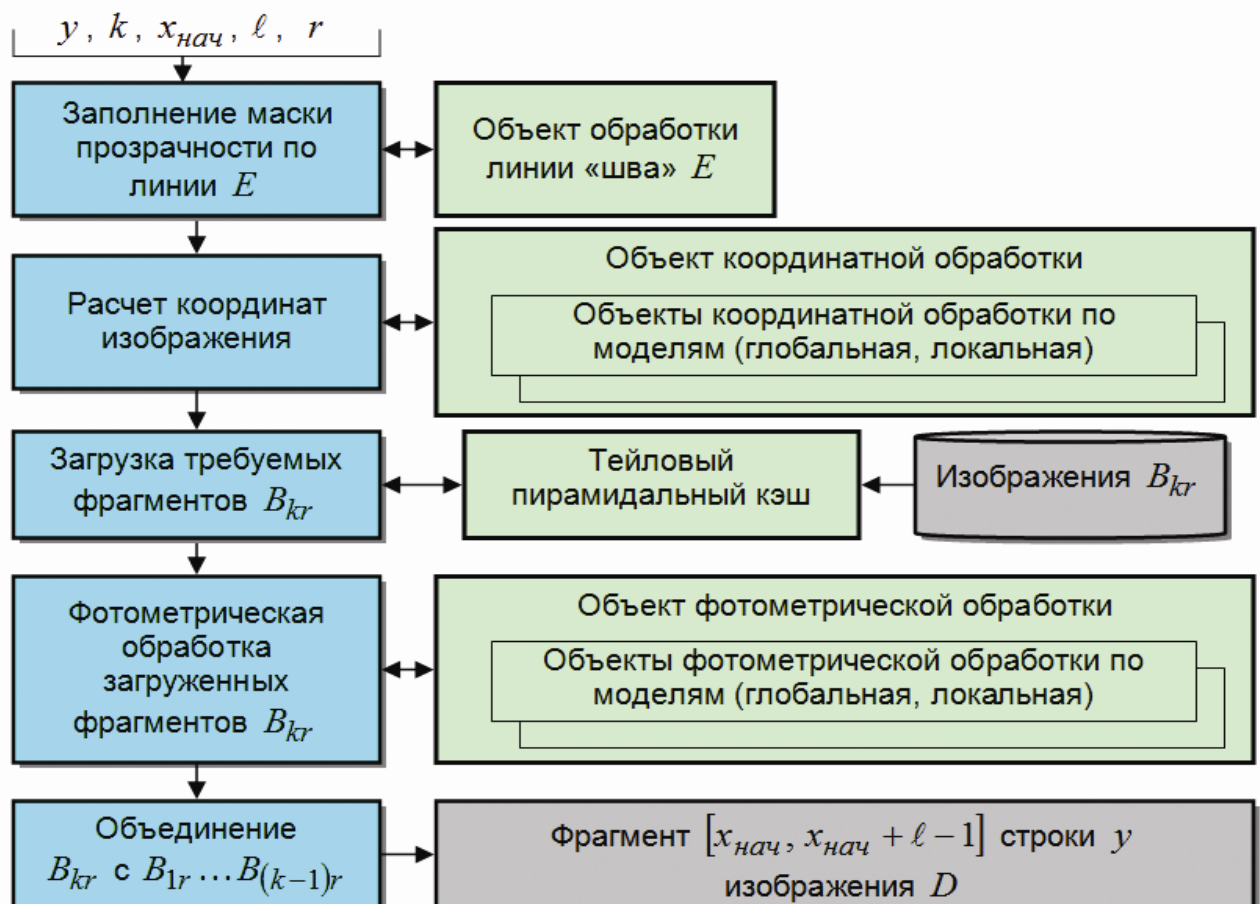


Рис. 4. Схема функционирования процессора синтеза мозаичного изображения

Физическое создание мозаичного изображения D в виде файла происходит за один проход обработки изображений B_{k1} , $k = \overline{1, K}$, по откорректированным на этапе анализа параметрам. Если не

требуется контроль над результатами обработки, то механизм виртуального синтеза не инициируется оператором, и формирование изображения D осуществляется по схеме, приведённой на рис. 2,а.



Важным элементом, позволяющим организовать технологию виртуальной обработки изображений B_k , является механизм формирования изображения D , описываемый выражениями (2) и (3). Этот механизм реализован в процессоре синтеза строк выходного изображения, путем одновременной обработки всех изображений B_k на основе функций F_k и Φ_k . Схема функционирования процессора приведена на рис. 4. Основной функционал процессора реализован в рекурсивной процедуре синтеза фрагмента строки y снимка D по данным изображений $B_0 \dots B_k$, в соответствии с которой сначала формируется фрагмент, заданный координатой начала $x_{нач}$, длиной ℓ и данными изображения B_k (верхнего слоя мозаичного снимка). Затем происходит формирование необходимых фрагментов видеоданных нижних слоев – изображений $B_0 \dots B_{k-1}$ для формирования участков, на которые не попадает изображение B_k , а также для реализации возможностей полупрозрачного отображения. Это позволит интерактивно управлять порядком объединения изображений путем исключения закрытых облачностью участков изображений с целью получения высококачественного изображения D .

Модель геометрического преобразования снимков

При разработке модели геометрического преобразования изображений B_{kr} необходимо решить две задачи:

- на основе прямых соотношений (1) получить функции обратного геометрического соответствия F_{km} , F_{kn} , используемые в выражениях (2), (3);
- выполнить локальное уточнение полученных уравнений по координатам ОИТ, идентифицированным в общих областях изображений.

Известно, что для спутниковых изображений земной поверхности прямые соотношения F_x , F_y , (здесь и далее для простоты индекс k опустим) имеют сложный вид и их аналитически обратить невозможно. Поэтому для получения обратных функций можно воспользоваться численными методами, рассмотренными в [1-3] и основанными на кусочно-билинейной или полиномиальной аппроксимации, $A: (F_x, F_y) \rightarrow (F_m, F_n)$, где A – оператор аппроксимации прямых функций координатного соответствия.

Для коррекции локальных взаимных геометрических искажений снимков были проанализированы модели «резинная плёнка» и триангуляционная. Установлено, что с точки зрения вычислительных затрат и учёта специфики функционирования процессора синтеза мозаичного снимка, триангуляционная модель наилучшим образом подходит для устранения остаточных геометрических рассогласований объединяемых изображений. В соответствии с этой моделью, вначале по алгоритму Делоне [4, 5] формируются q треугольников, вершинами которых являются одноименные точки из множества T . После чего функция координатных преобразований в выражении (2) определяется соотношениями вида:

$$m = a_q \times \begin{bmatrix} 1 \\ F_m(x, y) \\ F_n(x, y) \end{bmatrix}, \quad n = c_q \times \begin{bmatrix} 1 \\ F_m(x, y) \\ F_n(x, y) \end{bmatrix} \quad (4)$$

где q – номер треугольника, в который попадает точка (x, y) ; $a_q = [a_{0,q} \ a_{1,q} \ a_{2,q}]$, $c_q = [b_{0,q} \ b_{1,q} \ b_{2,q}]$, $q \in [1, Q]$ – векторы коэффициентов преобразований, рассчитываемых по координатам точек (m_{kj}, n_{kj}) и (x_{kj}, y_{kj}) .

Следует отметить две особенности, возникающие при использовании триангуляционной модели для синтеза мозаичного изображения. Во-первых, триангуляционная сеть, узлами которой являются ОИТ, не покрывает снимок целиком. Во-вторых, непосредственное использование триангуляционной модели предполагает для каждой точки определение номера треугольника, в который попадает точка (x, y) , что приводит к недопустимым временным затратам. Поэтому, для трансформирования всего снимка по триангуляционной модели, множество ОИТ дополняется четырьмя точками для углов снимка. А для выполнения скоростной массовой кусочно-линейной обработки сначала определяются треугольники, перекрывающие формируемую строку мозаичного изображения, и затем для каждого найденного треугольника выполняется преобразование координат попадающих в него точек.

Алгоритм глобального и локального яркостного совмещения снимков

Особенность задачи яркостного совмещения снимков, входящих в мозаику, состоит в рекурсивном воздействии на них рассчитываемых параметров яркостного преобразования. Поэтому рассмотрим алгоритм глобального яркостного комплексирования изображений, свободный от этого недостатка.

При выравнивании фотометрических характеристик снимков, входящих в мозаику, сначала найдём параметры глобального яркостного преобразования каждого изображения u_k , v_k , $k = \overline{1, K}$. Для этого проанализируем перекрытия каждой пары изображений в пределах мозаики и вычислим средние значения $\bar{b}_{ij}$ и СКО яркости σ_{ij} снимка B_i в области перекрытия со снимком B_j , $i = \overline{1, K}$, $j = \overline{1, K}$, $i \neq j$.

Затем снимки разобьём на группы таким образом, чтобы в группы входили все связанные между собой изображения. Здесь снимок считается связанным с другим снимком, если он имеет общую область с ним или со связанным с ним третьим снимком. Далее выполним независимое выравнивание яркости каждой группы. Допустим, что все K исходных снимков входят в одну группу. Для перекрытий значимой площади составляются уравнения:

$$\begin{cases} \bar{b}_{ij}u_i + v_i = \bar{b}_{ji}u_j + v_j, \\ \sigma_{ij}u_i = \sigma_{ji}u_j. \end{cases} \quad (5)$$

Чтобы исключить тривиальное решение $u_k = v_k = 0$, $k = \overline{1, K}$, яркость и контраст одного из снимков, например первого, принимаются эталонными и остаются неизменными: $u_1 = 1$, $v_1 = 0$. Коэффициенты u_k , v_k , $k = \overline{2, K}$ находятся из составленных уравнений при помощи метода наименьших квадратов.

Полученные коэффициенты изменяют в среднем яркость и контраст снимков. Чтобы этого избежать, составим дополнительные условия:

$$\sum_{k=1}^K v_k = 0, \quad \prod_{k=1}^K u_k = 1,$$

для выполнения которых рассчитанные коэффициенты нормируются:

$$u'_k = u_k \frac{1}{\sqrt[K]{\prod_{i=1}^K u_i}}, \quad v'_k = \left(v_k - \frac{1}{K} \sum_{i=1}^K v_i \right) \frac{1}{\sqrt[K]{\prod_{i=1}^K u_i}}.$$

После выравнивания яркости и контраста снимков основным препятствием для создания бесшовной мозаики являются яркостные отличия в локальных областях изображений, вызванные неравномерностью освещенности земной поверхности, локальными подобными отличиями и т.д. Для компенсации подобных отличий построим в плоскости мозаичного снимка равномерную решётку. В пределах каждой ячейки для всех изображений B_k найдём СКО яркости и средние значения яркости. Далее для каждой ячейки осредним полученные по всем изображениям значения и примем эти параметры в качестве эталонных. Для изображений B_k найдём коэффициенты линейных фотометрических преобразований, приводящие яркость и контраст в пределах ячеек к эталонным. Поскольку в части ячеек могут отсутствовать найденные эталонные значения или значения СКО и среднего значения яркости снимка, коэффициенты фотометрического преобразования не могут быть определены описанным способом. Для расчёта пропущенных значений коэффициентов фотометрического преобразования по имеющимся значениям выполним восстановление непрерывных функций значений коэффициентов с использованием модели «резиновая плёнка» и выборку значений полученных функций в местах пропусков.

Чтобы при обработке не возникали скачки яркости на границах ячеек, используем билинейную интерполяцию для определения коэффициентов фотометрического преобразования произвольного пикселя изображения.

Пример результатов фотометрического выравнивания приведён на рис. 5.

Задание произвольных границ «обрезки» изображений в мозаике выполняет оператор, исключая области с плохим геометрическим и фотометрическим совмещением или закрытые облачностью. Границы видимости снимка в мозаике удобно вводить в виде полигона, вершины которого указываются сколом «мыши». Попадание точки в полигон при заполнении строки мозаики видеоданными исходного снимка проверяется по критерию «чётности».

Следующим важным этапом локального яркостного совмещения снимков является операция устранения видимой линии «сшивки». «Шов», вызванный остаточными геометрическими и фотометрическими отличиями снимков, может быть сглажен за счёт плавного полупрозрачного перехода от одного снимка к другому. При этом для верхнего снимка определяется степень прозрачности каждого пикселя в окрестности «стыка» величиной d пикселей таким образом, чтобы пиксель на границе был прозрачным, а пиксель на расстоянии d пикселей от границы – непрозрачным. Пример работы алгоритма показан на рис. 6.

Заключение

На базе рассмотренных в статье структурных и алгоритмических решений спроектирован унифицированный программный модуль формирования мозаичных снимков.

Этот модуль позволяет формировать глобальные карты облачности подстилающей поверхности по данным от КА NOAA, MeteoSat, Meteor-M. Модуль с

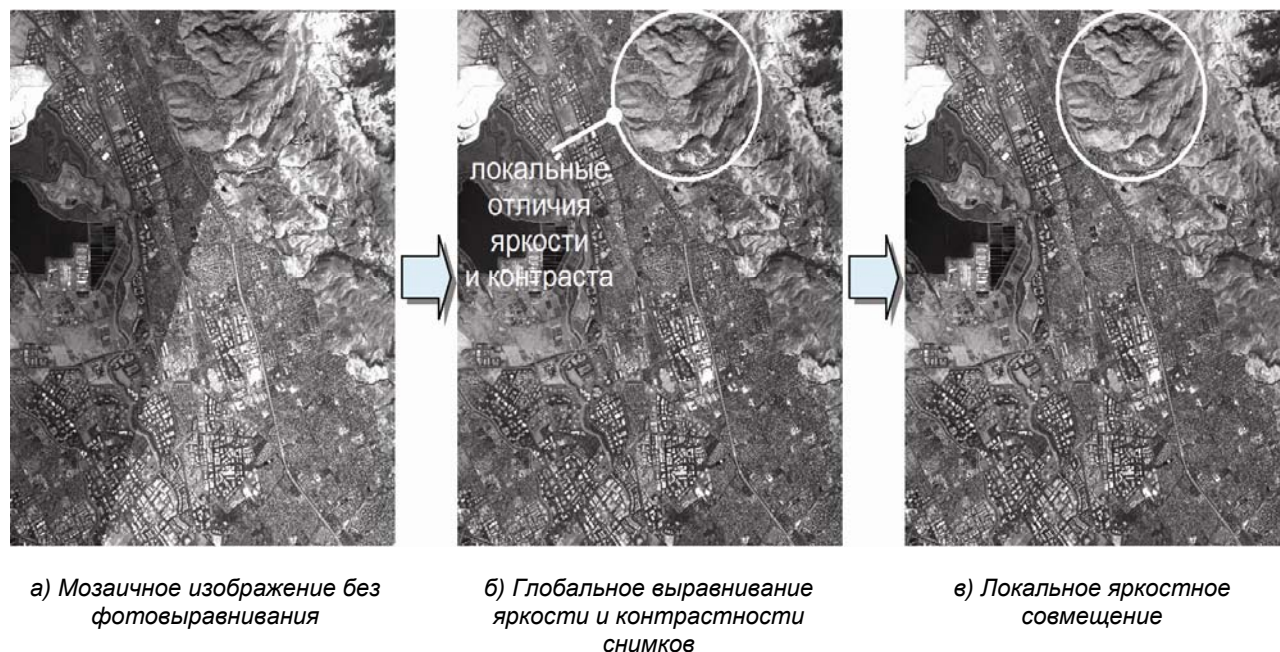


Рис. 5. Пример выполнения глобального и локального яркостного совмещения снимков

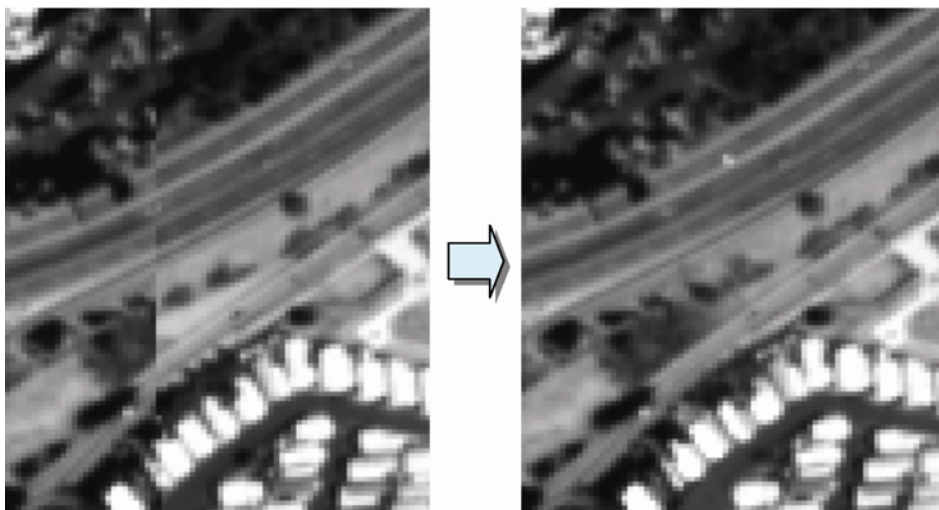


Рис. 6. Пример устранения яркостных различий вдоль линии «шивки»

успехом применяется и для синтеза высокодетальных космокарт по данным самолетного сканера Eagle и спутника «Ресурс-ДК». Получаемые при этом мозаичные снимки не уступают по качеству тем, которые формируются специализированными фотограмметрическими комплексами ERDAS, Photomod и ENVI. Однако учет специфики представления изображений и использование высокоскоростных алгоритмов обработки больших массивов данных позволяют создать на приемных центрах непрерывный технологический цикл обработки спутниковой информации.

Литература

1. Злобин В.К., Еремеев В.В., Кузнецов А.Е. Обработка изображений в геоинформационных системах: Учебное пособие. Рязань, 2006. 264. С. 80-88
2. Кузнецов А.Е., Назарцев В.В., Пресняков О.А. Получение мозаичных изображений от многокадровых систем наблюдения Земли. Материалы 16-й международной научно-технической конференции «Проблемы передачи и обработки информации в сетях и системах телекоммуникаций», Рязань, 2010. С. 113-114.
3. Vincent Tao, C., and Yong Hu, «A comprehensive study of the Rational function model for photogrammetric processing», PERS, Vol 67, No. 12, Dec 2001, 1347-1357 pp.
4. Ласло М. Вычислительная геометрия и компьютерная графика на C++: Пер. с англ. – М.: БИНОМ, 1997. С. 304.
5. L. Guibas and J. Stolfi, Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams, ACM Transactions on Graphics, 4 (2), 74-123 pp., 1985

THE ORGANIZATION OF REMOTE SENSING IMAGES MOSAICING PROCESS

Kuznetsov A.E., Presnyakov O.A.

The remote sensing images mosaicing technology based on virtual data processing procedure using pyramid image representation is proposed. The mathematical models of global and local seamless image geometric joining and intensity alignment are reviewed.

НОВОСТИ

Европейской университетской программы фирмы Texas Instruments

Новые обучающие материалы по процессору ОМАР-L138, совмещающему ядро DSP с плавающей точкой C6748 и ядро процессора ARM9, готовятся в тестовой версии. Ожидается, что данные материалы помогут пользователям перейти от C6713 DSK к новым платам «ОМАР-L138 Experimenter's Kit» с новым уровнем производительности. Чтобы получить тестовую версию материалов, обращайтесь по электронному адресу: i-bannov@ti.com

По адресу <http://www.ti.com/ww/ru/universities.htm> доступно большинство брошюр Европейской университетской программы на русском языке. Русскоязычный сайт университетской программы расширяется и модифицируется так, чтобы соответствовать основному англоязычному сайту. Если вы ещё не зарегистрированы в Университетской программе TI, вы можете сделать это по адресу: <https://focus-webapps.ti.com/ViewsFlash/servlet/viewsflash?cmd=showform&pollid=Russia!university>

УДК 681.323, 621.397.6

УЛУЧШЕНИЕ КАЧЕСТВА JPEG2000-ИЗОБРАЖЕНИЙ НА ОСНОВЕ МОДИФИЦИРОВАННОГО БИЛАТЕРАЛЬНОГО ФИЛЬТРА

Хрящев В.В., доцент кафедры динамики электронных систем Ярославского государственного университета им. П.Г. Демидова, к.т.н., e-mail: vhr@yandex.ru

Бекренев В.А., аспирант кафедры динамики электронных систем Ярославского государственного университета им. П.Г. Демидова, e-mail: vbekrenev@yandex.ru

Соловьев В.Е., аспирант кафедры динамики электронных систем Ярославского государственного университета им. П.Г. Демидова, e-mail: dcslab@uniyar.ac.ru

Никитин А.Е., аспирант кафедры динамики электронных систем Ярославского государственного университета им. П.Г. Демидова

Ключевые слова: вейвлет-преобразование, JPEG2000, звон, размытие, оценка качества изображений, билатеральный фильтр, постобработка сжатых изображений.

Введение

На сегодняшний день перспективными технологиями сжатия изображений остаются алгоритмы с использованием вейвлет-преобразования. Среди них следует выделить стандарт JPEG2000 [1, 2], превосходящий классический алгоритм JPEG с точки зрения субъективных и объективных критериев оценки качества восстановленного изображения.

JPEG2000-изображения обладают двумя типами искажений: размытие и искажение типа «звон» [3]. Данное обстоятельство делает актуальным вопрос об объективной оценке качества сжатого изображения. Традиционный подход к оценке качества основан на определении степени близости искаженного изображения к оригинальному. При этом оценка носит обобщенный характер и часто плохо коррелирует с визуально воспринимаемым качеством [4]. Современные методы оценки качества позволяют оценивать характер и уровень вносимых искажений. Как следствие, результаты, полученные на их основе, лучше коррелируют с субъективно воспринимаемым качеством и позволяют разрабатывать алгоритмы подавления искажений и улучшения качества восстановленных изображений [5, 6].

Следует заметить, что искажения размытия и звона имеют сильную нелинейную зависимость друг от друга [7]. Поэтому важными представляются задачи подавления звона при сохранении уровня размытия и увеличение четкости изображения при сохранении уровня звона. Ниже предлагается возможное решение первой задачи.

Проблема удаления звона для JPEG2000-изображений рассматривается в [8-10]. В работе [8] значение каждого пикселя изображения представляется в виде функции, зависящей от значений соседних пикселей, находящихся в пределах заданного окна. Для сохранения однородности фона и резкости границ объектов на изображении применяется набор адаптивных алгоритмов удаления шума. Адаптивный нелинейный диффузный метод для уменьшения уровня звона предлагается в работе [9]. Диффузный коэффициент определяется как линейная комбинация

Разработан модифицированный билатеральный фильтр, позволяющий понизить уровень звона в сжатом изображении и, одновременно с этим, сохранить уровень размытия. Результаты проведенного моделирования показывают преимущество предложенного алгоритма по сравнению с классическим билатеральным фильтром на 1,5-3 дБ. Анализ работы фильтров производится также на основе метрик размытия и звона, позволяющих выполнять объективную оценку искажений в сжатых изображениях.

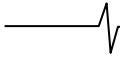
ция трёх функций принадлежности, которые соответствуют различным градиентным шкалам. Параметры подбираются посредством стохастической оптимизации, что позволяет адаптировать диффузный коэффициент к входному изображению. Алгоритм удаления звона, предложенный в [10], использует древовидную схему разбиения для постобработки восстановленных изображений пространственно-селективным образом. Стратегия голосования используется для определения набора морфологических фильтров, подавляющих звон.

Общим недостатком указанных выше алгоритмов является их высокая вычислительная сложность, что не позволяет использовать их в приложениях реального времени. В данной статье для решения поставленной задачи предлагается использовать билатеральный фильтр [11-13]. Этот алгоритм достаточно прост в реализации, обладает сравнительно низкой вычислительной сложностью и, учитывая природу звона, можно ожидать его эффективное подавление данным типом фильтра.

Работа построена следующим образом. Сначала описываются метрики объективной оценки качества изображений, используемые для определения уровня искажений, внесенных алгоритмом сжатия JPEG2000. Далее рассматривается классический билатеральный фильтр и предлагается его модификация, предназначенная для удаления артефактов звона. Приводится сравнительный анализ предложенного модифицированного билатерального фильтра с другими видами фильтрации на основе ряда критериев оценки качества. В заключении приводятся общие выводы относительно полученных результатов.

Метрики оценки уровня размытия и звона

В работах [14, 15] предложен подход, позволяющий оценивать размытие и звон в JPEG2000-изображениях. Доработанный и программно реализованный, он был положен в основу расчетов метрик размытия и звона. Расчёт метрики размытия (MP) производится следующим образом [7]:



1. Рассмотрим строку изображения $\{x_1, x_2, \dots, x_N\}$, где x_i – значение яркости пикселя, N – число пикселей в строке. Используя фильтр Собеля, выделим границы изображения, содержащиеся в строке. Получим массив значений, содержащий S положений границ в каждой из M строк изображения: $\{p_1, p_2, \dots, p_S\}$, $S < N$.

2. Для каждой точки p_s найдём положение ближайшего слева локального экстремума яркости l_s и ближайшего справа локального экстремума яркости r_s . Найдём разницу между определенными положениями: $w_s = r_s - l_s$. Полученное значение w_s назовём локальной метрикой размытия границы.

3. Метрику размытия определим как усреднённое значение локальных метрик размытия, вычисленных по всем строкам изображения:

$$MP = \frac{\sum_{j=1}^M \sum_{s=1}^S w_{s,j}}{\sum_{j=1}^M S_j}.$$

Следует отметить, что описанный выше алгоритм может иметь эталонную и неэталонную реализации. Эталонная реализация использует для определения границ исходное, а неэталонная – декодированное изображение.

Для оценки уровня звона воспользуемся результатами, полученными на этапе вычисления метрики размытия.

1. Предположим, что ширина каждой границы в каждой строке изображения известна и сохранена в переменной $w_{s,j}$. Вычислим локальную метрику звона вокруг границы изображения $(w_{ring})_s$, определяемую положением p_s , следующим образом:

$$(w_{ring})_s = |\max(x_i - \tilde{x}_i) - \min(x_i - \tilde{x}_i)| \times |w_f - l_s| + |\max(x_j - \tilde{x}_j) - \min(x_j - \tilde{x}_j)| \times |w_f - r_s|,$$

$$i \in [p_s - w_f, l_s], j \in [r_s, p_s + w_f],$$

где x_i – значение яркости пикселя оригинального изображения, $\tilde{x}_i$ – значение яркости пикселя искажённого изображения, w_f – фиксированная ширина звона, равная 16 пикселям, которая была получена на основе анализа воздействия искажающих факторов на идеальный сигнал. Если отрезки $[p_s - w_f, l_s]$ и $[r_s, p_s + w_f]$ не существуют, то полагаем $(w_{ring})_s$ равной 0.

2. Усредним локальные метрики звона: – по тем p_s , для которых величина $(w_{ring})_s$ отлична от 0. [Мы получим метрику звона (МЗ), которая отражает амплитуду звона с учётом размера области его фактического проявления и не зависит от других артефактов сжатия]; по всем p_s . [Мы получим общую метрику звона (МЗ_{общ}), которая отражает амплитуду звона с учётом размеров области его теоретического проявления и зависит от других искажений, вызванных сжатием изображения]:

$$МЗ = \frac{\sum_{j=1}^M \sum_{s=1}^S (w_{ring})_{s,j}}{\sum_{p_s} 1}, \quad МЗ_{общ} = \frac{\sum_{j=1}^M \sum_{s=1}^S (w_{ring})_{s,j}}{\sum_{j=1}^M S_j}.$$

На основе проведенных экспериментов были найдены пороговые значения, что позволило произвести нормировку указанных метрик размытия и звона в интервал $[0, 1]$.

Заметим, что алгоритм вычисления метрики звона использует оригинальное изображение, т.е. допускает только эталонную реализацию. Он содержит все необходимые вычисления для определения уровня размытия, поэтому обе метрики могут быть вычислены одновременно.

Модифицированный билатеральный фильтр

Кратко рассмотрим алгоритм билатеральной фильтрации [11, 12]. Пусть x – координата рассматриваемого пикселя, и пусть $\Omega = \Omega_x(n)$ представляет множество соседних с x пикселей в окрестности размером $(2n+1) \times (2n+1)$, координаты которых будем обозначать как y . Вес каждого пикселя с координатами $y \in \Omega_x$ относительно x зависит от двух компонент – пространственной $w_{sp}(x, y)$ и радиометрической $w_{rad}(x, y)$: $w(x, y) = w_{sp}(x, y)w_{rad}(x, y)$, где

$$w_{sp}(x, y) = e^{-\frac{|x-y|^2}{2\sigma_{sp}^2}}, \quad w_{rad}(x, y) = e^{-\frac{|u_x - u_y|^2}{2\sigma_{rad}^2}},$$

u_x – значения пикселей с координатами x , u_y – значения пикселей с координатами y , σ_{sp} – пространственное среднеквадратичное отклонение, σ_{rad} – яркостное среднеквадратичное отклонение.

Эти веса подвергаются нормировке, поэтому пиксели восстановленного изображения $\tilde{u}_x$ вычисляются как

$$\tilde{u}_x = \frac{\sum_{y \in \Omega} w(x, y)u_y}{\sum_{y \in \Omega} w(x, y)}.$$

Проведём анализ влияния ряда сглаживающих фильтров на искажения, вносимые в процессе сжатия. Изображения, сжатые алгоритмом JPEG2000, обрабатываем линейным низкочастотным, медианным и билатеральным фильтрами [2]. С помощью метрик размытия и звона проведем контроль уровня искажений до и после процедуры фильтрации. Результаты исследования для тестового изображения «Лена» приведены на рис. 1.

Установлено, что наибольшее подавление звона достигается при обработке билатеральным фильтром. С другой стороны, именно этот тип фильтрации вносит наибольшее размытие границ, что ухудшает зрительное восприятие изображения [16, 17]. С учётом полученных результатов проведем модификацию билатерального фильтра. Будем обрабатывать не всё изображение (и даже не все окрестности границ), а только те фрагменты, в которых звон действительно проявляется. На основе метрики звона составим соответствующую карту звона для изображения, и далее будем обрабатывать только те пиксели, которые принадлежат этой карте. Данная схема обработки изображения представлена на рис. 2.

Работа модифицированного билатерального фильтра включает в себя следующие этапы:

1. Выделение всех границ в сжатом изображении и составление отрезков $[p_s - w_f, l_s] \cup [r_s, p_s + w_f]$ для каждого положения границы p_s .

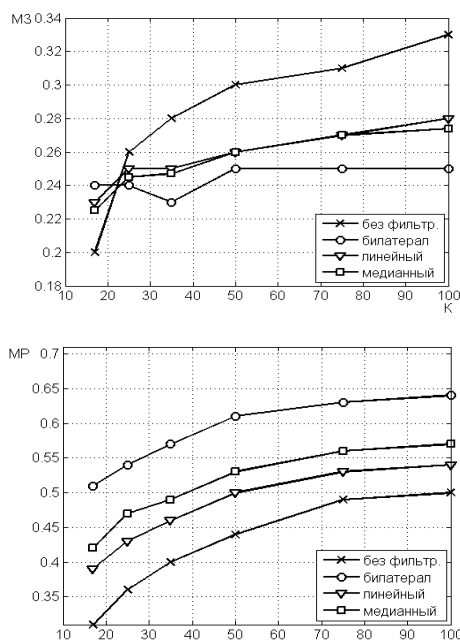


Рис. 1. Зависимости метрики звона (а) и метрики размытия (б) от коэффициента сжатия K до и после различных вариантов постобработки

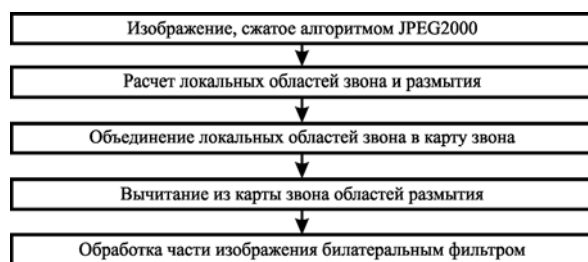


Рис. 2. Предлагаемый алгоритм постобработки JPEG2000-изображения

Нанесение отрезков на соответствующую карту – пустое изображение, размеры которого равны размерам исходного изображения. Поскольку метрика звона учитывает только вертикальные границы, следует применить эту же последовательность действий к горизонтальным границам.

- Удаление из карты звона окрестностей границ для сохранения уровня размытия изображения. Сделать это можно, объединяя все области, по которым подсчитывается метрика размытия, и вычитая полученное множество точек из карты звона. Пример итоговых карт звона для трёх тестовых изображений приведён на рис. 3.
- Фильтрация изображения билатеральным фильтром. Маска фильтра применяется только к тем пикселям, которые принадлежат карте звона.

Результаты моделирования

Реализация и исследование модифицированного билатерального фильтра проводились в среде Matlab. Наилучший уровень подавления звона достигается при следующих параметрах модифицированного билатерального фильтра [17]: размер маски фильтра 5×5 , $\sigma_{sp} = 3$ и

$$\sigma_{rad} = 0,5.$$

На рис. 4 приведены зависимости метрики звона и метрики размытия от коэффициента сжатия K при обработке билатеральным и модифицированным билатеральным фильтрами. Как следует из представленных зависимостей, желаемая цель подавить звон, не увеличив размытие изображения, достигнута.

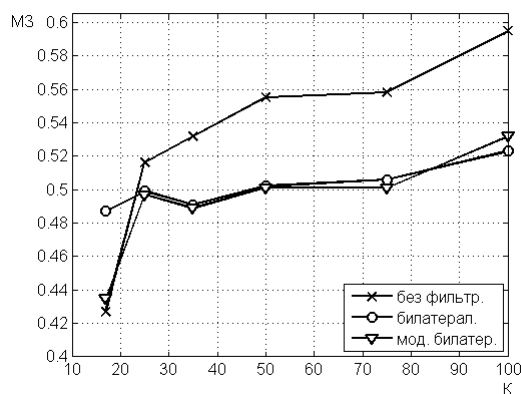
Сравнительный анализ работы билатерального и модифицированного билатерального фильтров по метрике пикового отношения сигнал/шум (ПОСШ) приведён на рис. 5.

Как следует из приведённых зависимостей, преимущество модифицированного билатерального фильтра достигает 1,5–3 дБ, причём наибольшая разница наблюдается при малых коэффициентах сжатия, что важно для практических приложений алгоритма JPEG2000.

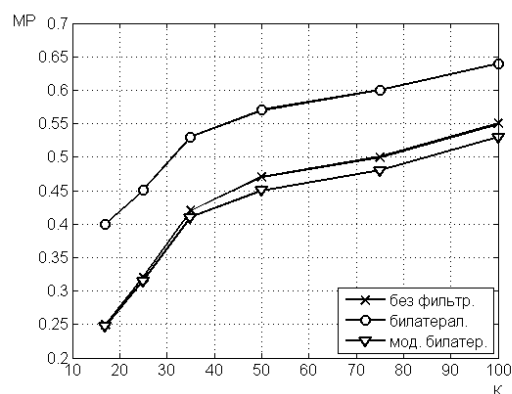
Следует также отметить, что вычислительная слож-



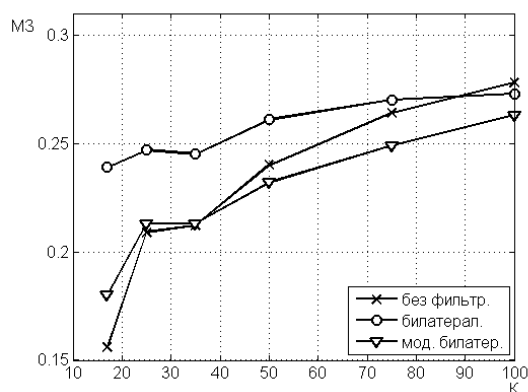
Рис. 3. Результаты вычисления карты звона для тестовых изображений «Лена», «Скарлет» и «Перцы»: оригинальные изображения (а-в), соответствующие им карты звона (г-е)



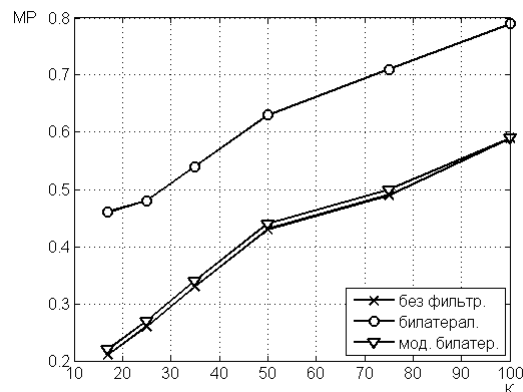
а)



б)

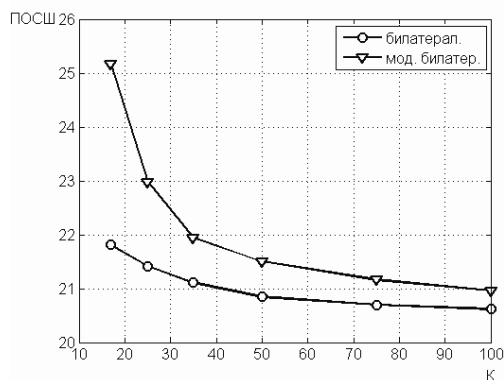


в)

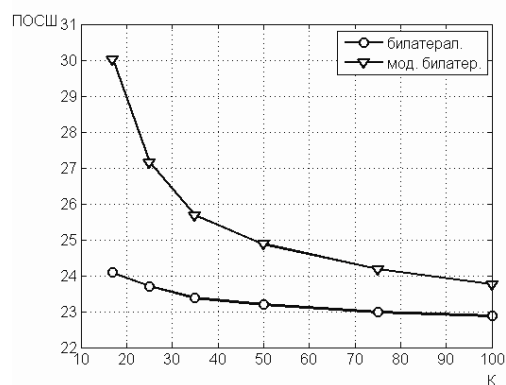


г)

Рис. 4. Зависимости метрики звона (M3) и метрики размытия (MP) от коэффициента сжатия K для тестовых изображений «Бабуин» (а, б) и «Барбара» (в, г) до и после билатеральной фильтрации



а)



б)

Рис. 5. Зависимость ПОСШ от коэффициента сжатия K для тестовых изображений «Бабуин» (а) и «Барбара» (б) при использовании классического и модифицированного билатерального фильтров.

ность модифицированного билатерального фильтра ниже, чем у классического билатерального фильтра, т.к. обработке подвергается только часть пикселей изображения.

Заключение

Рассмотрена возможность использования билатерального фильтра для снижения уровня звона в JPEG2000-изображениях. В ходе предварительных исследований установлено, что билатеральный фильтр позволяет эффективно устранить звон и, одновременно с этим, увеличивает уровень размытия JPEG2000-изображения. С целью сохранения

уровня размытия обрабатываемого изображения выполнена модификация классического билатерального фильтра.

Модификация билатерального фильтра заключается в составлении карты звона – пространственной области проявления звона для данного изображения. Применение разработанного фильтра приводит к снижению уровня звона в изображении, при этом размытие границ не усиливается. Результаты моделирования показывают преимущество модифицированного билатерального фильтра по сравнению с классическим билатеральным на 1,5-3 дБ применительно к задаче постобработки сжатых изображений.

Предложенный модифицированный била-теральный фильтр может быть использован в системах обработки и передачи визуальной информации и видеоархивирования.

Литература

1. Taubman D.S., Marcellin M.W. JPEG2000: Image Compression Fundamentals, Standards, and Practice // Norwell, MA: Kluwer, 2001.
2. Гонсалес Р., Вудс Р. Цифровая обработка изображений // М.: Техносфера, 2005.
3. Wang Z., Bovik A.C. Modern image quality assessment // Synthesis lectures on image, video & multimedia processing. Morgan & Claypool, 2006.
4. Wang Z., Bovik A.C. Mean squared error: Love it or leave it? – A new look at signal fidelity measures // IEEE Signal Processing Magazine, 2009. V. 26, N. 1, 98-117 pp.
5. Зараменский Д.А., Приоров А.Л., Хрящев В.В. Незатонная оценка качества изображений, сжатых на основе вейвлет-преобразования // «Успехи современной радиоэлектроники», 2009. №7. С. 28-34.
6. Nosratinia A. Postprocessing of JPEG-2000 Images to Remove Compression Artifacts // IEEE Signal Processing Letters, V. 10, 296-299 pp., 2003.
7. Зараменский Д.А., Бекренев В.А., Соловьев В.Е. Оценка уровня размытия и звона в изображениях стандарта JPEG2000 // Докл. 11-й междунар. конф. «Цифровая обработка сигналов и её применение» (DSPA-2009), Москва, 2009. Т. 2. С. 476-479.
8. Shen M., Kuo C. Artifact reduction in low bit rate wavelet coding with robust nonlinear filtering // IEEE Second Workshop on Multimedia Signal Processing, 480-485 pp., 1998.
9. Yao S., Lin W., Lu Z., Ong E., Yang X. Adaptive nonlinear diffusion processes for ringing artifacts removal on JPEG 2000 images // IEEE Int. Conf. on Multimedia and Expo (ICME), V. 1, 691-694pp., 2004.
10. Chen Y., Chang Y., Yen W. Design of a De-Ringing Filter for Wavelet-Based Compressed mage // Proc. of the 23rd Int. Technical Conf. on Circuits/Systems, Computers and Communications (ITC-CSCC), 1265-1268 pp., 2008.
11. Tomasi C., Manduchi R. Bilateral filtering for gray and color images // Sixth Int. Conf. on computer vision, 1998. 839-846 pp.
12. Elad M. On the origin of the bilateral filter and ways to improve it // IEEE Trans. on image processing, 2002. V. 11, № 10. 1141-1151 pp.
13. Szeliski R. Computer Vision: Algorithms and Applications – Springer, 2010.
14. Marziliano P., Dufaux F., Winkler S., Ebrahimi T. Perceptual blur and ringing metrics: Application to JPEG2000 // Signal Process.: Image Commun., Feb. 2004. V. 19, №2. 63-172 pp.
15. Simoncelli E.P. Statistical models for images: Compression, restoration and synthesis // IEEE Asilomar Conf. on Signals, Systems, and Computers, Nov. 1997. V. 1. 673-678 pp.
16. Khryashchev V., Priorov A., Shmaglit L. JPEG2000 Ringing Artifact Reduction by Smart Bilateral Filter // Proc. of the 2010 International Conference on Image Processing, Computer Vision & Pattern Recognition (ICPV 2010), Las Vegas Nevada, USA, 2010, 622-627 pp.
17. Nikitin A., Solovyev V., Khryashchev V., Priorov A. Adaptive Bilateral Filter for JPEG2000 Deringing // Proc. of the 2011 International Conference on Image Processing, Computer Vision & Pattern Recognition (ICPV 2011), Las Vegas Nevada, USA, 2011, 144-149 pp.

JPEG2000 IMAGES QUALITY ENHANCEMENT BASED ON MODIFIED BILATERAL FILTER

**V.V. Khryashchev, V.A. Bekrenev, V.E. Solovyev,
A.E. Nikitin**

Modified bilateral filter was developed. Proposed filter allows to eliminate ringing artifacts in compressed image while keeping blurring on the same level. Experimental results prove that modified bilateral filter performs better than original bilateral filter on 1,5-3 dB. Filter performance analyses was based on blurring and ringing level metrics, which allow to assess objective quality of JPEG2000 compressed images.

НОВОСТИ

Европейской университетской программы фирмы Texas Instruments

Новый обучающий диск по процессорам TMS320C55xx – «C5505 Teaching ROM» – доступен для скачивания с web-сайта: <http://www.ti.com/ww/eu/university/roms.html>. В двадцати главах рассматривается введение в теорию цифровой обработки сигналов, включая КИХ- и БИХ-фильтры, адаптивные фильтры МНК, алгоритм Герцеля, сжатие речи и вейвлет-преобразование, с акцентом на практическую реализацию на базе процессоров TMS320C5505 и TMS320C5515. Каждая глава сопровождается полностью рабочими примерами Сиг-кодов, исполняемых на платах TSM320C5505 и TMS320C5515 USB Stick. Дополнительно приводится ряд забавных примеров программного обеспечения: вы сможете говорить «голосом инопланетяни-

на» или играть на электрогитаре с эффектами эха и фазовых искажений.

С того же сайта может быть скачено **руководство «Yeditepe DSK Lab Manual»**, которое поможет вам быстро начать работу со стартовыми наборами разработчика на процессоре **TMS320C6713** в средах разработки Code Composer Studio версий 3,3 и 4,1. Руководство подготовлено Университетом «Yeditepe University», Стамбул, Турция, 2010 год. Полезно для перехода на использование новой версии среды CCS 4,1. Включает разделы, посвященные описанию сред CCS 3.3 и CCS 4,1, аудиоэффектам, спектральному анализу, фильтрации обработке изображений и использованию прерываний на процессорах указанного семейства.

КОМБИНИРОВАННЫЙ АЛГОРИТМ СЕГМЕНТАЦИИ ИЗОБРАЖЕНИЙ ПОЛУФАБРИКАТОВ КОЖИ

Бабаян П.В., к.т.н., начальник лаборатории Рязанского государственного радиотехнического университета, e-mail: aitu@rsreu.ru

Суходольская Н.В., инженер Рязанского государственного радиотехнического университета, e-mail: aitu@rsreu.ru

Ключевые слова: сегментация, поиск пороков, метод k -средних, пороговая обработка, кожа.

Введение

Одним из наиболее трудоёмких технологических процессов при производстве кожи является обработка полуфабрикатов. Она включает порядка десяти основных производственных этапов, на трёх из которых осуществляется сортировка полуфабрикатов. Сорт кожи зависит от дефектов, которыми она поражена (эти дефекты принято называть пороками), их площади и расположения. Фактически, чтобы определить сорт кожи, необходимо вначале выделить пороки кожи и оценить их характеристики. Поскольку сортировка производится вручную, на её результаты большое влияние оказывает субъективность человеческого восприятия. Отсюда возникает задача автоматизации данного процесса с целью улучшения качества сортировки [1].

На сорт полуфабриката кожи влияют следующие параметры: наличие пороков, расположение пороков, тип пороков. Насчитывается около 30 различных типов пороков, основными из которых являются бактериальность, дыра, прорезь, свищ, вортистость и сдир. Разрабатываемая система должна автоматически выделять пороки на изображениях полуфабриката, однако, кожа обладает большой неоднородностью, и пороки одного и того же типа на различных полуфабрикатах могут выглядеть по-разному. К тому же, на изображении кожи могут иметь место перепады яркости, возникшие по причине неоднородности освещения, т.е. на полуфабрикате может находиться как порок, так и участок кожи, близкий по яркости к этому пороку. И наоборот, фон может оказаться близким по яркости к полуфабрикату кожи. По этим причинам невозможно получить эталонные изображения или эталонные текстуры пороков. Выделение и оценивание параметров пороков на изображениях полуфабрикатов является задачей сегментации. Классические методы сегментации в такой ситуации не показывают желаемых результатов, поэтому необходимо разработать алгоритм, учитывающий описанные особенности.

В настоящей работе описан алгоритм сегментации изображений полуфабрикатов кожи, основанный на комбинированном использовании методов k -средних и пороговой обработки.

В настоящей работе описан алгоритм сегментации изображений полуфабрикатов кожи, основанный на комбинированном использовании методов k -средних и пороговой обработки. Для устранения проблем неоднородности освещения используется разбиение изображения полуфабриката на квадратные блоки. Особенностью разработанного алгоритма является то, что все блоки по некоторым параметрам разделяются на три типа, для каждого из которых используется соответствующий метод сегментации. Представлены результаты экспериментальных исследований комбинированного алгоритма на реальных изображениях полуфабрикатов.

Алгоритм сегментации

Для устранения проблем неоднородности освещения предлагается разбить все изображение полуфабриката на квадратные блоки шириной w , в которых неоднородность освещения является незначительной. Все эти блоки классифицируются на три типа, для каждого типа используется свой метод сегментации.

Вначале вычисляется ряд параметров, используемых для классификации блоков. С этой целью используется метод k -средних. Он разбивает множество элементов векторного пространства на заранее известное число кластеров k . В результате работы алгоритма, в соответствии с числом кластеров, для каждой области вычисляется математическое ожидание яркости области, дисперсия и вес (доля). Зная эти параметры, можно построить график аппроксимации плотности распределения яркости смеси k гауссоид [2].

Вернемся к описанию алгоритма сегментации. Параметры классификации, используемые в алгоритме, представлены в таблице 1.

Заметим, что локальными считаются параметры, вычисляемые для каждого блока. Глобальные параметры вычисляются для всего изображения.

Для вычисления параметров, описывающих характер границ (G и N_g), выполняется сегментация блока по методу Отсу [3]. В итоге получается бинарное изображение, и данные параметры вычисляются для каждого сегмента полученного изображения. Использование параметров, описывающих характер границ, обусловлено тем, что среднее значение градиента должно быть достаточно велико, если этот сегмент соответствует пороку, а неоднородность изображения, связанная с перепадом яркости, но не относящаяся к пороку, будет иметь размытую границу. Кроме того, количество точек границы порока, как правило, значительно меньше, чем у ложно выделенного сегмента.

Зная все необходимые параметры, можно отнести блоки к одному из трёх типов (таблица 2).

Таблица 1.

Параметры, используемые для классификации блоков

Обозначение параметра	Описание параметра	Способ вычисления	Признак
m_K	Математическое ожидание яркости кожи	Метод k-средних, при $k = 2$	глобальный
m_F	Математическое ожидание яркости фона		глобальный
G	Среднее значение градиента на границах сегментов	Используется бинарное изображение, полученное при сегментации по методу Оtsu	локальный
N_g	Количество граничных точек сегмента		локальный
m_1, m_2	Математическое ожидание яркости. Описывает бимодальную гистограмму	Метод k-средних при $k = 2$	локальные
N_s	Количество сегментов	Разметка бинарного изображения	локальный

Таблица 2.

Типы блоков

Тип блока	Характеристики	Условие
I тип	Содержит непораженную кожу и пороки	$A = (0,005w < N_g < 0,025w) \wedge (T_1 < G < T_2) \wedge (m_1 - m_2 > T_3)$
II тип	Содержит только непораженную кожу или кожу и порок, но кожа занимает большую часть площади	$\bar{A} \wedge B$ $B = (m_1 - m_2 < T_3) \wedge (G < T_2)$
III тип	Может содержать непораженную кожу и пороки. Присутствуют значительные перепады яркости	$\bar{A} \wedge \bar{B}$

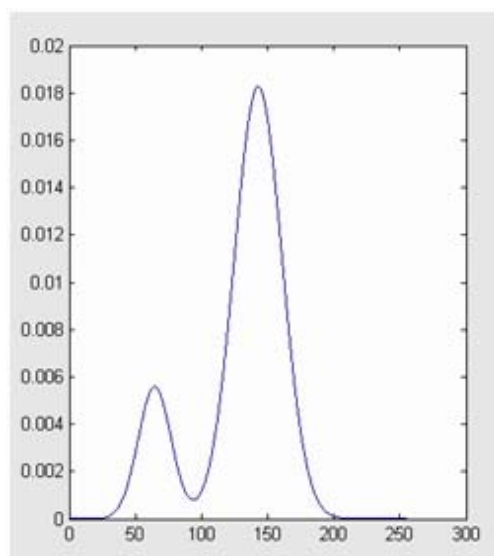


Рис. 1. График аппроксимации плотности распределения яркости смеси гауссоидов и параметры для блока I типа

$$(m_1 = 63,4; m_2 = 141,9; N_g = 283; N_s = 1; G = 12,8)$$

Для блоков I типа, как правило, аппроксимация плотности распределения яркости смеси гауссоидов имеет два пика (рис. 1), удалённых друг от друга на достаточное расстояние, что соответствует двум различающимся по яркости областям в блоке, причём доля точек, соответствующих каждой области, достаточно велика. Количество граничных точек для блоков I типа мало по сравнению с размером блока, следова-

тельно, с большой вероятностью, области в блоке имеют четко выраженные границы. Всё это свидетельствует о том, что блок содержит как кожу, так и поражённый порок, вызванные неравномерностью освещения. Для сегментации блоков I типа используется метод Оtsu.

Для блока II типа аппроксимация плотности распределения яркости смеси гауссоидов (рис. 2) имеет обычно один ярко выраженный пик. Блок содержит только непораженную кожу, или непораженную кожу и порок, но доля точек, относящихся к пороку, мала по сравнению с размером блока. В подобном случае для сегментации целесообразно использовать следующий алгоритм.

Для принятия решения вычисляется среднее значение яркости во всем блоке – m . Затем можно выделить три случая:

- Первый – $m > m_K$. При выполнении этого условия, считаем, что весь блок соответствует непораженному участку кожи.
- Второй – $\frac{m_F + m_K}{2} < m < m_K$. В данном случае, все точки, яркость которых меньше m_F , относятся к порокам, а остальные к непоражённой коже.
- Третий – $m < \frac{m_F + m_K}{2}$. В этом случае все точки, яркость которых меньше m_K , считаются пороками и наоборот, те точки, яркость которых больше m_K , рассматриваются участками, свободными от повреждений.

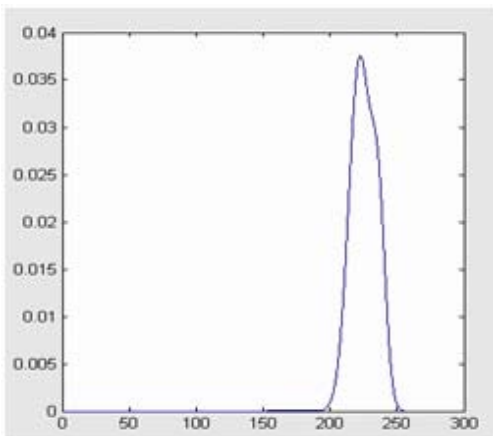


Рис. 2. График аппроксимации плотности распределения яркости смеси гауссоид и параметры для блока II типа

$$(m_1 = 235,5; m_2 = 221,3; N_g = 175; N_s = 0)$$

Аппроксимация плотности распределения для блоков III типа имеет, как правило, два ярко выраженных пика (рис. 3). Исходя из этого, для предварительной сегментации используется метод Оtsu. Полученное бинарное изображение параметризуется [4] и обрабатывается в отдельности каждый его сегмент. Если среднее значение градиента на границе такого сегмента больше некоторой фиксированной величины, значит, присутствует значимый перепад яркости, и сегмент соответствует пороку. В противном случае, сегмент стоит классифицировать как участок полуфабриката не пораженный пороками. Такой сегмент удаляется из изображения.

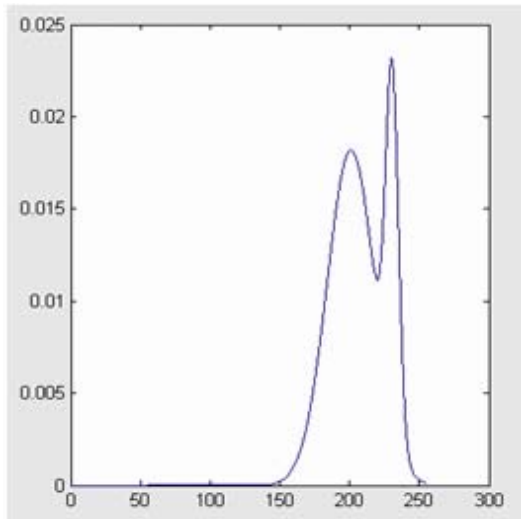


Рис. 3. График аппроксимации плотности распределения яркости смеси гауссоид и параметры для блока III типа

$$(m_1 = 200,267; m_2 = 229,89; N_g = 636; N_s = 39; G = [2,6; 2,9; 1,3; 1; 1,1; 0,5; \dots]).$$

После того, как все блоки будут отсегментированы, процесс обработки заканчивается. Из полученных блоков восстанавливается целостное изображение.

Экспериментальные исследования

Исследования разработанного алгоритма были проведены на реальных изображениях

полуфабрикатов кож, полученных из видеосюжетов, снятых на производстве. Исходя из требований, предъявляемых к качеству сортировки, доля неверно отсегментированных пикселей на изображении полуфабриката не должна превышать 7%. Удовлетворительные результаты были достигнуты при следующих параметрах:

$$T_1 = 1,5; T_2 = 2; T_3 = 17; w = 143.$$

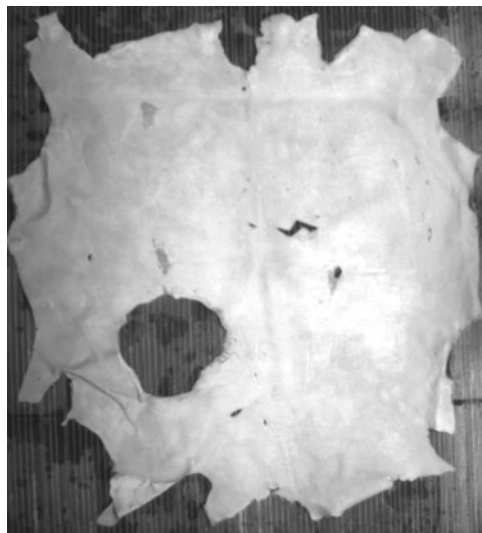


Рис. 4. Полуфабрикат кожи



Рис. 5. Результат сегментации

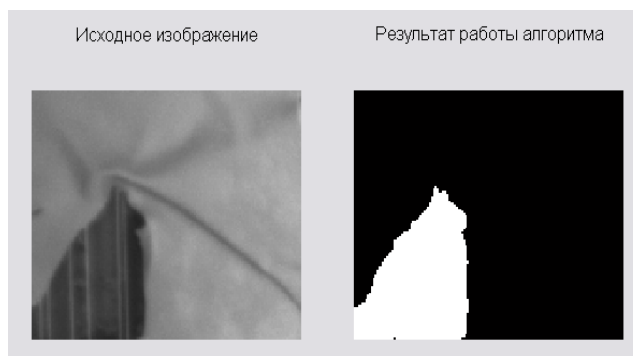


Рис. 6. Результат сегментации блока I типа

На рис. 4, 5, 6 приведены примеры работы алгоритма на блоках различных типов.

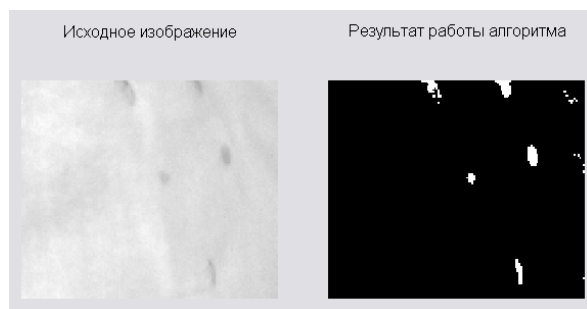


Рис. 7. Результат сегментации блока II типа



Рис. 8. Результат сегментации блока III типа

Для оценки точности работы алгоритма изображения блоков, полученные в результате сегментации, сопоставлялись с изображениями, на которых сегментация была проведена вручную. Для эксперимента использовались 50 блоков. Процент неверно отсегментированных пикселей усреднен по всем изображениям.

Результаты сегментации по представленному алгоритму сравнивались с результатами сегментации по методу Отсу и сегментации по водоразделам с использованием маркеров [5].

Результаты представлены в таблице 3.

Заключение

Таким образом, алгоритм работает с достаточно высокой точностью и имеет характеристики значительно лучшие, чем у других методов сегментации, с которыми проводилось сравнение.

Предполагается использование разработанного алгоритма в системе технического зрения для сортировки полуфабрикатов кожи.

Исследования проводились при поддержке гранта Президента РФ для молодых учёных (МК-20.2011.10).

Результаты эксперимента

Таблица 3.

	Доля неверно отсегментированных пикселей (%)
Комбинированный алгоритм	2,72
Для блоков I типа	4,65
Для блоков II типа	0,21
Для блоков III типа	1,08
Метод Отсу	6,1
Сегментации по водоразделам с использованием маркеров	10,3

Литература

1. Бабаян П.В., Серёгина Н.В. Система технического зрения для определения сорта полуфабриката кожи // Обработка информации в автоматических системах: Сборник научных трудов. Рязань: РГРТУ, 2009. – С. 30-35.
2. Лагутин М.Б. Наглядная математическая статистика. М.: П-центр, 2003.
3. Гонсалес Р., Вудс Р., Цифровая обработка изображений. М.: ТЕХНОСФЕРА, 2006.
4. Степашкин А.И., Хлудов С.Ю. Выделение и параметризация связанных областей двумерного изображения // Обработка информации в системах управления: Межвузовский сборник научных трудов. – Рязань: РРТИ, 1989.
5. Бабаян П.В., Серёгина Н.В. Сегментация изображений полуфабрикатов кожи // Вестник Рязанского государственного радиотехнического университета. – 2010. – №4.

THE COMBINED ALGORITHM OF IMAGE SEGMENTATION OF A HALF-FINISHED LEATHER

Babayan P. V, Sukhodolskaya N.V.

In the presented work the algorithm of image segmentation of the leather based on the combination of K-mean and thresholding methods is described. To overcome the problem of the illumination heterogeneity the observed image of leather is divided on square regions. The main feature of the developed algorithm is that all blocks can be classified into three types depending on several parameters. For each type of the block the specified method of the segmentation is used. The proposed algorithm shows better results in comparison with other considered segmentation approaches.

Уважаемые коллеги!

Для тех, кто не успел оформить подписку на первое полугодие 2012 года через ОАО «Роспечать», сохраняется возможность приобретения журналов непосредственно в редакции по адресу: 107031, г. Москва, Рождественка, 6/9/20, стр. 1, Российское научно-техническое общество радиотехники, электроники и связи им. А.С. Попова, или оформить Заказ в соответствии с требованиями, выставленными на сайте журнала: www.dsra.ru.

**Справки по телефонам: (495) 621-71-08, 621-06-10.
Факс: (495) 621-16-39. E-mail: nto.popov@mtu-net.ru.**

РЕАЛИЗАЦИЯ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ ПЕРЕМЕННОЙ ТОЧНОСТИ НА ПЛИС

Вичужанин В.В., доцент «кафедры Информационных технологий» Одесского национального морского университета, e-mail: vint532@yandex.ru

Сегодня проектировщики для реализации сложных алгоритмов цифровой обработки сигналов (ЦОС) применяют программируемые логические интегральные схемы (ПЛИС). Такому подходу способствуют развитие технологии программируемой логики, сопровождаемое как появлением новой номенклатуры ПЛИС, так и открывающимися возможностями реализации новых классов алгоритмов с различными требованиями к точности систем ЦОС.

Регулярный рост производительности современных ПЛИС в настоящее время происходит за счет уменьшения линейных размеров элементарных вентилей, выполняемых сегодня по 28 нм технологии [1]. Это дает производителям ПЛИС возможность увеличивать количество конфигурационных логических блоков в кристалле и приводит к значительному снижению стоимости кристаллов ПЛИС. Такая тенденция сохранится и в дальнейшем, стимулируя увеличение производительности вычислительных средств, осуществляющих ЦОС, что приведет к дальнейшему развитию применений ЦОС в различных областях радиолокации и гидролокации, медицинской визуализации изображений, цифровом аудио- и телевизионном вещании и т.д. Неотъемлемой частью этого процесса будет увеличение доли ПЛИС как основы для решения задач ЦОС, требующих обеспечения высокой помехоустойчивости систем обработки данных, необходимой точности и разрешающей способности, простого сопряжения подсистем обработки сигналов, стабильности параметров тракта обработки информации.

В различных технических приложениях для высокопроизводительной цифровой обработки сигналов (радиолокационные системы, беспроводные станции связи, медицинские приложения с высокоточной фильтрацией) часто необходима точность ЦОС выше, чем обычный диапазон в 18 бит. Требования к точности варьируются, прежде всего, на стадиях проектирования КИХ-фильтров, при реализации быстрого алгоритма вычисления дискретного преобразования в БПФ, при обработках адаптивных алгоритмов фильтрации и т.д.

Используемая в ряде ПЛИС DSP архитектура 18x18 или 18x25 фиксированной точности не позволяет обеспечить качественное функционирование КИХ-фильтров, а также реализацию БПФ. Использование DSP блока 18x25 для реализации операции 9x9 или каскадное включение нескольких блоков DSP невысокой точности для поддержания более точного режима ЦОС повышает энергоемкость подобных решений [1,2,3].

Обеспечить высокоточный режим ЦОС можно на основе применения DSP блоков переменной точности. На рис. 1 показана блок-схема КИХ-фильтра, состоящая из умножителей, элементов задержки (регистров), сумматоров, реализованных с помощью DSP блоков переменной точности.

Проводится исследование использования ПЛИС для осуществления алгоритмов ЦОС переменной точности в системах передачи и обработки. Рассматривается обобщенная блок-схема КИХ-фильтра, реализация алгоритма вычисления дискретного преобразования в БПФ на ПЛИС. Показана целесообразность использования алгоритмов КИХ-фильтров, БПФ с переменной точностью при реализации на ПЛИС.

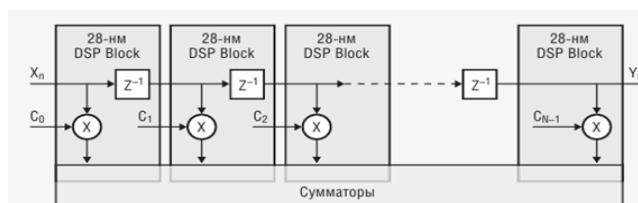


Рис. 1. Блок-схема КИХ-фильтра на DSP блоках

Следует отметить, что DSP блок при суммировании в режиме работы фиксированной 18-битной точности может быть настроен также и на умножение в автономном (независимом) режиме, как это показано на рис. 2. Однако в этом случае выходное разрешение не превышает 32 бит.

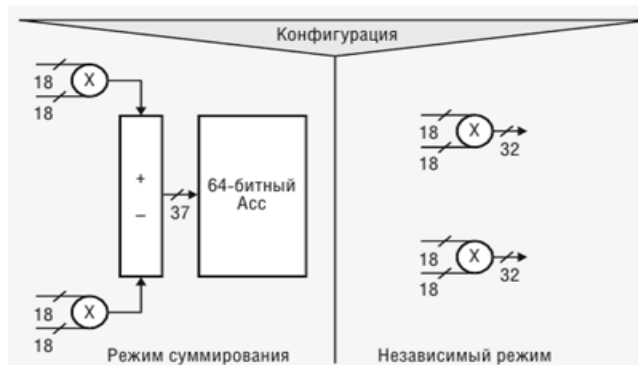


Рис. 2. Конфигурации DSP блока 18-битной точности

В режиме высокой точности следует настраивать каждый DSP блок переменной точности для реализации умножителя 27x27 или 18x36 так, как это показано на рис. 3.

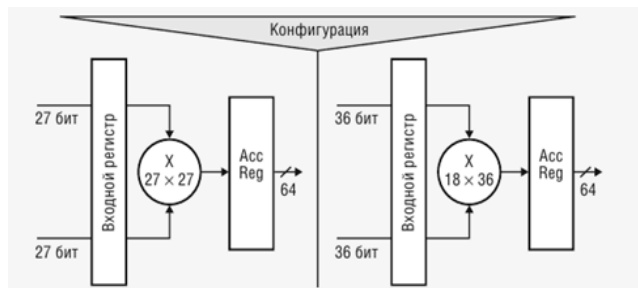


Рис. 3. Конфигурации DSP блока для режима работы высокой точности

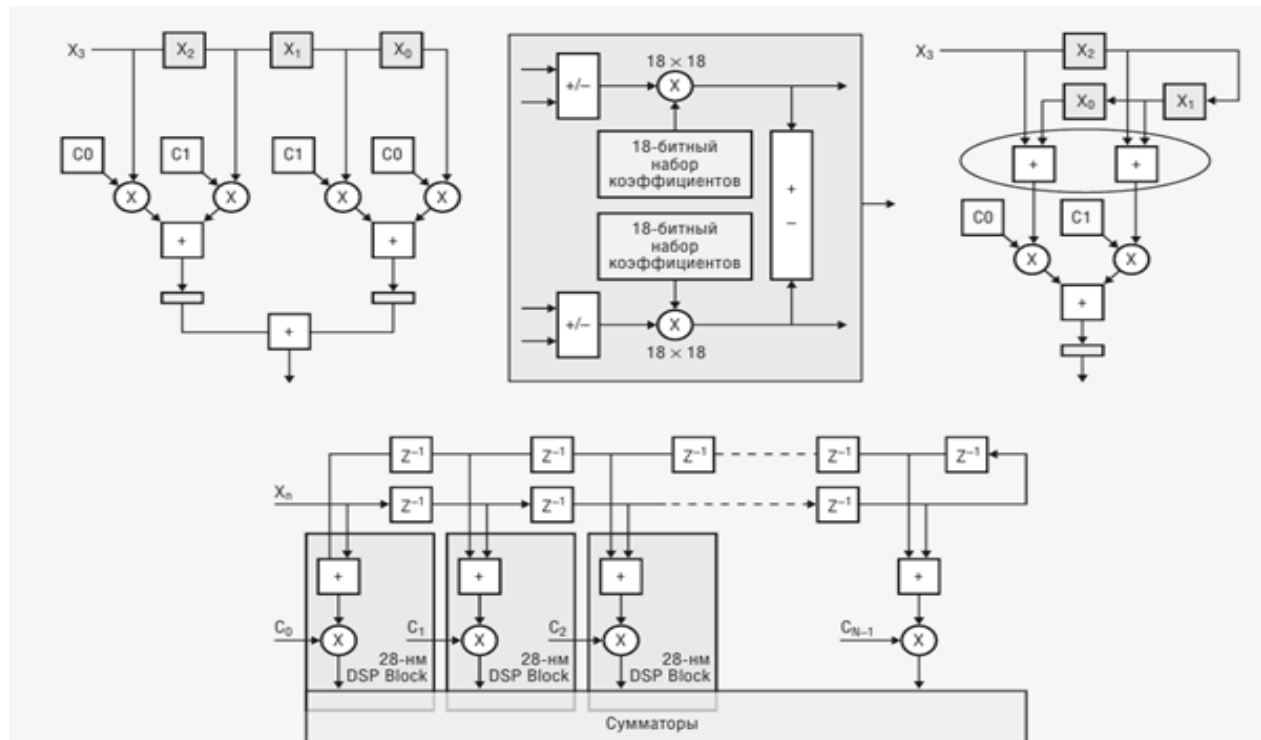


Рис. 4. Реализация симметричных КИХ-фильтров прямой формы в DSP блоке

Таким образом, конфигурацию каждого DSP блока переменной точности можно настроить с помощью двойного умножителя 18х18-бит (рис. 2) или высокоточного 27х27-разрядного умножителя (рис. 3). DSP блок выполняет различные операции умножения для каждого режима с точностью, как это показано в табл. 1.

Таблица 1
Точность режимов, поддерживаемых DSP блоком

Точность режима	Применение
3 независимых 9х9 режима	Низкая точность при фиксированной точке
2 суммирования 18х18 режима	Средняя точность при фиксированной точке
2 независимых 18х18 режима с разрешением 32-bit	Средняя точность при фиксированной точке
1 независимый 18х25 или 18х36 режим	Высокая точность при фиксированной точке
1 независимый 27х27 режим	Высокая точность при фиксированной точке или одинарная точность при плавающей точке

В DSP блоке переменной точности функции предварительного сумматора жестко закреплены внутри блока. При этом как предварительный сумматор, так и умножитель являются частью структуры сумматора, реализуемой в рамках DSP блока, а элементы задержки реализованы с использованием регистров за пределами DSP блока. Предварительные сумматоры в основном используются для построения симметричных КИХ-фильтров. Такая структура сумматора в DSP блоке переменной точности при реализации симметричных КИХ-фильтров прямой формы позволяет в два раза уменьшить число умножите-

лей. Для этого необходимо предварительно использовать две выборки данных с умножением их на общий коэффициент. В симметричных КИХ-фильтрах операции на двух умножителях можно заменить операцией, выполняемой на одном умножителе и одном сумматоре, как это показано на рис. 4. Точность выбранного режима определяется тем, будет использоваться двойной 18-битный или один 26-битный предварительный сумматор.

Алгоритм вычисления дискретного преобразования в быстрых преобразованиях Фурье в DSP блоке переменной точности реализуется с использованием комплекса умножителей. Причем DSP блок, оптимально поддерживая алгоритм вычисления дискретного преобразования в БПФ, реализует его в несколько этапов, обеспечивая тем самым более широкий динамический диапазон и низкий уровень шума. В целях обеспечения широкого динамического диапазона и низкого уровня шума при реализации алгоритма БПФ устанавливаются повышенные требования к точности DSP блока только со стороны умножителя. Для приложений высокой точности, в частности БПФ с плавающей точкой при использовании комплекса умножителей 27х27, необходим каскад из четырех DSP блоков переменной точности, настроенных в режим высокой точности.

Реализация рассмотренных алгоритмов ЦОС успешно может быть осуществлена в ПЛИС архитектуры FPGA. В 2010 г. фирма Altera представила ПЛИС семейства Stratix V (модели: Stratix V GT, Stratix V GX, Stratix V GS, Stratix V E) архитектуры FPGA, спроектированные по 28-нм техпроцессу [1,4]. В новой версии САПР Quartus II v 10.0 добавлена поддержка ПЛИС семейства Stratix V.

В ПЛИС для повышения точности обработки цифровых сигналов используется DSP блок переменной точности с 64-битной каскадной

шиной. Кроме того, в ПЛИС Stratix V имеются: 64-разрядный сумматор-накопитель; каскад сумматоров для реализации систолических КИХ-фильтров. Предусмотрены возможности увеличения числа независимых операций умножения, а также поддержка операций с плавающей точкой удвоенной точности. Использование 64-битной каскадной шины и сумматора-накопителя в ПЛИС Stratix V FPGA позволит поддерживать несколько уровней точности ее DSP блока. Архитектурные решения Stratix V обеспечивают реализацию структур высокопроизводительных КИХ-фильтров и БПФ.

В DSP блоке накопитель является неотъемлемой частью многих операций ЦОС. Его наличие способствует обеспечению высокой точности вычислений. В Stratix V используются 64-разрядные накопители, каждый из которых необходим для подсчета операций умножения и сложения. Кроме того, данное устройство обладает способностью выполнять округления конечного результата накопления.

На рис. 5 приведены блок-схемы КИХ-фильтра на ПЛИС Stratix V, использующие DSP блок переменной точности для фиксированного (18-битной точности) и высокоточного режимов работы. Результаты исследования КИХ-фильтра на ПЛИС в САПР Quartus II представлены на рис. 6.

Применением DSP блоков переменной точности при реализации КИХ-фильтра на ПЛИС достигаются следующие возможности:

- встроенные предварительные сумматоры могут быть использованы при реализации симметричного фильтра, что позволяет уменьшить число умножителей в два раза;
- коэффициенты регистра-хранения используются для хранения коэффициентов фильтра внутри DSP блока, что позволяет экономить количество регистров и объем памяти, увеличить тактовую частоту фильтра;
- два уровня сумматоров в одном блоке позволяют построить КИХ – фильтр прямой формы;
- регистры на выходе DSP блока позволяют реализовать каскад систолического фильтра.

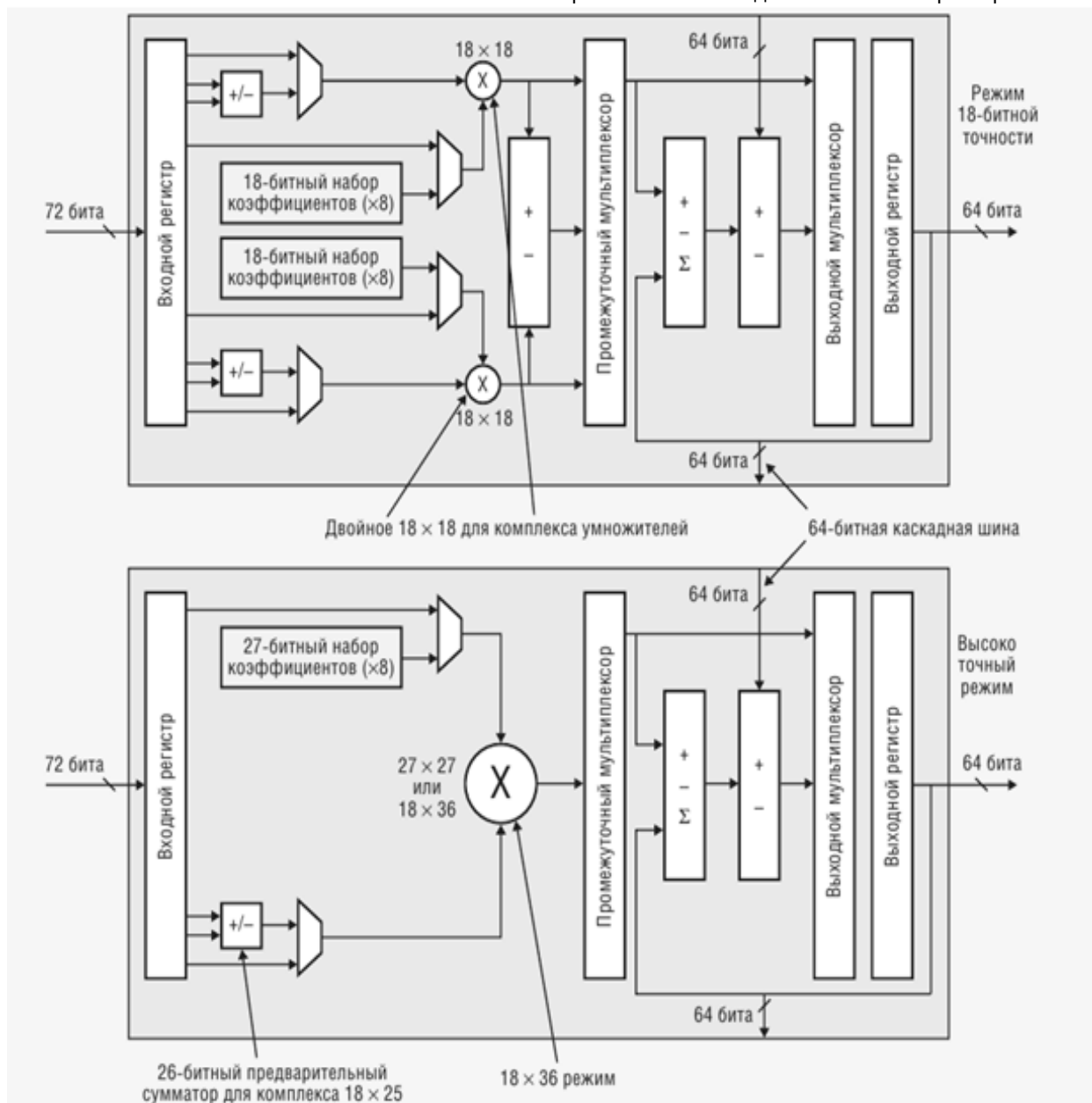


Рис. 5. Блок-схемы КИХ - фильтра на основе DSP блока переменной точности

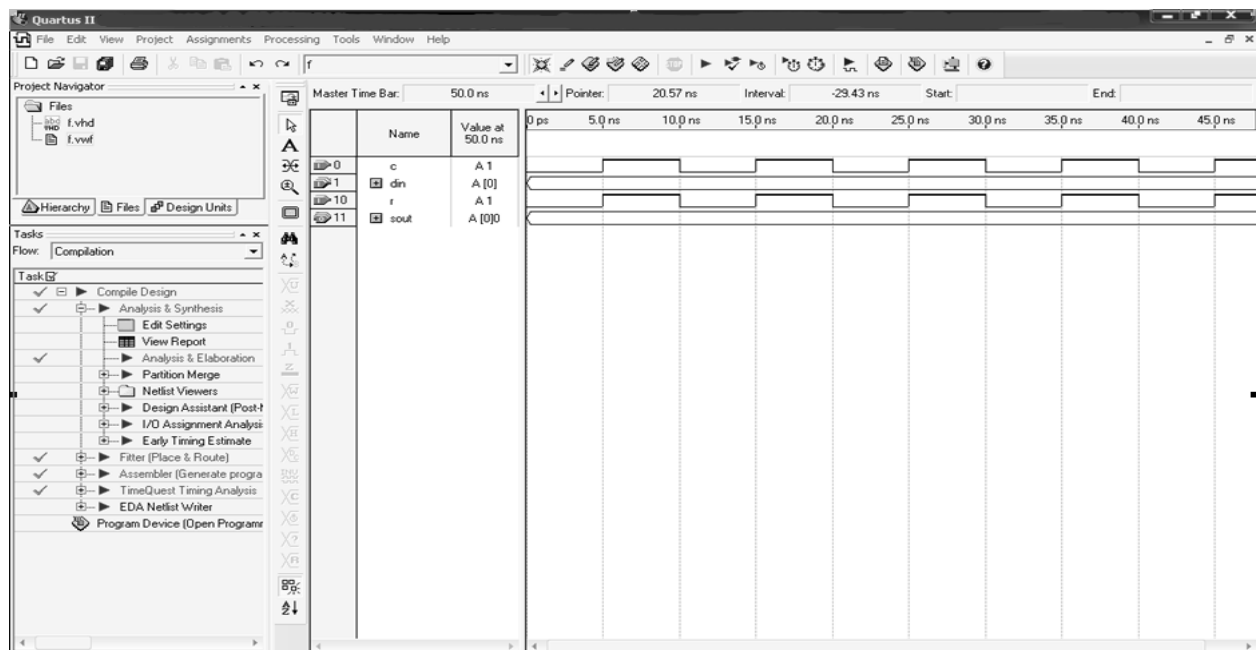


Рис. 6. Результаты моделирования КИХ – фильтра

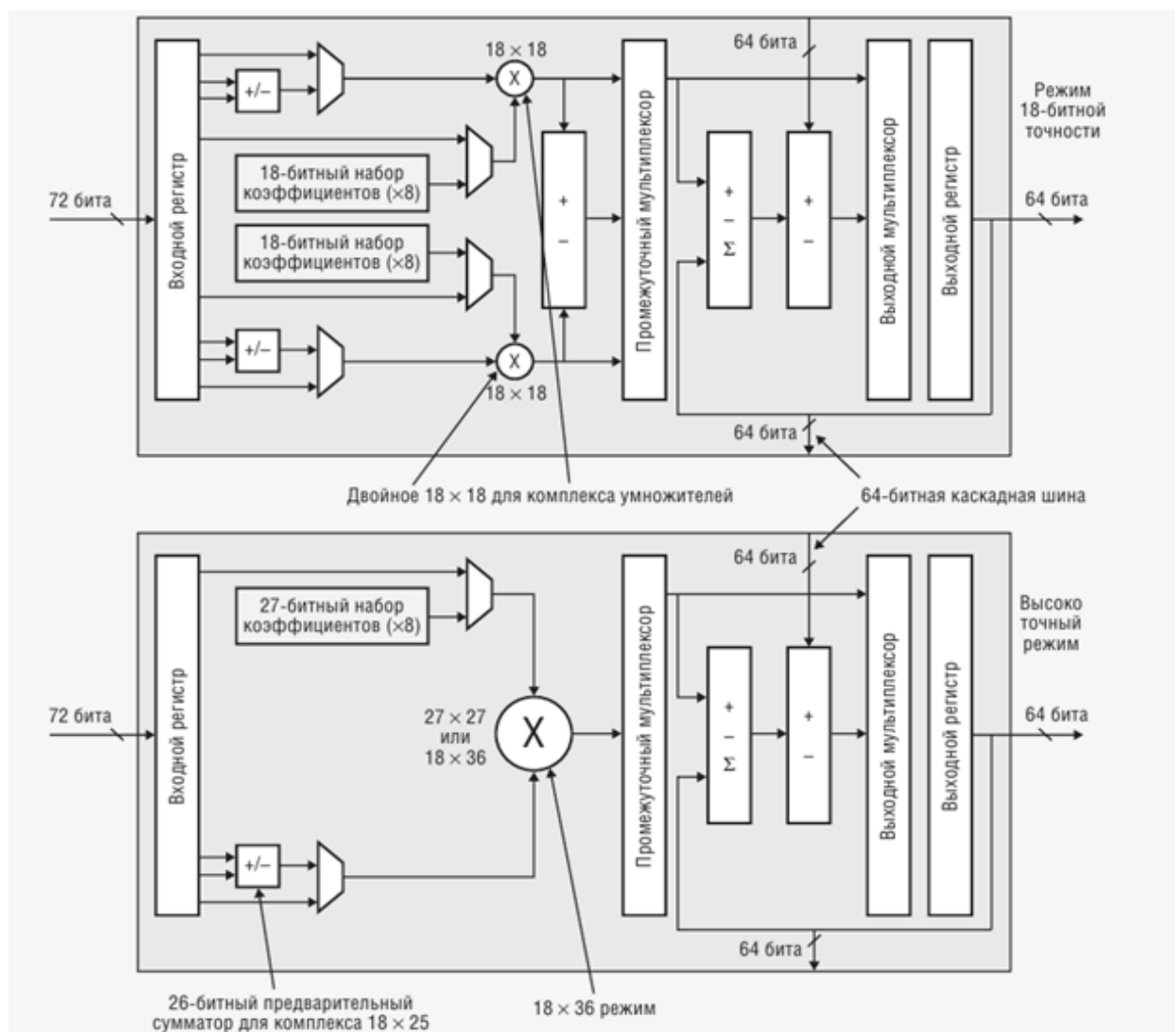


Рис. 7. Блок-схемы вычисления дискретного преобразования в БПФ на основе DSP блока переменной точности

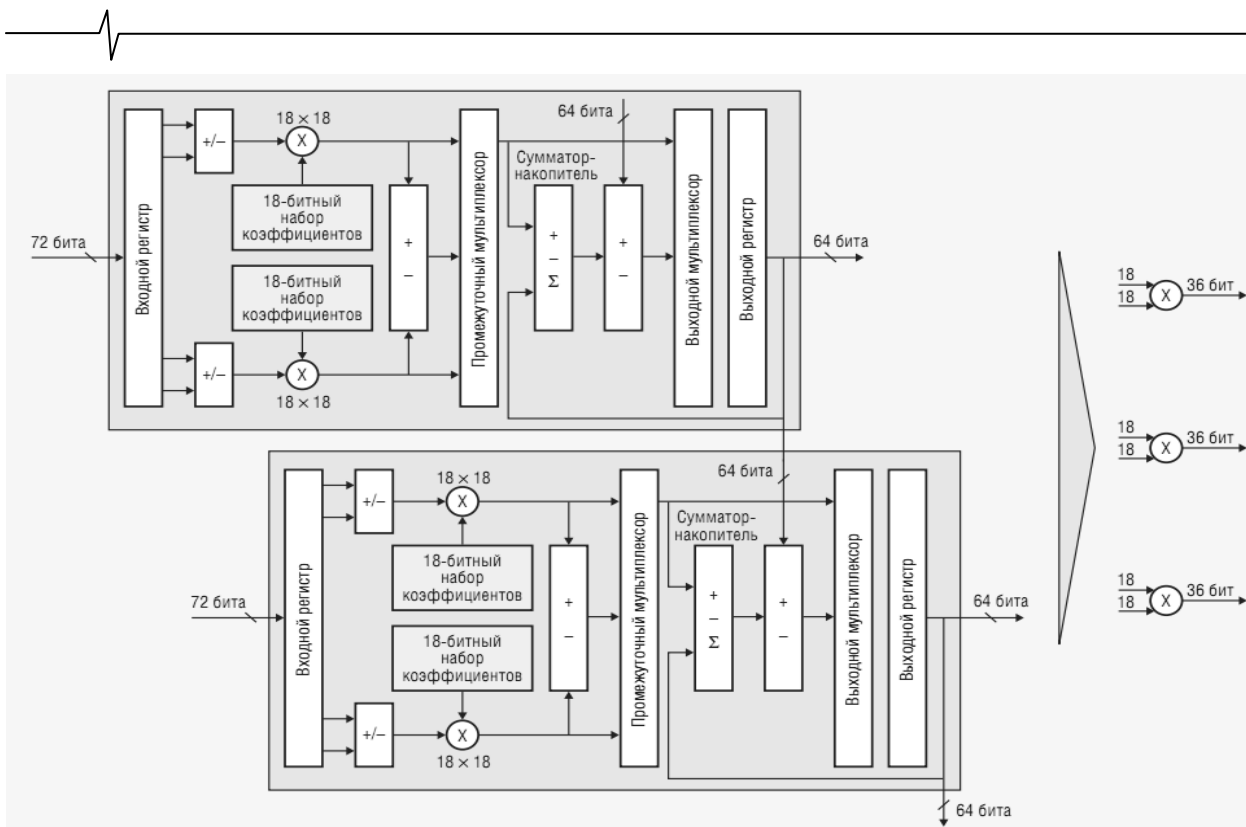


Рис. 8. Каскадное соединение двух блоков DSP переменной точности

В используемой архитектуре DSP блока для вычисления дискретного преобразования в БПФ применен 26-битный предварительный сумматор, позволяющий осуществить операции 18×25 . Возможность осуществить двойное 18×18 умножение позволяет получать действительные или мнимые числа в одном блоке. Реализация режима 18×36 обеспечивает высокую точность DSP блока (более широкий динамический диапазон и низкий уровень шума). Применение 64-разрядной каскадной шины позволяет соединять DSP блоки без потери точности. Особенности блок-схем реализации вычислений дискретного преобразования в БПФ на основе DSP блока переменной точности на ПЛИС Stratix V отражены на рис. 7.

Для расширения и поддержки необходимого диапазона точности используется каскадное соединение DSP блоков ПЛИС Stratix V посредством 64-битной каскадной шины. Использование подобного соединения в случае применения DSP блоков фиксированной точности при реализации, например трех независимых 18×18 умножителей требует четыре таких блока, а при использовании DSP блоков ПЛИС Stratix V для высокоточного режима необходимо только два блока с полным разрешением 36-бит (рис. 8).

Таким образом, при реализации алгоритмов ЦОС использование DSP блока переменной точности на ПЛИС Stratix V FPGA позволит обеспечить поддержание различных уровней точности, в том числе операции с плавающей точкой. Использование нескольких DSP блоков переменной точности не приводит к снижению точности алгоритмов цифровой обработки сигналов.

Разработчики устройств ЦОС могут в одном блоке реализовать сумматор 27×27 как для выполнения DSP приложений в высокоточных операциях с

фиксированной точкой, так и для дополнительных приложений DSP при осуществлении операций с плавающей точкой. Реализация КИХ-фильтров и БПФ структур с использованием схем DSP блоков переменной точности во многих случаях требует только половины ресурсов DSP по сравнению с конкурирующими решениями на DSP блоках фиксированной точности, что способствует снижению цены проектируемых устройств ЦОС.

Литература

1. 28-nm Variable-Precision DSP Block Architecture: <http://www.altera.com/technology/dsp/variable-precision/dsp-variable-precision.html>
2. Стешенко В.Б. ПЛИС фирмы Altera: элементная база, система проектирования и языки описания аппаратуры. – М.: Издательский дом «ДодэкаXXI», 2007. – С. 576.
1. Kevin Morris. A Perfect DSP Storm: BDTi + High Level Synthesis + FPGA / FPGA and Programmable Logic Journal. – №1. – 2010
4. Stratix V FPGAs: Built for Bandwidth: <http://www.altera.com/products/devices/stratix-fpgas/stratix-v/stxv-index.jsp>

IMPLEMENTATION OF DIGITAL SIGNAL PROCESSING VARIABLE PRECISION ON FPGA

V.V. Vychuzhanin

We study the use of FPGAs for DSP algorithms variable precision in data transmission systems and processing. The generalized block diagram of the FIR – filter realization algorithm for computing the discrete transformation in the FFT on the FPGA. The expediency of using algorithms FIR - Filter, FFT with variable accuracy accuracy for implementation on FPGAs.

УДК 00.004.021; 00.004.4; 00.004.9

ПРОГРАММНЫЙ КОМПЛЕКС ДЛЯ ОБРАБОТКИ ИЗОБРАЖЕНИЙ В МАССИВНО-МНОГОПОТОЧНОЙ CUDA-СРЕДЕ**Бибиков С.А.**, Самарский государственный аэрокосмический университет, *ssau@ssau.ru***Никоноров А.В.**, к.т.н., доцент Самарского государственного аэрокосмического университета, *E-mail: adminmcdk.org***Фурсов В.А.**, д.т.н., профессор Самарского государственного аэрокосмического университета, *E-mail: fursov@ssau.ru***Якимов П.П.**, магистрант Самарского государственного аэрокосмического университета, *E-mail: pavel.yakimov@hotmail.com***Постановка задачи**

Современная технология офсетной печати позволяет выпускать продукцию очень высокого качества. Возможна печать многокрасочных публикаций, информационной, рекламной продукции, а также репродукционное воспроизведение живописных полотен.

Необходимость проведения предпечатной обработки изображений при производстве цифровых репродукций произведений живописи связана со спецификой фотографирования оригиналов. В процессе фотографирования возникают следующие артефакты: множественные блики, затенения части холста рамкой и так называемые матовые блики. Артефакты обусловлены неровностями нанесения краски на холст и способом установки дополнительного освещения.

В данной работе рассматривается алгоритм цифровой обработки изображений для удаления множественных бликов, которые возникают вследствие того, что отдельные малые участки слоя краски обладают таким углом наклона поверхности, что падающий свет источника отражается и регистрируется камерой в виде пятен. На цифровом изображении такие артефакты выглядят как области небольшого размера, обладающие выделяющейся на общем фоне яркостью. Пример изображения с артефактами представлен на рис 1.



Рис. 1. Пример цифровой копии с точечными бликами.

Если обработку изображения будет проводить специалист по компьютерной графике с использованием стандартных возможностей программы Photoshop либо аналогичной, то для качественной ретуши изображения формата А3 ему понадобится примерно 2-3 часа времени. В настоящей работе описываются разработанные методы, алгоритмы и созданный на их

основе программный комплекс, обеспечившие возможность высокой степени автоматизации процессов распознавания, локализации и цветовой коррекции множественных точечных артефактов на цифровых изображениях произведений живописи в процессе их предпечатной подготовки.

Указанные свойства разработанного программного комплекса достигаются за счёт перспективной архитектуры, реализация которой основана на следующих соображениях. В последние годы наряду с совершенствованием и ростом производительности мощных суперкомпьютеров наметилась устойчивая тенденция развития технологий параллельных вычислений на персональных компьютерах. Эта тенденция обусловлена существенным технологическим прорывом, по крайней мере, в трёх направлениях.

Во-первых, это появление настольных CPU с возможностью параллельного выполнения программ. В настоящее время для персональных компьютеров доступны процессоры 7 с параллельным исполнением 8-ми потоков. Вторым технологическим прорывом является представленное в 2007 году компанией Nvidia альтернативное направление развития массивно-параллельных вычислений для настольных компьютеров с использованием GPU. Третий фактор, позволяющий всерьез говорить о настольных суперкомпьютерных системах, – это повышение скорости коммуникаций по системной шине для платформы x86 с развитием стандарта PCI Express. Скорость коммуникаций между узлами всегда была узким местом кластерных систем, снижающим их эффективность. Принятие подавляющим большинством производителей стандарта *Infiniband* позволяет строить весьма недорогие вычислительные системы со скоростью обмена данными между узлами до 5 GB/s.

Совокупность указанных факторов стала мотивом возрастающего интереса к созданию компактных, недорогих и, вместе с тем, высокопроизводительных систем обработки и анализа изображений нового поколения для систем видеоконтроля, видеонаблюдения, подготовки цифровых изображений к печати и др. В работах [1-3] рассматривались примеры создания компактных распределенных систем обработки изображений. В частности, в работе [1] описана реализация алгоритма коррекции цветных цифровых изображений в локальной сети компьютеров посредством специально разработанной GRID-системы, реализованной на Java с native-модулями. В этой работе показана возможность существенного ускорения одного из наиболее затратных по времени процессов подготовки цветных изображений к печати. В работах [4-5] приведены примеры реализации описанного выше алгоритма с помощью технологии CUDA.

В описываемом в настоящей статье программном комплексе высокое быстродействие и связанная с этим высокая степень автоматизации процессов распознава-

ния, локализации и цветовой коррекции множественных точечных артефактов на цифровых изображениях достигается за счёт построения эффективных рекуррентных процедур обработки данных.

Алгоритм устранения точечных бликов

Цифровое изображение может быть представлено как двумерная функция вида $f(x, y)$, где x и y – координаты элемента изображения на плоскости. Значение f задаётся парой координат (x, y) и называется интенсивностью или уровнем серого (яркости) изображения в этой точке. Значения функции яркости $f(x, y)$, как правило, принимают дискретные значения в диапазоне от 0 до 255 (8-битное кодирование), иногда – от 0 до 65535 (16-битное кодирование).

В настоящей работе цветные изображения представляются в пространстве RGB, поэтому в действительности мы имеем дело с тремя функциями яркости – $f_R(x, y)$, $f_G(x, y)$ и $f_B(x, y)$. При этом функция общей яркости или светлоты $L(x, y)$ определяется согласно [6], как

$$L(x, y) = 0,3f_R(x, y) + 0,59f_G(x, y) + 0,11f_B(x, y), \quad (1)$$

для которой используется нормированное значение яркости в диапазоне от 0 до 1.

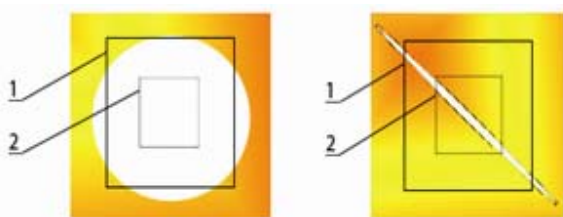
Задача цветовой коррекции множественных точечных бликов естественным образом распадается на следующие два этапа:

- обнаружение точечных артефактов на изображении;
- замещение цвета пикселей в области блика на некоторый цвет, близкий к цвету пикселей окрестности блика (компьютерное ретуширование).

Распознавание и локализация артефактов

Под точечным артефактом понимаются относительно небольшие компактные области с высокой (по относительному или абсолютному значению) яркостью. Поэтому распознавание и локализация артефактов проводится по значениям светлоты, описываемой значениями функции (1). При этом в качестве отличительных характеристик используются высокая относительная и/или абсолютная яркость области в совокупности с компактностью и небольшой плотностью области.

Для определения указанных характеристик используется алгоритм, осуществляющий «просмотр» всего изображения с использованием локальной области, включающей опорное окно и окно поиска. На рис. 2,а приведен пример, когда признаки наличия блика выполняются, а на рис. 2,б,в – случаи, когда условия наличия точечного блика не выполняются. Далее приводятся математически формализованные признаки существования точечного блика.



а) опорное окно б) окно поиска

Рис. 2. Слева пример нарушения размера, справа нарушение компактности

Пусть W_1 – область изображения, покрытая бликом, W_0 – некоторая окрестность области блика. Рассмотрим функцию принадлежности $I_0(x, y)$:

$$\left. \begin{aligned} I_0(x, y) &= 1, & (x, y) &\in W_0 \\ I_0(x, y) &= 0, & (x, y) &\notin W_0 \end{aligned} \right\}. \quad (2)$$

Тогда для блика W_0 должны выполняться следующие требования:

$$\max_{(x, y) \in W_1} f(x, y) \geq f_1, \quad (3)$$

$$\frac{\sum_{(x, y) \in W_1} f(x, y)}{\sum_{(x, y) \in W_0} f(x, y)} \geq M_1, \quad (4)$$

$$\sum I(x, y) \leq S_1, \quad (5)$$

$$D(W_1) \leq D_1. \quad (6)$$

Здесь $M(f(x, y))$ – среднее значение функции яркости в области, $D(W_1)$ – диаметр области, величина, характеризующая степень компактности области. Диаметр области определим, как максимальное евклидово расстояние между точками области:

$$D(W_1) = \max_{(x_1, y_1), (x_2, y_2) \in W_1} \|(x_1, y_1), (x_2, y_2)\|_2. \quad (7)$$

Соотношения (3) и (4) отражают требования к максимальной и относительной яркости блика, а (5) и (6) – требование малой площади и компактности области. Константы f_1 , M_1 , S_1 , D_1 являются определяющими для разрабатываемых алгоритмов и могут жестко задаваться экспертом, либо настраиваться в автоматизированном режиме.

Замещение артефактов

Множественные точечные блики характеризуются почти полной потерей информации о цвете, поэтому задача цветовой коррекции в данном случае сводится к замещению цвета пикселей в области блика на некоторый цвет, близкий к цвету пикселей в окрестности блика. В разработанной системе используется технология замещения пикселей блика, основанная на идентификации параметров модели замещения. Технология реализована в предположении, что цвет задан в пространстве RGB. Для определения значений цветовых компонентов в области блика строятся оценки коэффициентов поверхности M -го порядка, «опирающейся» на участки поверхности, образованной функциями яркости компонентов в окрестности блика:

$$z_k(x, y) = \sum_{i=1}^M \sum_{j=1}^M C_k^{ij} x^i y^j, \quad (8)$$

где $z_k(x, y)$ – значение k -го цветового компонента в точке (x, y) , C_k^{ij} – коэффициенты полинома, подлежащие оцениванию.

Замещение пикселей поверхности (8) может привести к тому, что блик останется заметным в силу гладкости поверхности. Поэтому на область блика дополнительно копируется текстура из окрестности блика. Затем выполняется размытие по Гауссу в области, чуть большей, чем область блика. Размытие позволяет избежать резких границ между ретушированным участком и его окрестностью.

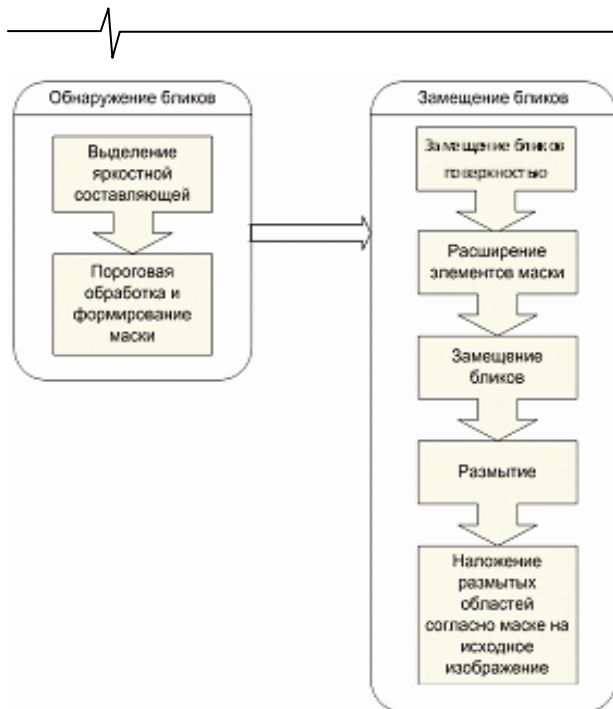


Рис. 3. Общая схема этапов локализации блика

На рис. 3 приведена общая схема этапов локализации блика и замещения пикселей в области блика.

Параллельная реализация алгоритма устранения точечных бликов

В приведённом выше алгоритме обнаружения и локализации бликов наиболее ресурсоёмкой в вычислительном отношении являются процедура «просмотра» изображения скользящим окном поиска. Известны быстрые рекуррентные (например, параллельно-рекурсивные) алгоритмы обработки изображений скользящим окном, реализованные на традиционных аппаратных средствах [6]. Создание таких же эффективных процедур на основе CUDA-технологий, реализуемых с использованием графических ускорителей NVIDIA, к сожалению, пока еще остается проблемной задачей. Связано это с тем, что вычислительная среда CUDA имеет более сложную, с точки зрения программирования, структуру по сравнению с многопоточной CPU средой или параллельной средой MPI. Основные структурные особенности среды CUDA следующие.

Во-первых, CUDA-среда имеет двухуровневую многомерную топологию исполнения потоков, которая может описывать десятки тысяч потоков. По аналогии с термином массивно-параллельная среда такую среду можно назвать массивно-многопоточной.

Во-вторых, потоки могут обмениваться данными только посредством сравнительно небольшого объема общей памяти. Такой обмен возможен в пределах небольшой области локальной связности – блока, включающего (для выпускаемых в настоящее время GPU) не более 512 потоков. Кроме того, когерентность данных в общей памяти достигается только путем приостановки всех потоков, т.е. упорядоченный обмен данными невозможен. Указанные ограничения создают серьезные проблемы при реализации в среде CUDA рекурсивных алгоритмов (сжатия данных, фильтрации и др.) [7].

Наконец, ещё одной важной особенностью является то, что в CUDA-среде можно работать с несколькими видами памяти, имеющими разную область видимости и скорость доступа. Крайне важным при доступе к глобальной памяти является согласование (coalescing) адреса памяти с номером активного потока. Если условие согласования выполняется, то

GPU может объединить несколько запросов в одну транзакцию [14]. Несоблюдение этого условия может привести к падению производительности почти в 8 раз [8, 9].

Многие базовые алгоритмы обработки данных требуют существенной переработки с учётом указанных особенностей. Например, в работе [10] рассмотрен адаптированный для CUDA-среды алгоритм префиксного суммирования. В настоящей статье, являющейся развитием работ [1-5], из всего комплекса рассмотренных в них задач выделены наиболее характерные и сложные для реализации на GPU-процессорах: обработка изображений скользящим окном и организация передачи видеоданных в реальном времени. В частности, рассмотрены основные особенности построения указанных алгоритмов и приводится краткое описание созданного в массивно-многопоточной CUDA-среде программного комплекса.

Продemonстрируем особенности реализации рекуррентных вычислительных алгоритмов в CUDA-среде на примере простейшего локального оконного фильтра, выполняющего расчёт среднего значения. Простота этого алгоритма позволяет наглядно продемонстрировать все основные особенности среды CUDA. Вместе с тем, иллюстрируемый на этом простом примере подход может быть легко применен к более сложным локальным оконным фильтрам, применяемым в различных технологиях обработки изображений.

Пусть алгоритм обработки скользящим окном состоит в вычислении для каждой точки изображения суммы значений отсчётов в некоторой окрестности (окне):

$$s_{x,y} = \sum_{i,j=1}^m L_{x+i,y+j}, \quad 0 < x, y \leq N, \quad (9)$$

где N – размер изображения, m – размер окна (для простоты полагаем, что изображение и окно квадратные). При этом потребуется $O_1 = N^2 m^2$ (10)

арифметических операций.

При рекуррентной реализации этих вычислений в CUDA-среде существенное значение играет согласование адреса памяти с номером активного потока. Будем считать обращение потока к глобальной памяти согласованным, если оно происходит к словам памяти размером 32 бита с адресами

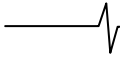
$$I(t, k) = 32 * n * k + t, \quad k = 0, 1, 2, \dots, \quad t = 0, 1, 2, \dots, \quad (11)$$

где $I(k, t)$ – адрес ячейки глобальной памяти, к которой производится обращение, n – натуральное число, характеризующее особенности конкретной задачи, связанные с количеством потоков в блоке; t – номер потока внутри блока, k – номер шага некоторой итерации алгоритма, одинаковый для всех потоков внутри блока. Приведённые в спецификации CUDA [8] требования согласованности несколько более сложные и к тому же различаются для различных реализаций стандарта CUDA. Тем не менее, легко проверить, что выполнение требования (11) для всех потоков внутри блока позволяют получить согласованный доступ к памяти в смысле спецификации [8].

Рекуррентный алгоритм, удовлетворяющий условию согласованности (11), может быть построен, например, как двухпроходный. При этом сначала выполняется вычисление частичных сумм по столбцам:

$$s_{x,y}^1 = s_{x,y-1}^1 - L_{x,y-m} + L_{x,y}, \quad s_{x,1}^1 = \sum_{j=1}^m L_{x,y+j}, \quad (12)$$

а окончательная сумма получается суммированием по строкам:



$$s_{x,y} = s_{x-1,y} - s_{x-1,y}^1 + s_{x,y}^1, \quad s_{1,y} = \sum_{j=1}^m s_{x+j,y}^1 \quad (13)$$

Вычисления по такой схеме требуют

$$O_1 = (2N - m)^2 \quad (14)$$

арифметических операций.

В CUDA-среде исходные данные располагаются в глобальной памяти таким образом, что строки изображения располагаются последовательно. Вычисление частичных сумм (12) удовлетворяет условию (11), если длина строк изображения кратна 32 пикселям. Если это не так, строки могут быть дополнены до требуемого размера, поэтому далее полагаем, что

$$N \bmod 32 = 0. \quad (15)$$

При последующем суммировании по строкам (13) требование (11) не выполняется, что приводит к существенному снижению скорости вычислений. Это подтверждается данными эксперимента. Рассмотрим модификацию алгоритма построчного суммирования, обеспечивающую согласованное обращение к глобальной памяти за счёт использования при проведении суммирования локальной и общей памяти потоков.

Текущая спецификация CUDA обеспечивает всего 16384 байт общей памяти, а каждый блок содержит не более 512 потоков. Небольшая часть этой памяти – около 20-30 байт расходуется на хранение переданных ядру аргументов. В таком объёме невозможно записать изображение целиком, поэтому предлагается итерационная схема блочной обработки изображения.

В каждом эпизоде обрабатывается блок из 32 строк изображения, при этом удобно включать в блок 32 потока. С учётом того, что данные хранятся в формате с плавающей точкой и одинарной точностью, для такой обработки потребуется $32 \times 32 \times 4 = 4096$ байт общей памяти на блок. Заметим, что в случае необходимости предлагаемая блочная реализация позволяет с использованием имеющегося объёма общей памяти выполнять вычисление нескольких показателей одновременно.

В предлагаемой схеме потоки группируются в блоки по 32 потока, общее число блоков, с учётом (15), равно $N/32$. Каждый поток текущего блока считывает в общую память данные из столбца изображения в глобальной памяти, т.е. каждый поток последовательно читает данные из соседних столбцов изображения. Таким образом, за одну итерацию все блоки обрабатывают 32 строки изображения, формируя в общей памяти фрагмент изображения размерностью $32 \times N$. Каждую итерацию чтения можно описать следующим выражением:

$$s[k * N + 32 * b + i] := d[K * N * 32 + k * N + 32 * b + i], \quad k = 0, 31, \quad (16)$$

где s – указатель на область общей памяти, d – указатель на данные изображения в глобальной памяти, k – номер шага итерации, K – номер итерации, b и i – индексы блока и потока соответственно.

После синхронизации потоков, каждый поток начинает суммирование данных согласно (13) «вдоль» строки изображения в общей памяти, при обращении к которой согласованности не требуется. Промежуточные данные сохраняются в локальной памяти потока. После выполнения вычислений потоки синхронизируются, а данные из общей памяти записываются в глобальную память по столбцам, аналогично операции чтения (16). При такой схеме поток читает и пишет в глобальную память данные «транспонированные» по отношению к тем, над которыми поток выполняет вычисления. Схема организации взаимодействия потоков с локальной и

глобальной памятью показана на рис. 4. Описанный рекуррентный алгоритм суммирования практически без существенных изменений может использоваться для реализации широкого круга параллельно-рекурсивных методов [11] обработки изображений.

Ниже приводятся результаты сравнения эффективности предложенных реализаций. Эксперименты проводились на тестовом изображении размером 1024×1024 для окон различных размеров. В первом случае декомпозиция проводилась на 2048 блоков по 512 потоков, часть которых запускалась не одновременно. Во втором и третьем случае запускалось одинаковое количество потоков – 1024, при этом они группировались в 32 блока по 32 потока в каждом. Тестирование проводилось для CPU PIV 3GHz и GPU Nvidia GF 9500. Результаты представлены в таблице 1.

Таблица 1.

Время выполнения различных алгоритмов для разных размеров окна (m).

	$m = 5$	$m = 9$	$m = 15$
1. CPU, не рекуррентный	283 мс	922 мс	2 531 мс
2. CPU, рекуррентный	142 мс	297 мс	733 мс
3. GPU, не рекуррентный	78 мс	262 мс	473 мс
4. GPU, рекуррентный	23 мс	25 мс	27 мс
5. GPU, рекуррентный с оптимизацией обращений к памяти	12 мс	12 мс	12 мс

Как видно из таблицы 1, результаты для рекуррентной CUDA-реализации практически не зависят от m , а зависят только от стратегии использования памяти.

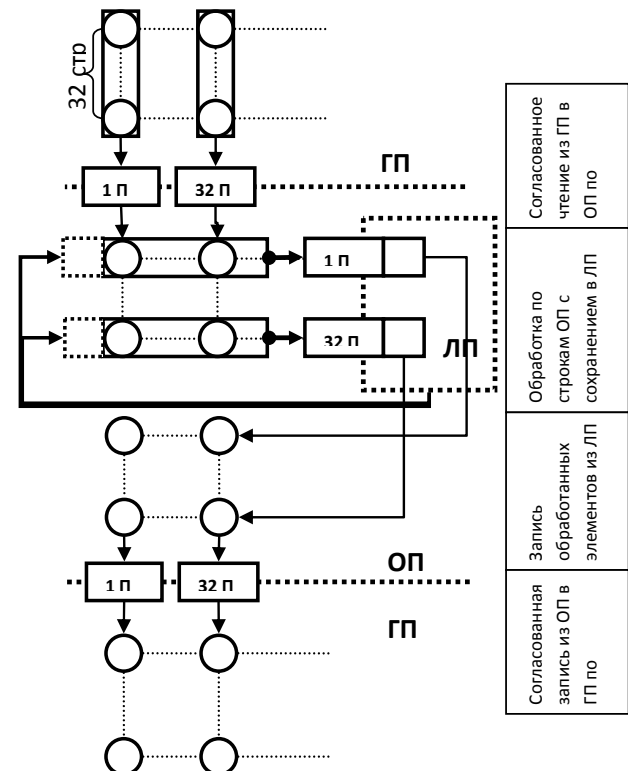


Рис. 4. Схема организации взаимодействия потоков с памятью



Рис. 5. Архитектура программного комплекса

Архитектура системы обработки изображений

Первый уровень – это интерфейс пользователя. Он может быть реализован в виде модуля расширения существующей системы, либо в виде независимого интерфейса работы с файлами растровых изображений в формате tiff или jpeg. Средний уровень – реализация графического интерфейса (GUI) обработки изображений представляет собой унифицированный графический интерфейс для любых режимов работы.

Графический интерфейс GUI связан с алгоритмической частью посредством динамической линковки. Тем не менее, в данном случае для создания GUI используется кроссплатформенная C++ библиотека. В настоящее время известно несколько таких библиотек. Среди них это QT, GTK и библиотека wxWidgets [12]. На наш взгляд, wxWidgets обладает большей концептуальной чистотой и гибкостью в использовании, что предопределило её использование в настоящей работе.

Третий уровень, не зависящий от других уровней, – это динамические библиотеки, в которых реализованы вычислительные алгоритмы. В настоящей работе описан алгоритм удаления точечных бликов. Однако архитектура позволяет подключать и другие алгоритмы обработки изображений. Если используемый алгоритм вычислительно сложен и доступны аппаратные средства, поддерживающие технологию CUDA, алгоритмы могут вызывать процедуры реализованные для GPU. Такой подход позволяет использовать для вычислений не только GPU Nvidia, но и GPU видеопроцессоры AMD (ATI). Заметим, что вычислительные процедуры могут также выполняться в нескольких потоках на CPU с использованием OpenMP или OpenCL [13]. Кроме того, система легко масштабируется с использованием обычных схем декомпозиции изображений.



Рис. 6. Графический интерфейс программного комплекса

В процессе работы системы модуль GUI вызывается динамически основным модулем расширения в качестве модуля расширения. В случае автономной работы программного комплекса GUI является стартовой точкой приложения и работает как самостоятельный исполняемый файл.

Наиболее гибкой платформой, позволяющей вести разработку модулей расширения существующих систем, а также разработку под платформу CUDA, является C. Кроме того, C позволяет создавать многопоточные и распределённые программы с использованием технологий OpenMP и MPI. Поэтому, несмотря на существование других более комфортных для разработки систем платформ, все описанные алгоритмы реализованы на C. Интерфейсная часть программы выполнена на C++.

Реализация программного комплекса и результаты экспериментов

Описанная выше схема взаимодействия потоков с памятью при реализации рекуррентного алгоритма использовалась при создании последних версий программного комплекса локализации и коррекции бликов на цифровых изображениях репродукций произведений живописи. Первые версии этого программного комплекса описаны в работах [2-5]. Программный комплекс имеет широкий набор функций для работы с цифровыми изображениями. Приложение работает в режиме stand alone, т.е. не требует установки дополнительного программного обеспечения. Благодаря использованию кроссплатформенных библиотек wxWidgets комплекс работает как на платформе Windows, так и на Unix-подобных операционных системах.

На рис. 6 представлен вид GUI, разработанного приложения в режиме автоматизированного поиска точечных артефактов (точки на экране указывают найденные блики).

В последней версии параметры поиска могут задаваться как вручную, так и автоматически путём перехода в режим адаптивного определения параметров. Кроме того, имеется возможность выполнять масштабирование изображения, выбор области поиска, а также модифицировать маску с отмеченными артефактами, сформированную алгоритмом, вручную. На рис. 7 приведены изображения, иллюстрирующие результаты устранения бликов с использованием разработанного программного комплекса.

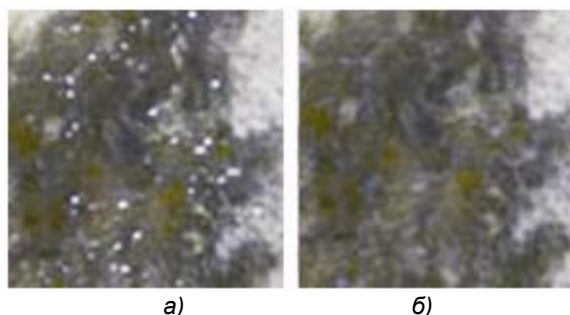


Рис. 7. Пример работы программной системы по устранению бликов на изображении «лес»: а) исходное, б) обработанное

В работах [4] и [5] авторами была показана возможность существенного ускорения алгоритмов обработки изображений в программной системе локализации и устранения бликов за счёт использования универсальных графических процессоров NVIDIA и технологии CUDA. В частности, на графических ускорителях NVIDIA GeForce 8400 и 9500 с 8 мультипроцессорами, было продемонстрировано более чем десятикратное ускорение CUDA реализации по сравнению с реализацией на CPU. В настоящей работе мы демонстрируем возможность дальнейшего повышения ускорения за счёт использования в программной системе оптимизированных рекуррентных алгоритмов.

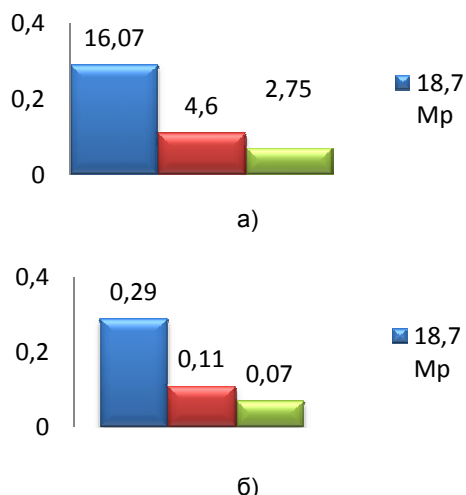


Рис. 8. Время расчёта оптимизированного алгоритма (с). а) алгоритм, опубликованный в работе [4]; б) результаты, оптимизированного рекуррентного алгоритма

Для экспериментальной проверки скорости использовали три изображения размером 3,2, 5,8 и 18,7 мегапикселей. Для расчётов использовалась видеокарта Nvidia GeForce 9500. На рис. 8,а приведено время вычислений для алгоритма из работы [4], на рис. 8,б приведено время вычислений для алгоритма поиска бликов на изображении, использующего оптимизированную рекуррентную

процедуру обработки скользящим окном. Из приведённых диаграмм видно, что в среднем, по трём экспериментам, достигнуто ускорение более чем в 45 раз.

Заключение

Подчёркнём, что полученное в эксперименте ускорение получено исключительно за счёт оптимизации CUDA алгоритма. Достигнутые показатели скорости обработки с использованием оптимизированной версии алгоритма поиска бликов открывают возможности для их более широкого применения. В частности, наряду с использованием этого алгоритма в задачах предпечатной обработки изображений, как это было сделано в [4], по-видимому, удастся использовать эту технологию также для обработки видео данных в реальном времени.

Работа выполнена при финансовой поддержке РФФИ (проект № 09-07-00269-а) и программы Президиума РАН

Литература

1. А.В. Никоноров, В.А. Фурсов. Предоставление сервиса управления цветопроизведением цветных изображений в сети интернет. Труды XIV Всероссийской научно-методической конференции «Телематика 2007», С.-Петербург, 18-21 июня, 2007, С. 412-413.
2. Никоноров А.В., Фурсов В.А. Распределенная вычислительная среда коррекции цветных изображений. Тр. XV Всероссийской научно-методической конференции «Телематика 2008», С.-Пб, 23-26 июня, 2008, С. 88-89.
3. В.А. Фурсов, А.В. Никоноров. Распределённый алгоритм коррекции точечных артефактов на цветных изображениях // Труды четвёртой международной конференции «Параллельные вычисления и задачи управления» (РАСО' 2008), Москва, 27-29 октября, 2008, 1087.
4. С.А. Бибииков, А.В. Никоноров, В.А. Фурсов, П.Ю. Якимов. Исследование эффективности технологии CUDA в задаче распределённой предпечатной подготовки цифровых изображений. Труды Всероссийской суперкомпьютерной конференции «Научный сервис в сети Интернет: масштабируемость, параллельность, эффективность», 21-26 сентября 2009 г., г. Новороссийск. – М.: Изд-во МГУ, 2009. – С. 204-207.
5. С.А. Бибииков, А.В. Никоноров, В.А. Фурсов, П.Ю. Якимов. CUDA-технология цветовой коррекции теневых искажений на цифровых фотокопиях произведений живописи. Сб. трудов международной научной конференции «Параллельные вычислительные технологии' 2010» Уфимский государственный авиационный технический университет, г. Уфа, 29 марта – 2 апреля 2010 г., С. 656.
6. Сойфер, В.А. и др., Методы компьютерной обработки изображений. – М.: Физматлит, 2003. – С. 784.
7. Nickolls J., Buck I., Skadron K., Scalable Parallel Programming with CUDA, // ACM Queue, VOL 6, No. 2 (March / April 2008) 40-53 pp. <http://www.mags.acm.org/queue/20080304/?u1=texterity>.
8. NVIDIA CUDA Programming Guide, 2010, 130 p.
9. NVIDIA CUDA Best Practices Guide, 2009, 68 p.
10. Harris M., Parallel Prefix Sum (Scan) with CUDA, 2008, 21 p.
11. Мурызин С.А., Сергеев В.В., Фролова Л.Г. Исследование эффективности двумерных параллельно-рекурсивных КИХ-фильтров // Компьютерная оптика. – М.: МЦНТИ, 1992. – Вып. – 12. – С. 65-71.
12. Smart, J. Cross-platform GUI Programming with wxWidgets / J. Smart – Prentice Hall, 2000. – 714 p.
13. Munshi, A. Kronos OpenCL Working Group, The OpenCL Specification. – 2009. – 314 p.