

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ

3/2012

НАУЧНО - ТЕХНИЧЕСКИЙ ЖУРНАЛ

ЦИФРОВАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ

тематический выпуск:

- КОДИРОВАНИЕ И СЖАТИЕ
- КОМПЕНСАЦИЯ ДВИЖЕНИЯ
- ТЕКСТУРНАЯ СЕГМЕНТАЦИЯ
- КОМПЛЕКСИРОВАНИЕ ИЗОБРАЖЕНИЙ
- АНАЛИЗ И ПОВЫШЕНИЕ КАЧЕСТВА
- СУБПИКСЕЛЬНАЯ КЛАССИФИКАЦИЯ
- ОБРАБОТКА НА ГРАФИЧЕСКИХ ПРОЦЕССОРАХ

ISSN 1684-2634



9 771684 263005 >

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ

Научно-технический журнал
№ 3/2012

Издается 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,
Ю.А. Брюханов, В.В. Еремеев, 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,
В.С. Сперанский, Ю.С. Шинаков

Адрес редакции:

107031 Москва, Рождественка, 6/9/20, стр. 1
Тел.: (495) 621-06-10, (495) 621-16-39
Факс: (495) 621-16-39
E-mail: rntores@mail.ru
vityazev.v.v@rsreu.ru
http://www.dspsa.ru

Издатель:

Российское научно-техническое общество
радиотехники, электроники и связи им. А.С. Попова
Компьютерная верстка: И.А. Благодарова
Дизайн: М.В. Аверин

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

Подписано в печать 28.09.12 г.
Формат 60x90/8.

Гарнитура «Agiat». Печать офсетная.
Бумага офсетная. Печ.л. 6,5. Тираж 700 экз.

Заказ № 5146. 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:

Умняшкин С.В., Шаронов И.О.

Использование шестиугольных блоков
для компенсации движения при видеокомпрессии 2

Радченко Ю.С., Миляев С.В.

О корреляции спектральных коэффициентов
обобщенных рядов Фурье в дискретной области 9

Мацыпаев Д.А., Броневич А.Г.

Нечёткая параметризованная модель классификации
блоков небинаризованной маски движения 15

Минаков Е.И., Серегин П.С.

Сравнительный анализ методов параллельной
реконструкции изображений
магнитно-резонансной томографии 23

**Алпатов Б.А., Мураев В.С.,
Стротов В.В., Фельдман А.Б.**

Исследование эффективности применения
алгоритмов анализа изображений в задаче навигации
беспилотных летательных аппаратов 29

**Еремеев В.В., Макаренко А.А.,
Москвитин А.Э., Юдаков А.А.**

Повышение четкости отображения объектов
на данных гиперспектральной съемки земной поверхности 35

Егошкин Н.А., Еремеев В.В., Москвитин А.Э.

Комплексирование изображений от линеек фотоприёмников
в условиях геометрических искажений 40

**Кочергин А.М., Кузнецов А.Е.,
Побаруев В.И., Шokol А.С.**

Технология визуального геодезического ориентирования
космических снимков земной поверхности 45

Тишкин Р.В., Юдаков А.А.

Субпиксельная классификация объектов
на космических гиперспектральных изображениях 49

Петров Е.П., Харина Н.Л., Харюшин В.Ф.

Математические модели и алгоритмы фильтрации
цифровых полутонных изображений
на основе сложных цепей Маркова 52

Толочко А.М., Борискевич А.А.

Метод быстрого вычисления LBP-признаков
области интереса видеоизображения 58

Жизняков А.Л., Привезенцев Д.Г.

Использование характера распределения самоподобия
в качестве признака цифрового изображения
в задаче классификации 64

Пантелеев А.Ю.

Цифровая обработка сигналов
на современных графических процессорах 68

Медведева Е.В., Курбатова Е.Е.

Метод текстурной сегментации изображений
на основе марковских случайных полей 76

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

ИСПОЛЬЗОВАНИЕ ШЕСТИУГОЛЬНЫХ БЛОКОВ ДЛЯ КОМПЕНСАЦИИ ДВИЖЕНИЯ ПРИ ВИДЕОКОМПРЕССИИ

Умняшкин С.В., д.ф.-м.н., профессор Национального исследовательского университета РФ «МИЭТ»,
e-mail: vrinf@miee.ru

Шаронов И.О., аспирант Национального исследовательского университета РФ «МИЭТ»,
e-mail: igor.sharonov@gmail.com

Ключевые слова: шестиугольные блоки, дискретное вейвлет-преобразование, компенсация движения, видеосжатие.

Предварительные замечания

Видео – это последовательность визуально связанных статических изображений, имеющих два вида избыточности: временную и пространственную. Временная избыточность появляется вследствие сходства соседних кадров видеопоследовательности. Пространственная избыточность является свойством статического изображения: близко расположенные пиксели внутри изображения сильно коррелированы. Процесс видеокодирования обычно сначала устраняет временную избыточность, а затем пространственную.

Ключевой кадр, или I -кадр, – это кадр, который сжимается как статическое изображение. Кодирование и декодирование I -кадров происходит независимо от остальных кадров видеоряда; помимо первого кадра, их периодически вставляют в видеопоток. Следующий кадр сравнивается с предыдущим I -кадром поблочко, чтобы найти похожие области на изображениях. С помощью векторов перемещений, которые характеризуют перемещения блоков кадра, сначала строится изображение из блоков предыдущего кадра. Полученное изображение вычитается из оригинального кадра и формируется разностное изображение, кодируемое далее как статическое. Кадры, полученные с помощью компенсации движения по одному предыдущему, называются P -кадрами. Далее, данный процесс повторяется. Чаще всего элементарной областью поиска движения являются небольшие блоки, на которые разбивается весь кадр. Самой лучшей фигурой блока являлся бы круг, так как при заданной площади фигуры он объединяет наиболее близко расположенные точки изображения, а, следовательно, и наиболее коррелированные наилучшим образом. Однако покрыть изображение одинаковыми круглыми блоками без перекрытия невозможно.

В современных видеокодеках при анализе межкадрового перемещения используются квадратные блоки. Однако шестиугольные блоки по форме ближе к кругу, чем квадратные, поэтому можно ожидать увеличение

Рассматривается алгоритм компенсации движения на основе шестиугольной формы блоков в приложении для сжатия видеопоследовательностей. Для сжатия разностного изображения, получаемого после компенсации движения, использован алгоритм SPIHT на основе дискретного вейвлет-преобразования (ДВП). Экспериментально показано, что применение шестиугольной формы блоков при компенсации движения позволяет повысить PSNR обработанной видеопоследовательности в среднем на 0.15

визуального качества видео при использовании шестиугольных блоков.

Качество видеокompрессии может быть измерено в результате сравнения оригинального кадра с восстановленным изображением. Для подсчета разницы обычно используется среднеквадратическое отклонение (СКО, mean square error, MSE). Как характеристику качества обработки всей последовательности чаще всего используют пиковое отношение сигнала к шуму (peak signal to noise ratio, PSNR):

$$PSNR = 10 \lg \frac{255^2}{\frac{1}{P} \sum_{k=1}^P MSE^2(I_k, \tilde{I}_k)} \quad (\text{дБ}), \quad (1)$$

где I – оригинальный кадр, \tilde{I} – восстановленный кадр, P – общее количество кадров, k – порядковый номер кадра.

Предлагаемый нами видеокодек, структурная схема которого изображена на рис 1, представляет собой развитие [1], связанное с заменой в алгоритме компенсации движения квадратных блоков на шестиугольные. Для сжатия I -кадров и разностного изображения использовался метод SPIHT [2], а для энтропийного кодера и декодера был выбран арифметический кодер [3].

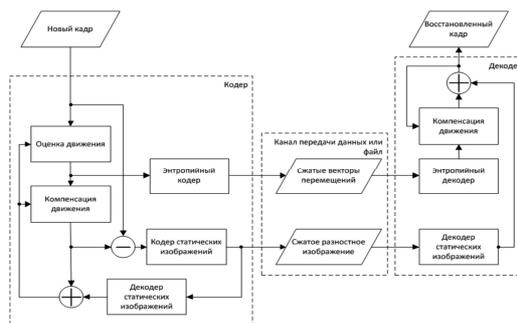


Рис. 1. Типичная схема кодирования/декодирования видеоряда

Блочная компенсация движения

Современные видеокодеки [4] используют различные методы для повышения визуального качества и степени сжатия видео. Чтобы уменьшить временную избыточность, применяется процедура компенсации движения, которая использует векторы перемещений блоков изображения, найденные на этапе оценки движения. Векторы перемещений получаются в результате деления области изображения на блоки одинаковой формы, обычно квадратные, и сравнения этих блоков с блоками на опорном кадре, чаще всего предыдущем и уже обработанном (рис. 2). Анализируемый блок попиксельно смещается в области поиска опорного кадра, пока не будет найдено оптимальное положение, которое определяется минимальным значением СКО. Вектор перемещения определяет оптимальное положение блока.

Существует множество методик уменьшения времени поиска векторов перемещений, см., например, [5, 6, 7]. В [1] разработаны шаблоны поиска для оценки движения, которые имеют решетчатую структуру (рис. 3). Конкретный шаблон применяется в зависимости от некоторых условий. Поиск по шаблону происходит только в закрашенных позициях рис. 3. Если найденный вектор попал в зону 1 или 2, то производится уточняющий поиск по окружающим незакрашенным позициям.

При восстановлении кадра в алгоритме блочной компенсации движения чаще всего используется перекрывающаяся весовая функция, накладываемая на блоки [8]. Такая технология называется блочной компенсацией движения с перекрытием (overlapped block motion compensation, ОБМС) и обеспечивает устранение нежелательных блочных артефактов на изображении. Также ОБМС используется и при оценке движения, чтобы получить более точные векторы перемещений.

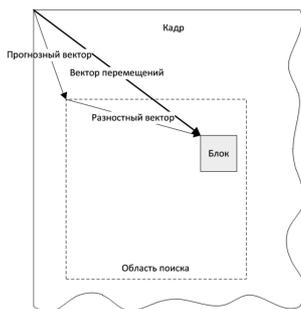


Рис. 2. Процесс оценки движения

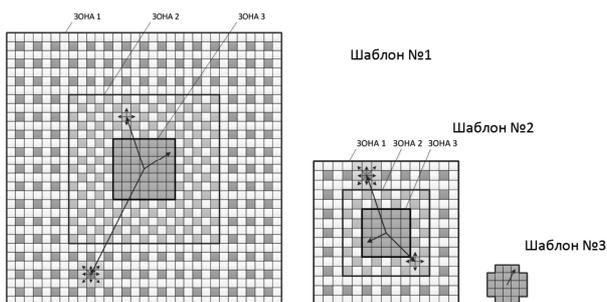


Рис. 3. Шаблоны поиска векторов перемещений блоков

Чтобы обеспечить баланс между качеством и битовым размером видео при поиске перемещенных блоков изображения, применяют процедуру минимизации RD-функции (rate-distortion, RD-optimization) [9]:

$$J(\vec{r}) = D(\vec{r}) + \lambda \cdot R(\vec{r}), \quad (1)$$

где $J(\vec{r})$ – это функция Лагранжа, у которой требуется найти минимум, \vec{r} – вектор перемещения блока, D – ошибка приближения блока кодируемого кадра блоком опорного кадра (обычно применяется СКО), R – количество битов, необходимых для энтропийного кодирования вектора \vec{r} , λ – коэффициент Лагранжа, который подбирается эмпирически. Для оптимального вектора перемещения \vec{r}^* функция $J(\vec{r}^*)$ достигает своего минимума, и соответствующий блок на опорном кадре считается наиболее подходящим для приближения текущего блока в обрабатываемом кадре.

Большинство векторов перемещений \vec{r} либо имеют нулевую длину, либо совпадают с вектором перемещения соседнего блока. Для учета данного факта применяются схемы предсказания вектора перемещения. Область поиска смещается на прогнозный вектор перед тем, как произвести поиск вектора перемещения. В таком случае вектор перемещения получается более коротким и может быть сжат более эффективно. Прогнозный вектор строится на основе анализа векторов соседних блоков, см., например, [10].

Предлагаемый нами видеокодек использует для сжатия разностного изображения кодер неподвижных изображений, основанный на дискретном вейвлет-преобразовании (ДВП). Имеется множество работ, посвященных видеокодированию [11, 12, 13], в которых используется ДВП, но алгоритм блочной компенсации движения в них основан на традиционной квадратной форме блоков.

Предлагаемый метод

Как уже упомянуто ранее, основой предлагаемого подхода является использование шестиугольной формы блоков при оценке движения, а не традиционной квадратной.

Шестиугольный блок

Предлагаемая шестиугольная форма блоков имеет ту же площадь, что и квадратные блоки 8x8, то есть 64 пикселя, и вкладывается в прямоугольник 8x10 пикселей (рис. 4а). Блок горизонтально вытянут, так как в видеопоследовательностях обычно преобладают горизонтальные перемещения объектов.

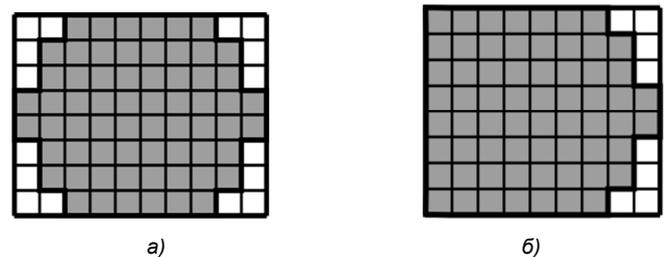


Рис. 4. а) пиксельная реализация шестиугольной формы; б) пиксельная реализация граничного шестиугольного блока

Разместить шестиугольные блоки по всей области изображения без пустот по краям невозможно. Поэтому используется пиксельное расширение границ изображения (рис. 5). Чтобы уменьшить количество блоков на изображении, используется специальный граничный блок на левой и правой сторонах кадра (рис. 4б).

Для исследования различных вариантов расширения изображения был проведен ряд экспериментов. Например, изображение расширялось добавлением пикселей со значением 0, 127 и 255. Другой вариант расширения заключался в зеркальном отображении изображения вдоль границы. Результаты показали, что лучшее качество восстановленного видео достигается при копировании последнего пикселя границы в область расширения (рис. 5). Такой именно вариант используется далее в нашей работе.

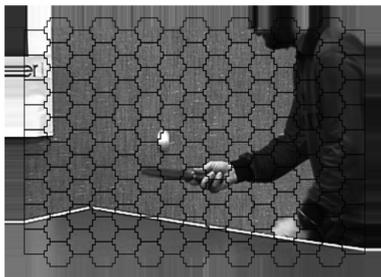


Рис. 5. Покрытие изображения шестиугольными блоками

Сегментация кадра шестиугольными блоками

Можно заметить, что при разбиении изображения на шестиугольные блоки структура расположения блоков различается по горизонтальной и вертикальной осям (рис. 5). Поэтому понятия «строка» и «столбец» должны быть определены заново. «Столбец» определяется как совокупность блоков, расположенных друг над другом по вертикальной оси (рис. 6а), а «строка» состоит из блоков, которые смещены друг относительно друга на половину размера блока по горизонтальной оси. После определения строки и столбца блока, можно найти все соседние блоки.

Обычно блоки имеют 6 соседей (см. рис. 6а):

- два соседа располагаются в левом верхнем и правом верхнем углах;
- два соседа располагаются в левом нижнем и правом нижнем углах;
- один сосед находится над рассматриваемым блоком;
- один сосед находится под блоком.

В [1] предлагается специальная группировка квадратных блоков по слоям (используется 3 слоя). Применение такой схемы позволяет уменьшить время поиска вектора перемещения, вследствие уменьшения области поиска для каждого следующего слоя, и повысить степень предсказания векторов. Следуя этой идее, опишем, как предлагается реализовать трехслойную схему для шестиугольных блоков.

Каждому шестиугольному блоку присваивается номер $L \in \{1, 2, 3\}$, соответствующий слою, к которому относится этот блок. Так, номер слоя для первого столбца блоков на рис. 6б может быть вычислен по формуле:

$$L = \text{mod}(i + 1, 3) + 1,$$

где i – индекс строки блока (нумерация строк и столбцов начинается с 0).

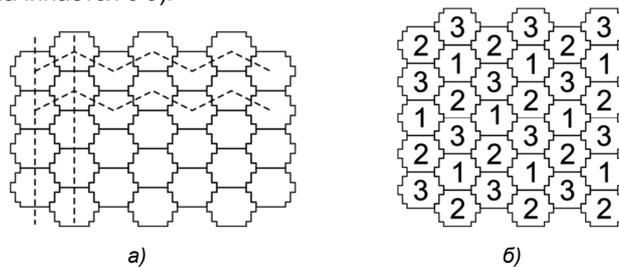


Рис. 6.

- а) строки и столбцы структуры шестиугольных блоков;
б) распределение блоков по трем слоям

Номер слоя для блоков из второго столбца на рис. 6б определяется как

$$L = \text{mod}(i + 2, 3) + 1$$

Правило вычисления номера слоя для третьего столбца имеет тот же вид, что и для первого; для четвертого – что и для второго, и т.д.

Номер L слоя блока с произвольными координатами (i, j) может быть вычислен следующим образом:

$$L = \text{mod}(i + \text{mod}(j, 2) + 1, 3) + 1$$

У блоков второго слоя есть 3 соседа из первого слоя. Блоки первого и второго слоев являются соседями для блоков третьего слоя рис. 6б. Количество блоков по слоям распределено примерно одинаково, поэтому на каждый слой приходится примерно по 1/3 площади всего изображения.

Алгоритм компенсации движения

После разбиения изображения на шестиугольные блоки поиск векторов перемещений производится слой за слоем.

Первый слой. Среди блоков первого слоя выбираются 1/6 блоков, для которых находятся векторы перемещений с помощью шаблона поиска № 1 (рис. 3). На основе найденных векторов, путем медианной фильтрации, строится глобальный вектор перемещения кадра. Далее его используют как прогнозный вектор для поиска векторов перемещений для первого слоя, используя тот же первый шаблон поиска.

Второй слой. На втором слое для поиска векторов перемещений используется шаблон поиска № 2 (рис. 3). Прогнозный вектор получается в результате медианной фильтрации векторов соседних блоков из первого слоя (рис. 7а). Область поиска затем смещается на вектор прогноза, и производится поиск векторов по шаблону поиска № 2.

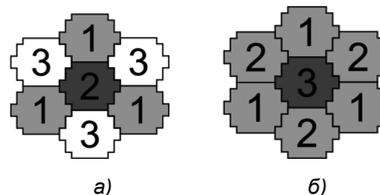


Рис. 7 а) формирование вектора прогноза;
б) формирование вектора прогноза для блоков 2-го слоя для блоков 3-го слоя

Третий слой. Используется шаблон поиска № 3, см. рис. 3. Прогнозный вектор строится на основе медианной фильтрации векторов всех соседних блоков (рис. 7б).

Во время поиска используется RD-оптимизация с применением ОВМС для подавления эффекта блочности изображения. В выходной битовый поток выдаются не сами векторы перемещений, а их отклонения относительно вектора прогноза, см. рис. 2. Для оценки битовых затрат R на кодирование вектора используется частота ν его появления в накопленной при обработке видеопоследовательности статистике, а именно, $R = -\log_2 \nu$. Последний шаг компенсации движения включает в себя вычисление разностного изображения. После вычитания предсказанного кадра из оригинального, диапазон разницы значения пикселей становится от -255 до +255. Для приведения разностного изображения к 8-битовому формату значения разностного изображения отображаются из отрезка [-255; +255] в отрезок [0; +255] с помощью следующей функции отображения [15]:

$$Y = \left[\frac{255}{2} \left(1 + \sin \left(\frac{\pi}{2} \times \frac{X}{255} \right) \right) \right],$$

где $X \in [-255; 255]$ и $Y \in [0; 255]$. Далее итоговое разностное изображение сжимается алгоритмом SPIHT [0].

Устранение эффекта блочности

Для шестиугольных блоков была разработана специальная весовая матрица (маска) A , которая уменьшает видимый блочный эффект после применения процедуры компенсации движения и повышает качество обработки изображений по критерию PSNR. При построении подходящей матрицы шестиугольных блоков выдвигались следующие требования:

- i. матрица симметрична по горизонтальной и вертикальной осям (поэтому достаточно рассматривать только $\frac{1}{4}$ матрицы, см. рис. 8а);
- ii. все матрицы для любого блока из любого слоя одинаковы;
- iii. матрица превышает размер блока на 2 пикселя с каждой стороны;
- iv. матрица строится на основе 2-х коэффициентов a и b , которые принадлежат отрезку [0; 1];
- v. равенство обоих коэффициентов a и b нулю означает, что блок не имеет весовой маски;
- vi. сумма весовых элементов матрицы с наложенными на нее элементами матриц соседних блоков дает единицу в каждой позиции элементов;
- vii. значения весов матрицы не должны убывать при перемещении от границ матрицы к ее центру.

Далее, в силу свойства (i), будет рассматриваться только $\frac{1}{4}$ весовой матрицы A (левая верхняя четверть), обозначим ее

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 & a & a & a \\ 0 & 0 & ab & a & b & b & b \\ 0 & 0 & a & b & 1 & 1 & 1 \\ 0 & ab & b & 1 & 1 & 1 & 1 \\ 0 & a & b & 1 & 1 & 1 & 1 \\ a & b & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Коэффициенты a и b имеют ограничение $a \leq b$. В области пересечения рассматриваемой матрицы с ее соседями находятся только три соседних матрицы (рис. 8б, левый верхний квадрант).

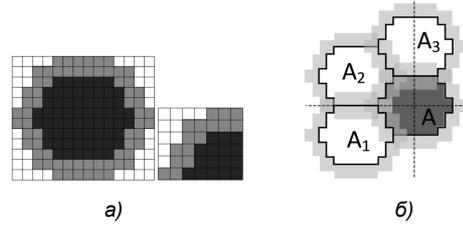


Рис. 8 а) весовая функция для шестиугольного блока; б) пересечение весовых матриц ее левая верхняя четверть

Очевидно, что весовая матрица для каждого соседнего блока (рис. 8б) – это та же самая матрица A . Зоны матриц A_1 , A_2 и A_3 (см. рис. 8б), перекрывающиеся с матрицей A , выглядят так:

$$A_1 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a & a & 0 & 0 & 0 & 0 & 0 \\ b & b & a & ab & 0 & 0 & 0 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 1 & 1 & 1 & b & a & 0 & 0 \\ 1 & 1 & 1 & 1 & b & a & 0 \\ 1 & 1 & 1 & 1 & b & a & 0 \\ 1 & 1 & 1 & b & a & 0 & 0 \\ 1 & 1 & 1 & b & ab & 0 & 0 \\ 1 & 1 & b & a & 0 & 0 & 0 \end{pmatrix}, \text{ и}$$

$$A_3 = \begin{pmatrix} 0 & ab & b & 1 & 1 & 1 & 1 \\ 0 & 0 & a & b & 1 & 1 & 1 \\ 0 & 0 & ab & a & b & b & b \\ 0 & 0 & 0 & 0 & a & a & a \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Обозначим за S матрицу суммы матриц A_1 , A_2 , A_3 и A :

$$S = A + A_1 + A_2 + A_3 = \begin{pmatrix} 1 & 1+ab & 1+b & 1+b & 1+2a & 1+a & 1+a \\ 1 & 1 & 1+ab+a & 1+a+b & 1+2b & 1+a+b & 1+b \\ 1 & 1 & 1+ab+a & 1+a+b & 1+2b & 1+a+b & 1+b \\ 1 & 1+ab & 1+b & 1+b & 1+2a & 1+a & 1+a \\ 1+a & 1+2a & 1+b & 1+b & 1+ab & 1 & 1 \\ 1+a+b & 1+2b & 1+a+b & 1+ab+a & 1 & 1 & 1 \end{pmatrix}$$

Все элементы матрицы S больше либо равны 1, но по свойству (vi) они должны быть строго равны 1. Деление каждого элемента a_{ij} матрицы A на соответствующие элементы s_{ij} матрицы S приводит к итоговой матрице A^* :

$$A^* = \begin{pmatrix} a_{ij} \\ s_{ij} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & \frac{a}{1+2a} & \frac{a}{1+a} & \frac{a}{1+a} \\ 0 & 0 & \frac{ab}{1+ab+a} & \frac{a}{1+a+b} & \frac{b}{1+2b} & \frac{b}{1+a+b} & \frac{b}{1+b} \\ 0 & 0 & \frac{a}{1+ab+b} & \frac{b}{1+a+b} & \frac{1}{1+2b} & \frac{1}{1+a+b} & \frac{1}{1+b} \\ 0 & \frac{ab}{1+ab} & \frac{b}{1+b} & \frac{1}{1+b} & \frac{1}{1+2a} & \frac{1}{1+a} & \frac{1}{1+a} \\ 0 & \frac{1}{1+2a} & \frac{b}{1+b} & \frac{1}{1+b} & \frac{1}{1+ab} & 1 & 1 \\ a & \frac{b}{1+a+b} & \frac{1}{1+2b} & \frac{1}{1+ab+a} & 1 & 1 & 1 \end{pmatrix}$$



Для поиска оптимальной весовой матрицы A^* задавались различные коэффициенты a и b . Если матрица не удовлетворяла свойству (vii), то она отбрасывалась; иначе проводилось кодирование тестовых видеопоследовательностей для получения характеристик PSNR. Матрица, для которой характеристики оказывались наилучшими, считалась оптимальной.

На рис. 9 приведена найденная оптимальная весовая матрица A^* , для которой в экспериментах достигается наилучшее качество видео. Проведенные эксперименты показали, что оптимальная матрица определяется значениями коэффициентов $a = 0,36$ и $b = 0,73$.

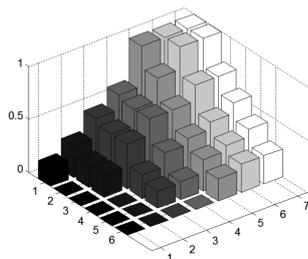


Рис. 9. Найденная оптимальная весовая матрица

Экспериментальные результаты

Сравнение разработанного видеокodeка, основанного на шестиугольных блоках, и видеокodeка, основанного на квадратных блоках [1], проводилось на одинаковых видеопоследовательностях и при следующих условиях:

I. коэффициент Лагранжа в (1) был выбран равным 100;

II. для обоих codeков использовались одинаковые шаблоны поисков для слоев из [1] (см. рис. 3);

III. оптимальные весовые матрицы для шестиугольной и квадратной формы блоков были найдены варьированием 2-х коэффициентов – параметров a, b ;

IV. в обоих случаях для кодирования векторов перемещений использовался один и тот же арифметический кодер с тремя статистическими моделями, с отдельной моделью для каждого слоя;

V. I-кадры кодировались одинаковым способом, с тем же битовым бюджетом и кадровым PSNR.

Для оценки видеокodeков сравнивались значения PSNR при одинаковом количестве битов на пиксель (bits per pixel, bpp):

$$R = \frac{B}{MN} \cdot (\text{bpp}),$$

где B – это количество битов для кодирования сжатого кадра, M и N – разрешение кадра в пикселях.

Битовые затраты R на кодирование сжатого кадра можно представить следующим образом:

$$R = V + S,$$

где V – количество битов на пиксель, потраченное при кодировании векторов перемещений, S – количество битов на пиксель для сжатия разностного изображения. Векторы перемещений сжимаются без потерь. Поэтому количество битов на пиксель, выделяемое для сжатия разностного изображения:

$$S = R - V.$$

Для полноты анализа результатов эксперимента мы приведем также значение битовых затрат на кодирование векторов перемещений V . Стандартная видео последовательность «Mobile», с разрешением 352x288, была обработана видеокodeком с шестиугольной формой блоков и видеокodeком с традиционной квадратной формой блоков.

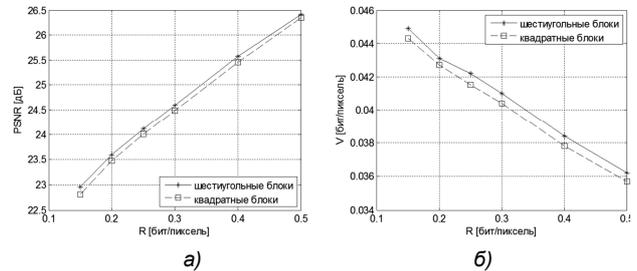


Рис. 10 а) PSNR, видеопоследовательность Mobile;
б) битовые затраты векторов перемещений, видеопоследовательность Mobile

Рис. 10а иллюстрирует изменение качества (в терминах PSNR) восстановленного видеоряда в зависимости от битовых затрат. Рис. 10б показывает, что битовые затраты кодирования векторов перемещений для шестиугольных блоков выше, чем для квадратных. Это означает, что оставшееся количество битов, которое используется для сжатия разностного изображения, для шестиугольной формы блоков меньше, чем для квадратной. Однако, итоговый график на рис. 10а говорит о лучшем качестве восстановленного видео именно при использовании шестиугольных блоков.

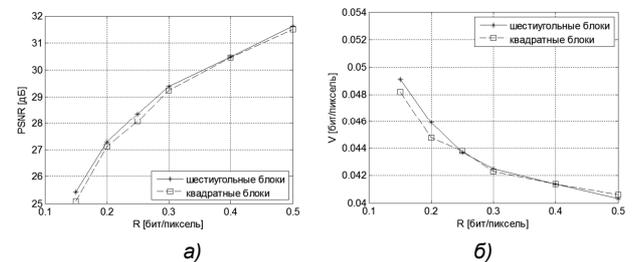


Рис. 11 а) PSNR, видеопоследовательность;
б) битовые затраты векторов перемещений, видеопоследовательность Tennis

Аналогичную ситуацию можно наблюдать на рис. 11а и 11б, на которых представлены результаты обработки видео последовательности «Tennis» разрешением 352x244.

В некоторых случаях (видео последовательность «Flower Garden» с разрешением 352x288) разработанный видеокodeк показывает примерно одинаковые PSNR по сравнению с традиционным видеокodeком, однако визуальное качество при использовании шестиугольных блоков более высокое. Битовые затраты векторов перемещений шестиугольных блоков в таких случаях все равно остаются выше.

На рис. 12 представлены восстановленные кадры после кодирования видеокodeком с квадратными блоками и разработанным видеокodeком на основе шестиугольной формы блоков, а также оригинальный кадр (81-й кадр из широко распространенной тестовой последовательности Tennis).



a)



б)



в)

Рис. 12 а) оригинальный кадр последовательности Tennis;
 б) пример восстановленного;
 в) пример восстановленного квадратными блоками кадра
 ($R=0,15$ bpp) шестиугольными блоками кадра ($R=0,15$ bpp)

Ценой за полученные улучшения является снижение производительности кодека. Процесс оценки движения на основе шестиугольных блоков примерно в 1,3 раза медленнее, чем аналогичный процесс для квадратных блоков. Первая причина заключается в том, что количество шестиугольных блоков в кадре больше на 3-5%, по сравнению с количеством квадратных блоков на кадре такого же разрешения. Вторая причина – распределение количества шестиугольных блоков на кадре по слоям составляет соотношение 1:1:1, в то время как соотношение количества квадратных блоков по слоям составляет 1:1:2. Поэтому шестиугольные блоки дольше обрабатываются на первом и втором слоях, из-за

больших размеров шаблонов поиска на этих слоях.

Заключение

В настоящей работе был предложен алгоритм компенсации движения на основе шестиугольной формы блоков. Для поиска векторов перемещений применялась трехслойная схема кодирования с различными шаблонами поиска из [1]. Результатом проведенных исследований является экспериментально подтвержденное улучшение визуального качества восстановленного видеоряда. Значение PSNR после перехода к шестиугольным блокам при фиксированных битовых затратах повысилось в среднем на 0.15 дБ. Следует отметить, что использование формы блоков близкой к шестиугольной дает хорошие результаты не только в алгоритмах компенсации движения [14].

Недостатками предложенной схемы, использующей шестиугольные блоки для оценки и компенсации движения, является падение производительности на этапе компенсации движения, в худшем случае на 20-30%.

Возможным направлением дальнейшей работы в развитии предложенной методики компенсации движения на основе шестиугольных блоков является применение субпиксельной компенсации движения [15], адаптации шестиугольной решетки, предложенной в [16], оптимизация шаблонов поиска для слоев с целью увеличения производительности.

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. Said, A. & Pearlman, W. A., «A New, fast and efficient image codec based on set partitioning in hierarchical trees,» IEEE Trans. Circuits Syst. Video Technol., vol. 6, pp. 243–250, June 1996.
3. Witten, I.H., Neal, R.M., Cleary, J.G., Arithmetic coding for data compression // Communications of the ACM, vol.30, no.6, pp. 520-540, June 1987.
4. «Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264/ISO/IEC 14 496-10 AVC», in Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVTG050, 2003.
5. Hashimaa, S.M., Mahmoud, I.I. & Elazm, A.A., «Experimental comparison among Fast Block Matching Algorithms (FBMAs) for motion estimation and object tracking», National Radio Science Conference (NRSC), 2011 28th , vol., no., pp.1-8, 26-28 April 2011.
6. Lei, J., Zhang J., Liu, J.& Cao, X., «An improved motion estimated three-step search algorithm based on H264», Computer technology and development, 2010, 12(20):5-7
7. Zhu, M.K.K., «A new diamond search algorithm for fast block matching motion estimation», IEEE Trans. Image Processing vol.9(2) 287-290 2000(2).
8. Song, B.C., Shin-Cheol, J. & Yanglim, C., «High-resolution image scalar using hierarchical motion estimation and overlapped block motion compensation», Consumer

Electronics, IEEE Transactions on , vol.56, no.3, pp.1579-1585, Aug. 2010

9. En-Hui Yang & Xiang Yu, «Rate Distortion Optimization for H.264 Interframe Coding: A General Framework and Algorithms», Image Processing, IEEE Transactions on, vol.16, no.7, pp.1774-1784, July 2007

10. Marcelino, S., Faria, S., Assuncao, P., Moiron, S. & Ghanbari, M., «Efficient MV prediction for zonal search in video transcoding», Multimedia Signal Processing (MMSP), 2010 IEEE International Workshop on, vol., no., pp.228-232, 4-6 Oct. 2010

11. Cagnazzo, M., Castaldo, F., Andre, T., Antonini, M. & Barlaud, M., «Optimal Motion Estimation for Wavelet Motion Compensated Video Coding», Circuits and Systems for Video Technology, IEEE Transactions on , vol.17, no.7, pp.907-911, July 2007

12. Ben Fradj, B. & Zaid, A.O., «Scalable video coding using motion-compensated temporal filtering», Visual Information Processing (EUVIP), 2011 3rd European Workshop on, vol., no., pp.50-55, 4-6 July 2011

13. Suliman, A. & Li, R., «Motion compensation in redundant wavelet domain», Southeastcon, 2011 Proceedings of IEEE, vol., no., pp.390-394, 17-20 March 2011

14. Al-Asmari & Awad, Kh., «Video compression using a semi-hexagonal shaped absolute moment block truncation

coding». Proceedings of 5th world Multiconference on systems, cybernetics, and informatics (SCI 2001) held in Orlando, Florida, U. S. A. July 22-25, 2001

15. Умняшкин С.В., Табориский А.Е. Метод обработки Р-кадров для видеокодека на основе дискретного вейвлет-преобразования. ЦОС, №3 – 2011.

16. He, X. & Jia, W., «Hexagonal structure for intelligent vision» Proc. 1st Int. Conf. Inf. Commun. Tech., 2005, pp. 52–64.

THE USE OF HEXAGONAL BLOCKS FOR MOTION COMPENSATION IN VIDEO COMPRESSION

Umnyashkin S., Sharonov I.

This paper investigates the motion compensation algorithm based on hexagonal form of blocks in application of video compression. The algorithm SPIHT based on discrete wavelet transform (DWT) has been used for the residual image compression obtained after the motion compensation procedure. The experiments showed that the usage of hexagonal form of blocks for the motion compensation has an average improvement of 0.15 dBs in terms of PSNR of the processed video sequence.

Уважаемые коллеги!

Приглашаем Вас принять участие в формировании тематических выпусков журнала «Цифровая обработка сигналов» и размещению рекламы продукции (услуг) Вашей организации на его страницах. В случае положительного решения просим представить в редакцию журнала Ваши предложения по плановому размещению информационных материалов и макет рекламы продукции (услуг) с указанием желаемого её месторасположения: обложка (2-я, 3-я или 4-я стр.), цветная внутренняя полоса (объем полосы).

Журнал «Цифровая обработка сигналов» издается с 1999 года. Выходит ежеквартально, тиражом – 700 экз. Распространяется по подписке через агентство «Роспечать» в России, СНГ и странах Балтии (индекс 82185), а также на Конференции и Выставке «Цифровая обработка сигналов и ее применение – DSPA'» (г. Москва). Более подробную информацию о журнале можно найти на сайте: www.dsps.ru.

Научно-технический журнал «Цифровая обработка сигналов» включен в Список изданий, рекомендуемый ВАК РФ для публикации результатов научных исследований соискателями ученой степени доктора и кандидата технических наук в области радиотехники, связи, вычислительной техники, электроники, информационно-измерительных и управляющих систем.

Планируемые сроки издания отдельных номеров журнала:

№ 4 – декабрь 2012 г. Тематический выпуск: «ЦОС в радиотехнике и системах телекоммуникаций».

№ 1 – март 2013 г. Тематический выпуск: «Теория и методы цифровой обработки сигналов».

№ 2 – июнь 2013 г. Специальный выпуск: «По материалам 15-й международной научно-технической конференции «Цифровая обработка сигналов и ее применение–DSPA'2013»».

№ 3 – сентябрь 2013 г. Тематический выпуск: «Цифровая обработка изображений».

Ориентировочная стоимость рекламных услуг:

4-я (внешняя) страница цветной обложки – 25 тысяч рублей.

2-я и 3-я (внутренние) страницы цветной обложки – 15 тысяч рублей.

1/2 цветной внутренней полосы – 8 тысяч рублей.

Ждем Ваших предложений. С наилучшими пожеланиями, зам. главного редактора д.т.н., профессор Витязев Владимир Викторович, телефон 8-903-834-81-81.

Предложения прошу направлять по адресу: E-mail: vityazev.v.v@rsreu.ru или info@dsps.ru

УДК 621.397

О КОРРЕЛЯЦИИ СПЕКТРАЛЬНЫХ КОЭФФИЦИЕНТОВ ОБОБЩЕННЫХ РЯДОВ ФУРЬЕ В ДИСКРЕТНОЙ ОБЛАСТИ

Радченко Ю.С., д.ф.-м.н., профессор кафедры радиофизики, Воронежский государственный университет, e-mail: ysradchenko@yandex.ru

Миляев С.В., аспирант кафедры радиофизики, Воронежский государственный университет, e-mail: sergey.milyaev@mail.ru

Ключевые слова: обработка изображений, сжатие сигналов, ортогональные дискретные преобразования, корреляция спектральных коэффициентов, неравномерная сетка отсчетов.

Введение

Сжатие цифровых изображений и видео основывается на дискретных ортогональных преобразованиях сигналов и последующей обработке полученных спектров. Различные дискретные преобразования обеспечивают разную степень декорреляции спектральных коэффициентов, что влияет на степень сжатия изображений и видео. С другой стороны корреляция спектральных коэффициентов в различных блоках позволяет использовать внутрикадровый прогноз для сокращения информационного потока на выходе кодера сигнала. Кроме того, знание величины корреляции спектральных мод необходимо для расчета количества информации в обрабатываемом изображении, позволяющего оценить близость характеристик реального кодера к теоретическому пределу.

Среди различных блочных дискретных преобразований сигналов с равномерной сеткой отсчетов наибольшее распространение получили косинусное и целочисленное псевдокосинусное преобразования [1]. Вместе с тем, было разработано дискретное полиномиальное преобразование, одним из вариантов которого является дискретное чебышевское преобразование GDCT [2, 3], использующее неравномерную сетку отсчетов.

Общее выражение для внутриблочной корреляционной матрицы спектральных коэффициентов для дискретных преобразований с равномерной сеткой приведено в [4]. Однако, расчет корреляционных моментов для спектральных коэффициентов разных блоков, а также расчет внутриблочной и межблочной корреляции спектральных мод при неравномерной сетке отсчетов остался открытым. Данная работа посвящена решению этой задачи.

Расчет внутриблочной корреляции

Рассмотрим одномерный сигнал, заданный на дискретном множестве отсчетов:

$$S_i = S_{0i} + \sigma \cdot X_i, \quad i = 1..N, \quad (1)$$

где S_{0i} – детерминированная величина, σ – СКО случайного отсчета, X_i – случайная величина, распре-

Выполнен расчет коэффициента корреляции спектральных коэффициентов дискретного косинусного преобразования, псевдокосинусного преобразования, дискретного чебышевского преобразования, внутри одного блока и между соседними блоками изображения. Проанализированы зависимости нормированных коэффициентов внутриблочной и межблочной корреляции спектральных коэффициентов от величины интервала корреляции. Показано, что дискретное чебышевское преобразование обеспечивает большую декорреляцию коэффициентов по сравнению с дискретным косинусным преобразованием. Полученные результаты были экспериментально подтверждены оценкой корреляции спектральных мод на реальном изображении.

ленная по нормальному закону:

$$X_i \sim N(0,1), \quad \langle X_i \cdot X_j \rangle = R(j-i).$$

Для конкретизации расчетов используем корреляционную функцию, которая применяется для описания гауссовских марковских процессов:

$$R(j-i) = \exp\left(-|j-i| \cdot \frac{\Delta\tau}{\tau_0}\right) = \rho^{|j-i|}, \quad \rho = \exp\left(-\frac{\Delta\tau}{\tau_0}\right), \quad (2)$$

здесь $\tau_0, \Delta\tau$ – параметры корреляционной функции.

Пусть $\{\varphi_k(i)\}$ – ортогональный векторный базис, определяющий матрицу дискретного ортогонального преобразования. Тогда спектральные коэффициенты разложения дискретного сигнала по этому базису будут иметь следующий вид:

$$C_k = \sum_{i=1}^N \varphi_k(i) \cdot S_i = \sum_{i=1}^N \varphi_k(i) \cdot S_{0i} + \sigma \cdot \sum_{i=1}^N \varphi_k(i) \cdot X_i, \quad (3)$$

где N – число дискретных отсчетов сигнала (1). Рассмотрим вначале статистические свойства спектральных коэффициентов дискретных отсчетов сигналов, заданных на равномерной сетке. С такими векторами отсчетов работают дискретное преобразование Фурье, дискретное косинусное и псевдокосинусное преобразование, дискретное преобразование Уолша-Адамара и ряд других.

В случае дискретного косинусного преобразования различные спектральные коэффициенты для одного блока будут выражаться следующим образом:

$$C_k = \gamma \cdot \sum_{i=1}^N S_i \cdot \cos\left(\pi \cdot k \cdot \frac{i-0.5}{N}\right) = \gamma \cdot M_k + \gamma \cdot \sigma \cdot \xi_k, \quad (4)$$

здесь γ – нормировочная константа ДКП, $k = 0..N-1$,

$$M_k = \sum_{i=1}^N S_{0i} \cos\left(\pi \cdot k \cdot \frac{i-0.5}{N}\right),$$

$$\xi_k = \sum_{i=1}^N X_i \cos\left(\pi \cdot k \cdot \frac{i-0.5}{N}\right) \quad (5)$$

- детерминированная и случайная составляющие спектрального коэффициента.

Выражение для дисперсии случайной составляющей спектральных коэффициентов имеет вид:

$$D(\xi_k) = \left\langle \left(\sum_{i=1}^N X_i \cdot \cos\left(\pi \cdot k \cdot \frac{i-0.5}{N}\right) \right)^2 \right\rangle = \quad (6)$$

$$\sum_{i=1}^N \sum_{j=1}^N \rho^{|j-i|} \cdot \cos\left(\pi \cdot k \cdot \frac{i-0.5}{N}\right) \cdot \cos\left(\pi \cdot k \cdot \frac{j-0.5}{N}\right)$$

Корреляционный момент случайных компонент ξ_k и ξ_m будет выглядеть следующим образом:

$$K_{dct}(k, m) = \langle \xi_k \cdot \xi_m \rangle =$$

$$= \sum_{i=1}^N \sum_{j=1}^N R(j-i) \cos\left(\pi \cdot k \cdot \frac{i-0.5}{N}\right) \cdot \cos\left(\pi \cdot m \cdot \frac{j-0.5}{N}\right) =$$

$$= \sum_{i=1}^N \sum_{j=1}^N \rho^{|j-i|} \cos\left(\pi \cdot k \cdot \frac{i-0.5}{N}\right) \cdot \cos\left(\pi \cdot m \cdot \frac{j-0.5}{N}\right). \quad (7)$$

Общее выражение для нормированного коэффициента корреляции между спектральными коэффициентами одного блока имеет вид:

$$R_{dct}(k, m) = \frac{K_{dct}(k, m)}{\sqrt{D(\xi_k) \cdot D(\xi_m)}}. \quad (8)$$

На рис. 1а и 1б представлены графики зависимости нормированного коэффициента корреляции спектральных коэффициентов четных и нечетных мод как функции коэффициента корреляции соседних отсчетов сигнала ρ для дискретного косинусного преобразования 8×8 .

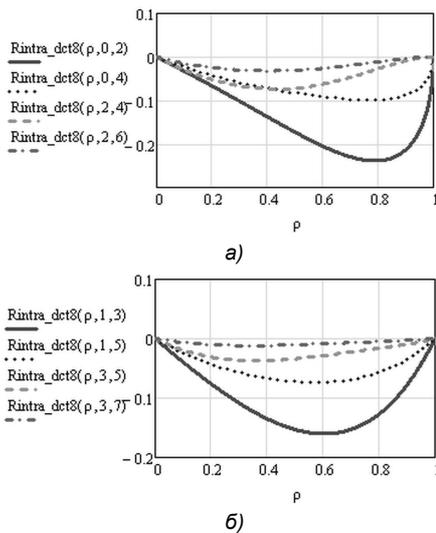


Рис. 1. Графики зависимости нормированного коэффициента корреляции спектральных коэффициентов внутри блока сигнала от корреляции соседних отсчетов сигнала для дискретного косинусного преобразования, а) – для четных мод, б) – для нечетных мод

Расчеты показывают, что взаимная корреляция между четными и нечетными коэффициентами отсутствует. Наблюдается некоторая корреляция только между ближайшими четными или нечетными коэффициентами. С увеличением разности $|k - m|$ взаимная корреляция становится пренебрежимо малой. Наибольший коэффициент корреляции между спектральными коэффициентами одного блока не превосходит по модулю 0.25 – 0.3.

В современных алгоритмах сжатия видео широкое применение находит псевдокосинусное целочисленное дискретное преобразование для блоков размера 4×4 , матрица преобразования которого имеет следующий вид [5]:

$$H = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{pmatrix}.$$

Графики нормированного коэффициента корреляции для спектральных коэффициентов целочисленного ДКП показаны на рис. 2. На рис. 2а изображена зависимость нормированного коэффициента корреляции нулевой моды и первой (сплошная линия) и нулевой моды и второй (пунктирная линия). На рис. 2б изображена зависимость нормированного коэффициента корреляции первой моды и третьей. На обоих графиках точечными линиями отмечены зависимости нормированного коэффициента корреляции для дискретного косинусного преобразования.

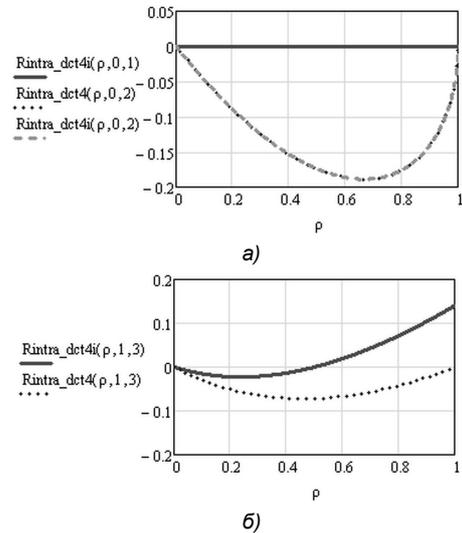


Рис. 2. Графики зависимости нормированного коэффициента корреляции спектральных коэффициентов внутри блока сигнала от корреляции соседних отсчетов сигнала для целочисленного дискретного псевдокосинусного преобразования, а) – для четных мод, б) – для нечетных мод

Закономерности в поведении корреляционных моментов от номера спектральных коэффициентов псевдокосинусного и косинусного преобразований, в целом, аналогичны. Имеет место некоторое различие численных значений $R_{km}(\rho)$, а также отличие в поведении кривых при $\rho \rightarrow 1$.

Рассмотренные выше случаи относились к дискретным преобразованиям сигналов, отсчеты которых выполнены по равномерной сетке. Однако возможны ортогональные преобразования, для которых сетка отсчетов берется неравномерной [2, 3, 6, 7]. Одним из таких преобразований с неравномерной дискретизацией является чебышевское преобразование – GDCT [2, 3].

Прямое и обратное преобразование GDCT определяется соотношениями

$$C = [C_{mk}] = \Phi S, S = \Psi^T C, \quad (9)$$

где матрицы прямого и обратного чебышевского преобразования имеют вид

$$\Phi = \sqrt{\frac{2}{N}} \begin{bmatrix} \sqrt{0.5} \\ \cos\left(\pi m \frac{(i+0.5)}{N}\right) \\ \vdots \\ \sqrt{0.5} \end{bmatrix}, \quad \Psi = \sqrt{\frac{2}{N}} \begin{bmatrix} \sqrt{0.5} \\ \cos\left(m \cdot \arccos\left(\frac{2n-1}{L-1}\right)\right) \\ \vdots \\ \sqrt{0.5} \end{bmatrix}. \quad (10)$$

$i = 0..N-1, m = 0..M-1 \leq (N-1), n = 0..L-1, N$ – число сэмплов (вторичных отсчетов) сигнала, $N \leq L, L$ – размер исходного и восстановленного блока. Дискретные чебышевские отсчеты сигнала (сэмплы) образуют вектор $S = [s_i] = [s(x_i)], i = 0..N-1$. Точки чебышевских отсчетов формируются по закону

$$x_i = 0.5 \cdot (L-1) \cdot \left(1 + \cos\left(\pi \frac{(i+0.5)}{N}\right)\right) \quad (11)$$

GDCT (чебышевское) преобразование требует перехода от первичной равномерной дискретизации сигнала с шагом $\Delta\tau$ к неравномерному сэмплированию по закону расположения нулей полинома Чебышева $T_N(z)$:

$z_i = \cos\left(\pi \frac{(i+0.5)}{N}\right), i = 0..(N-1)$ [2, 3]. В этом случае расположение сэмплов на интервале определения сигнала задается законом $x_i = (L-1)\Delta\tau \frac{(1+z_i)}{2}$. В таком

случае корреляционная матрица сэмплов внутри блока сигнала принимает вид

$$R_{i,j} = \rho^{|i-j|} \left[0.5(L-1) \left(\cos\left(\pi \frac{i+0.5}{N}\right) - \cos\left(\pi \frac{j+0.5}{N}\right) \right) \right]. \quad (12)$$

Здесь параметр ρ аналогичен введенному в (2). GDCT коэффициенты определяются выражением

$$C_k = \gamma \sum_{i=0}^{N-1} S_i \cos\left(\pi k \frac{i+0.5}{N}\right) = \gamma M_k + \gamma \sigma \xi_k, \quad (13)$$

которое аналогично (4) с тем отличием, что сэмплы S_i вычислены в точках x_i .

Моментные функции случайной компоненты спектра имеют вид

$$\langle \xi_k \rangle = 0,$$

$$D(\xi_k) = \left\langle \left(\sum_{i=0}^{N-1} X_i \cos\left(\pi k \frac{i+0.5}{N}\right) \right)^2 \right\rangle =$$

$$\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} R_{i,j} \cos\left(\pi k \frac{i+0.5}{N}\right) \cos\left(\pi k \frac{j+0.5}{N}\right). \quad (14)$$

Внутриблочная корреляция GDCT мод определяется моментом

$$K_{gdct}(k, m) = \langle \xi_k \xi_m \rangle = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} R_{i,j} \cos\left(\pi k \frac{i+0.5}{N}\right) \cos\left(\pi m \frac{j+0.5}{N}\right) \quad (15)$$

Соответственно нормированный коэффициент внутриблочной корреляции равен

$$R_{gdct}(k, m) = K_{gdct}(k, m) / \sqrt{D(\xi_k)D(\xi_m)}.$$

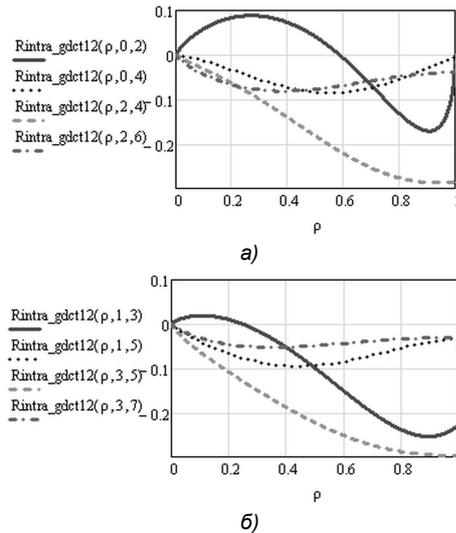


Рис. 3. Графики зависимости нормированного коэффициента корреляции спектральных коэффициентов внутри блока сигнала от корреляции соседних отсчетов сигнала для дискретного чебышевского преобразования, а) – для четных мод, б) – для нечетных мод

На рис. 3а и 3б представлены графики зависимости нормированного коэффициента корреляции спектральных коэффициентов четных и нечетных мод как функции коэффициента корреляции соседних отсчетов сигнала ρ для дискретного чебышевского преобразования при отношении $L/N = 12/8$. Из представленных на рис. 3 графиков видно, что GDCT обеспечивает большую декорреляцию коэффициентов преобразования по сравнению с ДКП. По сравнению с ДКП может иметь место, для ряда спектральных коэффициентов, особенность в поведении $R_{km}(\rho)$ при $\rho \rightarrow 1$. Эта особенность обусловлена тем, что при $\rho \rightarrow 1 D(\xi_k) \rightarrow 0, K_{km} \rightarrow 0$. Раскрытие неопределенности в нормированном коэффициенте корреляции типа 0/0 может привести к небольшим конечным значениям. У спектральных коэффициентов GDCT, так же как и при ДКП, четные и нечетные моды не коррелированы между собой. Небольшая корреляция имеет место только между ближайшими четными или нечетными спектральными коэффициентами.

Расчет межблочной корреляции

Исследуем теперь случай межблочной корреляции спектральных коэффициентов для различных видов дискретных преобразований. Пусть задан сигнал S_i , имеющий вид (1). В случае дискретного косинусного преобразования спектральные коэффициенты для соседних

блоков будут выражаться следующим образом:

$$C_k = \gamma \cdot \sum_{i=1}^N S_i \cdot \cos(\pi \cdot k \cdot \frac{i-0.5}{N}) = \gamma \cdot M_k + \gamma \cdot \sigma \cdot \xi_k \quad (16)$$

$$C_m = \gamma \cdot \sum_{i'=1+N}^{2 \cdot N} S_{i'} \cdot \cos(\pi \cdot m \cdot \frac{i'-0.5}{N}) = \gamma \cdot M_m + \gamma \cdot \sigma \cdot \eta_m$$

Здесь γ, σ аналогичны введенным выше,

$$M_k = \sum_{i=1}^N S_0(i) \cdot \cos(\pi \cdot k \cdot \frac{i-0.5}{N}),$$

$$M_m = \sum_{i'=1+N}^{2N} S_0(i') \cdot \cos(\pi \cdot m \cdot \frac{i'-0.5}{N}) \quad (17)$$

– детерминированные составляющие,

$$\xi_k = \sum_{i=1}^N X_i \cdot \cos(\pi \cdot k \cdot \frac{i-0.5}{N}),$$

$$\eta_m = \sum_{i'=1+N}^{2N} X_{i'} \cdot \cos(\pi \cdot m \cdot \frac{i'-0.5}{N}) \quad (18)$$

– случайные составляющие спектральных коэффициентов. В предположении о стационарности последовательностей X_i в пределах обоих блоков, дисперсии случайных составляющих имеют вид, определяемый формулой (6).

Взаимная корреляционная функция для случайных компонент ξ_k и η_m будет выглядеть следующим образом:

$$K_{dct}^{inter}(\xi, \eta) = \sum_{i=1}^N \sum_{j=1+N}^{2N} \rho^{|j-i|} \cdot \cos(\pi \cdot k \cdot \frac{i-0.5}{N}) \cdot \cos(\pi \cdot m \cdot \frac{j-0.5}{N}) = \sum_{i=1}^N \sum_{j=1}^N \rho^{|N+j-i|} \cdot \cos(\pi \cdot k \cdot \frac{i-0.5}{N}) \cdot \cos(\pi \cdot m \cdot \frac{j-0.5}{N}) \quad (19)$$

Общий вид выражения для коэффициента корреляции между спектральными коэффициентами соседних блоков определяется соотношением (8).

На рис. 4а и 4б представлены графики зависимости коэффициента корреляции между одинаковыми модами и между четными и нечетными модами соседних блоков от параметра корреляционной функции ρ для ДКП 8×8. Из графиков видно, что с ростом номера моды коэффициент корреляции быстро убывает. Так наиболее значимым, кроме коэффициента корреляции C_0 , является коэффициент корреляции C_1 , однако даже он имеет значения меньше 0.3. Таким образом, межблочной корреляцией всех коэффициентов, кроме C_0 , на практике можно пренебречь. Особый интерес представляет нулевой спектральный коэффициент C_0 разложения (16), который пропорционален нормированной сумме всех элементов вектора дискретного сигнала, и искажения которого при передаче наиболее сильно сказываются впоследствии на восстановленном сигнале. Из графика на рис. 4а видно, что при значениях параметра ρ меньших 0,8 коэф-

фициент корреляции нулевых спектральных коэффициентов соседних блоков не превосходит 0,3, что, как принято считать на практике, соответствует отсутствию линейной зависимости между этими величинами. Из рис. 4а следует, что межблочное предсказание DC коэффициента C_0 имеет смысл только при значении коэффициента межпиксельной корреляции $\rho \geq 0.9$.

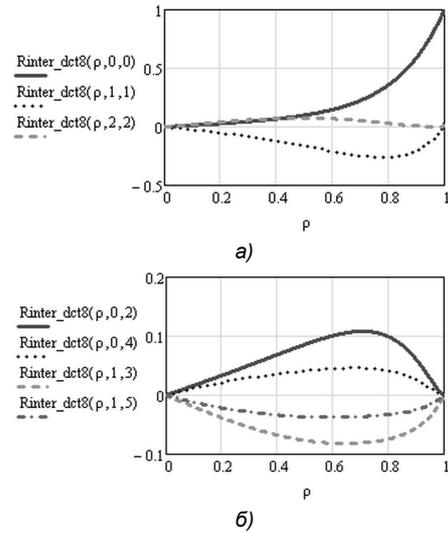


Рис. 4. Графики зависимости нормированного коэффициента корреляции спектральных коэффициентов между соседними блоками сигнала от корреляции соседних отсчетов сигнала для дискретного косинусного преобразования,

а) – для одинаковых мод, б) – для четных и нечетных мод

Графики нормированного коэффициента корреляции спектральных мод соседних блоков для спектральных коэффициентов целочисленного ДКП показаны на рис. 5. На рис. 5а изображена зависимость нормированного коэффициента корреляции одинаковых мод для соседних блоков. На рис. 5б изображена зависимость нормированного коэффициента корреляции между четными и нечетными модами соседних блоков.

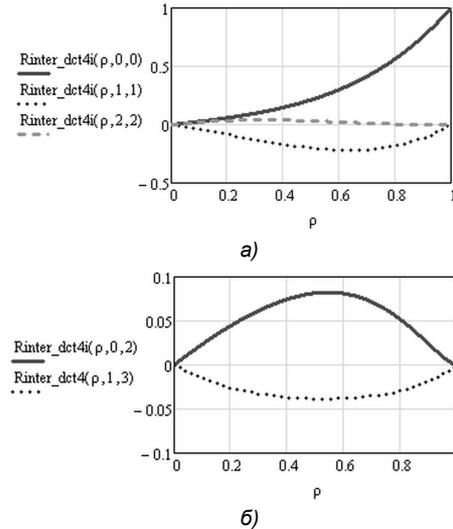


Рис. 5. Графики зависимости нормированного коэффициента корреляции спектральных коэффициентов между соседними блоками сигнала от корреляции соседних отсчетов сигнала для целочисленного дискретного косинусного преобразования,

а) – для одинаковых мод, б) – для четных и нечетных мод

При рассмотрении преобразования GDCT корреляционная матрица сэмплов между соседними блоками принимает следующий вид:

$$R(j, i) = \rho \left| 0.5(L-1) \left(\cos\left(\pi \frac{i+0.5}{N}\right) - \cos\left(\pi \frac{j+0.5}{N}\right) \right) + L \right|, \quad i, j = 0..N-1. \quad (20)$$

Межблочная корреляция GDCT мод соседних блоков определяется моментом

$$K_{gdct\ km}(\xi, \eta) = \langle \xi_k \eta_m \rangle = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} R(i, j) \cos\left(\pi k \frac{i+0.5}{N}\right) \cos\left(\pi m \frac{j+0.5}{N}\right). \quad (21)$$

Соответственно нормированный коэффициент межблочной корреляции равен

$$R_{gdct}^{inter}(k, m) = K_{gdct\ km}(\xi, \eta) / \sqrt{D(\xi_k)D(\eta_m)}, \quad (22)$$

где $D(\xi)$ и $D(\eta)$ определяются выражением (14).

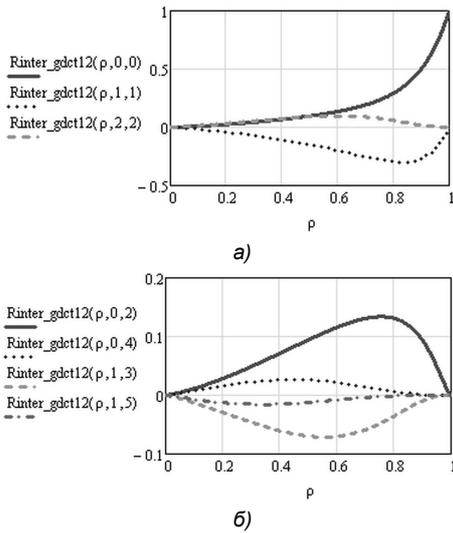
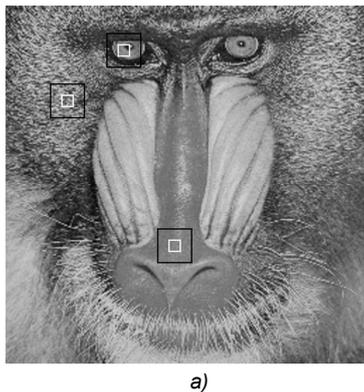
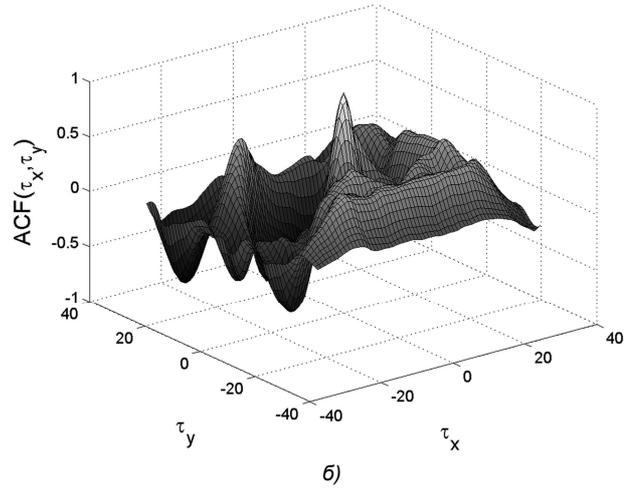


Рис. 6. Графики зависимости нормированного коэффициента корреляции спектральных коэффициентов между соседними блоками сигнала от корреляции соседних отсчетов сигнала для дискретного чебышевского преобразования, а) – для одинаковых мод, б) – для четных и нечетных мод

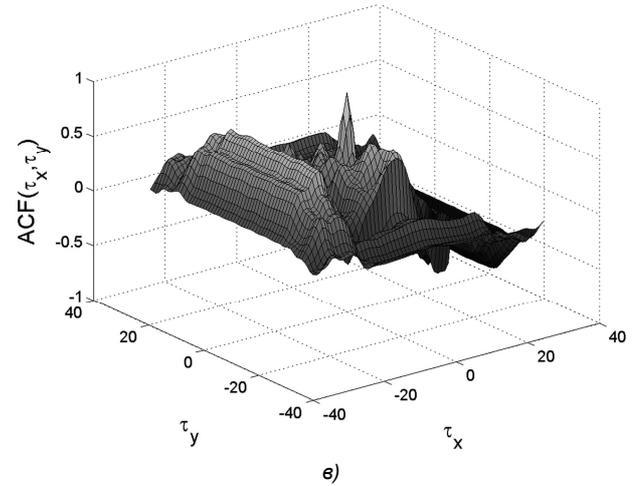
На рис. 6а и 6б представлены графики зависимостей коэффициента корреляции между одинаковыми модами и между четными и нечетными модами соседних блоков от параметра корреляционной функции ρ для преобразования GDCT при отношении $L / N = 12 / 8$.



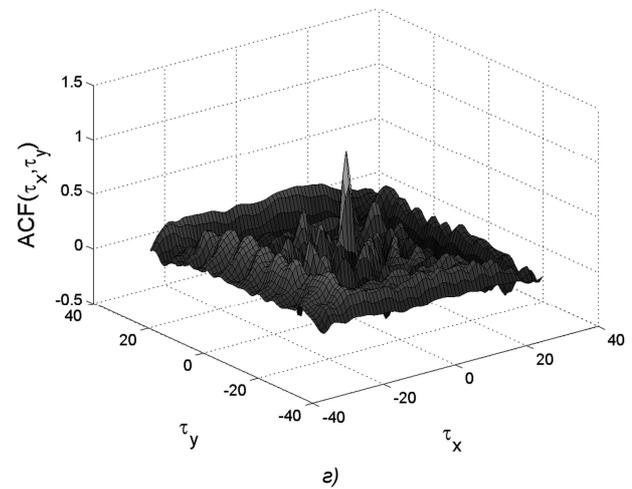
а)



б)



в)



г)

Рис. 7. Реальное изображение с областями различной степени корреляции, а) – исходное изображение с отмеченными областями, в которых проводилась оценка корреляции, б) – автокорреляционная функция для макроблока изображения в области высокой корреляции («глаз»), в) – автокорреляционная функция для макроблока изображения в области средней корреляции («нос»), г) – автокорреляционная функция для макроблока изображения в области низкой корреляции («шерсть»)

На рис. 6а представлена зависимость коэффициента корреляции между нулевыми модами соседних блоков преобразования GDCT. Сравнивая с аналогичным гра-



фином для преобразования ДКП на рис. 4а, видно, что при одинаковых значениях коэффициента корреляции между соседними отсчетами корреляция нулевых спектральных коэффициентов соседних блоков при GDCT преобразовании меньше, чем при ДКП.

Расчет корреляции спектральных мод на реальном изображении

На рис. 7а показано изображение «mandrill» с отмеченными черными прямоугольниками областями высокой («глаз»), средней («нос») и низкой («шерсть») корреляции. Белыми прямоугольниками выделены макроблоки, для которых строились автокорреляционные функции в областях высокой (рис. 7б), средней (рис. 7в) и низкой (рис. 7г) корреляции. По построенным автокорреляционным функциям было установлено, что интервалы корреляции в этих областях равны 0.9, 0.8, и 0.7 соответственно. Расчет корреляции нулевых спектральных мод C_0 дискретного косинусного преобразования 8×8 для блоков показывает, что ее значение равно 0.6 в области высокой межпиксельной корреляции, 0.47 в области средней межпиксельной корреляции и 0.19 в области низкой межпиксельной корреляции. Таким образом, полученные зависимости корреляции спектральных мод от интервала корреляции для модельного сигнала подтверждены экспериментальными результатами оценки корреляции спектральных мод, полученными на реальных изображениях.

Заключение

В работе были выполнены численные расчеты внутриблочной и межблочной корреляции для дискретного косинусного преобразования, дискретного чебышевского преобразования и целочисленного дискретного псевдокосинусного преобразования. Показано, что преобразование GDCT обеспечивает большую декорреляцию коэффициентов по сравнению с ДКП. Численные расчеты показали, что взаимная корреляция четных и нечетных мод рассмотренных преобразований равна нулю, ненулевой корреляцией обладают только ближайшие четные или нечетные моды. С увеличением разности индексов мод их взаимная корреляция стремится к нулю. Определены параметры корреляционной функции сигнала (коэффициент межпиксельной корреляции ρ), при которых целесообразен межблочный прогноз DC коэффициентов C_0 . Полученные результаты были экспериментально

подтверждены оценкой корреляции спектральных мод на реальном изображении.

Литература

1. Ричардсон Я. Видеокодирование. H.264 и MPEG-4 стандарты нового поколения. – М: Техносфера, 2005. – 368 с.
2. Радченко Ю.С. Алгоритм сжатия изображений на основе полиномиальных преобразований // Цифровая обработка сигналов. – 2002. – №1. – С. 2-6.
3. Радченко Ю.С. Метод сжатия и восстановления изображений на основе быстрых чебышевских преобразований // Автометрия. – 2002. – № 4. – С. 32-40.
4. Умняшкин С.В. Анализ эффективности применения ортогональных преобразований для кодирования дискретных сигналов с коррелированными отсчетами // Цифровая обработка сигналов. – 2008. – №4. – С. 15-18
5. H.S. Valvar, A. Hallapuro, M. Karczewicz Low-complexity Transform and Quantization in H.264/AVC //IEEE Trans on Circuit and System for Video Techn. - 2003. v13, №7. pp. 598-603.
6. W. Lin and D. Li, «Adaptive downsampling to improve image compression at low bit rates» // IEEE Trans. Image Process., 2006, vol. 15, №. 9. pp. 2513–2521.
7. Petre Stoica, Jian Li, Hao He «Spectral Analysis of Nonuniformly Sampled Data: A New Approach Versus the Periodogram» // IEEE Trans. on Signal Proc, v. 57, № 3, 2009. pp. 843-858

FOURIER SERIES SPECTRAL COEFFICIENTS CORRELATION IN DISCRETE SPACE

Radchenko Y.S., Milyaev S.V.

In this paper correlation coefficient of discrete cosine transform, integer cosine transform, and discrete Tchebyshev transform coefficients in continuous space in image single block and between neighbor blocks is estimated. Functional relation of spectral coefficients normalized intra-block and inter-block correlation coefficient from correlation distance is analyzed. It is shown that discrete Tchebyshev transform spectral coefficients are less correlated than discrete cosine transform spectral coefficients. Established results are proved by estimation of spectral coefficients correlation on a real image.

УДК 004.392

НЕЧЁТКАЯ ПАРАМЕТРИЗОВАННАЯ МОДЕЛЬ КЛАССИФИКАЦИИ БЛОКОВ НЕБИНАРИЗОВАННОЙ МАСКИ ДВИЖЕНИЯ

Мацыпаев Д.А., Таганрогский технологический институт Южного федерального университета, аспирант кафедры высшей математики, e-mail: dmitry.matsypaev@gmail.com

Броневиц А.Г., ОАО «Научно-исследовательский и проектно-конструкторский институт, автоматизации и связи на железнодорожном транспорте», главный научный сотрудник, доктор физико-математических наук, e-mail: brone@mail.ru

Ключевые слова: видеопоток, маска движения, небинаризованная, параметризованная модель, функционал Тверского, оценка надежности.

Введение

Предметом исследований, рассматриваемых в данной статье, является задача выделения движущихся областей в видеопотоке, который получен при помощи статической монокулярной камеры с фиксированным фокусным расстоянием. Суть предложенного алгоритма заключается в анализе небинаризованной маски движения. Такая маска движения содержит больше информации, чем её бинарный аналог после применения порогового преобразования. Небинаризованная маска движения разбивается на прямоугольные блоки, для каждого из которых считается соответствующее значение статистики, усредняющей меру движения внутри блока. Переход от пиксельного представления маски движения к блочному представлению позволяет сократить размерность области поиска и снизить влияние шумов на результат. В то же время использование блочного представления маски движения ограничивает геометрическую точность результата линейным размером блока. Для анализа функции распределения значений статистики блоков предлагается параметризованная математическая модель движущихся и покоящихся блоков. В данной модели вводятся функции принадлежности блоков данным множествам, предлагается критерий чёткого разграничения между двумя множествами. Далее вводятся характеристики, которые помогают оценить степень достоверности полученных результатов. Предложенная модель и её дальнейшие усовершенствования могут быть использованы в качестве компонента модуля видеоанализа в рамках системы видеонаблюдения.

Краткая постановка задачи

Пусть имеется последовательность кадров видеоряда $\{F^{(k)}\}_{k=0,1,2,\dots}$, которая была получена при помощи статической монокулярной видеокамеры с фиксированным фокусным расстоянием. Будем считать каждый кадр полутоновым изображением с разрешением $w_F \times h_F$ пикселей. Необходимо обнаруживать движущиеся объ-

Рассматривается задача выделения движущихся областей в видеопотоке. Предлагается алгоритм выделения движущихся объектов с использованием небинаризованной маски движения. Небинаризованная маска движения разбивается на прямоугольные блоки, для каждого из которых рассчитывается соответствующее значение статистики, усредняющей меру движения внутри блока. Для анализа функции распределения значений статистики блоков предлагается параметризованная математическая модель движущихся и покоящихся блоков. При помощи данной модели вводятся функции принадлежности блоков данным множествам, предлагается критерий чёткого разграничения между двумя множествами. Вводятся характеристики, которые помогают оценить степень достоверности полученных результатов.

екты на каждом кадре видеоряда и сопровождать их в видеопотоке. Основными этапами решения этой задачи являются:

- 1) построение маски движения для текущего кадра видеопоследовательности;
- 2) сегментация маски движения и выделение на ней областей, соответствующих движущимся объектам или группам объектов;
- 3) идентификация и сопровождение обнаруженных объектов.

Более подробно постановка задачи представлена в [8]. В рамках данной статьи рассматриваются первый и второй из выше перечисленных этапов решения задачи. В качестве маски движения предлагается использовать абсолютную разность двух соседних кадров видеоряда без применения порогового преобразования. Получаемая в результате небинаризованная маска движения разбивается на прямоугольные блоки, для каждого из которых рассчитывается значение введённой статистики. Анализ функции распределения данной статистики позволяет построить критерий разграничения блоков по признаку наличия движения. Выделение движущихся областей на маске движения осуществляется на основе отношения связности между движущимися блоками.

Небинаризованная маска движения

Общеизвестным и интуитивно понятным считается факт, что информации, содержащейся непосредственно в самом кадре $F^{(k)}$, недостаточно для того, чтобы локализовать области движения на нём. Необходимым условием для выделения движения является возможность доступа к информации, содержащейся в двух по-

следних кадрах видеоряда. В настоящее время наиболее популярными являются алгоритмы, в рамках которых используются статистические модели заднего плана для каждого пикселя кадра. Для построения и обновления таких моделей при анализе каждого следующего кадра требуется учитывать информацию, содержащуюся в нескольких десятках или сотнях последних кадров видеоряда [1, 3, 5, 7].

Независимо от природы используемого подхода, результаты работы таких алгоритмов для кадра $F^{(k)}$ представляются в виде маски движения $M^{(k)}$. Маска движения является полутоновым изображением с разрешением $w_F \times h_F$. Интенсивность пикселя в $M^{(k)}$ отражает степень принадлежности соответствующего пикселя из $F^{(k)}$ множеству движущихся пикселей. Широко распространённым является стремление получить бинарную маску движения путём применения порогового преобразования, что при удачном результате позволяет сразу же приступить к непосредственной сегментации движущихся объектов. В то же время использование бинарной маски движения имеет ряд недостатков, связанных с потерей существенной доли важной информации, которая содержалась в маске движения непосредственно перед её бинаризацией [8].

В [5] в качестве простейшего способа для построения маски движения рассматривается использование порогового преобразования для абсолютной разности интенсивностей соответствующих пикселей на двух соседних кадрах:

$$M_{ij}^{(k)} = \begin{cases} I_{\max}, & |F_{ij}^{(k)} - F_{ij}^{(k-1)}| \geq h \\ 0, & |F_{ij}^{(k)} - F_{ij}^{(k-1)}| < h \end{cases}$$

Значение $I_{\max} > 0$ задаёт диапазон, в котором изменяется величина интенсивности пикселя изображения. Обычно $I_{\max} = 2^d - 1$, где d - глубина представления цвета полутонового изображения в битах. При анализе видео практически всегда приходится работать с кадрами, глубина цвета которых равна $d = 8$ бит, что соответствует $I_{\max} = 255$. Очевидно, что

$$0 \leq |F_{ij}^{(k)} - F_{ij}^{(k-1)}| \leq I_{\max}. \quad \text{Соответственно,}$$

$0 \leq M_{ij}^{(k)} \leq I_{\max}$. Величина h является параметром порогового преобразования.

Очевидным преимуществом данного метода является его высокое быстродействие. В то же время в [5] подчёркивается, что этот метод не применим на практике вследствие слишком высокой чувствительности к значению h . Отметим также, что для формулы с пороговым преобразованием не предлагается никакой робастной эвристики для выбора значения h . Таким образом, при решении задачи не используется большое количество априорной информации, которая может существенно ограничить пространство поиска.

В этой статье мы постараемся показать, что абсолютная разность между двумя соседними кадрами может содержать в себе достаточно информации для устойчивой и аккуратной локализации движущихся облас-

тей на видео, если вместо применения порогового преобразования работать с небинаризованной маской движения

$$M_{ij}^{(k)} = |F_{ij}^{(k)} - F_{ij}^{(k-1)}|, \quad (1)$$

используя методы обобщённой теории информации [4] и априорную информацию, которая следует из постановки задачи.

Разбиение маски движения на блоки фиксированного размера и расчёт статистик для блоков

Использование формулы (1) для расчёта маски движения в идеальном случае отсутствия шумов на кадрах видеопоследовательности $\{F^{(k)}\}_{k=0,1,2,\dots}$ позволяет для

всех пикселей маски $M^{(k)}$ построить отношение порядка, упорядочив их по возрастанию соответствующего значения интенсивности. Введение такого отношения порядка очень важно для разграничения пикселей на множества движущихся и покоящихся. Между тем, в реальных приложениях шум присутствует всегда. Наличие шума влечёт большое количество ложных срабатываний при обнаружении движущихся пикселей.

Другая проблема попиксельного анализа заключается в том, что даже при самых низких разрешениях кадров видеопоследовательности (320×240 , 640×480) количество пикселей исчисляется сотнями тысяч. Одним из стандартных требований к алгоритмам видеоанализа является способность работать в режиме реального времени при достаточно высоком значении частоты смены кадров. Это накладывает серьёзные ограничения на сложность и эффективность применяемых алгоритмов. В то же время, априори известно, что движущиеся пиксели являются частями движущихся объектов в кадре и соответственно группируются в кластеры на основе геометрической близости, а шумовые пиксели представляются в виде одиночно располагающихся выбросов. Подобная априорная информация часто используется в компьютерном зрении, например при медианной фильтрации шумов или при уменьшении разрешения исходного изображения или видео. Определение областей движения обычно производится через выделение областей-кандидатов и на основе дополнительных статистических критериев. В общем случае подобный подход позволяет получить не только более робастные результаты, но и существенно сократить время анализа.

Одним из подходов, учитывающих групповые характеристики пикселей, является разбиение изображения на прямоугольные непересекающиеся области, которые называются блоками. Блочный подход хорошо зарекомендовал себя при сжатии и кодировании видео. В частности, в таком стандарте сжатия видео как H264 разбиение кадров на блоки размером от 4×4 до 16×16 позволяет определять векторы движения и достаточно точно выделять области движения на видео [6]. В работе [9] описан алгоритм рекурсивного блочного разбиения бинарной маски движения. Анализ полученных результатов показал, что данный алгоритм хорошо отделяет blobs на маске движения, если на ней отсутствуют шум, тени и «призраки». В работе [8] была рассмотрена другая модификация блочного метода (использовались

блоки фиксированного размера) применительно к небинаризованной маске движения. Был предложен критерий разграничения неподвижных и движущихся блоков, основанный на применении энтропии Шеннона при разграничении множества блоков, на котором была введена аддитивная мера количества движения. Также в [8] была поставлена задача обнаружения движения на видео и предложены простейшие критерии качества полученного разбиения. Главным недостатком предложенного метода оказался тот факт, что оптимальное значение параметра метода, от которого зависит разграничение движущихся и неподвижных блоков, может варьироваться в достаточно широком диапазоне значений при анализе кадров, принадлежащих одной видеопоследовательности. Тем не менее, анализ результатов показал перспективность обработки небинаризованной маски движения и использования блочного метода.

Недостатком такого подхода является ограничение геометрической точности результата размером анализируемых областей. С другой стороны, при решении далеко не каждой задачи необходима пиксельная точность, а геометрическую точность можно варьировать, изменяя размер блока.

Будем использовать процедуру разбиения маски движения $M^{(k)}$ на блоки фиксированного размера $w_B \times h_B$, аналогичную той, что представлена в [8]. Введём статистику для блока $B \subset M^{(k)}$, равную среднему значению интенсивности пикселей в блоке B :

$$I(B) = \frac{1}{w_B h_B} \sum_{i=0}^{w_B-1} \sum_{j=0}^{h_B-1} B_{ij}. \quad (2)$$

Очевидно, что множество значений статистики $I(B)$ ограничено $0 \leq I(B) \leq I_{\max}$.

Относительно статистики I мы также делаем предположение, что она достаточно хорошо упорядочивает блоки маски движения по степени принадлежности к объектам переднего и заднего планов. Чем больше значение статистики I , тем с большей достоверностью соответствующий блок относится к переднему плану. Выполнение данного предположения также означает, что для всех блоков $B \in U$ существует некоторое пороговое значение $I_{th} \in [0, I_{\max}]$, которое зависит от параметров алгоритма и для которого верно:

1. если $I(B) \geq I_{th}$, то блок B является движущимся блоком,
2. если $I(B) < I_{th}$, то блок B является покоящимся блоком.

Отметим также, что введение блоков размером $w_B \times h_B$ позволяет сократить количество анализируемых объектов на изображении с разрешением $w_F \times h_F$ приблизительно в $w_B h_B$ раз. Число получающихся при разбиении блоков равно $n \approx \frac{w_F h_F}{w_B h_B}$. Это оказывается весь-

ма существенным вкладом в увеличение производительности в случае использования алгоритмов с суперлинейной арифметической сложностью. Ниже мы будем

сортировать блоки по значению рассчитанной статистики I . Арифметическая сложность наилучших алгоритмов сортировки равна $O(n \log n)$. Это означает, что при размере блоков 16×16 и разрешении изображения в 640×480 сортировать блоки примерно в 450 раз эффективнее по быстродействию, чем сортировать соответствующие пиксели.

Параметризованная модель заднего и переднего планов

Будем называть множество блоков, соответствующих покоящимся на видеокadre пикселям, множеством блоков заднего плана и обозначать как U_{BG} . Множество блоков, соответствующих областям движущихся пикселей, будем называть множеством блоков переднего плана и обозначать как U_{FG} . Всё множество имеющихся блоков будем обозначать символом U .

Выше уже было изложено предположение о том, что статистика I в (2) хорошо разделяет движущиеся и покоящиеся блоки маски движения, полученной согласно выражению (1). Чем больше статистика I соответствующего блока, тем с большей степенью достоверности его можно отнести к множеству U_{FG} . Это предположение было подтверждено на практике во время проведённых экспериментальных исследований. Также в ходе эксперимента были отмечены ещё две закономерности:

- 1) блоки заднего плана в большинстве случаев количественно доминируют над блоками переднего плана;
- 2) распределение статистики I для блоков переднего плана доминирует над соответствующим распределением блоков заднего плана, занимая существенно более значительную часть диапазона значений $[0, I_{\max}]$.

Данные закономерности хорошо могут быть проиллюстрированы (см. рис. 1) типичным статистическим графиком функции распределения $F_I(x) = \left| \left\{ B \in U \mid I(B) \leq x \right\} \right| / |U|$.

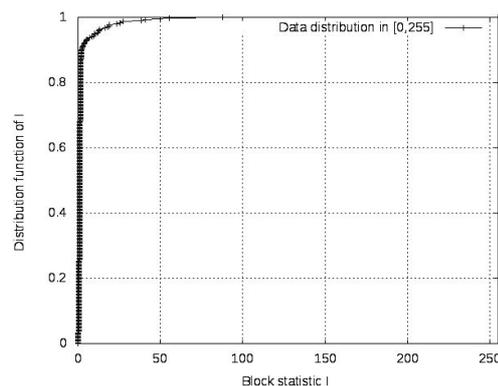
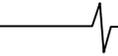


Рис.1. Функция распределения статистики I

Формализация приведённых выше закономерностей привела к созданию параметризованной модели заднего и переднего планов. Упорядочим блоки по возрастанию соответствующей им статистики I . Формализуем первую из закономерностей, изложенных выше. Для этого предположим, что к множеству U_{BG} относится не меньше, чем αn блоков, где $\alpha \in (0,1)$ - параметр модели. Поскольку мы имеем упорядоченный по статистике I мас-



сив блоков и предполагаем, что данная статистика хорошо разделяет блоки переднего и заднего плана, введём величину

$$I_\alpha = \min_{x \in [0, I_{\max}]} \{x \mid F_I(x) \geq \alpha\}.$$

Степень принадлежности блока B множеству U_{BG} будем измерять с помощью функции η_{BG} от статистики $I(B)$. Будем считать, что функция η_{BG} должна удовлетворять следующим условиям:

1. $\eta_{BG}(x)$ определена при $x \in [0, I_{\max}]$,
2. $\eta_{BG}(x) = 1, \quad x \in [0, I_\alpha]$,
3. $\eta_{BG}(I_{\max}) = 0$,

4. $\eta_{BG}(x)$ является монотонно невозрастающей функцией на $[0, I_{\max}] \quad \forall x, y \in [0, I_{\max}] \quad x \geq y \rightarrow \eta_{BG}(x) \leq \eta_{BG}(y)$.

Для формализации второй закономерности предположим, что распределение статистики I блоков переднего плана занимает долю $\beta \in (0, 1)$ диапазона значений $[0, I_{\max}]$, где β -параметр модели. Учитывая, что статистика I хорошо разделяет блоки переднего и заднего плана, введём величину

$$I_\beta = (1 - \beta)I_{\max}.$$

Степенью принадлежности блока B множеству U_{FG} будем считать значение функции η_{FG} от соответствующей статистики $I(B)$. Будем считать, что функция η_{FG} должна удовлетворять следующим условиям:

1. $\eta_{FG}(x)$ определена при $x \in [0, I_{\max}]$,
2. $\eta_{FG}(x) = 1, \quad x \in [I_\beta, I_{\max}]$,
3. $\eta_{FG}(0) = 0$,

4. $\eta_{FG}(x)$ является монотонно неубывающей функцией на $[0, I_{\max}] \quad \forall x, y \in [0, I_{\max}] \quad x \geq y \rightarrow \eta_{FG}(x) \geq \eta_{FG}(y)$.

Функционал Тверского для оценки степени схожести двух множеств и его применение в модели заднего и переднего планов

Как показал эксперимент, среди всего множества функций η_{BG} и η_{FG} , удовлетворяющих наложенным на них условиям, наиболее удачно согласуются с интуитивным представлением о множествах U_{BG} и U_{FG} функции, выведенные из параметризованного функционала Тверского. Параметризованный функционал Тверского используется в теории нечётких множеств [2, 4] для оценки степени схожести или сравнимости двух множеств [2]:

$$T_{a,b}(X, Y) = \frac{f(X \cap Y)}{f(X \cap Y) + af(X \setminus Y) + bf(Y \setminus X)},$$

где X, Y - множества, схожесть которых необходимо

оценить, \cup, \cap, \setminus - согласованные операции объединения, пересечения и разности двух в общем случае нечётких множеств, f - функционал, для которого верно $f(X \cup Y) = f(X) + f(Y) \quad \forall X, Y: X \cap Y = \emptyset, a, b \in R$ - параметры функционала Тверского.

Введём в рассмотрение множество g ограниченных чётких подмножеств действительной оси. Очевидно, что $\forall G_1, G_2 \in g$ справедливы следующие соотношения:

1. $G_1 \cup G_2 \in g$,
2. $G_1 \cap G_2 \in g$,
3. $G_1 \setminus G_2 \in g$.

Известно, что $\forall G_1, G_2 \in g \mid G_1 \cap G_2 = \emptyset \rightarrow L(G_1) + L(G_2) = L(G_1 + G_2)$, где L - мера Лебега. Поэтому можно определить функционал f как $f(G) = L(G)$ для всех $G \in g$. Очевидно, что $\forall \gamma, x \geq 0$ сегмент $[-\gamma/2, x] \in g$, а $L([-\gamma/2, x]) = x + \gamma/2$. Несложно также показать верность следующих соотношений:

1. $L([-\gamma/2, x] \cap [-\gamma/2, y]) = \min\{x, y\} + \gamma/2 \quad \forall x, y, \gamma \geq 0$,
2. $L([-\gamma/2, x] \setminus [-\gamma/2, y]) = \max\{x - y, 0\} \quad \forall x, y, \gamma \geq 0$.

Введём функцию $\varphi(x, y)$, численно равную значению функционала Тверского при $f(X) = L(X)$, $a = 1/2, b = 0$ от множеств $[-\gamma/2, x]$ и $[-\gamma/2, y] \quad \forall x, y, \gamma \geq 0$:

$$\begin{aligned} \varphi(x, y) &= T_{1/2, 0}([-\gamma/2, x], [-\gamma/2, y]) = \\ &= \frac{2 \min\{x, y\} + \gamma}{2 \min\{x, y\} + \gamma + \max\{x - y, 0\}}. \end{aligned}$$

Нетрудно видеть, что функция $\varphi(x, I_\alpha)$ как функция от аргумента x удовлетворяет первому, второму и четвёртому условиям, накладываемым на функцию $\eta_{BG}(x)$, но при этом не выполняется третье условие. Чтобы оно выполнялось, определим функцию $\eta_{BG}(x)$ как:

$$\eta_{BG}(x) = \frac{\varphi(x, I_\alpha) - \varphi(I_{\max}, I_\alpha)}{1 - \varphi(I_{\max}, I_\alpha)}.$$

Рассуждая аналогичным образом, введём функцию $\psi(x, y)$, численно равную значению функционала Тверского при $f(X) = L(X)$, $a = 0, b = 1/2$ от множеств $[-\gamma/2, x]$ и $[-\gamma/2, y] \quad \forall x, y, \gamma \geq 0$:

$$\begin{aligned} \psi(x, y) &= T_{0, 1/2}([-\gamma/2, x], [-\gamma/2, y]) = \\ &= \frac{2 \min\{x, y\} + \gamma}{2 \min\{x, y\} + \gamma + \max\{y - x, 0\}}. \end{aligned}$$

Функция $\psi(x, I_\beta)$ удовлетворяет первому, второму и четвёртому условиям, накладываемым на функцию

$\eta_{FG}(x)$, но не удовлетворяет третьему условию. Про-
нормируем значения функции $\psi(x, I_\beta)$ на сегменте
[0, I_{\max}] и определим функцию $\eta_{FG}(x)$ как:

$$\eta_{FG}(x) = \frac{\psi(x, I_\beta) - \psi(0, I_\beta)}{1 - \psi(0, I_\beta)}.$$

Продолжая аналогию, убеждаемся в том, что приве-
дѐнное выше выражение удовлетворяет условиям, на-
кладываемым на функцию $\eta_{FG}(x)$.

Параметр $\gamma > 0$ имеет калибровочный смысл и под-
бирается таким образом, чтобы выполнялись следую-
щие условия:

1. выражение для функции $\varphi(x, y)$ корректно при $x = 0, y = 0$,
2. $\varphi(0, 0) = 1$,
3. для двух фиксированных значений $x_0 \in (0, I_{\max}]$ и $\varphi_0 \in (0, 1)$ $\varphi(x_0, 0) = \varphi_0$.

Согласно третьему из выше приведѐнных условий,
величины x_0 и φ_0 однозначно определяют значение ка-
либровочного параметра γ модели переднего и заднего
планов как

$$\gamma = \frac{x_0 \varphi_0}{1 - \varphi_0}.$$

Несложно показать, что выбранное таким образом
значение калибровочного параметра γ удовлетворяет
также первому и второму условиям. В проведѐнных экс-
периментах для расчѐта параметра γ использовались
значения $x_0 = 1$ и $\varphi_0 = 2/3$ при $I_{\max} = 255$, что соот-
ветствует $\gamma = 2$.

**Вероятность принадлежности блока
множеству блоков переднего плана**

Очевидный теперь способ принятия решения относи-
тельно принадлежности блока B множеству блоков пе-
реднего плана U_{FG} состоит в определении статистики
 $I = I(B)$ и в проверке истинности неравенства:

$$\eta_{FG}(I) \geq \eta_{BG}(I). \tag{3}$$

Отметим, что в общем случае $\eta_{BG}(I) + \eta_{FG}(I) \neq 1$,
что не позволяет трактовать степени принадлежности
множествам U_{BG} и U_{FG} как соответствующие вероят-
ности. Между тем, вероятностная парадигма предостав-
ляет большое количество инструментов для оценки ка-
чества результатов и сложности исходных данных и ак-
тивно используется в области компьютерного зрения для
оценивания степени уверенности в результатах работы
алгоритмов.

Заметим, что согласно условиям, накладываемым на
функции $\eta_{BG}(x)$ и $\eta_{FG}(x)$, $0 \leq \eta_{BG}(I) + \eta_{FG}(I) \leq 2$
 $\forall I \in [0, I_{\max}]$. Введѐм характеристическую величину

$$\Delta(B) = 1 - \eta_{BG}(I(B)) - \eta_{FG}(I(B)).$$

Относительно этой величины примем следующие
предположения. Если $\Delta(B) > 0$, то она указывает на
количество недостающей информации в рамках модели
переднего и заднего планов для полного описания блока
 B и соответствующей ему статистике I . Если
 $\Delta(B) < 0$, то она указывает на то, что информация о
блоке B избыточна и противоречива. В случае же
 $\Delta(B) = 0$ описание блока B полно и непротиворечиво.

Для перехода к вероятностной парадигме воспользу-
емся таким инструментом обобщѐнной теории инфор-
мации как неточные вероятности [4]. Поскольку нас ин-
тересует вероятность принадлежности блока B множе-
ству блоков переднего плана U_{FG} , введѐм соответст-
вующую нижнюю вероятность как

$$\underline{p}_{FG}(B) = \underline{p}(B \in U_{FG}) = \min \{ \eta_{FG}(I(B)), 1 - \eta_{BG}(I(B)) \},$$

а соответствующую верхнюю вероятность как

$$\bar{p}_{FG}(B) = \bar{p}(B \in U_{FG}) = \max \{ \eta_{FG}(I(B)), 1 - \eta_{BG}(I(B)) \}.$$

Такой переход не противоречит определению неточ-
ных вероятностей, поскольку сегмент
[$\underline{p}(B \in U_{FG}), \bar{p}(B \in U_{FG})$] является выпуклым мно-
жеством [4]. Отметим также, что

$$\bar{p}(B \in U_{FG}) - \underline{p}(B \in U_{FG}) = |\Delta(B)|.$$

Пусть D - множество всех распределений вероятности
 p , удовлетворяющих введѐнным значениям нижней и
верхней вероятностей принадлежности блока B множе-
ству блоков переднего плана U_{FG} . Вероятностью при-
надлежности блока B множеству блоков переднего пла-
на U_{FG} будем считать ожидаемое значение вероятности
на множестве D :

$$p_{FG}(B) = p(B \in U_{FG}) = \int_{\underline{p}_{FG}(B)}^{\bar{p}_{FG}(B)} p \frac{dp}{\bar{p}_{FG}(B) - \underline{p}_{FG}(B)}.$$

Данное выражение путѐм элементарных преобразо-
ваний приведем к виду

$$p_{FG}(B) = \frac{\eta_{FG}(I(B)) + 1 - \eta_{BG}(I(B))}{2}. \tag{4}$$

Обратим внимание, что при $I(B) = 0$ $p_{FG}(B) = 0$, а
при $I(B) = I_{\max}$ $p_{FG}(B) = 1$. Заметим, что неравенство
 $p_{FG}(B) \geq 1/2$ равносильно неравенству (3). Вместе с
тем, выражение (4) позволяет заполнять множество бло-
ков переднего плана в соответствии с более общим кри-
терием

$$p_{FG}(B) \geq p_{th}, \tag{5}$$

где $p_{th} \in (0; 1]$.

Отметим также, что критерии (3) и (5) соответствуют выдвинутому выше предположению о том, что достоверность принадлежности блока B множеству блоков переднего плана U_{FG} и статистика I , определённая в (2), связаны друг с другом монотонно неубывающей функцией. Данное соответствие легко показать, если вспомнить, что функция $\eta_{BG}(x)$ монотонно не возрастает на $[0, I_{\max}]$, а функция $\eta_{FG}(x)$ монотонно не убывает на том же промежутке. Отсюда следует, что функция $(\eta_{FG} - \eta_{BG})(x)$ монотонно не убывает на $[0, I_{\max}]$, поскольку является разностью монотонно неубывающей и монотонно невозрастающей функций. Так как неравенство (3) можно переписать в виде $(\eta_{FG} - \eta_{BG})(x) \geq 0$, а неравенство (5) в виде $(\eta_{FG} - \eta_{BG})(x) \geq 2p_{th} - 1$, то необходимое соответствие можно считать доказанным.

Оценка надёжности результатов параметризованной модели переднего и заднего планов, оптимальные параметры модели, оценка качества маски движения

Введение аппарата неточных вероятностей позволяет оценить в битах общий уровень недоверия к результатам классификации блока B согласно следующей формуле [4]:

$$Mef(B) = S^*(D) = \sup_{p \in D} S(p),$$

где $S(p)$ - функционал, определяющий численное значение энтропии Шеннона для заданного распределения p из множества допустимых распределений D . Применительно к задаче классификации блока B с заданными классами U_{BG} и U_{FG} данное выражение имеет вид:

$$Mef(B) = \begin{cases} -\bar{p}_{FG} \log_2 \bar{p}_{FG} - (1 - \bar{p}_{FG}) \log_2 (1 - \bar{p}_{FG}), & \underline{p}_{FG} \leq \bar{p}_{FG} < 0.5 \\ 1, & \underline{p}_{FG} \leq 0.5 \leq \bar{p}_{FG} \\ -\underline{p}_{FG} \log_2 \underline{p}_{FG} - (1 - \underline{p}_{FG}) \log_2 (1 - \underline{p}_{FG}), & 0.5 < \underline{p}_{FG} \leq \bar{p}_{FG} \end{cases}$$

Зависимость величины Mef от статистики текущего блока можно проследить на рис. 2. Как и следовало ожидать, распределение данной величины имеет ярко выраженный максимум в той части диапазона $[0, I_{\max}]$, которая согласно используемой параметризованной модели блоков переднего и заднего плана соответствует блокам с переходными значениями статистики. Именно для таких блоков вероятность ошибки классификации наиболее велика. Обратим также внимание на то, что в крайних точках диапазона $[0, I_{\max}]$ неуверенность в результатах классификации ничтожно мала. Чтобы оценить уровень неопределённости, возникающий при обработке маски $M^{(k)}$, необходимо сложить неопределённости всех блоков, на которые была разбита $M^{(k)}$:

$$Mef(M^{(k)}) = \sum_{B \in M^{(k)}} Mef(B).$$

Эта величина является адекватным отражением на-

дёжности полученных при выделении блоков переднего плана результатом. Максимально возможная неопределённость при анализе маски движения равна количеству блоков $Mef_{\max} = n$ в битах.

Характеристической величиной маски движения $M^{(k)}$ будем считать

$$\Delta(M^{(k)}) = \frac{\sum_{B \in M^{(k)}} \Delta(B) Mef(B)}{Mef(M^{(k)})} = \frac{\sum_{B \in M^{(k)}} \Delta(B) Mef(B)}{\sum_{B \in M^{(k)}} Mef(B)}$$

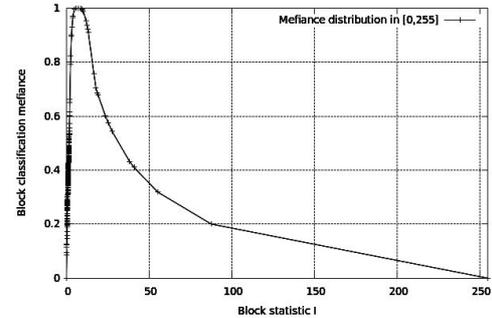


Рис. 2. Зависимость уровня недоверия к результатам классификации от статистики блока

Характеристическая величина маски движения является оценкой неполноты и противоречивости используемой параметризованной модели блоков переднего и заднего плана в масштабах одного кадра и может помочь при оптимизации параметров модели. Этот вопрос может стать предметом дальнейших исследований.

Конструирование блоков

Постановка задачи, подробно описанная в [8], подразумевает построение и анализ маски движения, а также выделение на ней блоков, которые соответствуют движущимся объектам или группам объектов. В [8] и [9] блобы строились на основе отношений 4- или 8-связности между движущимися блоками и представлялись в виде описанных вокруг найденных связанных компонент прямоугольников. В данной статье блобы представляются в виде выпуклых оболочек произвольной формы, описанных вокруг связанных подмножеств движущихся блоков. Множество движущихся блоков выбиралось согласно критерию (5) при различных значениях параметра p_{th} в диапазоне $[0.35, 0.8]$.

Результаты работы алгоритма

На рис. 3-5 представлен пример выделения движущихся объектов на одном из кадров видеопотока по маске движения. Небинаризованная маска движения на рис. 3 получена по формуле (1).

На рис. 4 представлены результаты анализа небинаризованной маски движения. Она разбита на блоки. Чем больше оцениваемая по формуле (4) степень принадлежности блока множеству блоков переднего плана, тем больше цвет границы блока отличается от фонового цвета рисунка. Согласно критерию (5) происходит отделение движущихся блоков от покоящихся. Из движущихся блоков конструируются блобы, выпуклые оболочки вокруг которых обозначены толстой линией более насыщенного цвета.

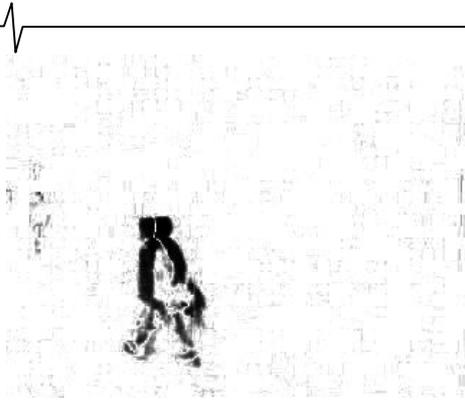


Рис. 3. Небинаризованная маска движения

На рис. 5 выпуклые оболочки блоков обозначены на исходном кадре видеопоследовательности, которому соответствует маска движения, представленная на рис. 3. Ещё несколько примеров выделения областей движения представлено в приложении к данной статье.

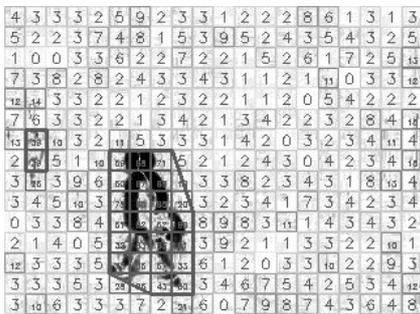


Рис.4. Результаты анализа небинаризованной маски движения

Для оценки качества предлагаемого алгоритма были вручную размечены три видеопоследовательности с разрешением 320×240, 384×288 и 848×560 пикселей. При проведении численных экспериментов размер блоков, на которые разбивалась маска движения, был равен 16×16 пикселей. Для сравнения результатов был выбран алгоритм смеси Гауссианов, описанный в работе [7] и реализованный в программной библиотеке OpenCV [1]. На рис. 6 представлены ROC-кривые обсуждаемого алгоритма и алгоритма из [7] для одной из размеченных видеопоследовательностей. Превосходство алгоритма из [7] в диапазоне чувствительности [0.13;0.97] объясняется блочной природой предлагаемого подхода, которая ограничивает снизу геометрическую точность результата величиной, производной от размеров блока.



Рис. 5 Выделение движущихся объектов на кадре видеоряда

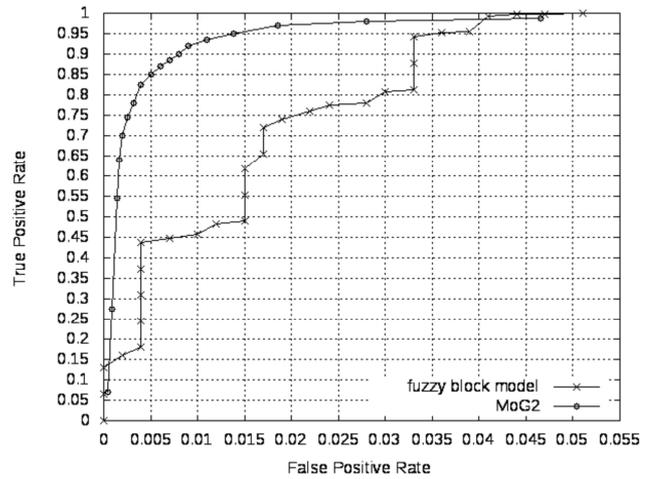


Рис.6. ROC-кривые для предлагаемого алгоритма (fuzzy block model) и алгоритма, описанного в [7] (MoG2)

Также было проведено сравнение быстродействия двух алгоритмов. В табл. 1 представлены сравнительные результаты анализа быстродействия представленного алгоритма и алгоритма из [7]. Из приведённых в таблице данных следует, что представленный алгоритм выполняется существенно быстрее своего аналога из [7]. Отметим, что указанная в таблице кадровая частота приведена без учёта операции чтения данных из видеопотока.

Таблица 1. Быстродействие предлагаемого алгоритма (fuzzy block model) и алгоритма, описанного в [7] (MoG2)

Видео	Разрешение	Fuzzy block model, кадров/сек.	MoG2, кадров/сек.
Видео 1	320x240	456.4	94.4
Видео 2	384x288	278.9	61.7
Видео 3	848x560	52.3	22.5

Заключение

В данной статье предложен алгоритм анализа небинаризованной маски движения для выделения движущихся объектов и групп объектов в видеопотоке. Главное преимущество небинаризованной маски движения состоит в том, что она может содержать в себе больше информации, чем её бинарный аналог. Другой особенностью алгоритма является разбиение маски движения на блоки прямоугольной формы и введение статистики, усредняющей интенсивность движения внутри блока. Далее на основе анализа функции распределения статистики блоков строится параметризованная модель движущихся и неподвижных блоков, которая позволяет оценить достоверность наличия движения в каждом из блоков. Предлагаются критерии разграничения движущихся и неподвижных блоков и вводятся величины, помогающие оценить надёжность полученных результатов.

Работа выполнена при финансовой поддержке РФФИ, проекты № 10-07-00135-а, 11-07-00591-а.



Рис.7. Примеры работы алгоритма

Литература

1. Bradski G., Kaehler A. Learning OpenCV. Computer vision with the OpenCV library. – Sebastopol: O'Reilly Media, Inc., 2008. – 555 p.
2. Cross V., Sudkamp T.A. Similarity and compatibility in fuzzy set theory: assessment and applications. Springer, 2002, 209 p.
3. P. KaewTraKuPong and R. Bowden, An improved adaptive background mixture model for real-time tracking with shadow detection, Proc. 2nd European Workshop on Advanced Video-Based Surveillance Systems, 2001.
4. Klir, G. J. Uncertainty and information. Foundations of generalized information theory. – New York: John Wiley & Sons, Inc., 2006. – 499 p.
5. Piccardi M. Background subtraction techniques: a review. The ARC Centre of Excellence for Autonomous Systems (CAS) Faculty of Engineering, UTS, April 15, 2004
6. Richardson I.E.G. H.264 and MPEG-4 video compression: video coding for next generation multimedia, John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, 2003
7. Zivkovic, Improved adaptive Gaussian mixture model for background subtraction, International Conference Pattern Recognition, UK, August, 2004
8. Мацыпаев Д.А., Броневи́ч А.Г. Анализ маски движения в задаче обнаружения и сопровождения движущихся объектов на основе энтропии Шеннона. Труды ОАО «НИИАС». Сборник научных трудов. Выпуск 9. М.: Изд-во ООО «Группа ИДТ», 2011, с. 255-273.
9. Мацыпаев Д.А. Алгоритм рекурсивного разбиения

маски движения для выделения движущихся объектов в видеопотоке // Доклады 12-й Международной конференции «Цифровая обработка сигналов и её применение» ДСПА-2010. -2010. – Т.2. – Стр. 114-116.

A FUZZY PARAMETERIZED MODEL FOR CLASSIFYING BLOCKS OF NON-BINARY MOTION MASK

Matsypaev D.A., Bronevich A.G.

The paper is dedicated to the problem of moving object detection in the video stream. The essence of the proposed algorithm is the analysis of non-binary motion mask. Such a mask contains more information than its binary analogue after threshold transformation applied. The initial step of the algorithm is to divide a non-binary motion mask on rectangular blocks and calculate the statistic for each block. The presented statistic averages motion intensity inside the corresponding block. The block representation of a motion mask allows us to reduce the dimension of search space and to suppress the influence of noises on the result. However, it limits a geometrical precision of the result with the linear size of the block. In the paper a fuzzy parameterized model of moving and stationary blocks is also proposed for analyzing the distribution function of the introduced statistic. In this model the membership functions for moving and stationary blocks are introduced, which are then used for discriminating moving and stationary objects. Measuring the confidence degree for classification results is produced by the justified upper entropy functional from the theory of imprecise probability.

УДК 389.001

СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ ПАРАЛЛЕЛЬНОЙ РЕКОНСТРУКЦИИ ИЗОБРАЖЕНИЙ МАГНИТНО-РЕЗОНАНСНОЙ ТОМОГРАФИИ

Минаков Е.И., д.т.н., профессор кафедры радиоэлектроники Тульского государственного университета, e-mail: MRTTULA@yandex.ru

Серегин П.С., аспирант кафедры радиоэлектроники Тульского государственного университета

Ключевые слова: магнитно-резонансная томография, параллельная реконструкция изображений в магнитно-резонансной томографии, метод сжатых измерений.

Введение

Магнитно-резонансная томография (МРТ) является современным и быстроразвивающимся методом медицинской визуализации.

Долгое время двумерное преобразование Фурье использовалось для реконструкции МРТ-изображений. Однако ввиду физических принципов МРТ, сбор данных в разных проекциях без акселерации для последующей Фурье реконструкции занимает десятки минут. Длительное время сканирования плохо переносят тяжелобольные пациенты. Помимо этого, визуализация путем Фурье-реконструкции подвижных органов (сосудов, сердца, легких и т.д.) с высоким разрешением во многих случаях становится невозможной ввиду наличия артефактов движений [15] (искажений изображений вследствие движения объекта исследования во время сканирования).

На смену классическому преобразованию Фурье пришел метод дробного преобразования Фурье [9, 14]. Данный метод требует меньшего числа исходных данных для реконструкции изображений (например, в 3/4, 5/6 или 7/8 раз). Метод дробного преобразования Фурье получил широкое распространение в промышленных МРТ-сканерах и используется по настоящее время.

Поиски более эффективных решений привели к появлению методов параллельной реконструкции изображений. Для использования параллельных методов реконструкции необходимо существенное изменение МРТ-систем: применение многоканальных антенных систем и приемных трактов. Использование методов параллельной реконструкции изображений позволяет в R-раз (где R – коэффициент акселерации) сократить время сбора данных МРТ (т.е. время МРТ-сканирования).

Очевидно, что при акселерации МРТ-сканирования реконструкция изображений выполняется по неполным исходным данным. Поэтому при использовании параллельных методов реконструкции изображений снижается отношение сигнал/шум (ОСШ) по сравнению с полным сканированием (т.е. сбором всех данных, необходимых для последующей Фурье-реконструкции) без акселерации. Компенсировать потерю ОСШ позволяет повышение постоянного магнитного поля МРТ-системы (вследствие того, что эффект ядерно-магнитного резо-

Проводится сравнительный анализ методов параллельной реконструкции изображений магнитно-резонансной томографии (МРТ) на основе моделирования. Рассматривается возможность регуляризации параллельной реконструкции изображений МРТ методом сжатых измерений. Особое внимание уделяется исследованию регуляризации метода SPACE-RIP. Кроме того, проводится сравнительный анализ методов параллельной реконструкции с другими методами регуляризации, в частности, с методом регуляризации на основе сингулярного разложения SVD.

нанса в МРТ-системах работает лучше с повышением напряженности поля), однако это сопровождается рядом сложностей и прежде всего значительным ростом стоимости подобных систем.

Параллельная реконструкция изображений подвижных объектов часто выполняется при достаточно высоких значениях коэффициента акселерации R (когда коэффициент акселерации сопоставим или превышает величину квадратного корня из числа антенных элементов МРТ-системы). Следствием этого является появление дополнительных искажений на изображениях и даже срыв реконструкции. Поэтому поиск методов реконструкции изображений МРТ, которые требовали бы меньшего числа исходных данных (т.е. способных работать при высоких коэффициентах акселерации) для качественной реконструкции изображений, является одной из важнейших и весьма актуальной задачей МРТ.

Существует достаточно много методик параллельной МРТ-реконструкции [15, 17], в основе которых лежит уравнение (1) [20]:

$$S(K_x, K_y) = \iint C(x, y) \rho(x, y) e^{-j(xK_x + yK_y)} dx dy, \quad (1)$$

где x, y – координаты декартового пространства изображений; K_x – координата K -пространства в направлении частотного кодирования; K_y – координата K -пространства в направлении фазового кодирования; $S(K_x, K_y)$ – полный уровень сигнала, принятого антенной системой (K -пространство), $C(x, y)$ – диаграмма направленности приемной антенной системы; $\rho(x, y)$ – функция распределения плотности водорода в исследуемом объекте (изображение объекта, которое нужно получить).

Наиболее распространенными из них являются SENSE, GRAPPA, PILS, SMASH, SPACE-RIP и др. Их общей особенностью является использование (1) с учетом различных дополнительных предположений о взаимосвязи диаграмм направленности антенн (т.е. чувствительности каналов) и исследуемого объекта.



Основная идея метода SMASH заключается в том, что интерпретация принятых МР-сигналов и оцениваемых диаграмм направленности антенн может непосредственно воспроизвести недостающие шаги фазового кодирования [15, 17]. В методике PILS, в отличие от SENSE и SMASH, основную роль играет представление о локализованности диаграмм направленности антенн в пределах их размеров. В методе GRAPPA используются автокалибровочные измерения [7, 11, 15, 17].

Для сравнительного анализа методов параллельной реконструкции использовалась библиотека Pulsar [6] для математического пакета Matlab. Реконструкция выполнялась с 8-канальной антенной системы для исследования головного мозга. Результат моделирования представлен в табл. 1. Под изображением приводится оценка погрешности реконструкции. Из табл. 1 видно, что при коэффициенте акселерации $R=6$ в методах SENSE и SPACE-RIP наблюдается срыв акселерации – изображение становится неразличимым. В то же время в методе PILS даже при $R=2$ наблюдаются сильные геометрические искажения, которые делают изображение непригодным для медицинского описания.

При изменении коэффициента акселерации, как видно из табл. 1, соотношение сигнал/шум наименее сильно изменяется при реконструкции с применением методики GRAPPA. Этот факт может объясняться применением данных автокалибровки. Наихудшие результаты по соотношению сигнал/шум и визуализации дают методики SMASH и PILS, что можно объяснить неравномерностью применения заложенных в их концепцию гипотез на примере конкретных исходных данных.

Из остальных методик реконструкции с незначительным преимуществом перед методом SENSE проявляет себя методика SPACE-RIP, что позволяет оценить ее в лучшую сторону. Далее в статье на ее примере будет рассмотрена возможность повышения качества реконструкции изображений.

Постановка задачи

В МРТ-системах сигналы, содержащие простран-

ственную информацию о плотности распределения водорода в исследуемом объекте, принимаются антенными системами. Далее сигналы усиливаются и происходит их дискретизация в АЦП. Таким образом, методы реконструкции и цифровой обработки МР-сигналов работают с дискретными данными.

Для построения МРТ-изображений необходима процедура реконструкции, так как принятые в антеннах МР-сигналы характеризуют объект исследования в частотной области. Поэтому дискретные отсчеты МР-сигналов заносят в массив данных, известный под названием K -пространство (Фурье-пространство или частотная область). Таким образом, задача МРТ-реконструкции состоит в преобразовании массива K -пространства в привычное человеку пространственное двумерное изображение исследуемого объекта (т.е. построение распределения плотности водорода).

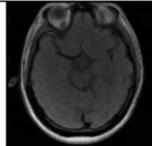
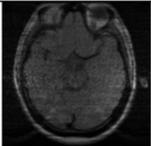
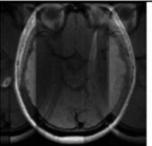
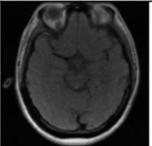
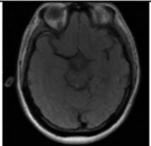
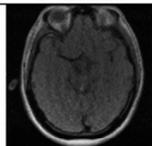
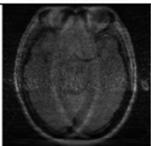
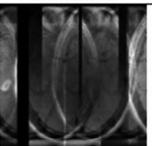
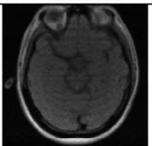
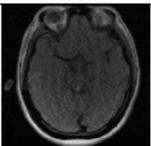
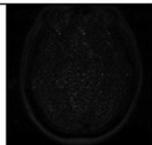
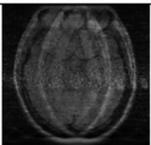
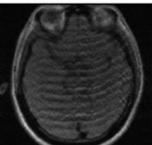
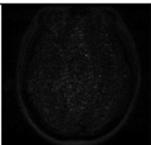
Основной моделью реконструкции изображений МРТ является двумерное обратное преобразование Фурье(1). Метод SPACE-RIP (Sensitivity Profiles From an Array of Coils for Encoding and Reconstruction in Parallel, дословный перевод: метод кодирования диаграмм направленности массива антенных систем МРТ и их параллельная реконструкция) [16] основывается на следующем приближении выражения (1):

$$S_l(x, \omega) = \int_{-\infty}^{\infty} C_l(x, y') \rho(x, y') e^{-j\omega y'} dy', \quad (2)$$

где x, y – координаты декартового пространства изображений; ω – координата K -пространства в направлении фазового кодирования; $S_l(x, \omega)$ – преобразованное K -пространство наблюдаемых сигналов с антенной системы номер l (в направлении частотного кодирования матрицы K -пространства выполнено одномерное обратное преобразование Фурье); $\rho(x, y)$ – реконструированное изображение, $C_l(x, y)$ – диаграмма направленности антенны канала l .

Таблица 1

Результат моделирования различных методов параллельной реконструкции изображений МРТ

	SENSE	SMASH	PILS	GRAPPA	SPACE-RIP
R=2					
	$\delta=0.0411$	$\delta=0.36089$	$\delta=0.4500$	$\delta=0.06833$	$\delta=0.0414$
R=4					
	$\delta=0.1531$	$\delta=0.4032$	$\delta=0.917$	$\delta=0.13769$	$\delta=0.1587$
R=6					
	$\delta=0.93872$	$\delta=0.46598$	$\delta=1.26$	$\delta=0.2499$	$\delta=1.063$

Уравнение (2) представляет собой интегральное уравнение Фредгольма первого рода относительно плотности распределения водорода $\rho(x, y')$. При больших степенях акселерации решение выражения (2) приобретает характер некорректно поставленной задачи, т.е. реконструкция изображения МРТ при высоких степенях акселерации не выполняется. Как известно, для таких классов задач применяются разнообразные методы регуляризации.

Одним из современных и активно развивающихся в других областях техники методом является метод сжатых измерений (compress sensing, или CS). Данный метод показал хорошую эффективность в приложениях, когда речь идет о работе с данными, степень редкости которых не удовлетворяет теореме Котельникова [8]. Далее будет показано исследование нескольких постановок метода сжатых измерений для регуляризации методов параллельной реконструкции изображений (для улучшения устойчивости и точности работы метода при работе с высокими степенями акселерации), а также сравнительный анализ с другими постановками регуляризации методов параллельной реконструкции (в частности, с методом регуляризации на основе сингулярного разложения SVD).

Регуляризация методом сжимающих измерений параллельных методов реконструкции МРТ-изображений

В последние несколько лет практическая эффективность метода сжатых измерений и, в частности, методика на основе l_1 -минимизации была успешно доказана [1, 2]. Реконструкция МРТ-изображений с применением метода сжатых измерений показана в работе [8].

В работе [19] проведен сравнительный анализ l_1 и l_0 -минимизации непосредственно для реконструкции МРТ-изображений. Автор [19] пришел к следующему выводу: метод l_0 -регуляризации не дает гарантии достижения минимума функции в отличие от l_1 -минимизации, которая это гарантирует. Однако l_0 -регуляризация способна работать при гораздо более разреженных данных, чем l_1 -минимизация. Результаты, приведенные в [19], показывают, что l_0 -минимизация имеет существенные преимущества перед l_1 -минимизацией на ряде сильно разреженных данных. Однако l_0 -реконструкция требует огромного числа вычислений, и часто ее решение из-за ряда факторов становится затруднительным. Поэтому в данной статье будет использоваться l_1 -минимизация.

Постановка задачи, согласно методу сжатых измерений, предполагает поиск максимально редкого вектора, удовлетворяющего заданному соотношению. Пусть имеется операторное уравнение вида

$$y = \Phi x, \quad (3)$$

где Φ – некоторый оператор, y – вектор известных наблюдаемых данных.

В терминах концепции сжатых измерений уравнение (3) с учетом требования редкости примет вид задачи оптимизации:

$$\|x\|_1 \rightarrow \min, \quad \Phi x = y. \quad (4)$$

Таким образом, в слабой постановке, удовлетворяющей практически потребностям (при наличии шума), (4) можно переписать так:

$$\|x\|_1 \rightarrow \min, \quad \|\Phi x - y\|_2 < \varepsilon, \quad (5)$$

где ε – величина, которая управляет точностью реконструкции к результатам измерений.

Исследование МРТ-изображений показало, что данные, как правило, не разрежены в любой заданной области преобразования, но действительно склонны к проявлению высокой сжимаемости.

Выполним предварительную дискретизацию (2) для того, чтобы было возможно перейти к (5). Для этого выберем некоторую систему функций, $\{\Omega_k(y)\}_{k=1}^{\infty}$, образующую базис в функциональном пространстве, которое допускает существование интегрируемых функций.

Также допустим, что система $\{\Omega_k(y)\}_{k=1}^{\infty}$ является ортонормальной, т.е. между функциями имеет место следующее соотношение:

$$\int_{-\infty}^{\infty} \Omega_i(y)\Omega_j(y)dy = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}. \quad (7)$$

Представим приближенно плотность распределения водорода и диаграммы направленности антенных каналов как результат разложения по системе $\{\Omega_k(y)\}_{k=1}^N$, где N – некоторый конечный номер, который может быть достаточно большим. Воспользуемся следующим приближением:

$$\rho(x, y) \approx \sum_{k=1}^N \alpha_k(x)\Omega_k(y),$$

$$C_l(x, y)e^{-j\omega y} \approx \sum_{m=1}^N \hat{C}_l(m, x)e^{-jm\omega}\Omega_m(y). \quad (8)$$

Подставляя приближенные представления (8) в (2), получим следующее:

$$S_l(x, \omega) = \int_{-\infty}^{\infty} C_l(x, y')\rho(x, y')e^{-j\omega y'}dy' =$$

$$= \int_{-\infty}^{\infty} \sum_{m=1}^N \hat{C}_l(m, x)e^{-jm\omega}\Omega_m(y) \cdot \sum_{k=1}^N \alpha_k(x)\Omega_k(y)dy' \quad (9)$$

Перегруппируем правую часть (9) таким образом, чтобы можно было внести базисные функции под знак интеграла:

$$\int_{-\infty}^{\infty} \sum_{m=1}^N \hat{C}_l(m, x)e^{-jm\omega}\Omega_m(y) \cdot \sum_{k=1}^N \alpha_k(x)\Omega_k(y)dy' =$$

$$= \sum_{m=1}^N \sum_{k=1}^N \alpha_k(x)\hat{C}_l(m, x)e^{-jm\omega} \int_{-\infty}^{\infty} \Omega_k(y)\Omega_m(y)dy \quad (10)$$

Далее, пользуясь свойством ортонормальности (8), замечаем, что ненулевыми останутся только те слагаемые, индексы базисных функций которых будут совпадать. Поэтому приближение интегрального уравнения (2) примет вид:

$$S_l(x, \omega) = \sum_{m=1}^N \hat{C}_l(m, x)\alpha_m(x)e^{-jm\omega}. \quad (11)$$

Группируя уравнения для каждого канала антенны в систему с учетом изменения параметра ω , получим сис-

тему уравнений, которая решается относительно коэффициентов $\alpha_m(x)$. По их значениям в свою очередь из (8) вычисляется плотность распределения водорода.

Пусть величина ω меняется в диапазоне $\omega = \omega_1, \omega_2 \dots \omega_L$. Рассмотрим (9) в матричной форме:

$$\hat{S}(x) = \hat{A}\hat{\alpha}, \quad (12)$$

где

$$\hat{S}(x) = (S_1(x, \omega_1), S_1(x, \omega_2) \dots S_1(x, \omega_L), S_2(x, \omega_1), S_2(x, \omega_2) \dots S_2(x, \omega_L), \dots S_k(x, \omega_L))^T$$

$$\hat{\alpha} = (\alpha_1(x), \alpha_2(x), \dots \alpha_N(x)).$$

Задачу (12) можно решать различными способами. Наиболее распространенными из них являются прямые методы, позволяющие переформулировать (12) как задачу метода наименьших квадратов. Это, кроме гибкой содержательной постановки, предоставляет возможность использования различных методов решения систем линейных уравнений. Для данного класса задач наиболее популярными из них являются два метода: метод сингулярного разложения (SVD-метод) и метод регуляризации Тихонова, заключающиеся в приведении системы линейных уравнений к некоторому специальному виду, что дает возможность решить их с высокой точностью при, как правило, плохой обусловленности главной матрицы системы [5, 7, 10, 12, 13].

С точки зрения соотношения (3), оператор \hat{A} является матрицей, связывающей вектор измерений $\hat{S}(x)$ и неизвестное решение $\hat{\alpha}$. В этом случае постановка метода сжатых измерений примет следующий вид:

$$\|\hat{\alpha}\|_1 \rightarrow \min, \quad \|\hat{S}(x) - \hat{A}\hat{\alpha}\|_2 < \varepsilon. \quad (13)$$

Требование редкости для $\hat{\alpha}$ может носить и содержательный характер, обусловленный тем, что при сканировании может иметь место прореживание данных. В этом случае требование редкости с учетом потерь информации помогает найти корректное решение задачи реконструкции.

Отметим, что, кроме (13), возможна альтернативная постановка, заключающаяся в изменении характера целевой функции и выбранных ограничений. Она заключается в том, чтобы, как и в методе наименьших квадратов, минимизировать норму отклонения $\|\Phi x - y\|_2$, но при условии, что вектор x должен быть достаточно редким [9]. В этом случае с учетом того, что редкость вектора мы будем оценивать как норму $\|\cdot\|_1$, задачи (2) и (5) могут быть переформулированы следующим образом:

$$\|\Phi x - y\|_2 \rightarrow \min, \quad \|x\|_1 \leq \tau, \quad (14)$$

где τ – показатель, учитывающий задаваемую в задаче редкость вектора. Задавая различную величину τ , можно контролировать степень точности решения задачи реконструкции методом сжимающего считывания. При этом постановку (14) иногда называют «лассо»-постановкой, что обусловлено аналогией с задачей о методе наименьших квадратов.

Таким образом, в постановке метода сжатых изме-

рений предлагаются следующие варианты решения задачи SPACE-RIP реконструкции:

– вариант строгой постановки задачи с ограничениями-равенствами:

$$\|\hat{\alpha}\|_1 \rightarrow \min, \quad \hat{S}(x) = \hat{A}\hat{\alpha}; \quad (15)$$

– вариант ослабленной постановки задачи с учетом возможной погрешности реконструкции:

$$\|\hat{\alpha}\|_1 \rightarrow \min, \quad \|\hat{S}(x) - \hat{A}\hat{\alpha}\|_2 < \varepsilon; \quad (16)$$

– вариант «лассо»-постановки:

$$\|\hat{S}(x) - \hat{A}\hat{\alpha}\|_2 \rightarrow \min, \quad \|\hat{\alpha}\|_1 \leq \tau. \quad (17)$$

Алгоритмы реализации методов сжимающих измерений

Совсем недавно исследователями были предложены новые числовые методики решения задачи l_1 -минимизации: предобусловленный метод сопряженных градиентов (PCG), метод итераций Брегмана [12], или метод пространства со шкалой обратных величин, и метод повторно взвешенных наименьших квадратов.

В качестве алгоритма решения задач (15) – (17) в данной статье использовался метод базиса преследования (basis pursuit) из библиотеки SPGL1 [18] для математического пакета Matlab.

Моделирование методов параллельной реконструкции. Оценка изображений

Качество МРТ-изображений принято оценивать следующими параметрами [3]: пространственное разрешение, коэффициент сигнал/шум (SNR), коэффициент контраст/шум, уровень артефактов (искажения изображения, AP). Более подробно расчет данных параметров описан в [15].

Погрешность использования метода оценивалась следующим образом:

$$\delta = \frac{\|A - A_{эм}\|_2}{\|A_{эм}\|_2} \quad (18)$$

где A – матрица интенсивностей реконструированного изображения; $A_{эм}$ – матрица интенсивностей эталонного изображения; δ – погрешность реконструкции (безразмерная величина, уменьшение которой свидетельствует о более качественной реконструкции).

При выполнении CS-реконструкции по методике (16) погрешность принималась $\varepsilon=0,01$. В качестве исходных данных были выбраны данные сканирования головного мозга с разрешением 256×256 точек, взятые с восьмиканальной антенной системы. Параметр регуляризации λ в методе сингулярного разложения принимался равным 0,05 и 0,1. Результаты изменения погрешности при различных коэффициентах акселерации приведены в табл. 2.

С целью повышения качества реконструкции задачу (12) можно переписать в виде:

$$\hat{A}^T \hat{S}(x) = \hat{A}^T \hat{A}\hat{\alpha} \quad (19)$$

и применить методики (13) и (14) уже для постановки (19).

Были получены следующие результаты.

Таблица 2
Сравнительный анализ методов реконструкции изображений МРТ

Погрешность метода, δ	R=2	R=4	R=6
Метод CS (13)	0,0255	0,0912	0,7133
Метод CS (14), $\varepsilon=0,1$	0,0255	0,0912	0,7136
SVD-разложение, $\lambda=0,05$	0,0255	0,1039	0,5615
SVD-разложение, $\lambda=0,1$	0,0255	0,2145	0,5671
Метод псевдообращения без регуляризации	0,0255	0,0912	0,7335

Таблица 3
Сравнительный анализ методов реконструкции изображений МРТ

Погрешность метода, δ	R=2	R=4	R=6
Метод CS (13)	0,0255	0,1365	0,6836
Метод CS (14), $\varepsilon=0,1$	0,0255	0,1340	0,6826
SVD-разложение, $\lambda=0,05$	0,0255	0,1039	0,5615
SVD-разложение, $\lambda=0,1$	0,0255	0,2145	0,5671
Метод псевдообращения без регуляризации	0,0255	0,0912	0,7335

Отдельно была исследована «лассо»-постановка (17) при разных значениях величины τ . Результаты изменения погрешности при коэффициенте акселерации R=2 приведены в табл. 4.

Таблица 4
Исследование «лассо»-постановки

Величина τ	32 π	64 π	128 π	250 π
Погрешность метода, δ	0,6813	0,3068	0,0255	0,0255

Результаты расчета реконструкции по модели (19) с применением выражения (17) при $\tau = 250\pi$ при разных

коэффициентах акселерации приведены в табл. 5.

Таблица 5
Исследование «лассо»-постановки

Погрешность метода, δ	R=2	R=4	R=6
Метод CS, «лассо»-модель (15)	0,0255	0,1028	0,5523

Изображения после выполнения параллельной реконструкции представлены в табл. 6.

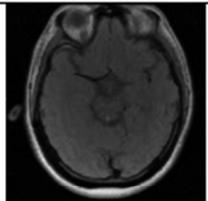
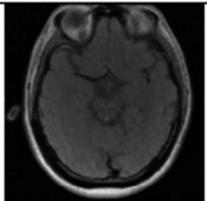
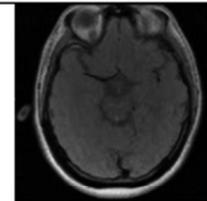
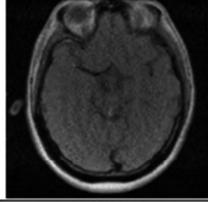
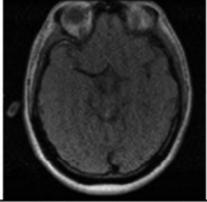
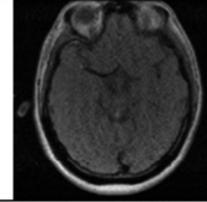
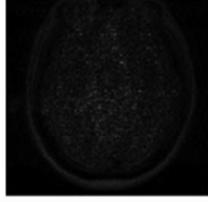
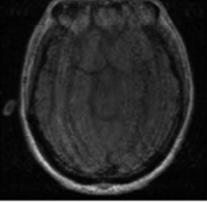
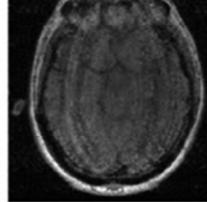
Заключение

Исследование методов параллельной реконструкции изображений МРТ показало эффективность методов SPACE-RIP и GRAPPA. Однако, при некоторых исходных данных базовые методы приводят к сильным погрешностям при реконструкции. Минимизировать погрешности при реконструкции позволяют методы регуляризации, одним из которых является метод сжатых измерений.

Из предложенных постановок метода сжатых измерений наиболее гибкими показали себя модели (13) и (15), причем точность последней определяется величиной τ , которую можно интерпретировать как радиус шара в пространстве l_1 , в котором решается задача метода наименьших квадратов. Применение методики «лассо» (17) к модели (19) в случае больших значений коэффициента акселерации позволяет сократить погрешность реконструкции изображений на 30-40% по сравнению с предыдущими оценками из табл. 2 и 4.

Из табл. 6 видно, что при коэффициенте акселерации 6 в регуляризованных методах параллельной реконструкции получаются более устойчивые изображения, чем при стандартной реконструкции.

Таблица 6
Сравнение результатов реконструкции МРТ изображений

	SPACE-RIP	SPACE-RIP+SVD	SPACE-RIP+CS
R=2			
	$\delta=0.0414$	$\delta=0.04137$	$\delta=0.04137$
R=4			
	$\delta=0.1587$	$\delta=0.15806$	$\delta=0.1468$
R=6			
	$\delta=1.063$	$\delta=0.34795$	$\delta=0.3325$



Дальнейшее улучшение качества МРТ-изображений возможно путем использования более эффективных импульсных последовательностей, модифицированных траекторий заполнения К-пространства[3] и методов фильтрации.

Усовершенствование алгоритмов параллельной реконструкции изображений позволяет повысить устойчивость и снизить искажения изображений при высоких степенях акселерации МРТ-сканирования (где R сопоставима с квадратным корнем из числа каналов антенных систем МРТ). Это позволяет расширить область применения МРТ-систем во врачебной практике для задач визуализации динамических органов (легких, сердца, сосудов).

Литература

1. Граничин, О.Н. Рандомизация измерений и L_1 -оптимизация / О.Н. Граничин // Стохастическая оптимизация в информатике. 2009. № 5. С. 3-23.
2. Кашин, Б.С. Замечание о задаче сжатого измерения / Б.С. Кашин, В.Н. Темляков // Математические заметки. – 2007. — Том 82. — Выпуск 6. — С.829-837.
3. A Statistical Approach to SENSE Regularization With Arbitrary k-Space Trajectories / L. Ying, Bo Liu, M.C. Steckner, G. Wu // Magnetic Resonance in Medicine. – 2008. – Volume 60. – Issue 2. – P. 414-421.
4. Combined compressed sensing and parallel MRI compared for uniform and random cartesian undersampling of k-Space [Электр. ресурс] / D.S. Weller, J.R. Polimeni, L. Grady, L.L. Wald. – Режим доступа: http://www.rle.mit.edu/stir/documents/wellerpgwag_icassp2011.pdf.
5. Fast Regularized Parallel Imaging in an Mr Image-Guided Therapy Application / W. S. Hoge, R. Chu, F. Jolesz, E. Samset // In Proc of 41st Asilomar Conf on Signals, Systems, and Computers. – 2007. – Nov 4-7. – P. 1869-1873.
6. Jim, X. Ji. PULSAR: A MATLAB Toolbox for Parallel Magnetic Resonance Imaging Using Array Coils and Multiple Channel Receivers / X. Ji Jim, Jong Bum Son, Swati D. Rane // Magnetic Resonance Engineering. – 2007. – V.31B (1). – P. 24-36.
7. Lin, Fa-Hsuan. Regularized Grappa Reconstruction / Fa-Hsuan Lin // United States Patent Lin. Patent No: US 7,394,252 B1. – 2008. – Date of Patent: Jul. 1. – 21 p.
8. Lustig, M. Sparse MRI: The application of compressed sensing for rapid MR imaging / M. Lustig, D. Donoho, J.M. Pauly // Magnetic Resonance in Medicine. – 2007. – Vol. 58. – P. 1182-1195.
9. Narayanan, Ashok V. The fractional Fourier transform: theory, implementation and error analysis / V. Ashok Narayanan, K.M.M. Prabhu // Microprocessors and Microsystems. – 2003. – N 27. – P. 511-521.
10. On the Regularization of Sense and Space-Rip in Parallel mr Imaging / W. Scott Hoge, Dana H. Brooks, Bruno Madore, Walid Kyriakos // IEEE International Symposium on Biomedical Imaging. – 2004. – 4 p.
11. Petr, Jan. Parallel magnetic resonance imaging reconstruction. PhD Thesis Proposal [Электр. ресурс] / Jan Petr. – Режим доступа: <ftp://cmp.felk.cvut.cz/pub/cmp/articles/petr/Petr-TR-2005-03.pdf>.
12. Regularized Sensitivity Encoding (SENSE) Reconstruction Using Bregman Iterations / Bo Liu, Kevin King, Michael Steckner, Jun Xie // Magnetic Resonance in Medicine. – 2009. – Volume 61. – Issue 1. – P. 145-152.
13. Rust, B. W. Residual periodograms for choosing regularization parameters for ill-posed problems / Bert W Rust, Dianne P O'Leary // Inverse Problems. – 2008. – N 24. – 30 p.
14. Saxena, R. Fractional Fourier transform: A novel tool for signal processing / R. Saxena, K. Singh // Indian Inst. Sci. – 2005. – Jan.-Feb. – 85. – P. 11-26.
15. Schoenberg, S.O. Parallel imaging in clinical MR applications / S. O. Schoenberg, O. Dietrich, M.F. Reiser // Medical Radiology. Diagnostic Imaging and Radiation Oncology. – Springer, 2006. – 564 p.
16. Sensitivity profiles from an array of coils for encoding and reconstruction in parallel (SPACE RIP) / W. Kyriakos, L. Panych, D. Kacher, C. Westin // Magnetic Resonance in Medicine. – 2000. – Vol. 44. – P. 301-308.
17. Smash, Sense, Pils, Grappa: how to choose the optimal method / M. Blaimer, F. Breuer, M. Mueller, R.M. Heidemann // Top Magnetic Resonance Imaging. – 2004. – Vol. 15. – N 4. – P. 223-236.
18. SGPL1: A solver for large-scale sparse reconstruction. – Режим доступа: <http://www.cs.ubc.ca/labs/scl/spgl1/>
19. Trzasko, J. Highly undersampled magnetic resonance image reconstruction via homotopic L0-minimization / J. Trzasko, A. Manduca // IEEE Trans Med Imaging. – 2009. – N 28. – P. 106-121.
20. Weller, D. S. Accelerating Magnetic Resonance Imaging by Unifying Sparse Models and Multiple Receivers / Daniel S. Weller // Massachusetts Institute of Technology. – 2012. – 148 p.

COMPARATIVE ANALYSIS OF PARALLEL IMAGE RECONSTRUCTION METHODS IN MAGNETIC RESONANCE IMAGING

Minakov E.I., Seregin P.S.

The article presents a comparative analysis of methods for parallel image reconstruction in MRI-based modeling. Here is considering the possibility of regularization of parallel image reconstruction of magnetic resonance imaging by compressed sensing method. Special emphasis is placed on the SPACE-RIP regularization method. In addition, in this paper was performed a comparative analysis with other productions of the regularization techniques of parallel reconstruction, in particular, with the regularization method, based on the singular value decomposition SVD.

УДК 004.932

ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ ПРИМЕНЕНИЯ АЛГОРИТМОВ АНАЛИЗА ИЗОБРАЖЕНИЙ В ЗАДАЧЕ НАВИГАЦИИ БЕСПИЛОТНЫХ ЛЕТАТЕЛЬНЫХ АППАРАТОВ

Алпатов Б.А., д.т.н., проф. заведующий кафедрой Автоматики и информационных технологий в управлении (АИТУ) Рязанского государственного радиотехнического университета (РГРТУ)

Муравьев В.С., к.т.н., старший научный сотрудник кафедры АИТУ РГРТУ, e-mail: aitu@rsreu.ru

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

Фельдман А.Б., младший научный сотрудник, аспирант кафедры АИТУ РГРТУ, e-mail: aitu@rsreu.ru

Ключевые слова: беспилотный летательный аппарат, навигация, анализ изображений, опорные участки, алгоритмы SIFT и ORB.

Введение

Аппаратура навигации играет ключевую роль в реализации автоматических и автоматизированных режимов полета беспилотных летательных аппаратов (БПЛА). В современном поколении навигационного оборудования, как правило, используют схему комплексирования измерений спутниковой и инерциальной навигационных систем. Это позволяет достичь высокой точности позиционирования при хороших массогабаритных показателях. Между тем, широко применяемые на практике спутниковые системы навигации не лишены недостатков, главным из которых является существенное ухудшение точности позиционирования в условиях плохого приема навигационных сигналов. Инерциальные системы, в свою очередь, быстро накапливают ошибки без введения спутниковой коррекции.

Альтернативным источником навигационных данных может служить видеосъемка, выполняемая с борта БПЛА. Вопросы, связанные с теоретическими и практическими аспектами использования видеоданных для навигации БПЛА, достаточно широко отражены в современной научной печати [1–11]. Применяемые методы можно условно разделить на две группы. Методы первой группы предполагают сопоставление данных текущей съемки с эталонными спутниковыми снимками или аэрофотоснимками. Методы второй группы основываются на принципах счисления пути, для чего выполняется сопоставление следующих подряд кадров текущей видеосъемки. Таким образом, в обоих случаях для определения навигационных параметров необходимо решать задачу сопоставления изображений. При этом важно учитывать возможные изменения ракурса съемки и наличие мешающих факторов.

Высокой точности и надежности сопоставления можно добиться, если вместо исходных изображений рассматривать совокупность выделенных на них особых точек, каждой из которых по определенному правилу ставится в соответствие дескриптор – описатель, мало зависящий от воздействия искажений. Подобный описатель вычисляется на основе распределения яркости

Рассмотрены результаты исследований, посвященных изучению возможностей применения алгоритмов анализа изображений SIFT и ORB в задаче навигации беспилотных летательных аппаратов. Исследования выполнены с помощью специализированного программного обеспечения, осуществляющего моделирование условий бортовой видеосъемки, оценку навигационных параметров, расчет критериев эффективности алгоритмов анализа изображений.

изображения в окрестности рассматриваемой точки. Сопоставив особые точки текущего изображения с точками эталонного изображения, можно определить параметры, необходимые для их геометрического совмещения.

Для выделения и описания особых точек изображений на данный момент предложено немало алгоритмов. Они различаются как по показателям эффективности, так и по быстродействию. Надо заметить, что эти качества вступают в противоречие, поэтому выбор подходящего алгоритма для реализации в аппаратуре реального времени является актуальным предметом для исследования. В связи с этим в данной работе выполнена оценка эффективности применения алгоритмов SIFT [12] и ORB [13] в задаче навигации БПЛА.

Краткое описание метода определения географических координат и курса БПЛА

Для определения навигационных параметров БПЛА в данной работе использовался подход, согласно которому на предварительно сформированных эталонных изображениях (ЭИ) местности размечаются опорные участки (ОУ) – характерные объекты и регионы наблюдаемой местности, которые впоследствии отыскиваются на текущем изображении (ТИ). Каждый ОУ задается в виде ориентированного прямоугольника на эталонном снимке местности с известными географическими координатами точек и характеристиками пространственного разрешения. Таким образом, после разметки ОУ заданы географические координаты его центра, размеры и ориентация.

Пусть в процессе определения координат БПЛА не совершает резких маневров, оптическая ось видеодатчика отклоняется от надира незначительно (на величину не более 5°), а перепад высот рельефа в поле зрения пренебрежимо мал в сравнении с высотой полета. Исходя из этого, без существенной потери точности можно

считать, что геометрические преобразования изображений ОУ ограничиваются сдвигом, поворотом относительно центра и изменением масштаба. Задача распознавания и локализации ОУ на ТИ является наиболее трудной в реализации описываемого подхода. Именно для ее решения применяются алгоритмы SIFT и ORB, являющиеся предметом исследования. При этом сохраняется общая для данного подхода логика обработки текущих и эталонных данных.

Перед началом полета на всех изображениях ОУ производится выделение особых точек с последующим расчетом дескрипторов. Это позволяет сформировать эталонную базу особых точек (БОТ), связанных с опорными участками. В ходе полета аналогичные операции выполняются с ТИ. Следующим шагом является сопоставление особых точек ТИ и точек из БОТ.

Для каждого ОУ число сопоставленных пар точек сравнивается с пороговым значением. Это позволяет принять решение о присутствии рассматриваемого ОУ на ТИ. Кроме того, соответствие между точками позволяет оценить положение, размеры и ориентацию изображения ОУ в текущем кадре. Для снижения влияния ошибочных сопоставлений особых точек используется метод робастного оценивания RANSAC [14]. С целью исключения аномальных ошибок локализации и для уточнения положения опорных участков используются методы калмановской фильтрации.

Последним шагом является расчет координат и курса БПЛА, для чего данные о положении опорных участков на ТИ соотносятся с их фактическим местоположением в системе географических координат.

Алгоритмы анализа изображений

Дадим краткую характеристику алгоритмов SIFT и ORB, исследованию эффективности которых посвящена настоящая работа.

Алгоритм SIFT. Для детектирования особых точек в алгоритме SIFT [12] строится масштабная гауссова пирамида [15] и вычисляется разность между изображениями смежных уровней. Особые точки выбираются среди экстремумов разностных изображений. Положение экстремума по пространственным координатам и масштабу определяется с субдискретной точностью посредством квадратичной интерполяции.

Каждый экстремум подвергается ряду проверок. Во-первых, выполняется сравнение величины отклика в рассматриваемой точке с порогом, что позволяет снизить чувствительность детектора к яркостным искажениям и действию шумов. Во-вторых, отбрасываются экстремумы, лежащие на границах объектов. Такие экстремумы могут смещаться вдоль границы, следовательно, также являются нестабильными. Алгоритм демонстрирует высокую стабильность результатов детектирования при наличии аффинных геометрических преобразований и яркостных искажений.

Для вычисления дескриптора особой точки с ней связывают определенное направление. Для этого в окрестности рассматриваемой точки на соответствующем масштабе пирамиды строится специальная гистограмма. Каждый её отсчет является суммой модулей градиентов, направление которых попадает в заданный ин-

тервал. Отыскивается направление, соответствующее максимуму гистограммы. Точное положение максимума определяется при помощи квадратичной интерполяции по ближайшим столбцам. Эту величину принимают за направление, связанное с данной особой точкой.

Далее формируется квадратное окно, центр которого располагается в рассматриваемой особой точке, а ориентация совпадает с сопоставленным ей направлением. Окно равномерно разбивается на регионы, в каждом из которых вычисляется гистограмма, аналогичная той, что использовалась для определения направления в особой точке, но с меньшей разрядностью. Для формирования дескриптора задаются правилом обхода регионов и последовательно выписывают отсчеты соответствующих гистограмм. Таким образом, привязка дескриптора к определенному уровню гауссовой пирамиды обеспечивает инвариантность к изменению масштаба, а использование направления, связанного с особой точкой, – к повороту изображения. Использование информации о градиенте для кодирования дескриптора обеспечивает низкую чувствительность к изменению уровня яркости. В последующем для сопоставления особых точек, вычисляется Евклидова норма расстояния между их дескрипторами.

Алгоритм ORB. Алгоритм SIFT демонстрирует отличные результаты в задачах обнаружения, распознавания и локализации объектов на изображении, но как видно из представленного описания требует значительных по объемам вычислений. В связи с этим, продолжают попытки создания более простых алгоритмов детектирования точек и вычисления дескрипторов, обеспечивающих при этом достаточную инвариантность к искажениям. Одним из них является рассматриваемый в данной работе алгоритм ORB [13].

На первом этапе для детектирования особых точек строится масштабная гауссова пирамида изображений. Затем на каждом масштабном уровне определяются экстремумы функции яркости. Для решения этой задачи применяется алгоритм FAST [16], согласно которому для каждой точки изображения формируется окружность некоторого радиуса и подсчитывается число точек, лежащих на этой окружности и имеющих значения яркости меньше или больше яркости ее центра. Если таких точек найдется больше $3/4$ от их возможного количества, то центр окружности считается кандидатом на роль особой точки. Найденные кандидаты отбраковываются, если они лежат на границах объектов.

Для расчета дескриптора в окрестности особой точки на текущем масштабе выделяется область, для которой находится центр масс. Вектор, направленный из особой точки в сторону центра масс, будет задавать ориентацию особой точки. Для составления дескриптора необходимо сформировать квадратное окно, центрированное относительно ключевой точки и согласованное с ее ориентацией. В этом окне по заданному правилу выбирается набор пар точек, значения яркости в которых сравниваются между собой. Если яркость первой точки оказывается выше, в соответствующий элемент дескриптора записывается 1, в противном случае записывается значение 0. Составленные таким образом дескрипторы можно сопоставлять друг с другом по норме Хемминга.

Структура алгоритма ORB показывает, что он менее требователен к вычислительным ресурсам в сравнении с описанным ранее алгоритмом SIFT. Выигрыш в скорости вычислений определяется, прежде всего, более простой процедурой построения дескрипторов и механизмом вычисления нормы.

Программный комплекс для автоматизации экспериментальных исследований

Для автоматизации проведения исследований алгоритмов навигации БПЛА по данным видеонаблюдения авторами статьи был разработан программный комплекс (ПК) «Навигация» [17]. ПК «Навигация» обеспечивает подготовку исходных данных, имитацию бортовой видеосъемки, реализует алгоритмы определения навигационных параметров и позволяет оценивать их эффективность. Для имитации видеосъемки используются средства программного авиационного симулятора с открытым исходным кодом FlightGear. К проекту FlightGear добавлены файлы, которые реализует специализированный сетевой сервис. С его помощью из внешнего приложения можно менять параметры моделирования, устанавливать позицию и свойства виртуальной видеокамеры, считывать видеоизображения.

Использование FlightGear имеет определенные преимущества по сравнению с традиционными средствами 3D моделирования. Во-первых, не требуется специализированная проработка ландшафтов. Во-вторых, имитация условий наблюдения (освещенности, погоды, облаков) осуществляется средствами авиасимулятора. В-третьих, облегчается переход от моделирования видеосъемки к решению задач управления БПЛА. Для FlightGear в открытом доступе представлены различные модели летательных аппаратов, базы рельефа, фото-реалистичные текстуры.

ПК «Навигация» также позволяет использовать натурные снимки и видеосюжеты. Структура ПК «Навигация» показана на рис. 1. Прямоугольники, выполненные сплошной линией, соответствуют модулям с графическим пользовательским интерфейсом. Основные потоки данных показаны сплошными стрелками, дополнительные – пунктирными. Информация об эталонных снимках, маршрутах полета, опорных участках, параметрах моделирования и результатах полетов хранится в интегрированной базе данных.



Рис. 1. Структура ПК «Навигация»

Модуль для управления снимками, позволяет сформировать снимок с заданными параметрами с помощью симулятора или загрузить в интегрированную БД натурный снимок. В модуле для управления маршрутами

реализован наглядный механизм ввода маршрутов. Маршрут определяется как последовательность ключевых точек, для которых указываются географические координаты, высота и время прохождения. Промежуточные точки маршрута рассчитываются посредством сплайн-интерполяции. Опорные участки связаны с конкретным снимком. В ПК «Навигация» реализован инструментальный для их разметки и параметризации.

При выполнении полета можно настроить параметры моделирования (выбрать сезон года, время суток, погоду и пр.), записать исходное видео полета по маршруту или выполнить полет с оценкой навигационных параметров. Функции для определения координат и курса БПЛА реализованы в виде специальной программной библиотеки. Для реализации алгоритмов SIFT и ORB использованы средства библиотеки с открытым исходным кодом OpenCV.

Результаты полета фиксируются в БД, и в дальнейшем могут быть проанализированы в модуле статистической обработки результатов. Он позволяет оценить эффективность применения алгоритмов анализа изображений для одного или нескольких полетов из заданного набора. Имеется возможность отображения истинной траектории и результатов ее оценки, а также средства для построения графиков ошибок определения навигационных параметров.

Описание тестовой коллекции видеосюжетов

Для проведения исследований с помощью ПК «Навигация» был сформирован набор из 12 видеосюжетов длительностью 200-250 кадров. Были подготовлены эталонные снимки, маршруты полета и опорные участки местности. Высота полета БПЛА варьировалась от 600 до 1500 м. Поле зрения видеокамеры составляло 55x42. Исходные снимки были получены с высоты в 1500 и 2000 метров. Каждому эталонному изображению местности был сопоставлен один маршрут. Общее количество опорных участков для выбранных маршрутов лежало в диапазоне от 10 до 19, в каждом кадре наблюдалось не менее трех ОУ. Ориентация участков изменялась от 0 до 180 угловых градусов.

Примеры кадров из тестового набора видеосюжетов с результатами локализации ОУ показаны на рис. 2.

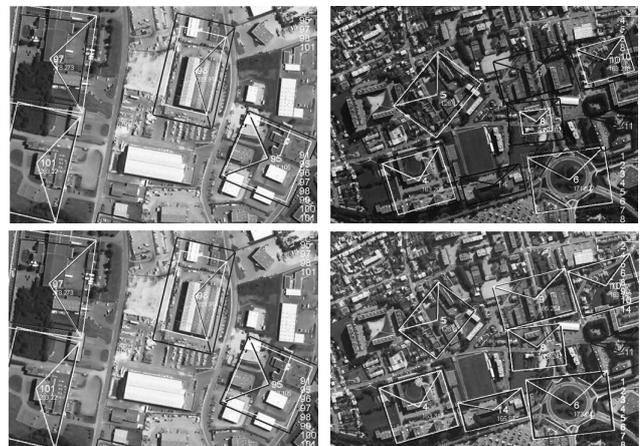


Рис. 2. Кадры из нескольких видеосюжетов с результатами локализации ОУ при использовании алгоритмов SIFT(вверху) и ORB(внизу)



На приведенном рисунке белым цветом показаны найденные опорные участки, черным – эталонные. Номер опорного участка в эталонной базе отображается рядом с его центром.

Критерии эффективности алгоритмов

Для оценки эффективности применения алгоритмов анализа изображений для навигации БПЛА в ходе экспериментов вычислялись следующие критерии.

Критерии надежности локализации ОУ: 1) частота (вероятность) правильного обнаружения ОУ – P_D ; 2) частота (вероятность) ложных тревог – P_{FA} .

Критерии точности локализации ОУ: 1) СКО ошибок локализации ОУ по осям координат – σ_x, σ_y ; 2) относительное время слежения за ОУ t_{tr} , рассчитываемое как отношение времени слежения за ОУ к полному времени его наблюдения в кадре; 3) нормированное СКО ошибки локализации ОУ $\delta = \sqrt{\sigma_x^2 + \sigma_y^2} / \bar{D}$,

где \bar{D} – средний размер диагонали опорного участка.

Критерии точности локализации БПЛА: 1) СКО определения широты (в метрах) – σ_{Lat} ; 2) СКО определения долготы (в метрах) – σ_{Lon} ; 3) СКО ошибки позиционирования БПЛА $\sigma_G = \sqrt{\sigma_{Lat}^2 + \sigma_{Lon}^2}$; 4) СКО ошибки определения истинного курса – σ_ψ .

Дополнительно было сопоставлено время обработки видеосюжетов каждым из алгоритмов.

Результаты экспериментов

Для заданного набора видеосюжетов было выполнено две серии экспериментов. В первой использовались детектор и дескриптор алгоритма SIFT, во второй – алгоритм ORB. Количественные показатели алгоритмов SIFT и ORB сведены в табл. 1 и 2 соответственно. Усредненные на наборе сюжетов показатели эффективности показаны в табл. 3.

Таблица 1.
Результаты экспериментов для алгоритма SIFT

№ эксп.	$P_D, \%$	$P_{FA}, \%$	σ_x , пикс.	σ_y , пикс.	$t_{tr}, \%$	$\delta, \%$	σ_{Lat} , м	σ_{Lon} , м	σ_G , м	$\sigma_\psi, ^\circ$
1	91,1	2,4	2,5	1,8	90,8	1,57	1,97	2,77	3,37	0,29
2	98,6	1,2	1,2	1,9	98,5	1,83	2,08	2,6	3,33	0,15
3	99,3	1,4	1,0	1,9	99,3	1,24	1,08	2,43	2,66	0,04
4	99,3	0,9	1,5	4,6	99,3	2,81	2,86	4,69	5,49	0,17
5	96,6	0,8	1,8	2,1	96,5	1,99	0,89	3,85	3,96	0,06
6	97,1	3,3	2,2	4,5	97,0	3,28	2,26	4,51	5,05	0,18
7	99,9	1,8	1,4	1,9	99,8	1,47	1,71	3,86	4,22	0,22
8	99,6	6,1	1,7	2,6	99,5	1,56	2,04	4,36	4,81	0,15
9	96,6	0,3	1,9	1,3	96,5	1,51	1,39	1,74	2,22	0,13
10	93,9	4,5	1,2	3,4	93,8	2,00	1,1	3,75	3,91	0,27
11	98,2	2,4	0,7	1,7	98,2	1,08	1,69	1,44	2,22	0,11
12	99,6	4,7	1,2	1,8	99,2	0,99	0,76	1,53	1,71	0,1

Таблица 2.
Результаты экспериментов алгоритма ORB

№ эксп.	$P_D, \%$	$P_{FA}, \%$	σ_x , пикс.	σ_y , пикс.	$t_{tr}, \%$	$\delta, \%$	σ_{Lat} , м	σ_{Lon} , м	σ_G , м	$\sigma_\psi, ^\circ$
1	70,9	0,8	5,5	3,8	65,7	3,40	2,11	2,49	3,27	0,35
2	67,4	0,2	1,2	2	59,1	2,00	2,26	2,34	3,30	0,31
3	97,9	1,1	1,3	2,2	96,7	1,40	1,11	2,79	3,00	0,1
4	98,3	4,3	2,9	5,8	97,2	3,87	3,98	5,38	6,69	0,35
5	87,3	1,1	2	2,6	79,9	2,35	1,02	4,06	4,18	0,17
6	88,0	1,1	2,5	5	86,4	3,67	2,16	4,73	5,20	0,25
7	95,2	0,2	1,7	2,2	91,9	1,65	2,25	4,32	4,87	0,26
8	92,8	7,4	2,3	3,4	90,8	2,06	3,02	5,11	5,93	0,25
9	94,1	0,3	1,9	1,4	90,6	1,57	1,43	1,52	2,08	0,26
10	84,6	2,3	2	3,9	84,3	2,37	1,26	3,92	4,12	0,3
11	89,4	0,3	0,9	1,9	86,8	1,19	1,74	1,38	2,22	0,21
12	97,0	6,4	1,6	2,2	95,9	1,28	0,87	1,78	1,97	0,13

Таблица 3.
Усредненные значения критериев эффективности для двух серий экспериментов

№ эксп.	Алгоритм	$P_D, \%$	$P_{FA}, \%$	σ_x , пикс.	σ_y , пикс.	$t_{tr}, \%$	$\delta, \%$	σ_{Lat} , м	σ_{Lon} , м	σ_G , м	$\sigma_\psi, ^\circ$
1	SIFT	97,48	2,48	1,53	2,46	97,37	1,78	1,65	3,13	3,58	0,16
2	ORB	88,58	2,13	2,15	3,03	85,44	2,23	1,93	3,31	3,89	0,25

Анализ результатов показывает, что оба подхода обеспечивают высокие характеристики обнаружения опорных участков и обеспечивают погрешность их локализации не выше нескольких пикселей. Однако алгоритм SIFT обладает несколько лучшими характеристиками обнаружения. Так, усредненная по выборке частота правильного обнаружения составляет 97,5% у алгоритма SIFT, против 88,6% у алгоритма ORB при весьма близких значениях частоты ложных тревог. В процессе настройки алгоритма ORB было замечено, что ряд параметров крайне чувствителен к размерам опорных участков и в идеале должен быть соотнесен с ними для достижения приемлемого качества обнаружения и локализации этих участков на наблюдаемом изображении. Подход на основе SIFT при фиксированных параметрах алгоритма является более робастным. Отклонение параметров не оказывало существенного влияния на итоговый результат.

В процессе исследований было отмечено, что протяженные прямоугольные участки, лежащие на границах кадра, обнаруживаются несколько хуже, чем квадратные, что может давать вклад в повышение частоты ложных тревог. Численное отличие в значениях частоты правильного обнаружения от относительного времени слежения за опорным участком может объясняться колебаниями их измеренных размеров. В виду этого, может приниматься решение о переключении алгоритма прослеживания в режим прогноза для данного опорного участка. Если по прошествии определенного времени ошибка измерения размеров не уменьшится, то рассматриваемый участок может быть снят со слежения. Очевидно, что при наличии маневров летательного аппарата точность прогнозирования будет снижаться.

На ряде сюжетов наблюдалось заметное снижение качества прослеживания опорных участков для метода

ORB. Данное обстоятельство обусловлено присутствием в кадре опорных участков разного размера и нестабильностью в определении их масштаба и ориентации. Немаловажен тот факт, что при использовании метода ORB намного чаще фиксировались ошибки в определении ориентации опорных участков в 180 градусов. На рис. 3 показаны графики ошибок позиционирования БПЛА при использовании двух сравниваемых алгоритмов.

Применительно к бортовым системам обработки информации важным элементом оценки эффективности применения алгоритмов является достигаемая при этом производительность. Оценить данную характеристику можно сравнивая среднее время обработки кадра. Измерение производительности выполнено на базе процессора Core 2 Duo E8500 с частотой 3,16 ГГц и 4 Гб оперативной памяти. Алгоритм ORB, в среднем оказывается в 2,5-3 раза быстрее SIFT, но проигрывает последнему в надежности локализации опорных участков.

Заключение

По результатам исследования можно сделать следующие выводы:

- алгоритм SIFT в целом превосходит сравниваемый подход по качеству обнаружения опорных участков на 9-10% и точности измерения географических координат и курса БПЛА на 25%;
- алгоритм SIFT уступает по быстродействию алгоритму ORB в несколько раз, обеспечивая при этом лучшую помехозащищенность и надежность работы;
- частота правильного обнаружения опорных участков составляет не менее 95% при частоте ложных тревог не более 2,5%;
- нормированное СКО ошибки локализации опорного участка не превышает 3,5%;

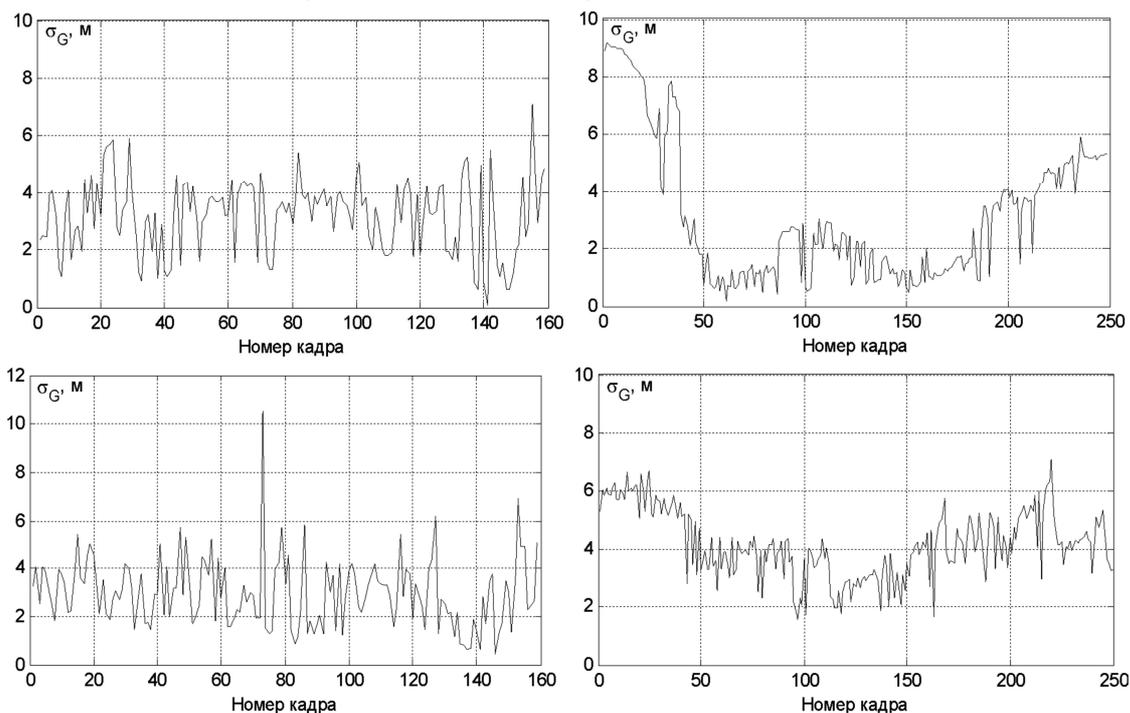


Рис. 3. Ошибка позиционирования БПЛА при использовании алгоритмов SIFT(вверху) и ORB (внизу)



– совокупная ошибка позиционирования БПЛА, характеризующая точность определения географических координат, не превышает 10 метров;

– обеспечиваемая точность определения истинного курса БПЛА не ниже 20 угловых минут, что при отсутствии накопления ошибки и частоте поступления измерений не реже нескольких раз в секунду не приведет к заметному отклонению летательного аппарата от заданной траектории.

Результаты проведенных исследований показывают, что оба сравниваемых алгоритма могут применяться в задаче навигации БПЛА, однако при использовании алгоритма SIFT удается добиться лучших показателей обнаружения опорных участков. Это означает, что подход на базе SIFT менее требователен к характеристикам ОУ и может применяться в более сложных условиях.

Исследования проводились при финансовой поддержке Министерства образования и науки РФ (госконтракт 07.514.11.4034).

Литература

1. Tightly-Coupled INS, GPS, and Imaging Sensors for Precision Geolocation / M. Veth, R. Anderson, F. Webber, M. Nielsen // Proceedings of the 2008 National Technical Meeting of The Institute of Navigation. – San Diego, 2008. – P. 487 – 496
2. Flight Test Results of a Video-Aided GPS/Inertial Navigation System / A. Brown, B. Bockius et al. // Proceedings of ION GNSS. – Ft. Worth, 2007. – P. 1111 – 1117
3. Conte G., Doherty P. Vision-Based Unmanned Aerial Vehicle Navigation Using Geo-Referenced Information // EURASIP Journal on Advances in Signal Processing. – 2009. Volume 2009. – P.10-1 – 10-18.
4. Wu A.D., Johnson E.N., Proctor A.A. Vision-Aided Inertial Navigation for Flight Control / Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit. – San Francisco, 2005. – P.1-13
5. Aerial SLAM with a Single Camera Using Visual Expectation / M.J. Milford, F. Schill et al. // 2011 IEEE International Conference on Robotics and Automation. – Shanghai, 2011. – P. 2506–2512
6. A Vision-Based Guidance System for UAV Navigation and Safe Landing using Natural Landmarks / A. Cesetti, E. Frontoni et al. // Journal of Intelligent Robotic System. – 2010. – № 57. – P. 233-257
7. Zhang J., Liu W., Wu. Y. Novel Technique for Vision-Based UAV Navigation // Proceedings of IEEE Transactions on Aerospace and Electronic System. – 2011. – Vol.47. № 4. – P. 2731–2741
8. Kendoul F., Fantoni I., Nonami K. Optic Flow-Based Vision System for Autonomous 3D Localization and Control of Small Aerial Vehicles // Robotics and Autonomous Sys-

tems. – 2009. – №57. –P. 591-602

9. Vision-Based Odometry and SLAM for Medium and High Altitude Flying UAVs / F. Caballero, L. Merino, J. Ferruz, A. Ollero // Journal of Intelligent Robotics System. – 2009. – Vol. 54., Issue 1-3 – P. 137 – 161

10. The Navigation and Control technology inside the AR.Drone micro UAV / P.-J. Bristean, F.Callou // Preprints of the 18th IFAC World Congress. – Milano, 2011. – P. 1477-1484.

11. Практическое применение технологий машинного зрения в решении задач распознавания, восстановления 3D, шивки карт, точного целеуказания, числения пути и навигации / А.Е. Семенов, Е.В. Крюков, Д.П. Рыкованов, Д.А. Семенов // Известия ЮФУ. Технические науки. – 2010. – №2. – С. 92-102

12. Lowe D. G. Distinctive Image Features from Scale-Invariant Keypoints // International Journal of Computer Vision. – 2004.– Vol. 60(2). – PP. 91-110

13. ORB: An efficient alternative to SIFT or SURF / Rublee E., Rabaud V., Konolige K., Bradski G.R. // Proceedings of ICCV. – 2011. – pp. 2564-2571

14. Fischler M.A., Bolles R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography // Comm. of the ACM – 2006 – №24 (6). – pp. 381–395.

15. Форсайт Д., Понс Ж. Компьютерное зрение. Современный подход. –М.: Вильямс, 2004. – 928 с.

16. Rosten E., Drummond T. Machine learning for high-speed corner detection // European Conference on Computer Vision, vol. 3951. – 2006. – PP. 430-443.

17. Алпатов Б.А., Бабаян П.В., Коблов Ю.С., Муравьев В.С., Стротов В.В., Фельдман А.Б. Автоматизация разработки и исследования алгоритмов машинного зрения для навигации беспилотных летательных аппаратов на базе специализированного программного комплекса // Известия ЮФУ. Технические науки. – 2012. – №3. – С.85-91.

RESEARCH OF EFFECTIVENESS OF IMAGE ANALYSIS ALGORITHMS IN THE PROBLEM OF UNMANNED AERIAL VEHICLES NAVIGATION

Alpatov B.A., Muraviev V.S., Strotov V.V., Feldman A.B.

In the article the results of studies on the possible application of image analysis algorithms SIFT and ORB in the problem of unmanned aerial vehicles navigation are described. The research has been carried out using specialized software performing modeling of onboard video receiving conditions, navigation parameters estimation, the calculation of the performance criteria of image analysis algorithms.

УДК 004.932

ПОВЫШЕНИЕ ЧЕТКОСТИ ОТОБРАЖЕНИЯ ОБЪЕКТОВ НА ДАННЫХ ГИПЕРСПЕКТРАЛЬНОЙ СЪЕМКИ ЗЕМНОЙ ПОВЕРХНОСТИ

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

Макаренков А.А., аспирант РГРТУ

Москвитин А.Э., к.т.н., в.н.с. НИИ «Фотон» РГРТУ

Юдаков А.А., аспирант РГРТУ

Ключевые слова: гиперспектральные изображения, комплексирование, спектральная характеристика, корреляция.

Введение

В ходе эволюции средств дистанционного зондирования Земли (ДЗЗ) прослеживается процесс повышения спектрального разрешения и расширения диапазонов длин волн, в которых производится съемка. При переходе от панхроматической съемки, при которой поток излучения регистрируется в широком диапазоне спектра (от ~0.4 до ~0.9 мкм), к мультиспектральным датчикам, фиксирующим излучение в нескольких спектральных диапазонах видимого (от ~0.4 до ~0.7 мкм), ближнего инфракрасного (от ~0.7 до 1.0 мкм) и инфракрасного излучения (от ~1.0 до ~2.5 мкм), появлялись новые возможности анализа характеристик объектов, для чего требуется создание принципиально новых алгоритмов обработки видеoinформации. Повышение спектрального разрешения, а вместе с ним и числа каналов изображения, создало предпосылки для решения задач выделения и классификации объектов снимка на качественно новом уровне. В последние годы активно ведутся разработки гиперспектральной аппаратуры (ГСА), позволяющей производить съемку одновременно в сотнях спектральных диапазонах. В этом случае каждая точка земной поверхности (пикселя изображения) сопровождается спектральной характеристикой (СХ), которая определяет физико-химические свойства наблюдаемых объектов. В настоящее время в практику ДЗЗ внедряются авиационные ГСА (AVIRIS, ProSpecTIR – США, НуМар – Австралия) и ГСА космического базирования (Hyperion, ARTEMIS – США, CHRIS – Евросоюз).

Наблюдаемые объекты земной поверхности обычно имеют сложную спектральную характеристику, поэтому в различных спектральных каналах гиперспектрального изображения (ГСИ) объекты могут отображаться совершенно по-разному: в одних каналах более светлыми, а в других – более темными по отношению к окружающим объектам. При этом различимость объектов в разных каналах ГСИ также не постоянна. Таким образом, отдельные каналы ГСИ не несут полной информации обо всех наблюдаемых объектах, а при традиционном подходе к объединению каналов – их усреднению – отдельные объекты существенно теряют контраст по отношению к исходным ГСИ. Это создает предпосылки к

Рассматривается задача комплексирования гиперспектральных снимков с целью формирования новых изображений, на которых с высокой четкостью отображаются все объекты. Представлены результаты обработки натурной гиперспектральной информации.

решению задачи комплексирования каналов ГСИ, т.е. формированию нового изображения, на котором с высокой четкостью представлены все наблюдаемые объекты.

Предпосылки для повышения по ГСИ четкости объектов

Рассмотрим съемку для следующего случая. Пусть имеется гиперспектральный датчик с K спектральными каналами и панхроматический датчик, снимающий в том же диапазоне длин волн. В результате съемки гиперспектральным датчиком двух точек земной поверхности будет сформировано два вектора яркостей $\mathbf{B}_1 = (B_{11}, B_{12}, \dots, B_{1K})$ и $\mathbf{B}_2 = (B_{21}, B_{22}, \dots, B_{2K})$. В результате съемки тех же точек панхроматическим датчиком будут сформированы два значения яркости

$B_1 = \frac{1}{K} \sum_{k=1}^K B_{1k}$ и $B_2 = \frac{1}{K} \sum_{k=1}^K B_{2k}$. Рассмотрим меры различимости этих точек на панхроматическом и гиперспектральном изображениях. Для панхроматического изображения в качестве меры различимости двух точек примем: $\ell = |B_1 - B_2| = \frac{1}{K} \left| \sum_{k=1}^K (B_{1k} - B_{2k}) \right| = \frac{1}{K} \left| \sum_{k=1}^K \Delta B_k \right|$. Для

гиперспектрального изображения в качестве меры различимости естественно принять расстояние между двумя точками в спектрально-пространственном пространстве, отнормировав его так, чтобы диапазон изменения совпадал с мерой

$\ell: d = \frac{1}{\sqrt{K}} \sqrt{\sum_{k=1}^K (B_{1k} - B_{2k})^2} = \frac{1}{\sqrt{K}} \sqrt{\sum_{k=1}^K \Delta B_k^2}$. Рассмотрим

отношение двух этих мер [1]:

$$\gamma = \frac{\ell}{d} = \frac{1}{\sqrt{K}} \left| \sum_{k=1}^K \Delta B_k \right| \bigg/ \sqrt{\sum_{k=1}^K \Delta B_k^2}.$$

Лишь при $x_1 = x_2 = \dots = x_s$: $\gamma = 1$, и гиперспектральная съемка не дает дополнительной информации для повышения четкости изображения. По мере взаимной компенсации приращений спектрально-пространственных компонент ($\ell \rightarrow 0$) создаются наилучшие условия для комплексирования гиперспектральных данных.

В настоящей статье рассматриваются различные

подходы к комплексированию каналов ГСИ с целью получения нового изображения, на котором с высокой четкостью отображаются все объекты, представленные в отдельных каналах гиперспектрального снимка. Алгоритмы основаны на анализе мер отличия спектральной характеристики от среднего уровня яркости, учете характера СХ, полиномиальном представлении СХ, комплексировании ГСИ на основе выделения границ. Проведен сравнительный анализ алгоритмов комплексирования с применением критерия точности передачи контуров. Приведены результаты апробации алгоритмов комплексирования на натурной гиперспектральной информации со съемочной системы ProСпекTIR-V.

Комплексирование ГСИ на основе мер отличия СХ от среднего уровня

Алгоритмы основаны на замене СХ в каждой точке ГСИ числом, которое характеризует степень отличия СХ от среднего значения яркости $\bar{B} = \frac{1}{K} \sum_{k=1}^K B_k$, соответствующего панхроматической съемке. В качестве такой меры может быть принято СКО:

$$\gamma_1 = \sqrt{\frac{1}{K} \sum_{k=1}^K (B_k - \bar{B})^2}. \quad (1)$$

В этом случае результат комплексирования B^* формируется как $B^* = \bar{B} + \gamma_1$. Мера (1) учитывает только степень отличия спектральной характеристики от среднего уровня яркости \bar{B} .

В качестве другой меры отличия спектральной характеристики от элемента панхроматического снимка можно принять спектральный угол α в k -мерном пространстве между векторами: $\mathbf{B}_{\Pi} = (\bar{B}, \bar{B}, \dots, \bar{B})$ и

$$\mathbf{B}_{\Gamma} = (B_1, B_2, \dots, B_K):$$

$$\gamma_2 = \cos(\alpha) = \frac{(\mathbf{B}_{\Pi} \cdot \mathbf{B}_{\Gamma})}{|\mathbf{B}_{\Pi}| \cdot |\mathbf{B}_{\Gamma}|} = \frac{\bar{B}}{\sqrt{\frac{1}{K} \sum_{k=1}^K B_k^2}}. \quad (2)$$

В данном случае: $B^* = \bar{B} \cdot \gamma_2$.

Меры (1) и (2) функционально связаны и отражают только отличие СХ от среднего уровня яркости точки \bar{B} . Можно показать, что $\gamma_1 = \bar{B} \sqrt{\gamma_2^2 - 1}$. Поэтому с точки зрения решения поставленной задачи меры γ_1 и γ_2 дают эквивалентные результаты. Например, две спектральные характеристики, имеющие разную форму, но имеют одно среднее значение (рис. 1), и γ_1 и γ_2 дают примерно одинаковую оценку.

Комплексирование ГСИ с учетом характера изменения СХ

Рассмотрим меру, учитывающую характер изменения СХ в зависимости от номера спектрального канала ГСИ:

$$\gamma_3 = \frac{\sum_{k=1}^K (K+1-k) B_k}{\sum_{k=1}^K k B_k}. \quad (4)$$

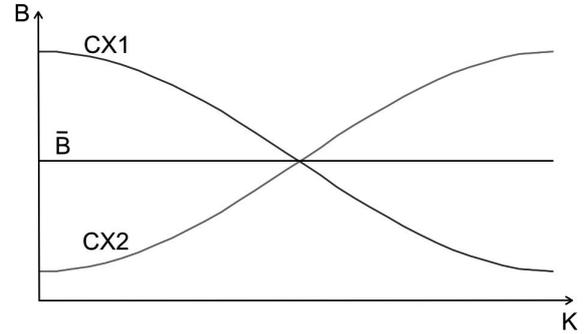


Рис. 1. Спектральные характеристики двух точек ГСИ

Мера γ_3 описывает характер изменения компонент B_1, B_2, \dots, B_K по мере изменения номера канала $k = \overline{1..K}$. Представим числитель (4) в виде:

$$\begin{aligned} \sum_{k=1}^K (K+1-k) B_k &= \\ &= (k+1) \sum_{k=1}^K B_k - \sum_{k=1}^K k B_k = (k+1) \bar{B} - \sum_{k=1}^K k B_k. \end{aligned}$$

Тогда:

$$\gamma_3 = \frac{(k+1) \bar{B}}{\sum_{k=1}^K k B_k} - 1.$$

Здесь, в отличие от γ_1 и γ_2 , сопоставляются \bar{B} и $\sum_{k=1}^K k B_k$. Последняя компонента реагирует на характер изменения спектральной характеристики по мере увеличения номера канала. Для представленных на рис.1 спектральных характеристик мера γ_3 будет иметь различную величину для СХ1 и СХ2.

Комплексирование ГСИ на основе полиномиального представления СХ

Другим подходом к анализу СХ является ее аппроксимация степенным полиномом и рассмотрение взаимного расположения полинома и среднего уровня яркости. Рассмотрим аппроксимацию СХ полиномом 3-го порядка. Здесь возможны три варианта взаимного расположения полинома и среднего уровня яркости: полином пересекает средний уровень \bar{B} в одной, двух и трех точках. В первом случае меру, описывающую характер изменения СХ, можно записать как

$$\gamma_4 = \begin{cases} -\sqrt{\frac{S_1 \cdot S_2}{n_1 \cdot n_2}}, & S_1 < 0; \\ \sqrt{\frac{S_1 \cdot S_2}{n_1 \cdot n_2}}, & S_1 > 0, \end{cases}$$

где $S_1 = \sum_{k=0}^{k_1} (B_k - \bar{B})$, k_1 – номер канала, в котором полином пересекает средний уровень яркости, B_k – яркость аппроксимированной полиномом СХ в канале k , \bar{B} – средний уровень яркости СХ исходного ГСИ, $S_2 = \sum_{k=k_1}^{K} (B_k - \bar{B})$, $n_1 = k_1$, $n_2 = K - k_1$.

В случае пересечения полинома и среднего уровня яркости в двух точках:

$$\gamma_4 = \begin{cases} -\sqrt[3]{\frac{S_1 \cdot S_2 \cdot S_3}{n_1 \cdot n_2 \cdot n_3}}, & S_1 < 0; \\ \sqrt[3]{\frac{S_1 \cdot S_2 \cdot S_3}{n_1 \cdot n_2 \cdot n_3}}, & S_1 > 0, \end{cases}$$

где $S_2 = \sum_{k=k_1}^{k_2} (B_k - \bar{B})$, k_2 – вторая точка пересечения полиномом уровня \bar{B} , $S_3 = \sum_{k=k_2}^{K} (B_k - \bar{B})$, $n_1 = k_1$,

$n_2 = k_2 - k_1$, $n_3 = K - k_2$.

При пересечения полиномом уровня \bar{B} в трех точках (рис. 2.)

$$\gamma_4 = \begin{cases} -\sqrt[4]{\frac{S_1 \cdot S_2 \cdot S_3 \cdot S_4}{n_1 \cdot n_2 \cdot n_3 \cdot n_4}}, & S_1 < 0; \\ \sqrt[4]{\frac{S_1 \cdot S_2 \cdot S_3 \cdot S_4}{n_1 \cdot n_2 \cdot n_3 \cdot n_4}}, & S_1 > 0, \end{cases}$$

где $S_1 = \sum_{k=0}^{k_1} (B_k - \bar{B})$, $S_2 = \sum_{k=k_1}^{k_2} (B_k - \bar{B})$,

$S_3 = \sum_{k=k_2}^{k_3} (B_k - \bar{B})$, $S_4 = \sum_{k=k_3}^{K} (B_k - \bar{B})$, $n_1 = k_1$,

$n_2 = k_2 - k_1$, $n_3 = k_3 - k_2$, $n_4 = K - k_3$.

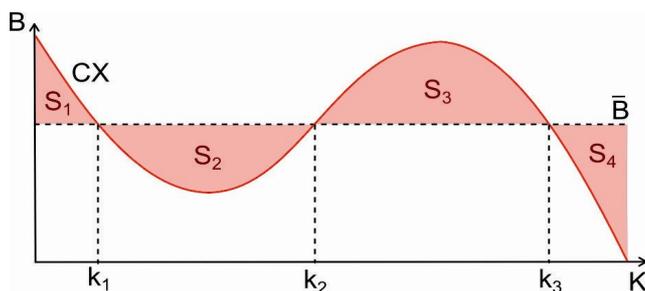


Рис. 2. Полиномиальное представление спектральной характеристики

Полиномиальное представление СХ позволяет снизить влияние шума, нелинейности передаточной характеристики датчика, атмосферных искажений. Это позволяет достичь лучшего качества комплексирования.

Формирование по ГСИ панхроматического изображения на основе выделения границ объектов

Вначале с использованием ГСИ выделяются границы объектов на основе корреляционного сопоставления спектральных характеристик [2]. Принадлежность элемента B_{mn} (m, n – соответственно номера элементов по строке и столбцу) определяется корреляционной мерой сходства его СХ с соседними элементами:

$$\tilde{\rho}_{mn} = \sqrt{\rho_{mn}^2 + \rho_{m,n+1}^2},$$

где $\rho_{mn} = \frac{B_{mn} \cdot B_{m+1,n} - \bar{B}_{mn} \cdot \bar{B}_{m+1,n}}{\sigma_{mn} \cdot \sigma_{m+1,n}}$, а черта над выраже-

ниями означает операцию усреднения по всем спектральным каналам.

Комплексирование ГСИ заключается в объединении каналов изображения по одному из алгоритмов, представленных выше. После этого выполняется совмещение скомплексированного ГСИ с результатом выделения границ с использованием меры

$$\gamma_5 = B^* + B^* \cdot (1 - \tilde{\rho}) = B^* (2 - \tilde{\rho}),$$

где $\tilde{\rho}$ – изображение границ объектов.

Результаты комплексирования натуральных ГСИ, полученных с авиационного датчика ProSpecTir-V (США), представлены на рис. 3: изображения A – результат усреднения каналов ГСИ, B – на основе полиномиального представления СХ, C – на основе выделения границ, D – по формуле (3). Из рисунка видно, что наилучший результат дает алгоритм комплексирования на основе полиномиального представления СХ.

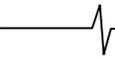
Заключение

При сравнении рассмотренных алгоритмов важным вопросом является численная оценка качества комплексирования. Поскольку целью поставленной задачи является повышение четкости отображения объектов, то для оценки качества этого процесса в работе [1] предложено использовать степень сохранения градиентов яркости по отношению к исходным изображениям. Предлагается оценивать не степень сохранения, а степень улучшения градиента.

Степень улучшения модуля градиента элемента (m, n) при переходе от изображения A к изображению F определим как

$$G^{AF}(x, y) = \begin{cases} \frac{g_F(m, n)}{g_A(m, n)}, & g_A(m, n) > g_F(m, n); \\ 1, & g_A(m, n) < g_F(m, n), \end{cases}$$

где $g_F(m, n)$ – модуль градиента в точке (x, y) (результат обработки изображения F оператором Собела).



Мера, оценивающая степень сохранения направления градиента при переходе от изображения A в изображение F , определяется как

$$A^{AF}(m, n) = \frac{\left| \alpha_A(m, n) - \alpha_F(m, n) \right| - \pi/2}{\pi/2},$$

где $\alpha_F(m, n) = \arctan(S_F^Y(m, n)/S_F^X(m, n))$ – направление градиента, $S_F^Y(m, n)$ – элемент (m, n) изображения, отфильтрованного вертикальным «окном» оператора Собела, $S_F^X(m, n)$ – элемент (m, n) изображения, отфильтрованного горизонтальным «окном» оператора Собела [3].

В [2] предлагается использовать нелинейные функции Q_G^{AF} и Q_A^{AF} , учитывающие чувствительность глаза к изменениям модуля и направления градиента. В этом случае:

$$Q^{AF}(m, n) = \sqrt{Q_G^{AF}(m, n) Q_A^{AF}(m, n)}.$$

Величина $Q^{AF} \in \{0, 0, 1, 0\}$, при этом 0 – полная потеря информации, 1 – полное сохранение градиента. По всему полю изображения, в случае сравнения двух исходных снимков A и B с результатом F , мера рассчитывается как взвешенная сумма:

$$Q^{AB/F} = \frac{\sum_{m=1}^M \sum_{n=1}^N [\omega_A(m, n) Q^{AF}(m, n) + \omega_B(m, n) Q^{BF}(m, n)]}{\sum_{m=1}^M \sum_{n=1}^N [\omega_A(m, n) + \omega_B(m, n)]}, \quad (5)$$

где $\omega_A(m, n)$ – информативность элемента (m, n) на изображении A . В качестве информативности в [2] пред-

лагается принять значение градиента в точке (m, n) . Результаты оценки качества комплексирования по (5) представлены в таблице:

Алгоритм комплексирования	Качество
СКО (γ_1)	0.12
Спектральный угол (γ_2)	0.09
Характер изменения СХ (γ_3)	0.30
Полиномиальное представление СХ (γ_4)	0.67
Выделение границ (γ_5)	0.14

Таким образом, рассмотрено несколько алгоритмов комплексирования гиперспектральных изображений. Представлено математическое обоснование целесообразности комплексирования каналов ГСИ для формирования нового изображения, на котором с высокой четкостью отображаются все объекты. Выполнена численная оценка качества функционирования алгоритмов с использованием критерия улучшения модуля градиента и сохранения его направления. Результаты работы алгоритмов апробированы на натуральных данных с авиационной гиперспектральной съемочной системы ProSpecTIR-V (США). На рис.3 на изображении B по сравнению с A наблюдается подчеркивание границ объектов за счет объединения скомплексированного изображения и коэффициентов корреляции СХ в граничных точках. На изображении B по сравнению с A с повышенной четкостью отображаются растительные объекты, улучшен контраст между строениями и дорожным покрытием (области явного улучшения выделены белыми окружностями). На изображении Γ с высоким контрастом представлены водные объекты, растительность – черным цветом.



а)



б)

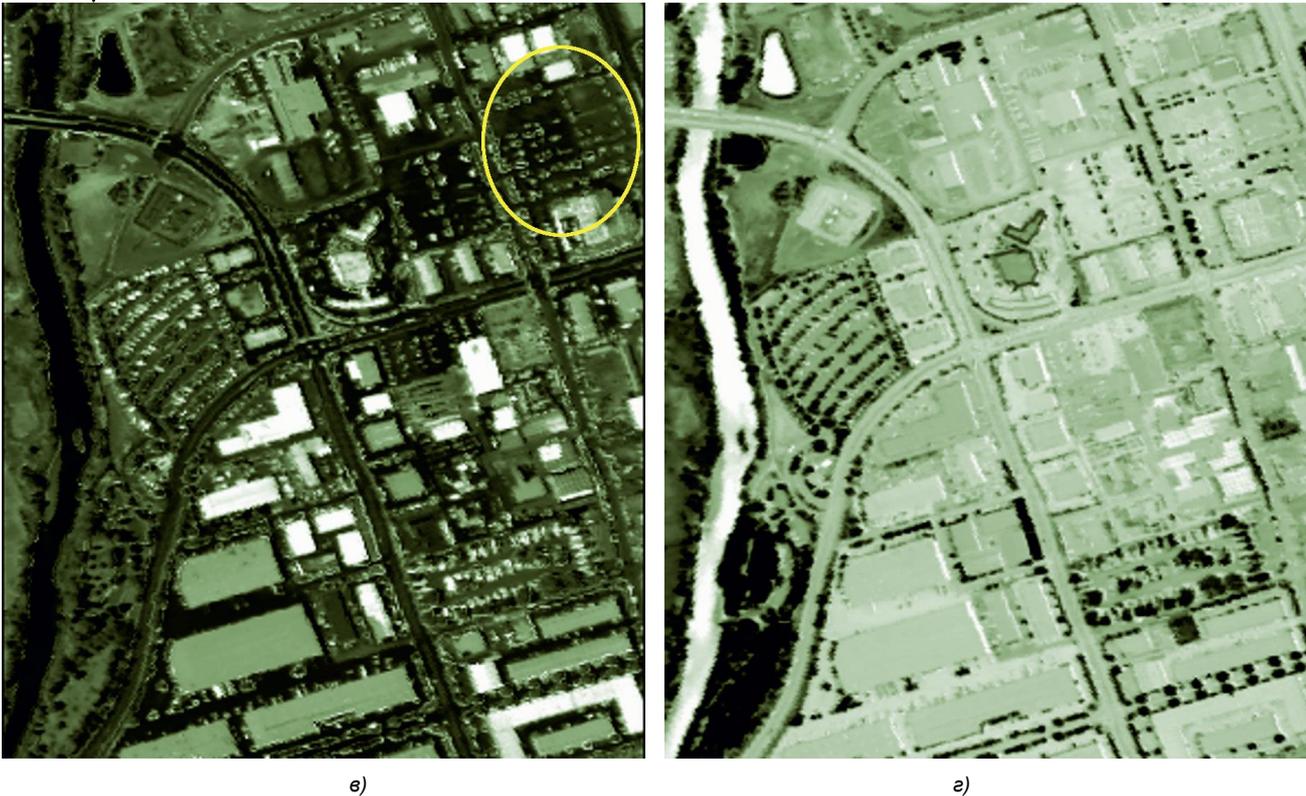


Рис. 3. Результаты комплексирования авиационного гиперспектрального снимка: а) усреднение каналов; б) полиномиальное представление CX ; в) комплексирование на основе выделения границ; г) комплексирование с учетом характера изменения CX

Литература

1. Антонушкина С.В., Еремеев В.В., Макаренков А.А., Москвитин А.Э. Особенности анализа и обработки информации от систем гиперспектральной съемки земной поверхности // Цифровая обработка сигналов. 2010. №4. С. 38-43.
2. Costas Xydeas, Vladimir Petrovic. Pixel-level image fusion metrics. Image fusion: algorithms and applications. – Great Britain: Academic Press, 2008. pp 429-450.
3. Претт У. Цифровая обработка изображения: Пер. с англ. – М.: Мир, 1982. – Кн.2 – 480 с.

OBJECTS READABILITY IMPROVING ON HYPERSPECTRAL IMAGERY OF EARTH SURFACE

*Eremeev V.V., Makarenkov A.A.,
Moskvitin A.E., Uydakov A.A.*

Hyperspectral imagery fusion for creation of new images with all objects presented with high readability were examined. Processing results of hyperspectral data were presented.

НОВЫЕ КНИГИ



Умняшкин С.В.

Теоретические основы цифровой обработки и представления сигналов:

Учебное пособие.

Второе издание, исправленное и дополненное

Москва: Техносфера, 2012 - 368 с.

Предназначено для студентов вузов, обучающихся по направлениям «Информатика и вычислительная техника» и «Прикладная математика». Включает в себя рассмотрение общих теоретических вопросов, связанных с цифровым представлением сигналов, основами анализа линейных дискретных систем. Значительное внимание уделено вопросам эффективного представления информации (сжатия данных) и использования вейвлет-преобразования. Может быть рекомендовано в качестве дополнительного материала для направлений подготовки радиотехнического и телекоммуникационного профиля.

КОМПЛЕКСИРОВАНИЕ ИЗОБРАЖЕНИЙ ОТ ЛИНЕЕК ФОТОПРИЁМНИКОВ В УСЛОВИЯХ ГЕОМЕТРИЧЕСКИХ ИСКАЖЕНИЙ

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

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

Москвитин А.Э., к.т.н., в.н.с. НИИ «Фотон» РГРТУ

Ключевые слова: комплексирование, линейка фотоприемников, пространственное, радиометрическое разрешение, геометрические искажения.

Введение

Современные системы дистанционного зондирования Земли (ДЗЗ) обычно строятся на базе линеек фотоприёмников [1]. Развертка изображений осуществляется в одном направлении за счет опроса чувствительных элементов линейки, а в другом – за счет механического сканирования [2] или за счет движения спутника. Один из путей повышения геометрического и радиометрического разрешения съемки является одновременное использование нескольких линеек фотоприёмников, установленных в фокальной плоскости видеодатчика [1].

На рис. 1 показан пример расположения в фокальной плоскости L одинаковых линеек на равном расстоянии друг от друга Δ с взаимными субпиксельными сдвигами на M (НС – направление сканирования).

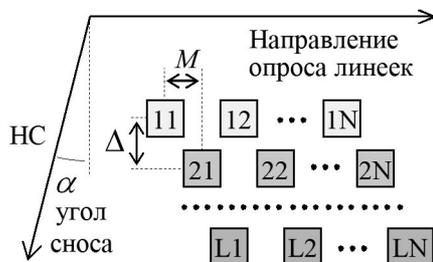


Рис. 1. Расположение линеек в фокальной плоскости

Весьма актуальной является задача комплексирования информации от нескольких линеек фотоприёмников с целью получения изображения максимального качества (радиометрического и пространственного разрешения). В некоторых системах ДЗЗ задача комплексирования может выступать как дополнительная. Например, современная гиперспектральная аппаратура строится на базе матрицы, содержащей до сотни и более линеек. Поэтому можно поставить задачу объединения информации от нескольких линеек (спектральных каналов) с целью повышения качества съемки (например, пространственного разрешения), конечно же за счет сокращения спектрального разрешения.

Общей особенностью систем ДЗЗ является то, что в процессе съемки развертка изображений происходит со значительными геометрическими искажениями. Обычно направление сканирования не перпендикулярно к ли-

Представлены модели и алгоритмы комплексирования изображений от субпиксельно смещенных линеек фотоприёмников с целью повышения пространственного и радиометрического разрешения в условиях геометрических искажений наблюдаемой сцены.

нейкам чувствительных элементов, угол сноса $\alpha \neq 0$. Для систем глобального наблюдения земли (например, геостационарная система «Электро-Л», тепловые каналы [3]) угол сноса обусловлен особенностями механического сканирования. Для низкоорбитальных систем характерно непостоянство угла сноса по полю изображения. Всё это значительно усложняет задачу комплексирования. В данной статье рассматривается решение задачи в условиях геометрических искажений (т.е. при $\alpha \neq 0$).

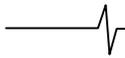
Постановка задачи комплексирования

Пусть L линеек фотоприёмников формируют дискретные изображения $B_l(m_l, n_l)$, $l = \overline{1, L}$. Требуется получить новое изображение $D(x, y)$, обладающее лучшими характеристиками по пространственному и радиометрическому разрешению. Взаимное расположение линеек, геометрию сканирования сцены и нового изображения $D(x, y)$ считаем полностью известными, например, в виде функций связи между системами координат (m_l, n_l) , $l = \overline{1, L}$, и (x, y) :

$$m_l = F_l(x, y), \quad n_l = G_l(x, y), \quad l = \overline{1, L}. \quad (1)$$

Вычисляемые по (1) координаты (m_l, n_l) , вообще говоря, не являются целочисленными. Яркости $B_l(m_l, n_l)$ определены только при $m_l, n_l \in Z$, где Z – множество целых чисел. В конечном счете $D(x, y)$ это дискретное изображение (по размерам большее исходных B_l). Однако дальнейшие выкладки станут проще, если поставить задачу получения непрерывного изображения $D(x, y)$. При этом для перехода к дискретному изображению достаточно взять отчеты $D(x, y)$ в узлах подпадающей сетки (периоды дискретизации следует выбирать в зависимости от реально достигнутого пространственного разрешения).

Достаточно произвольный выбор функций (1) подразумевает, что комплексирование может осуществляться одновременно с геометрической трансформацией. Если это не требуется, достаточно положить в (1) $x = m_1$,



$y = n_l$, при этом $D(x, y)$ геометрически идентично изображению от первой линейки, т.е. сформировано как бы новой более качественной линейкой, расположенной на месте первой.

Задача повышения радиометрического разрешения

Будем считать, что изображение от каждой линейки уже удовлетворительно по пространственному разрешению (т.е. шаг дискретизации подстилающий поверхности таков, что выполнены условия теоремы Котельникова), и рассмотрим задачу повышения только радиометрического разрешения. Этот случай наиболее прост: геометрически трансформируем изображения $B_l(m_i, n_l)$, $l = \overline{1, L}$, в плоскость (x, y) и усредним полученные изображения $D_l(x, y)$:

$$D(x, y) = \frac{1}{L} \sum_{l=1}^L D_l(x, y). \tag{2}$$

Трансформацию изображений удобно выполнять по обратному закону [1]:

$$D_l(x, y) = B_l^*(F_l(x, y), G_l(x, y)), \quad B_l^* = \mathfrak{R} B_l, \tag{3}$$

где \mathfrak{R} – оператор яркостной интерполяции, позволяющий из дискретного изображения B_l получить непрерывное изображение B_l^* . Существование оператора \mathfrak{R} следует из допущения об «удовлетворительном пространственном разрешении» каждого B_l . Напротив, на практике обычно применяются интерполяционные операторы, работа которых полагается «удовлетворительной», например, линейные:

$$B_l^*(m_i, n_l) = - \sum_{M, N \in Z} B_l(M, N) \rho(m_i - M) \rho(n_l - N), \tag{4}$$

где $\rho(t)$ – ядро интерполяционного оператора. Например, $\rho(t)$ для билинейной и бикубической интерполяции имеет вид

$$\rho_x(t) = \begin{cases} 1 - |t|, & |t| < 1, \\ 0, & |t| \geq 1, \end{cases}$$

$$\rho_k(t) = \begin{cases} 1 - 0.5|t| - t^2 + 0.5|t|^3, & |t| < 1, \\ 1 - 11/6|t| + t^2 - |t|^3/6, & 1 \leq |t| < 2, \\ 0, & |t| \geq 2. \end{cases} \tag{5}$$

Легко показать, что формула (1) является оптимальной в том смысле, что если отношения сигнал/шум (С/Ш) для изображений B_l одинаковы, то для $D(x, y)$ С/Ш достигает максимума. Если известен уровень шума каждого чувствительного элемента в виде дисперсии $\sigma_i^2(n_l)$, то можно получить более качественный результат.

Из (4) следует, что вызванная шумом дисперсия интерполированного отчета

$$\sigma_i^2(m_i, n_l) = \sigma^2 [B_l^*(m_i, n_l)] = \sum_{N \in Z} \sigma_i^2(N) \rho^2(n_l - N) \sum_{M \in Z} \rho^2(m_i - M). \tag{6}$$

Тогда оптимальное решение в плане С/Ш комплексированного изображения дает формула:

$$B(m, n) = \sum_{l=1}^L w_l(m, n) B_l^*(m_i, n_l) / \sum_{l=1}^L w_l(m, n), \tag{7}$$

где $w_l(m, n) = 1 / \sigma_i^2(m_i, n_l)$.

Для использования (7) изображения $B_l(m_i, n_l)$, $l = \overline{1, L}$, должны быть предварительно с высокой точностью выровнены по яркости, иначе изменение весов по полю снимка проявится в виде амплитудной модуляции изображения $D(x, y)$. Впрочем, поскольку изображения от отдельных линеек содержат практически одинаковый сюжет, это легко сделать путем выравнивания гистограмм [1].

Задача повышения пространственного разрешения

Пусть теперь требуется повысить пространственное разрешение. Сразу заметим, что использование формул (2)-(3) не даёт нужного результата: шаг дискретизации в направлении опроса линейки слишком велик, поэтому интерполяция по (3) не корректна. Построим иной интерполяционный оператор, учитывающий дискретизацию сцены несколькими линейками. Вообще говоря, вид интерполяционного оператора и качество интерполяции зависит от способа геометрической трансформации изображения (по прямому закону или по обратному) [1]. Для простоты будем решать задачу интерполяции в плоскости изображения первой линейки (что соответствует обработке по обратному закону).

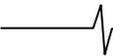
Будем считать, что чувствительные элементы линейки бесконечно малые. Конечные размеры пикселей эквиваленты включению НЧ - фильтра перед взятием выборок, поэтому в реальных системах возникает задача компенсации этих искажений. Эта задача в данной статье не рассматривается, некоторые подходы к её решению описаны в [4].

Рассмотрим плоскость (m, n) , где $m = m_i$, $n = n_l$. Обозначим

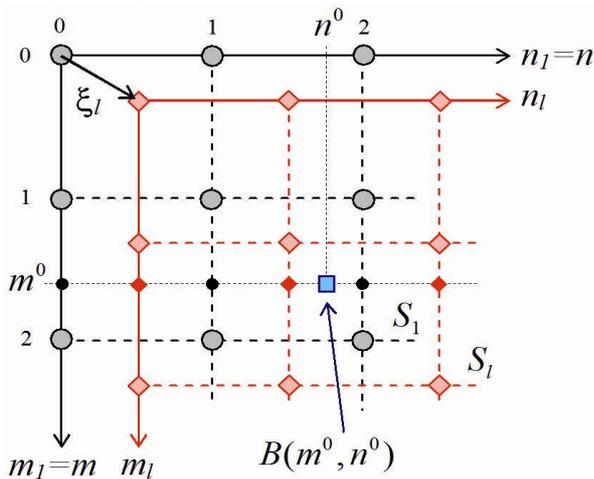
$$S_l = \{(m, n) : m = F(F^{-1}(M, N), G^{-1}(M, N)), n = G(F^{-1}(M, N), G^{-1}(M, N)), M, N \in Z\}$$

– множество узловых точек, порожденных l -й линейкой в плоскости (m, n) . Ставится задача восстановить непрерывное изображение $B(m, n)$ по его дискретным отчетам B_l в точках $(m, n) \in S_l$, $l = \overline{1, L}$. В общем случае эта задача оказывается достаточно сложной и имеет лишь приближенное решение. Однако можно принять два допущения:

Во-первых, линейки одинаковы и опрашиваются с одной и той же частотой. Очевидно, это наиболее простой вариант построения аппаратуры датчика. Тогда сетки S_l , $l = \overline{1, L}$, имеют одинаковый период.



Во-вторых, линейки расположены достаточно близко друг к другу в том смысле, что за интервал времени, когда крайние линейки фиксируют один и тот же сюжет, угол α не успевает существенно измениться. Если линейки смещены всего на несколько (десятки) пикселей, то это допущение для реальных систем наблюдения всегда выполнено. Тогда в каждой локальной области сетки S_l , $l = \overline{1, L}$, можно считать параллельными друг другу. В рамках этих допущений вид сеток S_l значительно упрощается. Рассмотрим фиксированную точку (m^0, n^0) и поставим задачу определения яркости $B(m^0, n^0)$. При интерполяции используются локальные операторы, поэтому существенна форма сеток S_l только вблизи (m^0, n^0) . Очевидно $S_1 = Z^2$. По введенным допущениям $S_l \approx S_1 + \xi_l$ (рис. 2), где ξ_l – вектор плоскопараллельного сдвига, который можно определить как $\xi_l = (\theta_l, \phi_l)$, $\theta_l = m_l^0 - m_1^0$, $\phi_l = n_l^0 - n_1^0$, $m_l^0 = F_l(m^0, n^0)$, $n_l^0 = G_l(m^0, n^0)$. (8)



- отсчеты B_1 ◆ отсчеты B_l
- интерполированные отсчеты $B_1(n_1)$
- ◆ интерполированные отсчеты $B_l(n_1)$

Рис. 2. Отсчеты в плоскости (m, n)

Пусть в направлении сканирования требуемое разрешение обеспечивается выбором частоты опроса линеек, а задача повышения разрешения ставится только в направлении опроса линеек (в действительности повысить частоту опроса гораздо проще, чем изготовить линейку с большим числом элементов). Тогда интерполяция в направлении сканирования корректна, например, функция

$$B_l^*(m_l, n_l) = \sum_M B_l(M, n_l) \rho(m_l - M) \quad (9)$$

корректно определена для $(m_l, n_l) \in R \times Z$, где R – множество вещественных чисел. Отсюда следуют два следствия.

Первое. Наличие сдвигов θ_l сеток дискретизации в направлении сканирования никак не влияет на качество интерполяции. Это важно для аппаратной реализации,

так как позволяет выбирать частоту опроса линеек без строгого согласования со скоростью сканирования. Заметим, что при оптимальном выборе частоты интерполяция вообще не требуется.

Второе. Задача интерполяции сводится к одномерной на суперпозиции сеток $s_l = Z + \varphi_l$, $l = \overline{1, L}$, (см. рис. 2). Возможность и алгоритм её решения зависит от конкретных значений сдвигов φ_l .

Для расположения линеек, показанном на рис. 1, сдвиги $\varphi_l = (l-1)(M + \Delta \sin \alpha)$. То есть субпиксельный сдвиг линеек M и ненулевой угол сноса приводят к одинаковым результатам. Для повышения пространственного разрешения обычно линейки устанавливаются со сдвигом $M = 1/L$, однако тот же результат можно достичь при $M = 0$ и угле сноса $\alpha = \arcsin [1/(\Delta L)]$. Это дает реальную возможность решения задачи повышения пространственного разрешения в системах ДЗЗ, в которых субпиксельное смещение линеек изначально не предусмотрено (например, в гиперспектральной аппаратуре).

Наиболее просто задача повышения разрешения решается, когда отсчеты располагаются равномерно: $\varphi_l = (l-1)/L$ при $M + \Delta \sin \alpha = 1/L$. В общем случае отсчеты располагаются группами по L равноотстоящих отсчетов, рис. 3.

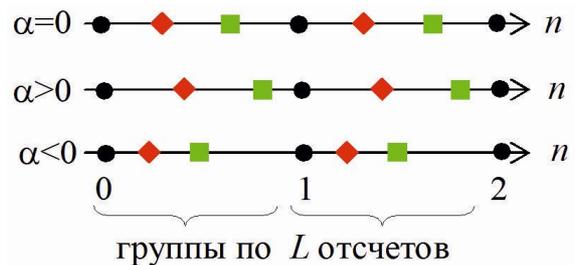


Рис. 3. Расположение отсчетов в зависимости от угла сноса ($L=3, M=1/L$)

Сначала построим интерполяционный оператор для равномерных отсчетов. Поскольку $\rho(t)$ – ядро интерполяционного оператора на единичной сетке, то $\rho(Lt)$ – ядро интерполятора для сетки с шагом $1/L$. Соответственно восстановление изображения выполняется как

$$B(m^0, n^0) = \sum_{l=1}^L \sum_N B_l^*(m_l^0, N) \rho(L(n^0 - N - \phi_l)) = \sum_{l=1}^L \sum_N B_l^*(m_l^0, N) \rho(L(n_l^0 - N)). \quad (10)$$

Интересно отметить, что (9), (10) отличаются от (4) только наличием множителя L .

При $\alpha \neq 0$ формула (10) не применима: во-первых, она не оптимальна, а во вторых, не корректно интерполирует даже поле постоянной яркости. Последний дефект можно исправить нормировкой:

$$B(m^0, n^0) = \frac{1}{w} \sum_{l=1}^L \sum_N B_l^*(m_l^0, N) \rho(L(n_l^0 - N)), \quad w = \sum_{l=1}^L \sum_N \rho(L(n_l^0 - N)). \quad (11)$$

Однако необходимое $w > 0$ гарантировано только при $L = 2$ и линейного ρ .

Формулу (11) можно использовать при не слишком значительных величинах α . Её достоинство в том, что используется уже имеющийся интерполяционный оператор. Однако это приближенное решение.

В работе [5] показано, что теорема, аналогичная теореме Котельникова, справедлива и для неравномерных отчетов при условии общего сохранения их плотности. Там так же получены формулы для интерполяционного оператора, обеспечивающего точное восстановление сигнала с финитным спектром. Применительно к рассматриваемой задаче это означает: если сетки s_l , $l = \overline{1, L}$, попарно различны, то можно повысить разрешение ровно в L раз (в том смысле, что верхняя пространственная частота изображения может быть выше в L раз при сохранении условия восстановимости изображения).

Однако использовать практически операторы из [5] нельзя: они имеют бесконечный носитель. Поэтому используем для синтеза интерполяционного оператора следующую аналогию. В практике обработки изображений для случая равномерных отчетов вместо идеального восстанавливающего ФНЧ используют полиномиальную аппроксимацию функции в окрестности восстанавливаемой точки. Будем также аппроксимировать дискретные отсчеты полиномом вблизи n^0 . Тут возможны различные варианты, поскольку расположение отсчетов вокруг n^0 имеет сложный характер. Однако именно ближайшие $2L$ отсчетов обладают полезным свойством: L отсчетов располагаются слева от n^0 и ровно L справа.

Введем полином $P(d) = \sum_{i=0}^{2L-1} a_i d^i$ и определим его коэффициенты из системы $2L$ уравнений:

$$\begin{cases} P(\lfloor n_l^0 \rfloor - n_l^0) = B_l^*(m_l^0, \lfloor n_l^0 \rfloor), l = \overline{1, L}; \\ P(\lfloor n_l^0 \rfloor + 1 - n_l^0) = B_l^*(m_l^0, \lfloor n_l^0 \rfloor + 1), l = \overline{1, L}, \end{cases} \quad (12)$$

где $\lfloor t \rfloor$ – максимальное целое, не превосходящее t . Тогда значение полинома $P(0)$ есть восстановленное значение яркости: $B(m^0, n^0) = P(0)$. Если разности $\lfloor n_l^0 \rfloor - n_l^0$ попарно различны, то система (12) имеет единственное решение (её определитель есть определитель Вандермонда). В противном случае задача повышения разрешения в L раз неразрешима (на практике это редкий случай, а при разумных ограничениях на α даже невозможный).

В вычислительном отношении эффективнее рассчитывать $P(0)$ по интерполяционной формуле Лагранжа:

$$P(0) = \sum_{t=0}^{2L-1} B_t \prod_{i \neq t} [d_i / (d_i - d_t)], \quad (13)$$

где B_t, d_t , $t = \overline{0, 2L-1}$, – аргументы полинома и его эталонные значения из (12).

Описанной процедуре также соответствует некоторый линейный интерполяционный оператор. На рис. 4

показаны импульсные характеристики этих операторов для $L = 2$, $\varphi_1 = 0.5$ и $\varphi_1 = 0.3$, а также для сравнения – идеальные фильтры из [5].

К сожалению, свойства интерполяционных операторов выше третьего порядка плохо исследованы даже для равномерного расположения отчетов.

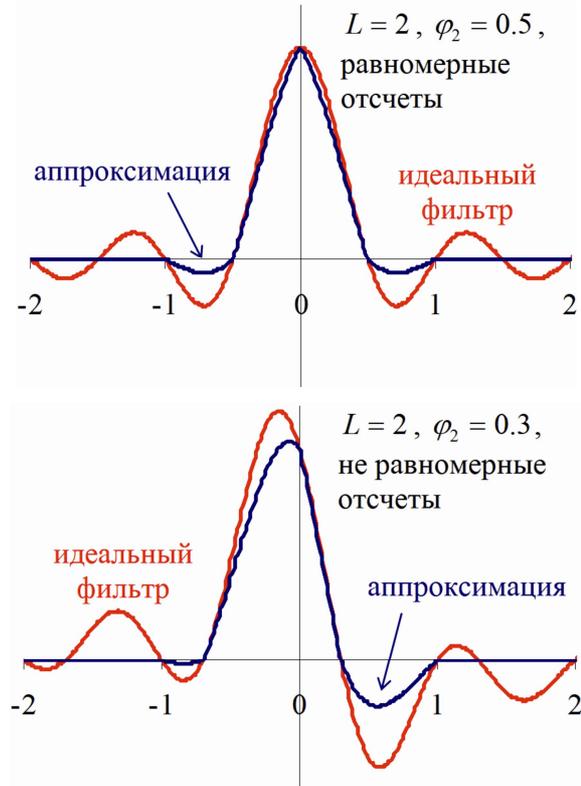


Рис. 4. Отклик идеального фильтра и его аппроксимация по (12)

Задача одновременного повышения радиометрического и пространственного разрешений

Часто необходимо повысить разрешение в $1 < K < L$ раз, а оставшуюся избыточность использовать для повышения радиометрического разрешения, т.е. отношения С/Ш. Общий подход может быть таким: разобьем линейки на группы по K линеек. В пределах каждой группы решим задачу повышения пространственного разрешения. Затем усредним L/K полученных результатов. Это показывает разрешимость задачи, по крайней мере при целых L/K .

Однако на практике удобнее обойтись без явного разбиения на группы. Так, в формуле (11) достаточно в аргументе функции ρ заменить множитель L коэффициентом K . В этом случае усреднение выполняется автоматически. Это также положительно отразится на работе интерполяционного оператора (11).

Формулу (12) также легко приспособить для одновременного решения задачи усреднения. В данном случае необходимо понизить в (12) порядок полинома с $2L-1$ до $2K-1$. Тогда система (6) станет переопределенной и её необходимо решать по МНК; повышение радиометрического разрешения достигается за счет фильтрующих свойства МНК.

Экспериментальное исследование

Выполнена апробация рассмотренных выше алгоритмов на тепловых изображениях от космической системы «Электро-Л». На рис. 5 представлены фрагменты восстановленного изображения в области сильных геометрических искажений ($\alpha \approx 4.6^\circ$, $\varphi_2 \approx 0.34$): А – изображение только от одной линейки (после интерполяции на более частой сетке); Б – результат объединения изображений от пары линеек по формуле (2); В – результат комплексирования по (11) для $L = K = 2$; Г – изображение, полученное по алгоритму (12). Сопоставляя А и Б, видим, что повышения пространственного разрешения не происходит, хотя радиометрическое улучшилось.

Изображение В имеет лучшее пространственное разрешение по отношению к А и Б, однако на В присутствуют артефакты (если α мал, они не возникают). Изображение Г в условиях геометрических искажений оказывается наиболее качественным.

Литература

1. Злобин В.К., Еремеев В.В. Обработка аэрокосмических изображений. М. Физматлит, 2006. 288 с.
2. Воронин А.А., Егошкин Н.А., Еремеев В.В., Москатиньев И.В. Геометрическая обработка данных от кос-

мических систем глобального наблюдения земли. // Вестник РГРТУ, 2009. №1. С. 12-17.

3. Гектин Ю.М., Еремеев В.В., Егошкин Н.А., Зенин В.А., Москатиньев И.В. Нормализация изображений от геостационарной космической системы наблюдения Земли. // Цифровая обработка сигналов, 2011. №3. С. 28-32.

4. Москвитин А.Э. Технологии и алгоритмы повышения качества изображений земной поверхности на основе комплексирования спектральной видеoinформации. Дисс. канд. тех. наук, Рязань, 2003. 130 с.

5. Хургин Я.И., Яковлев В.П. Фinitные функции в физике и технике. Изд. 2. М. Книжный дом «ЛИБРОКОМ», 2010. 416 с.

IMAGE FUSION FROM PHOTODETECTORS LINES IN CASE OF GEOMETRIC DISTORTION

Eremeev V.V., Egoshkin N.A., Moskvitin A.E.

Presents models and algorithms for aggregation of images from subpixel shifted photodetector arrays to increase the spatial and radiometric energy resolution in geometric distortion of the observed scene.

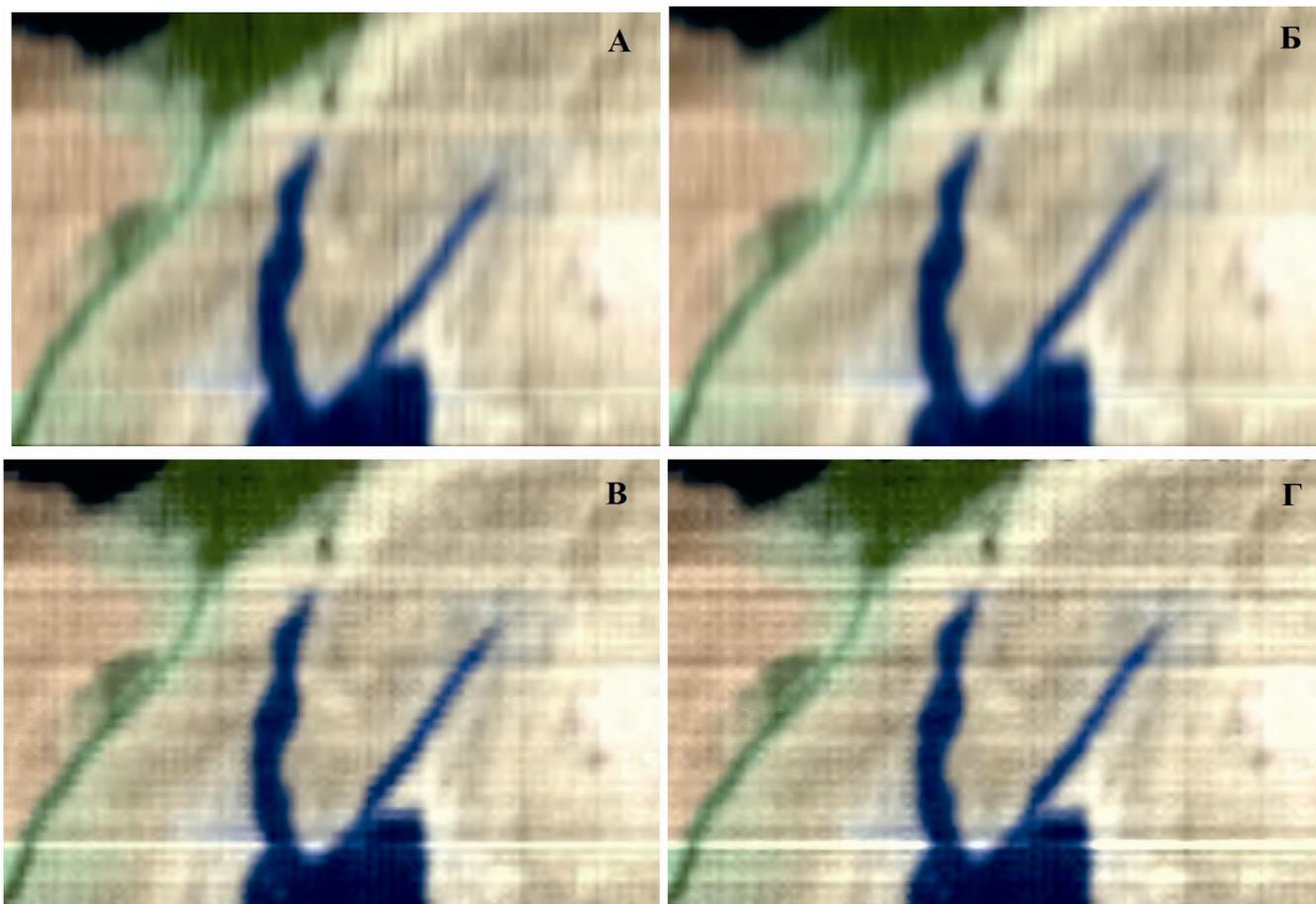


Рис. 5. Исходное изображение от одной линейки (А) и результаты комплексирования изображений от пары линеек различными алгоритмами (Б, В, Г)

УДК 004.932

ТЕХНОЛОГИЯ ВИЗУАЛЬНОГО ГЕОДЕЗИЧЕСКОГО ОРИЕНТИРОВАНИЯ КОСМИЧЕСКИХ СНИМКОВ ЗЕМНОЙ ПОВЕРХНОСТИ

Кочергин А.М., к.т.н., с.н.с. НИИ "Фотон" РГРТУ, e-mail: foton@rsreu.ru

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

Побаруев В.И., к.т.н., с.н.с. НИИ "Фотон" РГРТУ

Шокол А.С., аспирант, и.о. начальника НЦ ОМЗ ОАО «Российские космические системы», e-mail: ntsomz@ntsomz.ru

Ключевые слова: дистанционное зондирование земли, геодезическое ориентирование, технология, маршрутные съемки.

Рассматривается технология визуального геодезического ориентирования космических снимков земной поверхности и математическая модель, лежащая в ее основе.

Введение

Во всех системах дистанционного зондирования Земли (ДЗЗ) операции геодезического ориентирования протяженных маршрутов съемки отводится важная роль, поскольку требуется оперативно оценить точность геопривязки представленных на снимке объектов к местности. При этом если координатные рассогласования достаточно велики, то необходимо уточнить параметры внешнего ориентирования снимка с целью последующего адресного размещения информации в электронном каталоге и корректного формирования выходных информационных продуктов стандартных уровней обработки.

В системном плане операция геодезического ориентирования маршрутов съемки должна выполняться максимально быстро в комплексе предварительной обработки данных ДЗЗ. В комплексе создания стандартных продуктов так же может выполняться геодезическое ориентирование, но уже не целых маршрутов, а отдельных кадров с более высокой точностью, на основе большого количества опорных точек. Таким образом, задачей геодезического ориентирования маршрутов съемки по минимальному числу опорных точек является контроль точности и компенсация возможных грубых ошибок геопривязки видеоданных. Поскольку маршруты съемки представляют собой изображения земной поверхности сверхбольших объемов (до нескольких сотен гигабайт), то к технологии геодезического ориентирования предъявляются высокие требования по производительности и удобству в работе.

Технология визуального геодезического ориентирования

Анализ грубых ошибок координатной привязки целевой информации от спутниковых систем «Ресурс-01», «Метеор-М», «Ресурс-ДК» и др. показывает, что основным источником погрешностей являются неточности заданий начальных условий, описывающих положение КА в пространстве и задаваемых вектором $\boldsymbol{\eta}(t_0)$, времени формирования первой строки изображения t_0 и углового положения спутника. Причем грубые ошибки, вы-

званные неточным угловым положением КА, могут быть скомпенсированы поправкой к вектору $\boldsymbol{\eta}(t_0)$. Исходя из этого повысить точность геопривязки маршрута съемки можно, используя всего нескольких опорных точек (в пределе одну), для уточнения отдельных параметров кеплеровской орбиты спутника.

Если взять за основу технологию прецизионного геодезического ориентирования, используемую на этапе создания ортопланов, то выполнить контроль и коррекцию геопривязки маршрута съемки можно так, как это показано на рис. 1. В соответствии с этой схемой на основе углового положения сканера, задаваемого вектором $\mathbf{R}_c(t)$, для каждого пикселя исходного снимка

$B(m, n)$, $m = \overline{1, M}$, $n = \overline{1, N}$, рассчитываются функции вычисления геодезических координат:

$$\phi = f_\phi [\mathbf{R}_c, \boldsymbol{\eta}(t_0), m, n],$$

$$\lambda = f_\lambda [\mathbf{R}_c, \boldsymbol{\eta}(t_0), m, n]. \quad (1)$$

Далее, на экране монитора отображается исходное изображение $B(m, n)$ и геокодированная карта $E(\phi, \lambda)$. Оператор, просматривая снимок и карту, идентифицирует опорные точки (ОТМ) с координатами $(\phi, \lambda)_i$ и $(\phi^*, \lambda^*)_i$, $i = 1, 2, \dots$. Полученные рассогласования пересчитываются в линейные величины $(\Delta x, \Delta y)_i$, на основе которых вычисляются поправки к орбитальным параметрам $\Delta \boldsymbol{\eta}$ и времени съемки первой строки Δt_0 .

Нетрудно заметить, что данная технология весьма ресурсоемкая, поскольку оператору необходимо анализировать большие массивы видеоданных для поиска одноименных объектов на снимке и карте. При этом требования к числу ОТМ и месту их расположения в значительной степени влияют на точность геопривязки маршрута. Поэтому предлагается более эффективная - так называемая, технология визуального геодезического ориентирования, схема которой приведена на рис. 2. Основная ее идея состоит в том, что для контроля геопривязки используется сжатое изображение B^* , которое трансформируется в плоскость карты (снимок D_B). За-

тем снимок D_B и карта (изображение D_E) полупрозрачно совмещаются с коэффициентом полупрозрачности q . Оператор с помощью курсора плоско-параллельно смещает снимок относительно карты, добиваясь наилучшего совмещения одноименных объектов по всему изображению земной поверхности. Величина смещения $\Delta x, \Delta y$ используется для вычисления поправок $\Delta \eta, \Delta t_0$.

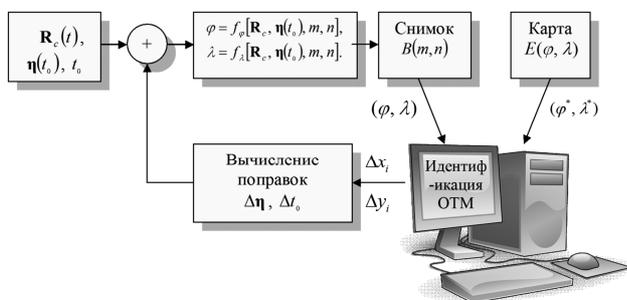


Рис. 1. Технологическая схема геодезического ориентирования на основе ОТМ

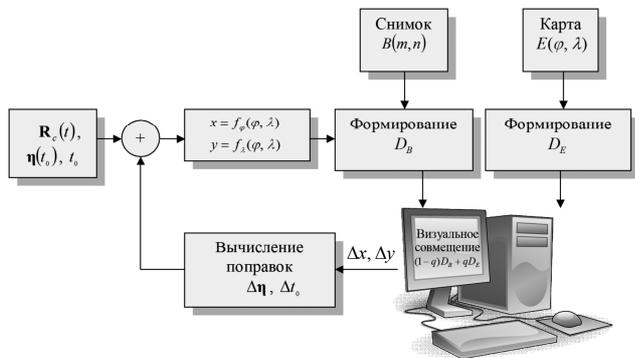


Рис. 2. Схема выполнения операций при визуальном геодезическом ориентировании

Таким образом, благодаря тому, что для анализа используется изображение меньшего объема B^* и не требуется находить отдельные ОТМ обеспечивается высокая автоматизация и оперативность процесса контроля и геодезического ориентирования маршрута съемки.

Математическая модель визуального геодезического ориентирования

Рассмотрим основные математические соотношения, лежащие в основе процедуры геодезического ориентирования.

Зададим географическими координатами $(\phi_{\min}, \lambda_{\min})$, $(\phi_{\max}, \lambda_{\max})$ область определения синтезируемого изображения

$$D(x, y), \quad x = \overline{1, X}, \quad y = \overline{1, Y},$$

где $X = (x_{\max} - x_{\min})/\mu$, $Y = (y_{\max} - y_{\min})/\mu$,
 $x_{\min} = f_x(\phi_{\min})$, $y_{\min} = f_y(\lambda_{\min})$, $x_{\max} = f_x(\phi_{\max})$,
 $y_{\max} = f_y(\lambda_{\max})$, μ - масштаб отображения,
 f_x, f_y - функции картографического преобразования.

Для проекции Меркатора

$$x = f_x(\phi) = \frac{R \cos \phi_c}{0,4343} \lg \operatorname{tg} \left(\frac{\pi}{4} + \frac{\phi}{2} \right), \quad (2)$$

$$y = f_y(\lambda) = R \cos \phi_c \lambda,$$

где ϕ_c - широта стандартной параллели, $\phi_c = (\phi_{\max} - \phi_{\min})/2$; R - радиус земного эллипсоида на широте ϕ_c .

Изображение B^* сформируем путем интегрального сжатия исходного снимка B с коэффициентом w :

$$B^*(m', n') = \frac{1}{w^2} \sum_{m=w(m'-1)+1}^{m'w} \sum_{n=w(n'-1)+1}^{n'w} B(m, n),$$

где m', n' - координаты элемента сжатого изображения. Величина коэффициента сжатия $w > 1$ определяется из условия: $MN/w^2 \leq V_{on}$, где V_{on} - объем доступной оперативной памяти. Далее выполним трансформирование сжатого изображения B^* в плоскость снимка D . При этом из-за масштабных изменений трансформирование по прямым зависимостям приводит к появлению на результирующем изображении «пустот», снижающих контраст и соответственно дешифрируемость изображения. Поэтому процесс геометрического преобразования изображения B^* выполним на основании обратных зависимостей:

$$B(x, y) = \begin{cases} B^*(m', n'), & (wm' \in [1, M]) \text{ и } (wn' \in [1, N]); \\ \text{иначе } b_{\max}, \end{cases}$$

где $m' = \operatorname{ent} [F_m(x, y)/w]$, $n' = \operatorname{ent} [F_n(x, y)/w]$;

F_m, F_n - функции, устанавливающие связь систем координат изображений B и D на основе выражений (1), (2).

Поскольку соотношения (1), (2), как правило, не допускают аналитического обращения, то в расчетах используем их полиномиальные аппроксимации. Для получения высокоточных полиномиальных приближений исходных функций F_m, F_n разработана трехэтапная процедура, основанная на использовании промежуточной проекции, обладающей минимальными искажениями относительно системы координат исходного снимка. На первом этапе рассчитываются географические координаты элемента (x, y) на основе аналитических зависимостей, полученных путем обращения выражений (2):

$$\phi = f_x^{-1}(\mu x + x_{\min}), \quad \lambda = f_y^{-1}(\mu y + y_{\min}) \quad (3)$$

На втором этапе выполняется пересчет координат (ϕ, λ) в косую равнопромежуточную цилиндрическую проекцию (КРЦП). Для этого осуществляем переход от геодезической (ϕ, λ) к геоцентрической (ϕ', λ') системе координат:

$$\phi' = \operatorname{arc} \sin \left(\sin \phi \sqrt{1 - e^2} / \sqrt{1 - e^2 \sin^2 \phi} \right), \quad (4)$$

$$\lambda' = \lambda,$$

где e - эксцентриситет земного эллипсоида. Далее оп-



ределяются значения широты ϕ_0 и долготы λ_0 полюса косой сферической системы координат:

$$\phi_0 = \frac{\pi}{2} - i, \quad \lambda_0 = L - \frac{\pi}{2} \quad (5)$$

где $L = \gamma - w_3 \frac{M}{2} t_c$, γ – долгота восходящего узла по отношению к гринвичскому меридиану на момент формирования первой строки сюжета t_0 , w_3 – скорость вращения Земли, t_c – период сканирования, i – угол наклона орбиты. Далее выполняется переход от прямых (ϕ', λ') к косым (ϕ'', λ'') сферическим координатам:

$$\begin{aligned} \phi'' &= \arcsin \left[\sin \phi' \sin \phi_0 + \cos \phi' \cos \phi_0 \cos (\lambda_0 - \lambda') \right], \\ \lambda'' &= \arcsin \operatorname{tg} \left[\frac{\cos \phi' \sin (\lambda_0 - \lambda')}{\sin \phi' \cos \phi_0 - \cos \phi' \sin \phi_0 \cos (\lambda_0 - \lambda')} \right]. \end{aligned} \quad (6)$$

После этого вычисляются координаты пикселя (u, v) в КРПЦ по формулам:

$$u = R_3 \lambda'', \quad v = R_3 \phi'' \quad (7)$$

где R_3 – радиус Земли на экваторе.

На третьем этапе выполняем пересчет значений u и v в систему координат изображения B :

$$\begin{aligned} m &= \sum_{\rho=0}^P \sum_{\nu=0}^{\rho} a_{\nu, \rho-\nu} u^{\nu} v^{\rho-\nu}, \\ n &= \sum_{\rho=0}^P \sum_{\nu=0}^{\rho} c_{\nu, \rho-\nu} u^{\nu} v^{\rho-\nu}, \end{aligned} \quad (8)$$

где $(a_{\rho\nu}, c_{\rho\nu})$ – коэффициенты полинома, рассчитанные по методу наименьших квадратов.

Таким образом, формулы (3)-(8) определяют вид искоемых функций F_m и F_n . Апробация данной технологии на протяженных маршрутах, полученных со спутника «Ресурс-О1», показала, что использование в формуле (8) полинома 3-й степени обеспечивает погрешность в пересчете координат не более 0,5 пикселя.

Рассмотрим теперь процесс нанесения на изображение D векторной электронной карты. В качестве картографической основы используем гидрографический слой, который характеризуется высокой временной устойчивостью и хорошо различим на космических снимках. Электронная карта представляется множеством линейных объектов $E = \{ (\phi_s^H, \lambda_s^H; \phi_s^K, \lambda_s^K); s = \overline{1, S} \}$, принадлежащих выбранному региону, где (ϕ_s^H, λ_s^H) и (ϕ_s^K, λ_s^K) – соответственно географические координаты начальных и конечных точек s -го объекта. Выполним пересчет элементов множества E в систему координат изображения D : $E \rightarrow E_d = \{ (x_s^H, y_s^H; x_s^K, y_s^K); s = \overline{1, S} \}$, где $x_s = (f_x(\phi_s) - x_{\min}) / \mu$, $y_s = (f_y(\lambda_s) - y_{\min}) / \mu$, и осуществим их векторно-растровое отображение: $D_E(x, y) = b_e$, если

$(x, y) \in (x_s^H, y_s^H; x_s^K, y_s^K)$, $s = \overline{1, S}$, где b_e – код яркости объектов карты.

Операция полупрозрачного совмещения снимка и карты описывается выражением

$$D = (1 - q)D_B + qD_E,$$

где q – коэффициент прозрачности, находящийся в диапазоне от 0 до 1 и устанавливаемый оператором. По результатам совмещения определяется взаимный сдвиг снимка и карты $(\Delta x, \Delta y)$, используемый для расчета поправок $\Delta \eta$.

При коррекции параметров орбиты из всех кеплеровских компонентов вектора $\eta(t_0) = [\eta_k(t_0)]$, $k = \overline{1, K}$, выбираются лишь те, ошибки которых вносят наибольшую погрешность географической привязки, а именно: время прохождения перицентра τ , долгота восходящего узла Ω и наклонение орбиты i . Их уточнение осуществляется на основе координат ОТМ, представленных в КРПЦ, которая обладает минимальными искажениями по отношению к системе координат снимка. Эта особенность позволяет обеспечить работоспособность алгоритма при малом числе ОТМ.

На основе соотношений (3)-(8) ОТМ пересчитываются в область определения исходного изображения и в систему координат КРПЦ. В результате формируются два множества $O_b = \{ m, n; m^*, n^* \}$, и $O_k = \{ u, v; u^*, v^* \}$. С учетом выражений (1) и (4)-(7) функциональная связь между компонентами этих множеств определяется как

$$\begin{aligned} u &= F_u(\eta(t_0), m, n) = \Phi_u(f_\phi, f_\lambda), \\ v &= F_v(\eta(t_0), m, n) = \Phi_v(f_\phi, f_\lambda), \end{aligned} \quad (9)$$

где Φ_u, Φ_v – функции координатных преобразований, определяемые формулами (4) - (7).

Обозначим элементы множества O_k в виде векторов $\mathbf{r} = (u, v)$ и $\mathbf{r}^* = (u^*, v^*)$. Тогда в общем случае:

$$\Delta \mathbf{r} = \mathbf{r}^* - \mathbf{r} = \mathbf{F}[\eta(t_0) + \Delta \eta, m, n] - \mathbf{F}[\eta(t_0), m, n],$$

где $\mathbf{F} = (F_u, F_v)$, $\Delta \eta = (\Delta \eta_k; k = \overline{1, K})$ – искомый вектор орбитальных поправок. Разложим правую часть данного соотношения в ряд Тейлора относительно компонентов вектора $\Delta \eta$, ограничившись линейными членами. В результате получим систему линейных уравнений относительно искомых величин $\Delta \eta_k$:

$$\Delta \mathbf{r}_j = \sum_{k=1}^K \frac{\partial \mathbf{F}[\eta(t_0) + \Delta \eta, m_j, n_j]}{\partial \eta_k} \Bigg|_{\Delta \eta=0} \cdot \Delta \eta_k, \quad (10)$$

$$k = \overline{1, K}.$$

Значения производных находим через конечные разности:

$$\frac{\partial \mathbf{F}(j)}{\partial \eta_k} = \frac{\mathbf{F}[\eta(t_0) + \varepsilon_k, m_j, n_j] - \mathbf{F}[\eta(t_0), m_j, n_j]}{\varepsilon_k},$$

где ε_k - достаточно малое приращение параметра η_k по отношению к $\Delta\eta_k$.

Система (10) при $J \geq K$ решается с помощью метода наименьших квадратов. Однако для $J = 1$ или $J = 2$ при $K = 3$ система не имеет решения. Работоспособность алгоритма обеспечивается в этом случае следующим образом.

Известно, что изменение времени прохождения перицентра τ , главным образом, оказывает влияние на значения координат в продольном направлении маршрута, а изменение долготы восходящего узла Ω - на смещениях в поперечном направлении. С учетом этого поправка $\Delta\eta_1 = \Delta\tau$ оценивается как

$$\Delta\tau = (u^* - u) / \frac{\partial F_u}{\partial \tau}.$$

Оставшееся приращение $\Delta\Omega$ определяем из уравнения

$$\Delta v = \frac{\partial F_v}{\partial \Omega} \Delta\Omega.$$

Полученные поправки $\Delta\eta = (\Delta\tau, \Delta\Omega)$ используются для коррекции вектора начальных условий $\Delta\eta(t_0)$.

Заключение

Рассмотренная технология уже много лет успешно используется в НЦ ОМЗ и ряде других приемных центров при первичной обработке и каталогизации данных КА «Ресурс-ДК», «Монитор-Э», «Метеор-М» № 1. На рис. 3 показан пользовательский интерфейс, обеспечивающий визуальное ориентирование снимка относительно карты. В качестве опоры используется векторная электронная карта DCW масштаба $1:10^6$. Достижимая при этом точность геопривязки снимка оказывается даже выше точности карты (например, для снимков КА «Ресурс-ДК» это не хуже 100м), поскольку интегральное совмещение всего маршрута с картой позволяет минимизировать ошибки представления на карте отдельных объектов.

THE TECHNOLOGY OF EARTH OBSERVATION IMAGES VISUAL GEODESIC ORIENTATION

*Kochergin A.M., Kuznetsov A.E.,
Pobaruev V.I., Shokol A.S.*

In the article are described the technology of Earth observation images visual geodesic orientation and mathematical model it based on.

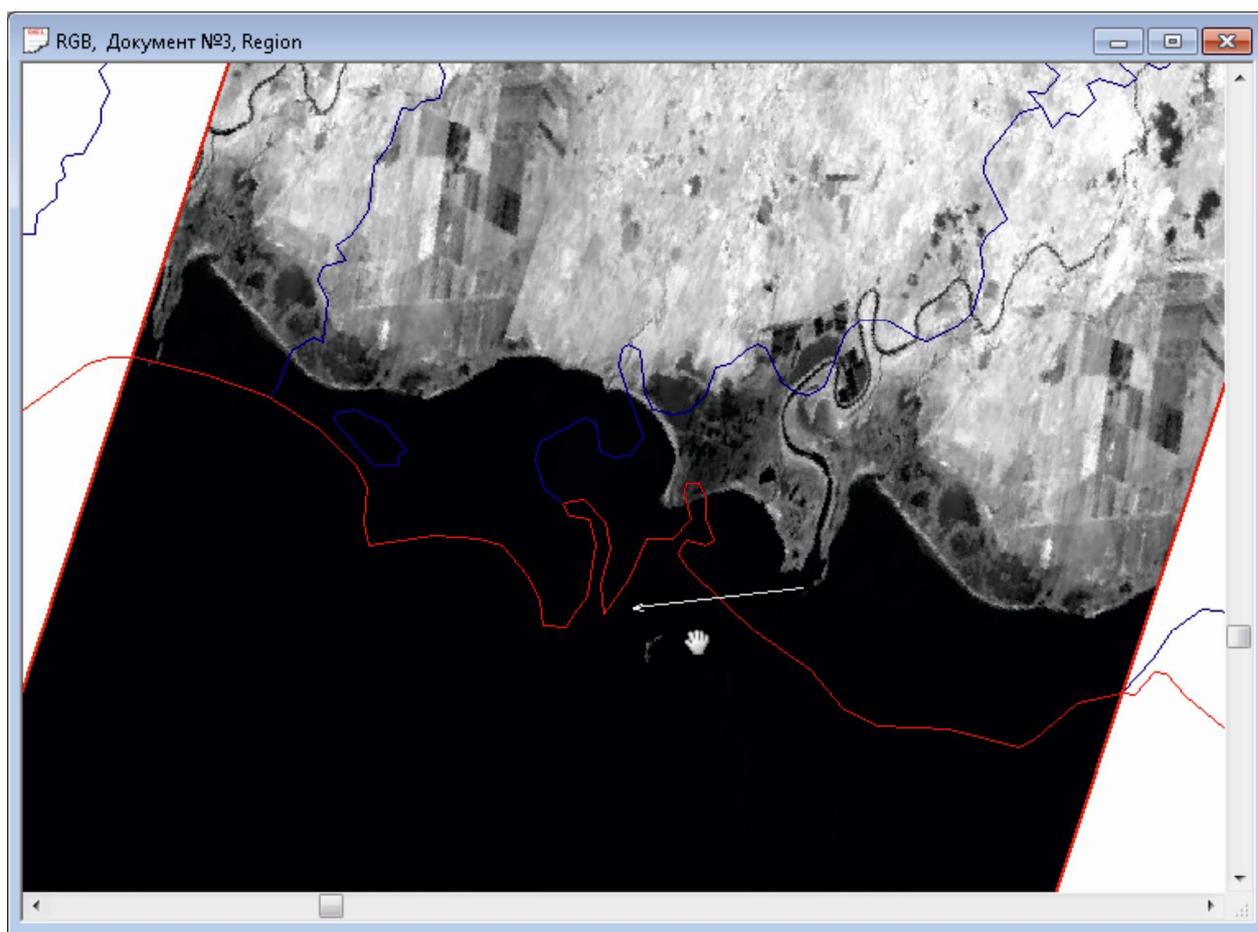


Рис. 3. Интерфейс пользователя по визуальному совмещению снимка и карты.
Ошибки геопривязки устраняются плоско-параллельным смещением снимка относительно карты

УДК 681.3

СУБПИКСЕЛЬНАЯ КЛАССИФИКАЦИЯ ОБЪЕКТОВ НА КОСМИЧЕСКИХ ГИПЕРСПЕКТРАЛЬНЫХ ИЗОБРАЖЕНИЯХ

*Тишкин Р.В., к.т.н., начальник отдела 4374 филиала ФГУП «ГНПРКЦ «ЦСКБ-Прогресс» - ОКБ «Спектр»,
e-mail: roman.tishkin@mail.ru*

Юдаков А.А., начальник отдела ФГУП «ГНПРКЦ «ЦСКБ-Прогресс», e-mail: antonyudakov@yandex.ru

Ключевые слова: сегментация изображений, гиперспектральные изображения, нечеткая логика.

Введение

В настоящее время при аэрокосмическом наблюдении Земли гиперспектральная съемка и анализ результатов получил большое развитие. Федеральное космическое агентство России планирует запуск космических аппаратов (КА) с аппаратурой гиперспектральной съемки [1]. Эти планы способствовали появлению большого количества публикаций, посвященных обработке гиперспектральных космических изображений. Однако, практически отсутствуют отечественные публикации, посвященные субпиксельной классификации объектов на гиперспектральных снимках (в оригинале spectralunmixing), в то время как зарубежные исследователи уделяют этому вопросу повышенное внимание [2]. Данная статья является попыткой изложить подходы к решению этой задачи.

Основа для решения поставленной задачи

Прострастенную разрешающую способность съемочной аппаратуры принято характеризовать наземным интервалом дискретизации (НИД) и мгновенным полем обзора (МПО)[3]. Известно, что на космических снимках можно обнаруживать объекты, линейные характеристики которых существенно меньше чем значения НИД и МПО данной съемочной аппаратуры.

Данный эффект возникает при сильной контрастности между объектом и окружающим его фоном. В [3] указывается, что яркий объект на темном фоне будет зафиксирован съемочной системой, даже в случае, если он занимает 0,4% МПО. Этим объясняется появление на высокодетальных снимках объектов, линейные размеры которых значительно меньше пространственного разрешения съемочной системы.

Известно, что даже при высокодетальной съемке поверхности Земли большинство объектов, попавших в МПО, неоднородны. В случае гиперспектральной съемки рассчитываемый спектральный состав пикселя является фактически комбинацией спектральных характеристик объектов, которые этот пиксель образуют. В связи с этим возникает ряд новых задач.

Во-первых, требуется «очистка» спектральной характеристики объекта для его более надежной классификации с использованием спектральных библиотек.

Во-вторых, привлекательной является возможность

Рассматриваются различные подходы к решению задачи субпиксельной классификации объектов на гиперспектральных космических изображениях. Предлагается использование алгоритмов, основанных на нечеткой логике для решения указанной задачи.

восстановления информации о компонентном составе пикселей изображения, т.е. субпиксельной классификации.

В данной работе под субпиксельной классификацией понимается процедура, в ходе которой измеренный спектр пикселя разбивается на набор составляющих спектров (конечных элементов, объектов) с определением соответствующих долей (показателей относительного содержания), которые показывают пропорции каждого конкретного объекта, входящего в состав пикселя. Конечные элементы могут соответствовать известным макроскопическим элементам среды, таким как: вода, почва, металл, растительность либо другим объектам.

Далее будут рассмотрены известные и предлагаемые подходы к решению указанной задачи.

Известные подходы

В настоящее время большинство подходов к субпиксельной классификации основано на предположении о линейном характере взаимодействия объектов, входящих в МПО съемочной аппаратуры.

Так же принято выделять два варианта проведения субпиксельной классификации: контролируемый (supervised) и неконтролируемый (unsupervised). Если принять во внимание, что отражающая поверхность представлена в виде смеси типа «шахматная доска» (рис. 1).

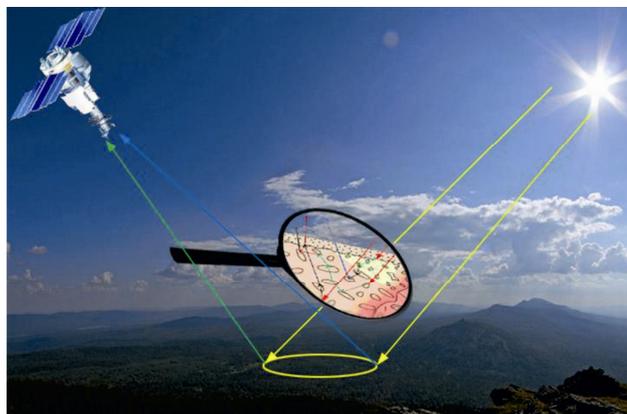


Рис. 1.

Любой определенный пакет падающего излучения взаимодействует только с одним компонентом и общая



площадь поверхности разделена пропорционально относительной распространённости конечных элементов, то отраженное излучение передает характеристики связанных сред в тех же пропорциях. В этом смысле существует линейное соотношение между относительной распространённостью элементов, составляющих отбражаемую область, и спектрами в отраженном излучении. Однако эта модель весьма упрощена.

Аналитические модели смешивания отдельных материалов составляют основу для развития методов восстановления оценок спектров составных элементов смешанных пикселей и их пропорций. Однако полная модель процесса смешивания является более сложной, чем простое описание взаимодействия смесей на поверхности. Модели смешивания могут также включать эффекты трехмерной топологии объектов в среде, такие как: высота деревьев, размер и плотность их листовых покровов, а также угол наблюдения датчика.

Основной предпосылкой моделирования смесей является тот факт, что в пределах данной среды на поверхности доминирует небольшое количество отдельных материалов, имеющих относительно постоянные спектральные свойства. Эти отдельные элементы (например, вода, трава, типы минералов) называются конечными элементами, и доля их содержания в смешанном пикселе называется относительной распространённостью. Из того, что большая часть спектральной изменчивости среды является последовательностью конечных элементов, присутствующих в переменных пропорциях, логически следует, что определенная комбинация их спектральных свойств позволяет моделировать спектральную изменчивость, наблюдаемую системой дистанционного зондирования.

В статье [5] предлагается подход на основе искусственных нейронных сетей (ИНС) для извлечения пропорций конечных элементов в пикселях из гиперспектральных данных. ИНС выполняет процедуру сокращения размерности данных и результирующее несмешивание. Архитектура сети выполняет обе операции в автоматическом режиме, однако сами операции должны быть проработаны по отдельности для исключения возможных ошибок. Результаты классификации показывают, что ИНС дает более точные оценки пропорций конечных элементов в пикселях, чем линейные подходы. Однако требование при обучении ИНС использовать «чистые» пиксели существенно затрудняет проведение процедуры обучения, так например, обучение ИНС по снимку уже становится невозможным. Фактически требуется либо наличие некоторой априорной информации, либо использование существующих спектральных библиотек.

Пути усовершенствования

Отличительной особенностью всех приведенных методов является то, что допускаются предположения о линейном законе «смешивания» компонентов пикселя. В общем случае это не так. Объекты, попавшие в МПО, могут быть совершенно разными по природе и соотношению в МПО. Для проведения субпиксельного анализа было бы полезно использовать данные съемочной ап-

паратуры с более высоким пространственным разрешением, чем разрешение гиперспектрометра.

В случае отечественного КА «Ресурс-П» данная задача облегчается тем, что КА оснащен как гиперспектральной, так и высокодетальной аппаратурой съемки Земли разработки ОАО «Красногорский завод». Наличие такой совокупности съемочной аппаратуры позволяет КА осуществлять синхронную съемку в одной полосе обзора как высокодетальной съемочной аппаратурой, так и гиперспектральной.

В итоге на обработку поступают два набора данных: высокодетальное панхроматическое изображение и набор гиперспектральных изображений со средним пространственным разрешением (гиперкуб). Перспективным представляется использование данных, полученных в оптическом диапазоне, но с высокой детальностью при субпиксельной классификации объектов на гиперспектральных изображениях. Однако, при таком подходе возникает неопределенность, характеризаемая следующими особенностями: неоднозначностью классификации субпиксельных элементов на гиперспектральном снимке (какие спектральные характеристики относить к пикселям высокодетального изображения) и неоднозначностью сравнения спектральных характеристик с характеристиками из спектральных библиотек.

Для снижения неоднозначности классификации субпиксельных элементов перспективным представляется следующий подход:

- проведение кластеризации сегмента высокодетального изображения, соответствующего аналогичному МПО гиперспектрального изображения, с разделением на оптимальное число кластеров;

- использование результатов кластеризации для разделения спектральной характеристики пикселя гиперспектрального изображения (количество новых спектральных характеристик должно соответствовать количеству кластеров).

Для проведения кластеризации предполагается использование нечетких множеств второго типа (type 2 fuzzy set, T2FS), предложенных в [6]. Понятие нечеткого множества 2-го типа в алгоритме нечеткой кластеризации элементов изображения следует использовать следующим образом. Экспоненциальный вес m определяется не точечным значением, а некоторым «нечетким» значением из интервала (m_1, m_2) , т.е. $m \in (m_1, m_2)$ [8].

T2FS \tilde{A} представляется в форме нечеткой системы $\tilde{A} = \{((x, u), \mu_{\tilde{A}}(x, u)) \mid \forall X \in A,$

$$\forall u \in J_x \subseteq [0, 1], \mu_{\tilde{A}}(x, u) = 1\},$$

в которой первичная функция принадлежности J_{x_i} образца x_i представляется интервальной функцией принадлежности со всеми вторичными степенями первичных функций принадлежности, равными 1. Для задания интервальной первичной функции принадлежности определим нижнюю и верхнюю интервальные функции принадлежности при значениях фазификатора m_1, m_2 . Первичные функции принадлежности, представляющие образец x_i , определяются следующим образом:

$$\mu_j^{(B)}(x_i) = \begin{cases} \frac{1}{\sum_{l=1}^k \left(\frac{D_{ji}}{D_{li}}\right)^{\frac{2}{m_1-1}}}, & \text{если } \frac{1}{\sum_{l=1}^k \left(\frac{D_{ji}}{D_{li}}\right)^{\frac{2}{m_1-1}}} > \frac{1}{\sum_{l=1}^k \left(\frac{D_{ji}}{D_{li}}\right)^{\frac{2}{m_2-1}}} \\ \frac{1}{\sum_{l=1}^k \left(\frac{D_{ji}}{D_{li}}\right)^{\frac{2}{m_2-1}}}, & \text{в противном случае} \end{cases} \quad (1)$$

$$\mu_j^{(H)}(x_i) = \begin{cases} \frac{1}{\sum_{l=1}^k \left(\frac{D_{ji}}{D_{li}}\right)^{\frac{2}{m_1-1}}}, & \text{если } \frac{1}{\sum_{l=1}^k \left(\frac{D_{ji}}{D_{li}}\right)^{\frac{2}{m_1-1}}} \leq \frac{1}{\sum_{l=1}^k \left(\frac{D_{ji}}{D_{li}}\right)^{\frac{2}{m_2-1}}}; \\ \frac{1}{\sum_{l=1}^k \left(\frac{D_{ji}}{D_{li}}\right)^{\frac{2}{m_2-1}}}, & \text{в противном случае} \end{cases} \quad (2)$$

В (1) и (2) фаззификаторы m_1 и m_2 представляют различные нечеткие степени принадлежности, использование некоторых приводит к различным целевым функциям, которые должны быть минимизированы с помощью алгоритма FCM. В результате выполнения алгоритма для каждого кластера сегментируемого изображения определяются два значения центра кластера $V_j^{(H)}$ и $V_j^{(B)}$ (в пространстве признаков сегментируемого изображения), поэтому в модернизированный алгоритм FCM предлагается ввести следующую процедуру расчета центров кластеров:

$$V_j = \frac{V_j^{(H)} + V_j^{(B)}}{2}. \quad (3)$$

Модернизацию центров кластеров следует выполнять в ходе работы алгоритма FCM. Для этого используются операции «понижения типа» с помощью обобщенного центроида, применение которых подобно методу центра тяжести в базовом алгоритме кластеризации. Понижение типа во время поиска центров кластеров реализуется с помощью итерационного алгоритма Карника-Менделя [8]. Основным недостатком данного подхода приходится признавать резкое увеличение сложности вычислений и, как следствие, возрастание времени обработки, что особенно критично при обработке многозональных спутниковых снимков. В связи с этим предлагается следующая модификация алгоритма FCM: определяются два значения центров каждого кластера $V_j^{(H)}$ и $V_j^{(B)}$, $j = \overline{1, k}$. Полученные значения участвуют в модернизации центров кластера в соответствии с формулой (3). Подобная процедура модернизации центров кластеров будет более «грубой» по сравнению с предлагаемым в [8] использованием итерационного алгоритма Карника-Менделя. Алгоритм Карника-Менделя обеспечивает последовательную модернизацию центров кластеров, однако в основном (итерационном) цикле алгоритма FCM необходима реализация еще одного итерационного алгоритма. Подобное обстоятельство резко увеличивает время обработки. Предлагаемая модернизация алгоритма позволяет ускорить результаты обработки по сравнению с базовым алгоритмом FCM

(уступая по эффективности алгоритму из [8]), сохраняя сопоставимую сложность вычислений.

В качестве критерия оптимальности кластеризации предлагается использовать индекс Хие-Бени [9]. Индекс Хие-Бени представляет собой функцию следующего вида:

$$HB = \frac{\sum_{i=1}^k \sum_{j=1}^N \mu_{ij}^2 \|V_i - X_j\|^2}{N \cdot \min_{i,j} \|V_i - V_j\|^2}, \quad (4)$$

где k – число кластеров, N – число объектов, μ_{ij} – функция принадлежности, V_i – координаты центра i -го кластера, X_j – некоторый j -й объект.

С помощью индекса Хие-Бени можно оценивать качество нечеткой кластеризации: чем меньше значение HB , тем разбиение элементов на классы выполнено качественнее.

Применение указанной теории к задаче субпиксельной классификации позволит проводить более качественную классификацию элементов гиперспектрального изображения.

Литература

1. Ахметов Р. Н., Стратилатов Н.Р. Новые технологии анализа и обработки данных ДЗЗ// Аэрокосмический курьер №6 2011г.
2. Geoscience and remote sensing v.49, p.1 Special Issue 2011г.
3. Шовенгердт Р.А. Дистанционное зондирование. Методы и модели обработки изображений. М.: Техносфера - 560с.
4. Keshava N., Mustard J.F. Spectral unmixing Signal processing Magazine, IEEE v.19 2002г. С. 44-57
5. Licciardi G.A., Del Frate F. Pixel Unmixing in hyperspectral data by Means of Neural Networks// Geoscience and remote sensing v.49, p.1 Special Issue 2011г p. 4163-4172.
6. Mendel, J.M., Type-2 Fuzzy Sets and Systems and Overview // IEEE Computational Intelligence Magazine 2(1):20-29, February 2007.
7. А.Н. Пылькин, Р.В. Тишкин Методы и алгоритмы сегментации изображений. М.: Горячаялиния-Телеком, 2010 – 92с.
8. Mendel Uncertain Rule-Based fuzzy logic systems: introduction and new direction, Prentice Hall, upper saddle river, NJ, 2001
9. Xei X.L., Beni G.A. Validity Measure for Fuzzy Clustering // IEEE Trans. on Pattern Anal. and Machine Intell. 3 (8). - 1991. – p. 841 - 846.

SUBPIXEL CLASSIFICATION OF THE OBJECTS ON HYPERSPCTR SPACE IMAGE

Tishkin R.V., Udakov A.A.

In article various approaches to the solution problem of subpixel classification (spectral unmixing) on hyperspectral space image. Use of algorithm based on fuzzy logic for the solution of the specified task is offered.

МАТЕМАТИЧЕСКИЕ МОДЕЛИ И АЛГОРИТМЫ ФИЛЬТРАЦИИ ЦИФРОВЫХ ПОЛУТОНОВЫХ ИЗОБРАЖЕНИЙ НА ОСНОВЕ СЛОЖНЫХ ЦЕПЕЙ МАРКОВА

Петров Е.П., д.т.н., профессор, зав. кафедрой радиоэлектронных средств (РЭС) Вятского государственного университета, e-mail: EPPetrov@mail.ru

Харина Н.Л., к.т.н., доцент кафедры РЭС Вятского государственного университета, e-mail: NataL_res@mail.ru

Харюшин В.Ф., аспирант кафедры РЭС Вятского государственного университета, e-mail: vladimir_vgu@mail.ru

Ключевые слова: цифровые полутоновые изображения, математическая модель, марковский процесс, сложные цепи Маркова, нелинейная фильтрация.

Введение

Случайные процессы, у которых каждая выборка процесса зависит от определенного числа выборок расположенных в ближайшей окрестности от нее могут быть представлены сложной цепью Маркова со связностью $m \geq 1$ между исследуемой выборкой и выборками окрестности [1-3].

Выбор размера m сложной цепи Маркова, аппроксимирующей реальный процесс, зависит от скорости убывания автокорреляционной функции (АКФ), имеющей экспоненциальный или близкий к нему характер. Чем быстрее убывает АКФ с удалением от исследуемой выборки, тем меньше размер m . При $m=1$ сложная цепь Маркова переходит в простую цепь Маркова, хорошо изученную в радиотехнических и других приложениях [2-6]. Большинство работ, посвященных аппроксимации простой цепью Маркова не содержат оценки адекватности реального случайного процесса его модели, что может привести к потере части информации, иногда достаточно большой. В этом случае целесообразно использовать сложные цепи Маркова связности m , определяемой величиной допустимых потерь.

Для оценки потерь информации при выборе связности сложной цепи Маркова необходимо разработать математическую модель (ММ), адекватную реальному случайному процессу, и на ее основе, используя теорию фильтрации условных марковских процессов, синтезировать алгоритмы фильтрации в условиях действия белого гауссовского шума (БГШ).

Математическая модель сложной цепи Маркова

Пусть последовательность бинарных импульсных сигналов $\mu_1, \mu_2, \dots, \mu_k$ - сложная цепь Маркова с двумя равновероятными ($p_1 = p_2$) состояниями M_1 и M_2 , в которой каждое последующее состояние зависит от m ($m \geq 1$) предыдущих с матрицей вероятностей переходов (МВП) вида:

$$\Pi_m = \begin{vmatrix} \underbrace{\pi_{ii\dots ii}}_m & \underbrace{\pi_{ji\dots ii}}_m & \dots & \underbrace{\pi_{jj\dots ji}}_m \\ \underbrace{\pi_{ii\dots ij}}_m & \underbrace{\pi_{ji\dots ij}}_m & \dots & \underbrace{\pi_{jj\dots jj}}_m \end{vmatrix} \quad (1)$$

Получена математическая модель (ММ) цифрового полутонového изображения на основе сложной цепи Маркова m -й связности. В основу модели положен энтропийный подход и принцип суперпозиции m простых цепей Маркова. Используя ММ и теорию фильтрации условных марковских процессов, синтезированы алгоритмы нелинейной фильтрации.

где $i, j = \overline{1, 2}; i \neq j; m \geq 1$.

Нормированная АКФ случайного процесса, представленного сложной m -связной цепью Маркова, имеет вид (рис. 1).

Представим сложную цепь Маркова суперпозицией из m одномерных простых цепей Маркова [7, 8]. В этом случае элементы МВП могут быть вычислены из аргумента логарифма в формуле (2) взаимной информации между состояниями m -связной сложной цепи Маркова.

$$I(\mu_k) = \frac{\prod_{s=1}^m p(\mu_k | \mu_{k-s}) \cdot \prod_{a=3}^m p(\mu_k | \overbrace{\mu_{k-q}, \mu_{k-s}, \dots, \mu_{k-t}}^a)}{\prod_{\gamma=2}^m p(\mu_k | \overbrace{\mu_{k-s}, \mu_{k-t}}^{\gamma}) \cdot \prod_{\beta=4}^m p(\mu_k | \overbrace{\mu_{k-s}, \mu_{k-t}, \dots, \mu_{k-r}}^{\beta})} \quad (2)$$

где α - нечетное, β - четное число; $q, s, t, \dots, r = \overline{1, m}$. Знак \prod в (2) означает последовательность произведений для сомножителей с несовпадающими индексами в аргументах.

МВП m -связной сложной цепи Маркова с двумя состояниями имеет вид:

$$\Pi_m = \begin{vmatrix} \underbrace{\pi_{ii\dots ii}}_m & \underbrace{\pi_{ji\dots ii}}_m & \dots & \underbrace{\pi_{jj\dots ji}}_m \\ \underbrace{\pi_{ii\dots ij}}_m & \underbrace{\pi_{ji\dots ij}}_m & \dots & \underbrace{\pi_{jj\dots jj}}_m \end{vmatrix}^T = \begin{vmatrix} \alpha_1 & \alpha_1 & \dots & \alpha_1 \\ \alpha'_1 & \alpha'_1 & \dots & \alpha'_1 \end{vmatrix}^T \quad (3)$$

где $i, j = \overline{1, 2}; i \neq j; m > 1; T$ - знак транспонирования.

Потери информации за счет неадекватной аппроксимации случайного процесса простой цепью Маркова можно вычислить по формуле (2) при различных $m \geq 1$. На практике удобнее оперировать с оценками потерь в отношении сигнал/шум по мощности сигнала. Для этой цели необходимо, используя теорию фильтрации услов-

ных марковских процессов и разработанную ММ сложной цепи Маркова, синтезировать алгоритмы фильтрации случайных бинарных коррелированных сигналов при аппроксимации их сложными цепями Маркова с различной связностью ($m \geq 1$) [7, 8].

Выполнение заданных ограничений на потери в отношении сигнал/шум по мощности гарантирует адекватность реального случайного процесса алгоритму фильтрации наименьшими вычислительными ресурсами.

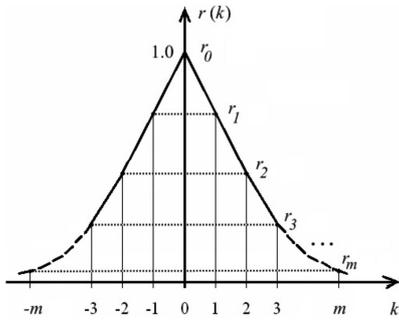
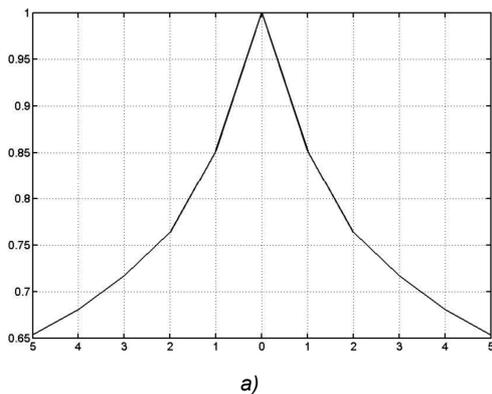


Рис. 1. Нормированная АКФ случайного процесса

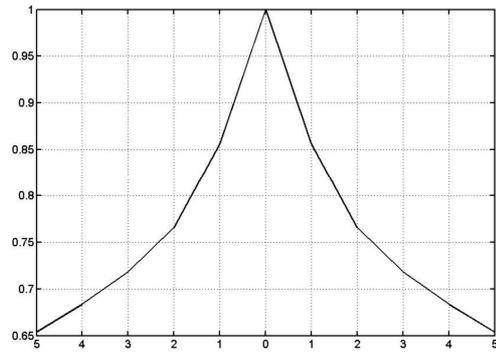


Рис. 2. 8-разрядное ЦПИ (473 × 990)

Оценим потери за счет упрощения аппроксимации реального случайного процесса простой цепью Маркова с двумя равновероятными состояниями. Для этого возьмем в качестве реального случайного процесса последовательность бинарных импульсных сигналов старшего разряда первой строки цифрового полутонового изображения (ЦПИ) (рис. 2), представленного 8-разрядными двоичными числами. Длина последовательности (строки) состоит из 938 импульсов. Автокорреляционная функция (АКФ) последовательности, представленная на рис. 3, имеет заметную корреляцию на интервале трех соседних импульсов по горизонтали и вертикали разрядного двоичного изображения (РДИ). Данный бинарный случайный процесс аппроксимируем одно-, двух- и трехсвязной цепью Маркова.



а)



б)

Рис. 3 а) АКФ старшего РДИ по горизонтали; б) АКФ старшего РДИ по вертикали

Алгоритмы одномерной фильтрации РДИ

На основе ММ (3) и теории фильтрации условных марковских процессов синтезированы алгоритмы фильтрации реального случайного процесса при трех аппроксимациях.

1. Односвязная цепь Маркова ($m = 1$):

$$u_k = [f_k(M_1) - f_k(M_2)] + u_{k-1} + z(u_{k-1}, {}^1\hat{\pi}_{ij}) \leq H \quad (4)$$

где

$$z(u_{k-1}, {}^q\hat{\pi}_{ij}) = \ln \frac{{}^q\hat{\pi}_{ii} + {}^q\hat{\pi}_{ji} \exp\{-u_{k-1}\}}{{}^q\hat{\pi}_{jj} + {}^q\hat{\pi}_{ij} \exp\{u_{k-1}\}} \quad (5)$$

${}^q\hat{\pi}_{ii} = 1 - {}^q\hat{\pi}_{ij}$ – оценки элементов МВП ($q = 1$;

$i, j = \overline{1, 2}$); $u_k = \ln \frac{P_{as}\{M_1(\mu_{k-1})\}}{P_{as}\{M_2(\mu_{k-1})\}}$ – логарифм отно-

шения апостериорных вероятностей состояний дискретного параметра импульсных сигналов; $[f_k(M_1) - f_k(M_2)]$ – разность логарифмов функций правдоподобия дискретного параметра бинарных сигналов; H – порог, вычисленный по критерию идеального наблюдателя [9].

2. Двусвязная ($m = 2$) сложная цепь Маркова:

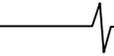
$$u_k = [f_k(M_1) - f_k(M_2)] + u_{k-1} + z(u_{k-1}, {}^1\hat{\pi}_{ij}) + u_{k-2} + z(u_{k-2}, {}^2\hat{\pi}_{ij}) - u_{k-2} - z(u_{k-2}, {}^3\hat{\pi}_{ij}) \leq H \quad (6)$$

где в функции (6) $q = \overline{1, 3}$ и ${}^3\Pi = {}^1\Pi \times {}^2\Pi$.

3. Трехсвязная ($m = 3$) сложная цепь Маркова:

$$u_k = [f_k(M_1) - f_k(M_2)] + u_{k-1} + z(u_{k-1}, {}^1\hat{\pi}_{ij}) + u_{k-2} + z(u_{k-2}, {}^2\hat{\pi}_{ij}) + u_{k-3} + z(u_{k-3}, {}^4\hat{\pi}_{ij}) + u_{k-3} + z(u_{k-3}, {}^7\hat{\pi}_{ij}) - u_{k-2} - z(u_{k-2}, {}^3\hat{\pi}_{ij}) - u_{k-3} - z(u_{k-3}, {}^5\hat{\pi}_{ij}) - u_{k-3} - z(u_{k-3}, {}^6\hat{\pi}_{ij}) \leq H \quad (7)$$

где в функции (7) $q = \overline{1, 7}$ и ${}^3\Pi = {}^1\Pi \times {}^2\Pi$, ${}^6\Pi = {}^1\Pi \times {}^4\Pi$, ${}^5\Pi = {}^2\Pi \times {}^4\Pi$, ${}^7\Pi = {}^1\Pi \times {}^2\Pi \times {}^4\Pi$.



В табл. 1 приведены результаты фильтрации реального случайного процесса, из которого следует, что при неудачном выборе аппроксимации реального процесса, потери в отношении сигнал/шум по мощности сигнала могут достигать значительной величины и тем большей, чем меньше отношение сигнал/шум на входе приемного устройства.

Однако одномерная цепь Маркова является ограниченной ММ, чтобы быть применимой к решению задач связанных с обработкой многомерных случайных процессов, например, изображений: статических – двумерная цепь Маркова, динамических – трех и большей размерности.

Таблица 1
Выигрыш при фильтрации ЦПИ на основе
одномерной сложной цепи Маркова различной связности

Зашумленный реальный процесс	$m = 1$	$m = 2$	$m = 3$
$\rho^2 = -6 \text{ дБ}$	7,47 дБ	8,55 дБ	8,81 дБ
$\rho^2 = -9 \text{ дБ}$	9,31 дБ	11,07 дБ	11,85 дБ
$\rho^2 = -12 \text{ дБ}$	10,84 дБ	13,63 дБ	14,85 дБ

ММ РДИ на основе простой двумерной цепи Маркова ($m = 1$)

Представим статическое ЦПИ набором РДИ, каждое из которых является двумерной цепью Маркова с двумя состояниями. Будем считать, что каждый элемент $\mu_{i,j}$ РДИ $\mu = \{\mu_{ij}, i, j \in \Phi_{n,k}\}$ (рис. 4) принадлежит двум независимым цепям Маркова с двумя равновероятными ($p_1 = p_2$) состояниями M_1, M_2 $i, j = \overline{1,2}; i \neq j$ и матрицами вероятностей переходов (МВП) из состояния M_i к соседнему состоянию M_j по горизонтали и вертикали БСП, соответственно:

$${}^1\Pi = \begin{pmatrix} {}^1\pi_{ii} & {}^1\pi_{ij} \\ {}^1\pi_{ji} & {}^1\pi_{jj} \end{pmatrix},$$

$${}^2\Pi = \begin{pmatrix} {}^2\pi_{ii} & {}^2\pi_{ij} \\ {}^2\pi_{ji} & {}^2\pi_{jj} \end{pmatrix}.$$

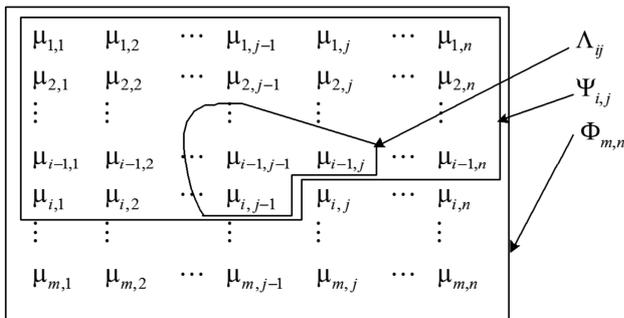


Рис. 4. Фрагмент ММ РДИ

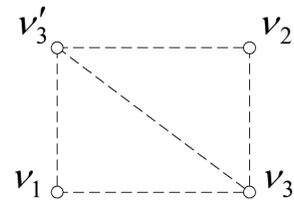


Рис. 5. Фрагмент области $\Lambda_{i,j}$ РДИ, где приняты обозначения $v_1 = \mu_{i,j-1}; v_2 = \mu_{i-1,j}; v_3 = \mu_{i,j}; v_3' = \mu_{i-1,j-1}$;

Если известны коэффициенты корреляции по строкам r_i^e и столбцам $r_i^e (i = \overline{1,m})$ РДИ, то элементы МВП (8) можно определить по формулам:

$${}^1\pi_{ii} = \frac{1+r_i^e}{2}, \quad {}^2\pi_{ii} = \frac{1+r_i^e}{2},$$

$${}^k\pi_{ij} = 1 - {}^k\pi_{ji}; \quad i, j = \overline{1,2}; \quad i \neq j$$

Если условная зависимость бинарных импульсных сигналов РДИ определена от левого верхнего сегмента, то сигнал с параметром $\mu_{i,j}$ зависит от случайных сигналов только некоторого подмножества $\Lambda_{i,j}$ этого сегмента, называемого окрестностью (рис. 5).

$$\Lambda_{i,j} = \left\{ \mu_{i,j-1}, \mu_{i-1,j}, \mu_{i-1,j-1} \right\} = \left\{ v_1, v_2, v_3' \right\}.$$

Для односвязной ($m = 1$) двумерной цепи Маркова окрестность $\Lambda_{i,j}$ примет вид (рис. 5):

$$\Lambda_{i,j} = \left\{ \mu_{i,j-1}; \mu_{i-1,j}; \mu_{i-1,j-1} \right\} = \left\{ v_1, v_2, v_3' \right\}.$$

Вероятности перехода состояний элементов окрестности $\Lambda_{i,j}$ относительно элемента v_3 образуют МВП вида:

$$\Pi = \begin{pmatrix} \pi_{iii} & \pi_{ijj} \\ \pi_{iji} & \pi_{ijj} \\ \pi_{jii} & \pi_{jjj} \\ \pi_{jji} & \pi_{jjj} \end{pmatrix} = \begin{pmatrix} \alpha_1 & \alpha_1' \\ \alpha_2 & \alpha_2' \\ \alpha_3 & \alpha_3' \\ \alpha_4 & \alpha_4' \end{pmatrix}, \quad i, j = \overline{1,2}; \quad i \neq j$$

Элементы матрицы Π (12) связаны с элементами матриц (8) следующими соотношениями:

$$\alpha_1 = \pi_{iii} = \pi(v_3 = M_i | v_1 = M_i; v_2 = M_i) = 1 - \frac{{}^1\pi_{ij} \cdot {}^2\pi_{ij}}{{}^3\pi_{ii}}$$

$$\alpha_1' = \pi_{ijj} = \pi(v_3 = M_j | v_1 = M_i; v_2 = M_i) = \frac{{}^1\pi_{ij} \cdot {}^2\pi_{ij}}{{}^3\pi_{ii}}$$

где ${}^3\pi_{ij}$ $i, j = \overline{1,2}; i \neq j$ - элемент дополнительной матрицы

$${}^3\Pi = {}^1\Pi \times {}^2\Pi = \begin{pmatrix} {}^3\pi_{ii} & {}^3\pi_{ij} \\ {}^3\pi_{ji} & {}^3\pi_{jj} \end{pmatrix}$$

Элементы строк матрицы Π удовлетворяют условию нормировки:

$$\alpha_l + \alpha_l' = 1, \quad l = \overline{1,4}$$

**ММ РДИ на основе сложной
двумерной цепи Маркова $m = 2$**

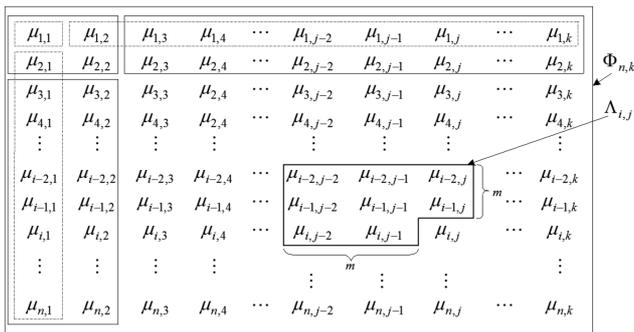


Рис. 6. Фрагмент двумерной ММ РДИ со связностью ($m = 2$)

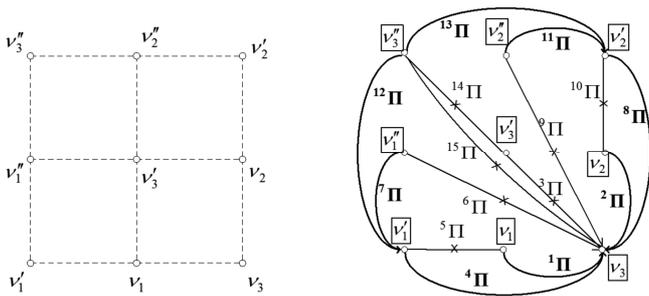


Рис. 7. Фрагмент области $\Lambda_{i,j}$ (рис. 6) с МВП

Для сложной двусвязной двумерной цепи Маркова ($m = 2$) окрестность, предшествующих формируемому состоянию v_3 , примет вид (рис. 7):

$$\Lambda_{i,j} = (v_1, v_2, v'_3, v'_1, v''_1, v'_2, v''_2, v''_3) \quad (16)$$

В этом случае корреляция ограничивается коэффициентом r_2^c (рис. 1) по горизонтали и r_2^e по вертикали.

Представим двусвязную двумерную цепь Маркова как суперпозицию двух двусвязных сложных цепей Маркова с МВП:

$$\begin{aligned}
 1^{\Pi} &= \begin{vmatrix} 1\pi_{ii} & 1\pi_{ij} \\ 1\pi_{ji} & 1\pi_{jj} \end{vmatrix}, \\
 4^{\Pi} &= \begin{vmatrix} 4\pi_{ii} & 4\pi_{ij} \\ 4\pi_{ji} & 4\pi_{jj} \end{vmatrix}, \\
 2^{\Pi} &= \begin{vmatrix} 2\pi_{ii} & 2\pi_{ij} \\ 2\pi_{ji} & 2\pi_{jj} \end{vmatrix}, \\
 8^{\Pi} &= \begin{vmatrix} 8\pi_{ii} & 8\pi_{ij} \\ 8\pi_{ji} & 8\pi_{jj} \end{vmatrix}.
 \end{aligned} \quad (17)$$

$$\begin{aligned}
 1\pi_{ii} &= \frac{1+r_1^c}{2}, & 4\pi_{ii} &= \frac{1+r_2^c}{2}, \\
 2\pi_{ii} &= \frac{1+r_1^e}{2}, & 8\pi_{ii} &= \frac{1+r_2^e}{2}, \\
 k\pi_{ij} &= 1 - k\pi_{ii}; & i, j &= \overline{1,2}; \quad i \neq j
 \end{aligned} \quad (18)$$

Тогда МВП двусвязной двумерной цепи Маркова примет вид:

$$\Pi = \begin{vmatrix} \pi_{iiii} & \pi_{iiij} \\ \pi_{iiij} & \pi_{iiij} \\ \vdots & \vdots \\ \pi_{jjjji} & \pi_{jjjji} \end{vmatrix} = \begin{vmatrix} \alpha_1 & \alpha'_1 \\ \alpha_2 & \alpha'_2 \\ \vdots & \vdots \\ \alpha_{16} & \alpha'_{16} \end{vmatrix}; \quad i, j = \overline{1,2}; \quad i \neq j. \quad (19)$$

Элементы матрицы Π (19) связаны с элементами матриц (18) следующими соотношениями:

$$\begin{aligned}
 \alpha_1 &= \pi_{iiii} = \pi(v_3 = M_i | v_1 = M_i; v_2 = \\
 &= M_i; v'_1 = M_i; v'_2 = M_i) = \\
 &= 1 - \frac{1\pi_{ij} \cdot 2\pi_{ij} \cdot 4\pi_{ij} \cdot 8\pi_{ij} \cdot 8\pi_{ij} \cdot 7\pi_{ij} \cdot 11\pi_{ij} \cdot 12\pi_{ij} \cdot 13\pi_{ij}}{3\pi_{ii} \cdot 5\pi_{ii} \cdot 6\pi_{ii} \cdot 9\pi_{ii} \cdot 10\pi_{ii} \cdot 15\pi_{ii} \cdot 14\pi_{ii}}, \quad (20)
 \end{aligned}$$

$$\begin{aligned}
 \alpha'_1 &= \pi_{iiij} = \pi(v_3 = M_j | v_1 = M_i; v_2 = \\
 &= M_i; v'_1 = M_i; v'_2 = M_i) = \\
 &= \frac{1\pi_{ij} \cdot 2\pi_{ij} \cdot 4\pi_{ij} \cdot 8\pi_{ij} \cdot 7\pi_{ij} \cdot 7\pi_{ij} \cdot 11\pi_{ij} \cdot 12\pi_{ij} \cdot 13\pi_{ij}}{3\pi_{ii} \cdot 5\pi_{ii} \cdot 6\pi_{ii} \cdot 9\pi_{ii} \cdot 10\pi_{ii} \cdot 15\pi_{ii} \cdot 14\pi_{ii}},
 \end{aligned}$$

где

$$\begin{aligned}
 3^{\Pi} &= 1^{\Pi} \times 2^{\Pi}, \quad 5^{\Pi} = 1^{\Pi} \times 4^{\Pi}, \quad 6^{\Pi} = \\
 &= 2^{\Pi} \times 4^{\Pi}, \quad 7^{\Pi} = 3^{\Pi} \times 4^{\Pi} = 1^{\Pi} \times 2^{\Pi} \times 4^{\Pi}, \\
 9^{\Pi} &= 1^{\Pi} \times 8^{\Pi}, \quad 10^{\Pi} = 2^{\Pi} \times 8^{\Pi}, \quad 11^{\Pi} = \\
 &= 3^{\Pi} \times 8^{\Pi} = 1^{\Pi} \times 2^{\Pi} \times 8^{\Pi},
 \end{aligned} \quad (21)$$

$$\begin{aligned}
 12^{\Pi} &= 5^{\Pi} \times 8^{\Pi} = 1^{\Pi} \times 4^{\Pi} \times 8^{\Pi}, \quad 13^{\Pi} = \\
 &= 6^{\Pi} \times 8^{\Pi} = 2^{\Pi} \times 4^{\Pi} \times 8^{\Pi}, \quad 15^{\Pi} = 4^{\Pi} \times 8^{\Pi}, \\
 14^{\Pi} &= 7^{\Pi} \times 8^{\Pi} = 1^{\Pi} \times 2^{\Pi} \times 4^{\Pi} \times 8^{\Pi}.
 \end{aligned}$$

Остальные элементы матрицы Π (19) вычисляются аналогично в соответствии с состояниями четырех бинарных импульсов в точках v_1, v'_1, v_2, v'_2 окрестности $\Lambda_{i,j}$ (16).

Для оценки потерь вызванных аппроксимацией возьмем РДИ старшего разряда реального 8-разрядного ЦПИ (рис. 2). АКФ по горизонтали (рис. 3а) и вертикали (рис. 3б) РДИ имеют заметные значения на интервале $m = 3$, что позволяет аппроксимировать РДИ сложной цепью Маркова со связностью $m = 3$.

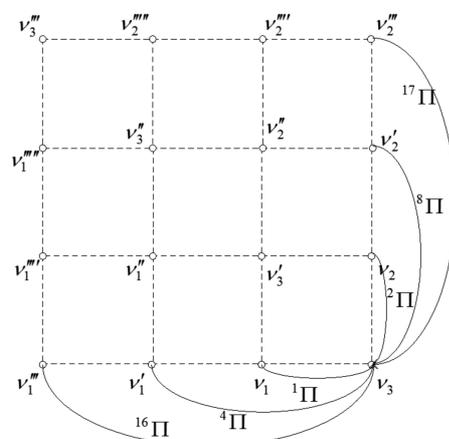
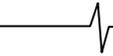


Рис. 8. Фрагмент области $\Lambda_{i,j}$ (рис. 6) со связностью $m = 3$ и МВП (19)



Для сложной трехсвязной двумерной цепи Маркова ($m = 3$) окрестность, предшествующих формируемому состоянию v_3 , примет вид (рис. 8):

$$\Lambda_{i,j} = (v_1, v_2, v_3, v'_1, v''_1, v'_2, v''_2, v'_3, v''_3, v'''_1, v''''_1, v'''_2, v''''_2, v'''_3, v''''_3) \quad (22)$$

Алгоритмы нелинейной фильтрации РДИ и ЦПИ

Используя ММ и теорию фильтрации условных марковских процессов построим алгоритмы фильтрации случайного бинарного коррелированного процесса при наличии белого гауссовского шума $n(t)$ с нулевым средним и дисперсией σ_n^2 при трех аппроксимациях.

1. Аппроксимация простой двумерной цепью Маркова ($m = 1$). Алгоритм фильтрации бинарных импульсных сигналов синтезирован в [7] и имеет вид:

$$u(v_3) = [f_{v_3}(M_1) - f_{v_3}(M_2)] + u(v_1) + z(u(v_1), \hat{\pi}_{ij}^1) + u(v_2) + z(u(v_2), \hat{\pi}_{ij}^2) - u(v'_3) - z(u(v'_3), \hat{\pi}_{ij}^3) \leq H \quad (23)$$

где

$$z(u_{k-1}, \hat{\pi}_{ij}^q) = \ln \frac{q \hat{\pi}_{ii} + q \hat{\pi}_{ji} \exp\{-u_{k-1}\}}{q \hat{\pi}_{jj} + q \hat{\pi}_{ij} \exp\{u_{k-1}\}} \quad (24)$$

$q \hat{\pi}_{ij} = 1 - q \hat{\pi}_{ji}$ - оценки элементов МВП ($q = 1; i, j = \overline{1, 2}$); $u_k = \ln \frac{P_{as}\{M_1(\mu_{k-1})\}}{P_{as}\{M_2(\mu_{k-1})\}}$ - логарифм отношения апостериорных вероятностей состояний дискретного параметра импульсных сигналов; $[f_k(M_1) - f_k(M_2)]$ - разность логарифмов функций правдоподобия дискретного параметра бинарных сигналов; H - порог вычисленный по критерию идеального наблюдателя [9].

2. Аппроксимация двумерной двусвязной цепью Маркова ($m = 2$). Алгоритм фильтрации бинарных импульсных сигналов имеет вид:

$$u(v_3) = [f_{v_3}(M_1) - f_{v_3}(M_2)] + u(v_1) + z(u(v_1), \hat{\pi}_{ij}^1) + u(v_2) + z(u(v_2), \hat{\pi}_{ij}^2) + u(v'_1) + z(u(v'_1), \hat{\pi}_{ij}^4) + u(v'_2) + z(u(v'_2), \hat{\pi}_{ij}^8) + u(v''_1) + z(u(v''_1), \hat{\pi}_{ij}^7) + u(v''_2) + z(u(v''_2), \hat{\pi}_{ij}^{11}) + u(v''_3) + z(u(v''_3), \hat{\pi}_{ij}^{12}) + u(v''_3) + z(u(v''_3), \hat{\pi}_{ij}^{13}) - u(v'_1) - z(u(v'_1), \hat{\pi}_{ij}^5) - u(v'_2) - z(u(v'_2), \hat{\pi}_{ij}^{10}) - u(v''_1) - z(u(v''_1), \hat{\pi}_{ij}^6) - u(v''_2) - z(u(v''_2), \hat{\pi}_{ij}^9) - u(v''_3) - z(u(v''_3), \hat{\pi}_{ij}^{14}) - u(v''_3) - z(u(v''_3), \hat{\pi}_{ij}^{15}) - u(v'_3) - z(u(v'_3), \hat{\pi}_{ij}^3) \leq H, \quad (25)$$

$$\begin{aligned} &\text{где в функции (25) } q = \overline{1, 15} \text{ и } {}^3\Pi = {}^1\Pi \times {}^2\Pi, \\ &{}^5\Pi = {}^1\Pi \times {}^4\Pi, \quad {}^6\Pi = {}^2\Pi \times {}^4\Pi, \\ &{}^7\Pi = {}^3\Pi \times {}^4\Pi = {}^1\Pi \times {}^2\Pi \times {}^4\Pi, \quad {}^9\Pi = {}^1\Pi \times {}^8\Pi, \\ &{}^{10}\Pi = {}^2\Pi \times {}^8\Pi, \quad {}^{11}\Pi = {}^3\Pi \times {}^8\Pi = {}^1\Pi \times {}^2\Pi \times {}^8\Pi, \\ &{}^{12}\Pi = {}^5\Pi \times {}^8\Pi = {}^1\Pi \times {}^4\Pi \times {}^8\Pi, \\ &{}^{13}\Pi = {}^6\Pi \times {}^8\Pi = {}^2\Pi \times {}^4\Pi \times {}^8\Pi, \quad {}^{15}\Pi = {}^4\Pi \times {}^8\Pi, \\ &{}^{14}\Pi = {}^7\Pi \times {}^8\Pi = {}^1\Pi \times {}^2\Pi \times {}^4\Pi \times {}^8\Pi. \end{aligned}$$

Алгоритм нелинейной фильтрации РДИ (25) можно упростить, сократив одинаковые слагаемые, а функции $z(\cdot)$ объединить под одним логарифмом.

3. Аппроксимация двумерной трехсвязной цепью Маркова ($m = 3$) синтезируется аналогично алгоритму (25), но из-за громоздкости записи не приводится. Алгоритм содержит 32 слагаемых со знаком «плюс» и 31 слагаемое со знаком «минус».

Учитывая, представление ЦПИ набором 8-и РДИ, было проведено моделирование реального ЦПИ при трех аппроксимациях сложной цепью Маркова со связностью $m = 1, 2, 3$. Результаты фильтрации реального ЦПИ искаженного БГШ $n(t)$ с нулевым средним и дисперсией σ_n^2 при отношении сигнал/шум по мощности

$\rho^2 = -6\text{дБ}, -9\text{дБ}, -12\text{дБ}$ приведены в табл. 3. Выигрыш вычислялся суммированием выигрышей, полученных для каждого РДИ:

$$\eta = \sum_{l=1}^g 10 \lg \left(\rho_{\text{вых}}^{(l)2} / \rho_{\text{вх}}^2 \right) \text{дБ},$$

где $\rho_{\text{вх}}^2, \rho_{\text{вых}}^{(l)2}$ - отношение сигнал/шум в l -ом РДИ на входе и выходе нелинейного фильтра, т.е. подключенного к выходу детектора, реализующего разность логарифмов функции правдоподобия.

Заключение

Результаты данной работы позволяют правильно выбрать связность сложной цепи Маркова, аппроксимирующей реальный случайный процесс, оценить степень адекватности его ММ, объем ресурсов, необходимых для реализации алгоритмов фильтрации.

Литература

1. Марков А.А. Избранные труды: Теория чисел. Теория вероятностей/ Под ред. проф. Ю.В.Линника.- М.: Изд-во академии наук, 1951. - 465 с.
2. Ching Wai-Ki, Michael K. Ng. Markov Chains: Models, Algorithms and Applications.- Springer Science+Business Media, Inc., 2006. - 211p.
3. Королук В.С., Турбин А.Ф. Полумарковские процессы и их приложения. - Киев: Изд. Наукова думка», 1986. - 182 с.
4. Амиантов И.Н. Избранные вопросы статистической теории связи. - М.: Сов. радио, 1971, - 416 с.
5. Петров Е.П., Харина Н.Л., Кононова В.Ю. Нелинейная фильтрация нестационарных цифровых случай-

ных полей. Т-Сотм. Телекоммуникации и транспорт. № 5, 2011. – С. 18-22.

6. Петров Е.П., Медведева Е.В., Харина Н.Л. Модели и алгоритмы цифровой обработки изображений: учебное пособие. – Киров: О-Краткое, 2008. – 88 с.

7. Петров Е.П., Харина Н.Л. Математическая модель последовательностей бинарных импульсных сигналов на основе сложных цепей Маркова// Сб. докл. XVIII МНТК

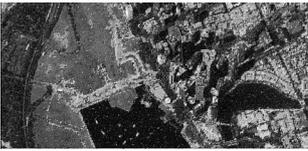
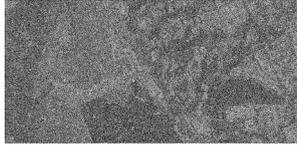
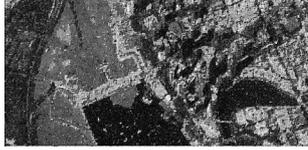
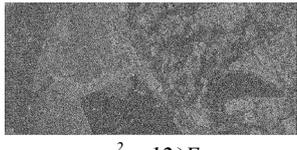
«Радиолокация, навигация, связь»: - Воронеж, т.1, 2012.- С.1-8.

8. Харина Н.Л. Математическая модель двоичных изображений на основе сложных цепей Маркова// Сб. докл. XVIII МНТК «Радиолокация, навигация, связь»: - Воронеж, т.1, 2012.- С.147-154.

9. Тихонов В.И. Статистическая радиотехника. - М.: Сов. радио, 1966, 679 с.

Таблица 2

Выигрыш при фильтрации ЦПИ на основе двумерной сложной цепи Маркова различной связности m

	$m = 1$	$m = 2$	$m = 3$
 $p^2 = -6 \text{ dB}$	 10,91 дБ	 11,39 дБ	 11,44 дБ
 $p^2 = -9 \text{ dB}$	 14,60 дБ	 15,64 дБ	 15,72 дБ
 $p^2 = -12 \text{ dB}$	 18,23 дБ	 20,30 дБ	 20,39 дБ

MATHEMATICAL MODELS AND ALGORITHMS OF THE FILTRATION OF DIGITAL HALF-TONE IMAGES ON THE BASIS OF THE COMPLICATED MARKOV CHAINS

Petrov E.P., Harina N.L., Kharushin V.F.

The mathematical model (MM) of digital halftone images, which is bases on the complicated Markov chain of m order

is received. The information approach and the principle of m superposition of simple Markov chains of is taken as the basis of the model. Using MM and the theory of filtration of causal Markov processes, algorithms of a nonlinear filtration are synthesized. Results of the given work allow to choose correctly order of the complicated Markov chain approximating real casual process, to estimate a degree of adequacy its to MM, volume of the resources necessary for realization of algorithms of a filtration.



Зеленоградскими предприятиями **ОАО НПЦ «ЭЛВИС»** и **ОАО «АНГСТРЕМ-Т»** при участии **ООО «НЕЛС»** разработана и освоена в серийном производстве новая система на кристалле – **СИГНАЛЬНЫЙ ПРОЦЕССОР 1892BM10A** со встроенной функцией **ГЛОНАСС/GPS** навигации и пониженным энергопотреблением.

Подробнее: <http://multicore.ru/index.php?id=1149>



IJEEE – Международный журнал по проблемам образования в области электротехники. Специальный выпуск *IJEEE* №3, июль 2012, выпуск 49 посвящен проблемам обучения в области реализации систем цифровой обработки сигналов.

Подробнее: <http://www.eee.manchester.ac.uk/aboutus/ijeee/>

МЕТОД БЫСТРОГО ВЫЧИСЛЕНИЯ LBP-ПРИЗНАКОВ ОБЛАСТИ ИНТЕРЕСА ВИДЕОИЗОБРАЖЕНИЯ

Толочко А.М., аспирант кафедры СиУТ Белорусского государственного университета информатики и радиоэлектроники, e-mail: Aljaksandr@gmail.com

Борискевич А.А., к.т.н., доцент кафедры СиУТ Белорусского государственного университета информатики и радиоэлектроники, e-mail: anbor@bsuir.by

Ключевые слова: видеоизображение, текстура, LBP-признаки, быстрые вычисления, примитивы, локальная окрестность.

Введение

Одной из важных характеристик визуальной информации, используемой зрительной системой человека и современными системами машинного зрения, является текстура. Она является характеристикой пространственной организации изменения интенсивности поверхности объекта, количественно определяющей такие свойства, как гладкость, грубость и регулярность. Основными подходами к описанию текстуры являются статистические, спектральные и структурные методы [1].

Среди множества известных методов статистического описания текстуры локальной области изображения следует выделить группу методов, основанных на вычислении LBP-признаков (Local Binary Pattern) [2, 3]. Устойчивость к изменениям условий освещения, простота реализации и компактность представления базового LBP-признака определили появление большого количества его модификаций, обладающих новыми свойствами: инвариантность к повороту ($LBP_{8,1}^{riu2}$ [4]), устойчивость к аддитивному шуму (TLBP [5]) и др.

Различные модификации LBP-признака, используемые для описания текстуры области изображения, находят широкое применение в различных прикладных задачах: обнаружение и распознавание лиц [6]; анализ, сегментация и классификация текстур аэрокосмических, промышленных и биомедицинских изображений [7]; моделирование заднего плана и обнаружение динамических объектов [8]; сопровождение динамических объектов в видеопотоке [9].

Для эффективного использования вычислительных ресурсов в современных системах обработки и анализа видеоизображения в режиме реального времени актуальной становится задача уменьшения времени вычисления текстурных признаков. В связи с этим, целью работы является разработка метода быстрого вычисления LBP-признаков.

Базовые методы формирования LBP-признаков

Базовый вариант $LBP_{8,1}$ [4] представляет собой 8-битное кодовое слово, бинарное представление которого основано на анализе круговой окрестности единич-

Предложен метод быстрого вычисления LBP-признаков изображения, основанный на выделении примитива регулярной вычислительной структуры, установлении свойства взаимности пороговой и модифицированной пороговой функций, определяющих правила сравнения соседних пикселей локальной окрестности, и правила обновления LBP-признаков. Он позволяет получить выигрыш в скорости вычисления на уровне 66-72% для инвариантного к повороту LBP-признака с размером примитива 3x3 по отношению к традиционному методу за счёт сокращения удельного количества операций сравнения и пикселей, к которым осуществляется доступ при вычислении LBP-признака.

ного радиуса кодируемого пикселя изображения. $LBP_{8,1}$ - признак вычисляется в соответствии с выражением:

$$LBP_{8,1} = \sum_{i=0}^8 g(I_p, I_c) 2^i, \quad (1)$$

где I_c - значение интенсивности центрального пикселя в окрестности, для которого рассчитывается значение $LBP_{8,1}$ -признака; I_p - значение интенсивности p -го элемента локальной круговой окрестности, $p = 0, 1, \dots, 7$.

Пороговая функция $g(I_p, I_c)$ значений интенсивностей элемента окрестности I_p и центрального пикселя I_c определяется выражением:

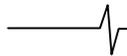
$$g(I_p, I_c) = \begin{cases} 1, & \text{при } I_p \geq I_c, \\ 0, & \text{при } I_p < I_c. \end{cases} \quad (2)$$

Локальная круговая окрестность центрального пикселя I_c представлена на рис. 1. Пусть I_c определяет значение интенсивности элемента изображения с координатами (x_c, y_c) ($I_c = \text{Img}(x_c, y_c)$), тогда координаты (x_p, y_p) p -го элемента круговой окрестности для формирования $LBP_{8,1}$ -признака задаются формулой:

$$\begin{aligned} x_p &= x_c + \sin(\pi(p-1)/4), \\ y_p &= y_c - \cos(\pi(p-1)/4). \end{aligned} \quad (3)$$

Значения интенсивностей элементов круговой окрестности, имеющих нецелочисленные координаты, рассчитываются посредством билинейной интерполяции. В соответствии с формулой (1), $LBP_{8,1}$ -признак может принимать одно из 256 значений из диапазона $[0, 255]$.

$LBP_{8,1}^{u2}$ -признак уменьшает пространство возможных значений $LBP_{8,1}$ -признака, используя свойство равно-



мерности (uniformity) локального паттерна. Степень равномерности $U(LBP_{8,1})$ [4], характеризующая количество переходов 0/1, 1/0 в циклическом представлении $LBP_{8,1}$ -признака, вычисляется с помощью соотношения:

$$U(LBP_{8,1}) = |g(I_7, I_c) - g(I_0, I_c)| + \sum_{p=1}^7 |g(I_{p-1}, I_c) - g(I_p, I_c)| \quad (4)$$

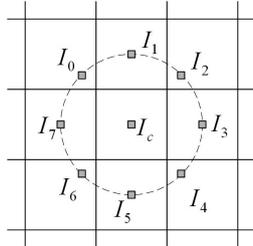


Рис. 1. Локальная круговая окрестность для формирования $LBP_{8,1}$ -признака центрального пикселя I_c

Равномерными считаются паттерны, имеющие не более двух бинарных переходов в циклическом представлении $LBP_{8,1}$ -признака. В случае невыполнения данного требования паттерн считается неравномерным. Замечено, что для реальных изображений почти 90% $LBP_{8,1}$ -признаков обладают свойством равномерности.

Выражение для вычисления $LBP_{8,1}^{u2}$ признака можно представить в виде

$$LBP_{8,1}^{u2} = \begin{cases} 0, & \text{при } LBP_{8,1} = 0; \\ 57, & \text{при } LBP_{8,1} = 255; \\ 8 \cdot \sum_{p=0}^7 g(I_p, I_c) - 7 + \lg_2(\text{idx}), & \text{при } LBP_{8,1} \neq 0, 255; \\ 58, & \text{при } U(LBP_{8,1}) > 2, \end{cases} \quad \text{при } U(LBP_{8,1}) \leq 2, \quad (5)$$

Здесь idx является решением модулярного уравнения вида:

$$\left(2^{\sum_{p=0}^7 g(I_p, I_c)} - 1 \right) \text{idx} \equiv LBP_{8,1} \pmod{255} \quad (6)$$

$LBP_{8,1}^{u2}$ -признак может принимать одно из 59 возможных значений. Кодирование группы равномерных $LBP_{8,1}^{u2}$ -признаков с одинаковой мощностью (количество единиц в бинарном представлении) только одним кодовым словом позволяет добиться инвариантности к повороту $LBP_{8,1}$ -признака. Выражение для расчёта инвариантного к повороту $LBP_{8,1}^{riu2}$ -признака имеет вид:

$$LBP_{8,1}^{riu2} = \begin{cases} \sum_{p=0}^7 g(I_p, I_c), & \text{при } U(LBP_{8,1}) \leq 2 \\ 9, & \text{при } U(LBP_{8,1}) > 2 \end{cases} \quad (7)$$

$LBP_{8,1}^{u2}$ и $LBP_{8,1}^{riu2}$ признаки естественным образом выявляют типовые особенности текстуры: точки, окончания линий, углы, края.

Круговой характер топологии анализируемой окрестности является условием обеспечения инвариантности $LBP_{8,1}^{riu2}$ -признака к повороту [4]. Вместе с тем широко применяется базовая версия LBP -признака [2], использующая квадратную топологию окрестности (рис. 2).

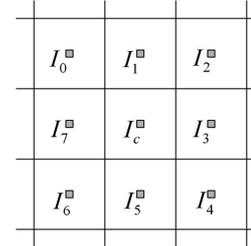


Рис. 2. Локальная квадратная окрестность для формирования LBP -признака центрального пикселя I_c

Использование квадратной топологии окрестности позволяет избежать вычислительно ёмкой процедуры расчёта интерполированных значений её элементов с нецелочисленными координатами. Таким образом, значения элементов квадратной окрестности представляют собой значения пикселей соседей центрального пикселя I_c в окрестности 3×3 пикселя. Выражения для расчёта LBP -признака аналогичны выражениям для расчёта $LBP_{8,1}$ -признака (1, 2).

Изменение пороговой функции (2) посредством сдвига порога на величину c позволяет обеспечить устойчивость признака к шуму. Пороговая функция $g_t(I_p, I_c)$ для модифицированного LBP -признака – Threshold LBP ($TLBP$) [4] – определяется выражением:

$$g_t(I_p, I_c) = \begin{cases} 1, & \text{при } I_p + c \geq I_c, \\ 0, & \text{при } I_p + c < I_c. \end{cases} \quad (8)$$

Метод повышения устойчивости признака к шуму посредством изменения пороговой функции (8) применим к представленным модификациям LBP -признака, образуя соответственно $TLBP_{8,1}$, $TLBP_{8,1}^{u2}$, $TLBP_{8,1}^{riu2}$ признаки.

Метод быстрого вычисления LBP -признаков

LBP -признаки с круговой топологией локальной окрестности, несмотря на ряд достоинств, обладают существенным недостатком, препятствующим реализации его быстрого вычисления: имеет место необходимость вычисления интерполированных значений элементов окрестности кодируемого пикселя, имеющих нецелочисленные координаты. Напротив, LBP -признаки с квадратной топологией локальной окрестности не требуют осуществления интерполяции значений элементов окрестности. При этом для базового варианта LBP -признака также применимо понятие равномерности паттерна, что позволяет сформировать LBP^{u2} и LBP^{riu2} -признаки.

Квадратная топология локальной окрестности позволяет реализовать метод быстрого вычисления LBP -признака. Метод быстрого вычисления LBP -признаков ($F-LBP$, Fast Local Binary Pattern) использует свойство взаимности пороговой функции: при известном результате операции сравнения аргументов a, b для пороговой функции $g(a, b)$ значение функции $g(b, a)$ также известно. Для расширенного представления пороговой функции (2) свойство взаимности может быть представ-

лено в форме табл. 1. Из табл. 1 видно, что каждому результату операции сравнения аргументов a, b из трёх возможных исходов ($a > b$, $a = b$, $a < b$) соответствует пара бинарных значений пороговых функций $g(a, b)$, $g(b, a)$. Таким образом, операция сравнения для получения значений пороговых функций $g(a, b)$, $g(b, a)$ при заданных аргументах a, b осуществляется единожды.

Таблица 1
Свойство взаимности пороговой функции $g(a, b)$

Результат операции сравнения аргументов a, b	Значение пороговой функции	
	$g(a, b)$	$g(b, a)$
$a > b$	1	0
$a = b$	1	1
$a < b$	0	1

Свойство взаимности для расширенного представления пороговой функции (8) $g_i(a, b)$, используемой для формирования модифицированных вариантов F -LBP-признаков (F -TLBP, F -TLBP^{u2}, F -TLBP^{riu2}), при положительных значениях порога c представлено в табл. 2. Следует отметить, что при нулевом значении порога $c = 0$ пороговые функции $g_i(a, b)$ и $g(a, b)$ идентичны.

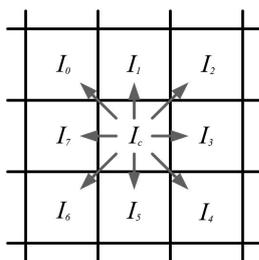


Рис. 3. Направление векторов вычисления операций сравнения в локальной окрестности

Использование свойства взаимности пороговой функции позволяет сократить удельное количество операций сравнения на один пиксель при вычислении LBP-признаков изображения (или его фрагмента). На рис. 3 представлена локальная окрестность размером 3x3 пикселей с указанием векторов вычислений операций сравнения в соответствии с пороговой функцией (от кодируемого центрального пикселя к элементам его окрестности). В соответствии с рис. 3 при вычислении LBP-признака центрального пикселя I_c требуется выполнение $N_{comp}^{LBP} = 8$ процедур сравнения и осуществление доступа к $N_{access}^{LBP} = 9$ элементам изображения (8 элементов локальной окрестности и, собственно, сам кодируемый пиксель). Несложно заметить, что выполнение операции сравнения пороговой функции дублируется: каждая из 8 операций сравнения осуществляется при расчёте LBP-признака центрального элемента анализируемого паттерна, а также при расчёте LBP-признака соответствующего элемента локальной окрестности.

Повышение скорости вычисления LBP-признаков изображения (или его фрагмента) достигается посред-

ством выделения примитива регулярной вычислительной структуры за счёт двух влияющих факторов:

- снижение удельного количества процедур сравнения на один пиксель;
- снижение удельного количества элементов изображения, к которым осуществляется доступ для формирования F -LBP-признака.

Таблица 2
Свойство взаимности модифицированной пороговой функции $g_i(a, b)$

Результат операции сравнения аргументов a, b	Значение пороговой функции	
	$g(a, b)$	$g(b, a)$
$ b - a \leq c$	1	1
$b - a > c$	0	1
$b - a < -c$	1	0

Вычислительный примитив представляет собой повторяющийся в пределах изображения фрагмент вычислительной структуры, определяющий правила доступа к пикселям изображения и правила осуществления операции сравнения при формировании $F_{x_{lm}}$ -LBP-признака. Параметры l, m характеризуют размерность вычислительного примитива.

На рис. 4 представлены три варианта вычислительного примитива различной размерности: 1x1 пиксель, 2x2 пикселей, 3x3 пикселей. Направление стрелок на рисунке определяет направление от элемента со значением интенсивности b к элементу с интенсивностью a пороговой функции $g(a, b)$ (или $g_i(a, b)$), осуществляющей сравнение двух соседних пикселей. Элементы изображения, к которым направлены стрелки, в совокупности с опорными элементами изображения, отмеченными символом h , образуют множество пикселей изображения P_{pe} , к которым осуществляется доступ при обработке примитивного элемента. На рис. 5 представлена регулярная вычислительная структура для фрагмента изображения размером 5x5 пикселей с примитивом размером 2x2 пикселей.

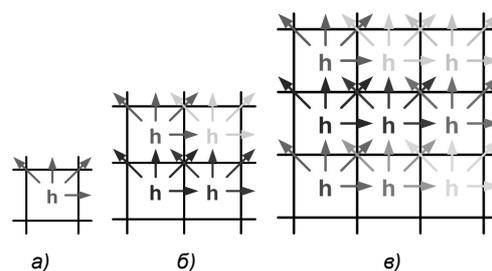


Рис. 4. Примитивы регулярной вычислительной структуры: а) 1x1 пиксель; б) 2x2 пикселей; в) 3x3 пикселей

Представленные на рис. 4 базовые вычислительные примитивы отличаются удельным количеством элементов изображения, к которым осуществляется доступ для формирования F -LBP-признака. Увеличение размерности примитива регулярной вычислительной структуры приводит к уменьшению удельного количества пикселей, к которым осуществляется доступ. При этом удельное количество процедур сравнения в пересчёте на один пиксель N_{comp}^{F-LBP} является одинаковым для всех размер-

ностей примитивов. Характеристики вычислительных примитивов представлены в табл. 3.

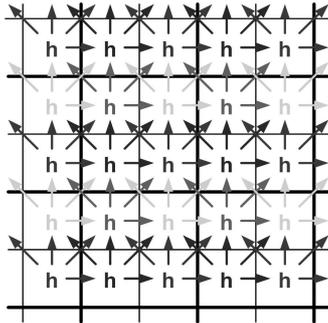


Рис.5. Регулярная вычислительная структура для фрагмента изображения

Вычислительная процедура обработки примитива регулярной вычислительной структуры включает следующие основные операции:

1. Осуществляется доступ к элементам изображения P_{pe} , необходимым для обработки вычислительного примитива.

2. Для каждого опорного компонента вычислительно-го примитива (отмеченного символом **h**) осуществляется сравнение пикселей по четырём направлениям в соответствии с пороговой функцией $g(a,b)$ (или $g_i(a,b)$), задаваемой табл. 1 (или табл. 2). По бинарным результатам i -й ($i=1..4$) операции сравнения осуществляется обновление $F-LBP$ -признаков соседних элементов в соответствии с побитовыми функциями $f_i^a(F-LBP_a)$ и $f_i^b(F-LBP_b)$ (табл. 4) по формулам:

$$F-LBP_a = f_i^a(F-LBP_a), \tag{9}$$

$$F-LBP_b = f_i^b(F-LBP_b). \tag{10}$$

Таблица 3

Характеристики базовых вычислительных примитивов

Размерность вычислительного примитива, $l \times m$	N_{comp}^{F-LBP}	N_{access}^{F-LBP}
1x1	4	5
2x2	4	11/4
3x3	4	19/9

При осуществлении обновления $F-LBP$ -признаков в соответствии с формулами (9), (10) функция $f_i^a(F-LBP_a)$ воздействует на один из четырёх младших битов бинарного представления LBP -признака пикселя с интенсивностью a , в то время, как функция $f_i^b(F-LBP_b)$ воздействует на один из четырёх старших битов бинарного представления LBP -признака пикселя с интенсивностью b . Вычисление признаков $F-LBP^{u2}$, $F-LBP^{riu2}$ ($F-TLBP^{u2}$, $F-TLBP^{riu2}$), реализованное на основании базового признака $F-LBP$ ($F-TLBP$) в соответствии с формулами (4)-(6), может быть эффективно осуществлено посредством таблиц поиска $LUT_{u2}(LBP)$, $LUT_{riu2}(LBP)$.

Результаты моделирования

Для оценки эффективности предложенного метода быстрого вычисления LBP -признаков осуществлена программная реализация на языке программирования C++ LBP -признаков: LBP^{u2} , LBP^{riu2} , $TLBP^{u2}$, $TLBP^{riu2}$, $F_{l \times m} - LBP^{u2}$, $F_{l \times m} - LBP^{riu2}$, $F_{l \times m} - TLBP^{u2}$, $F_{l \times m} - TLBP^{riu2}$. Размерность $l \times m$ вычислительного примитива в экспериментах представлена набором из значений: 1x1, 2x2, 3x3 пикселей. Реализованные функции вычисления LBP -признаков скомпилированы в тех-функции для использования в среде Matlab.

Представленные LBP -признаки рассчитывались для полутоновых изображений «Tbilisi», «Ankara», «Kopenhagen» размером $n_x \times n_y = 800 \times 600$ пикселей (рис. 6). Моделирование осуществлялось на стандартном ПК Celeron 1.86GHz, 2Gb. В серии из $N = 20$ запусков фиксировалось время T вычисления соответствующих типов LBP -признаков всех пикселей изображения. По серии измерений вычислялась средняя скорость $Rate$ расчёта LBP -признаков:

$$Rate = \frac{n_x \times n_y}{N} \sum_{i=1}^N T_i \text{ [пиксель/сек]}. \tag{11}$$

Таблица 4

Функции $f_i^a(F-LBP_a)$, $f_i^b(F-LBP_b)$ обновления LBP -признака

Вектор вычисления	$g(a,b), g_i(a,b)$	0	1	1
	$g(b,a), g_i(b,a)$	1	1	0
↖	$f_1^a(F-LBP_a)$	$F-LBP_a$ or 1		$F-LBP_a$ and 254
	$f_1^b(F-LBP_b)$	$F-LBP_b$ and 239	$F-LBP_b$ or 16	
↑	$f_2^a(F-LBP_a)$	$F-LBP_a$ or 2		$F-LBP_a$ and 253
	$f_2^b(F-LBP_b)$	$F-LBP_b$ and 223	$F-LBP_b$ or 32	
↗	$f_3^a(F-LBP_a)$	$F-LBP_a$ or 4		$F-LBP_a$ and 251
	$f_3^b(F-LBP_b)$	$F-LBP_b$ and 191	$F-LBP_b$ or 64	
→	$f_4^a(F-LBP_a)$	$F-LBP_a$ or 8		$F-LBP_a$ and 247
	$f_4^b(F-LBP_b)$	$F-LBP_b$ and 127	$F-LBP_b$ or 128	

Для каждого варианта быстрого вычисления LBP -признака оценивался выигрыш в скорости $\delta Rate$ по отношению к соответствующему базовому варианту LBP -признака по формуле:

$$\delta Rate = \frac{Rate_F - Rate_{Base}}{Rate_{Base}} \cdot 100\%, \quad (12)$$

где $Rate_F$ – средняя скорость вычисления LBP -признаков, соответствующая быстрому методу вычисления; $Rate_{Base}$ – средняя скорость вычисления LBP -признаков, соответствующая базовому варианту вычисления.

Величина сдвига порога функции $g_i(a,b)$ (8) в процессе моделирования составляла $c = 4$. Табл. 5, 6 содержат результаты оценки скорости $Rate$ и выигрыша в скорости $\delta Rate$ относительно базового варианта расчёта LBP -признаков для LBP^{riu2} , $F-LBP^{riu2}$, $TLBP^{riu2}$, $F-TLBP^{riu2}$ признаков при разных размерностях примитива регулярной вычислительной структуры. На рис. 7 представлено графическое изображение LBP^{riu2} , $TLBP^{riu2}$ -признаков для изображения «Копенгаген».

Таблица 5
Скорость вычисления LBP^{riu2} , $F-LBP^{riu2}$ -признаков

Наименование LBP -признака	«Tbilisi»		«Ankara»		«Kopenhagen»	
	$Rate$, пиксели/сек	$\delta Rate$, %	$Rate$, пиксели/сек	$\delta Rate$, %	$Rate$, пиксели/сек	$\delta Rate$, %
LBP^{riu2}	4570864	0	4681408	0	4577011	0
$F_{1 \times 1} - LBP^{riu2}$	5627498	23	6263555	34	6002274	31
$F_{2 \times 2} - LBP^{riu2}$	7035291	54	7433466	59	7131787	56
$F_{3 \times 3} - LBP^{riu2}$	7620634	67	8050563	72	7708744	68

Таблица 6
Скорость вычисления $TLBP^{riu2}$, $F-TLBP^{riu2}$ -признаков

Наименование LBP -признака	«Tbilisi»		«Ankara»		«Kopenhagen»	
	$Rate$, пиксели/сек	$\delta Rate$, %	$Rate$, пиксели/сек	$\delta Rate$, %	$Rate$, пиксели/сек	$\delta Rate$, %
$TLBP^{riu2}$	4764211	0	4685137	0	4756369	0
$F_{1 \times 1} - TLBP^{riu2}$	6101520	28	6009571	28	6234694	31
$F_{2 \times 2} - TLBP^{riu2}$	7379653	55	7253018	55	7559939	59
$F_{3 \times 3} - TLBP^{riu2}$	7936973	67	7799633	66	8187392	72



а)

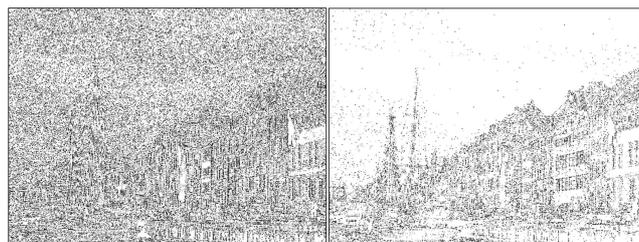


е)



б)

Рис. 6. Тестовые изображения:
а) «Tbilisi», б) «Ankara», в) «Kopenhagen»



а)

б)

Рис. 7. Графическое представление LBP -признаков для изображения «Копенгаген» а) LBP^{riu2} ; б) $TLBP^{riu2}$

Из табл. 5, 6 видно, что применение метода быстрого вычисления LBP -признаков с размером вычислительного примитива 3×3 позволяет обеспечить относительный выигрыш в скорости на уровне 66-72% для вариантов LBP -признака с обычной и модифицированной пороговой функцией. Для признаков $F_{l \times l} - LBP^{riu2}$, $F_{l \times l} - TLBP^{riu2}$ выигрыш в скорости вычисления составляет 23-34% и обеспечивается, с одной стороны, сокращением удельного количества операций сравнения с $N_{comp}^{LBP} = 8$ до $N_{comp}^{F-LBP} = 4$, с другой стороны, сокращением удельного количества пикселей, к которым осуществляется доступ при расчёте LBP -признака, с $N_{access}^{LBP} = 9$ до $N_{access}^{F-LBP} = 5$. Дальнейший выигрыш в скорости при увеличении размерности вычислительного примитива достигается за счёт сокращения удельного количества пикселей, к которым осуществляется доступ при расчёте LBP -признака.

Увеличение размерности примитива регулярной вычислительной структуры для обеспечения увеличения выигрыша в скорости вычисления повышает требования к размеру кэш-памяти процессора.

Заключение

Предложен метод быстрого вычисления $F_{l \times m} - LBP$ -признаков изображения, основанный на выделении примитива регулярной вычислительной структуры размером $l \times m$ пикселей, установлении свойства взаимности пороговой $g(a, b)$ и модифицированной пороговой $g_t(a, b)$ функций, определяющих правила сравнения соседних пикселей локальной окрестности, и правила обновления LBP -признаков. Примитив регулярной вычислительной структуры определяет правила доступа к пикселям изображения и правила осуществления операции сравнения в соответствии с пороговой функцией.

Установлено, что предложенный метод с размером примитива 3×3 для быстрого инвариантного к повороту LBP -признака ($F_{3 \times 3} - LBP^{riu2}$, $F_{3 \times 3} - TLBP^{riu2}$) позволяет получить выигрыш в скорости вычисления на уровне 66-72% по отношению к варианту без ускорения (LBP^{riu2} , $TLBP^{riu2}$) за счёт сокращения удельного количества операций сравнения с $N_{comp}^{LBP} = 8$ до $N_{comp}^{F-LBP} = 4$ и сокращения удельного количества пикселей, к которым осуществляется доступ при расчёте LBP -признака, с $N_{access}^{LBP} = 9$ до $N_{access}^{F-LBP} = 19/9$.

Данный метод может быть использован для решения задач извлечения текстурной информации из видеоизображения в режиме реального времени.

Литература

1. F. Tomita, S. Tsuji: Computer Analysis of Visual Textures, Kulwer Academic, Boston, 192 p. (1990)
2. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on feature distributions. Pattern Recognit. 29(1), 51–59 (1996)
3. Matti Pietikäinen, Abdenour Hadid, Guoying Zhao, Timo Ahonen: Computer Vision Using Local Binary Patterns. Springer, London (2011)
4. Ojala, T., Pietikäinen, M., Maenpää, M.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7), 971–987 (2002)
5. Jun Meng, Yumao Gao, Xiukun Wang, Tsauyoung Lin, Jianying Zhang, Face Recognition Based on Local Binary Patterns with Threshold, IEEE International Conference on Granular Computing, San Jose, California, August 14–August 16, (2010)
6. Di Huang, Caifeng Shan, Ardabilian, M., Yunhong Wang, Liming Chen, «Local Binary Patterns and Its Application to Facial Image Analysis: A Survey», Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, On page(s): 765 - 781 Volume: 41, Issue: 6, Nov. (2011).
7. Loris Nanni, Alessandra Lumini, Sheryl Brahnam: Local binary patterns variants as texture descriptors for medical image analysis. Artificial Intelligence in Medicine 49(2): 117-125 (2010).
8. Heikkilä, M., Pietikäinen, M.: A texture-based method for modeling the background and detecting moving objects. IEEE Trans. Pattern Anal. Mach. Intell. 28(4), 657–662 (2006).
9. Ning, J., Zhang, L., Zhang, D., Wu, C.: Robust object tracking using joint color-texture histogram. Int. J. Pattern Recognit. Artif. Intell. 23(7), 1245–1263 (2009).

FAST LBP-FEATURES COMPUTATION METHOD FOR VIDEO IMAGE ROI

Talochka A.M., Boriskevich A.A

A fast LBP-features computation method for the image based on forming a regular computation structure primitive, defining a mutuality property of the threshold- and the modified threshold-function determining pixel comparison rules in the local neighborhood, and update LBP-features rules is proposed. It allows us to obtain a computation rate gain equal about 66-72% for the Rotation Invariant LBP-feature in using 3×3 primitive in contrast with the conventional one due to specific amount reduction of the comparison operations and pixels that are used for the LBP-feature computation.

ИСПОЛЬЗОВАНИЕ ХАРАКТЕРА РАСПРЕДЕЛЕНИЯ САМОПОДОБИЯ В КАЧЕСТВЕ ПРИЗНАКА ЦИФРОВОГО ИЗОБРАЖЕНИЯ В ЗАДАЧЕ КЛАССИФИКАЦИИ

Жизняков А.Л., д.т.н., профессор, зав. кафедрой «САПР ЭС», зам. директора по научной работе Муромского института (филиала Владимирского государственного университета имени Александра Григорьевича и Николая Григорьевича Столетовых), e-mail: Ivovich@newmail.ru

Привезенцев Д.Г., аспирант кафедры «САПР ЭС» Муромского института (филиала Владимирского государственного университета имени Александра Григорьевича и Николая Григорьевича Столетовых), e-mail: dgprivezencev@mail.ru

Ключевые слова: цифровые изображения, классификация, самоподобие, распределение, признаки, фракталы.

Введение

В настоящее время одним из развивающихся и перспективных направлений в цифровой обработке изображений является применение фрактального анализа. Фракталы обладают свойствами самоподобия, что означает точное или вероятностное повторение свойств объекта при рассмотрении его в разных масштабах. Свойство самоподобия приводит к определенным закономерностям в статистическом поведении признаков изображений, в результате чего изображения можно с определенной точностью описать фрактальными признаками [1-4].

Любая из процедур обработки изображений опирается на модель класса изображений – формализованное описание, выполненное с определенной степенью абстрагирования. Роль модели изображения в процессе извлечения информации состоит в обеспечении адекватного описания существенных свойств класса изображений, позволяющего дать конструктивную основу для построения эффективных вычислительных процедур.

В частности для создания новых методов фрактальной обработки изображений необходимо создать фрактальную модель изображений. Так как для построения фракталов используются системы итерируемых функций [1, 2, 4], то одним из возможных способов описания изображения является использование систем итерируемых функций [5, 6].

Для качественной классификации изображений необходим набор признаков, однозначно характеризующих изображение. Чем обширнее этот набор, тем выше степень точности классификации [7]. Поэтому, кроме традиционных признаков, использование фрактальных характеристик изображений позволит снизить ошибки и существенно увеличить эффективность распознавания.

Самоподобие цифрового изображения

Фрактальная модель цифрового изображения предполагает представление изображения в виде дерева, листьями которого являются участки изображения (ран-

Приводится описание свойства самоподобия цифрового изображения. Предлагаются новые признаки изображений, характеризующие внутреннее распределение самоподобия и наиболее подходящие участки изображения. Описывается алгоритм формирования фрактальных признаков изображений: характерных участков изображения и характера распределения самоподобия. Приводятся результаты исследований возможности использования распределения самоподобия в задачах классификации изображений.

говые блоки R). Для каждого рангового блока определяется, какой другой участок изображения (доменный блок D) имеет схожие характеристики, после выполнения над ним заданных преобразований. К преобразованиям, выполняемым над доменными блоками, относятся сжатие, перемещение, изменения яркости и контраста точек.

Свойство самоподобия означает возможность описания одними элементами других элементов того же множества. В случае с изображением, свойство самоподобия означает возможность описания изображения его же участками. В случае полного самоподобия изображение может быть описано с помощью любого участка с нулевой погрешностью. Однако, в большинстве случаев, изображение не является полностью самоподобным. Из этого следует, что одни участки являются более самоподобными чем другие, и, следовательно, для изображения можно сформировать диаграмму или гистограмму распределения самоподобия внутри изображения.

Распределение самоподобия изображения

Согласно фрактальной модели, изображение разбивается на доменные блоки D , с помощью которых будет впоследствии описано изображение. Затем изображение разбивается на ранговые блоки R таким образом, чтобы каждый участок изображения был описан с помощью доменного блока с заданной точностью.

Для аппроксимации ранговых блоков доменными блоками применяется преобразование w , включающее операции масштабирования, переноса, изменения яркости:

$$R_i \approx w_i(D_i) = s_i(\tilde{w}_i(D_i)) + o_i,$$

В итоге такого описания получается список ранговых блоков, из которых строится изображение, для каждого из которых указывается доменный блок и преобразование [5, 6]:

$$f' = \sum_{i=1}^M (B_{n,m}^r)^* \left[s_i \left(\tilde{w}_i \left(B_{k,l}^d [f] \right) \right) + o_i \right],$$

где f – исходное изображение, $B_{k,l}^d$ – оператор извлечения блока изображения, который извлекает доменный блок изображения с индексом d , левый верхний угол которого находится в точке (k,l) ; $(B_{n,m}^r)^*$ – оператор вставки блока, который вставляет блок изображения размером $m \times n$ в место рангового блока с индексом r , левый верхний угол которого находится в точке с координатами (n, m) ; s_i, o_i – коэффициенты преобразования яркости изображения; \tilde{w}_i – аффинное преобразование; M – количество ранговых блоков на которое развивается изображение при заданных условиях фрактального разложения.

Из множества доменных блоков $D = \{D_i\}$ можно выделить подмножество используемых доменных блоков $Du = \{Du_i\} \in D$, отражающее самоподобные участки изображения. Исходя из этого, можно использовать в качестве характеристики самоподобия изображений отношение числа используемых доменных блоков к общему количеству:

$$K = \frac{|Du|}{N_D}.$$

где $|Du|$ – количество доменных блоков, используемых для восстановления изображений, N_D – общее количество доменных блоков.

Каждый доменный блок из множества Du используется для формирования фрактального кода определенное количество раз, т.е. каждому доменному блоку можно сопоставить число, равное количеству раз использования данного блока во фрактальном коде, которое означает, сколько других участков изображения подобны этому. Разместив эти числа на изображении в местах расположения соответствующих доменных блоков, можно увидеть наиболее часто встречающиеся участки (рис. 1).

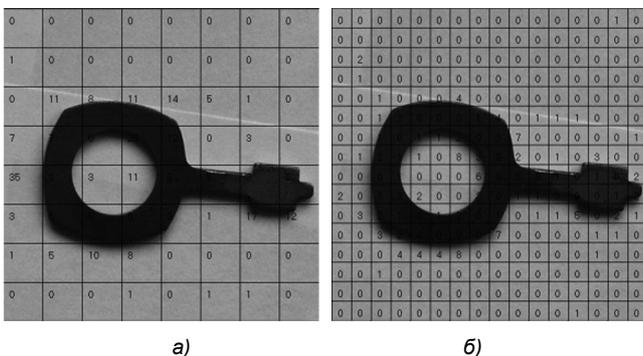


Рис. 1. Частота использования доменных блоков тестового изображения: а) первого уровня; б) второго уровня

Если используется несколько уровней доменных блоков, то для отражения подобия участков изображения данные по доменным блокам каждого уровня необходимо отобразить на одном изображении (рис. 2).

Представив частоту использования доменных блоков в виде карты высот и проведя аппроксимацию, можно получить трехмерную поверхность $Z(x, y)$, где $x = 1 \dots W, y = 1 \dots H, Z(x, y) = 1 \dots \max(Du_i)$, значение которой в каждой точке отражает степеню подобия участка изображения (рис. 3).

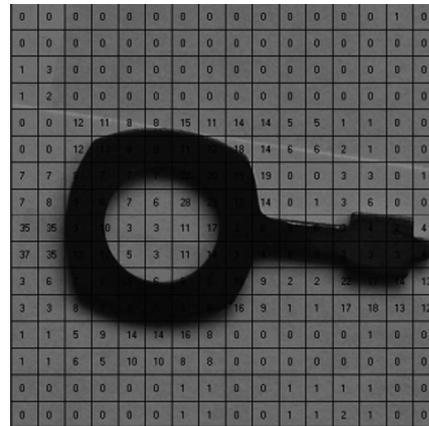


Рис. 2. Частота использования всех доменных блоков на тестовом изображении

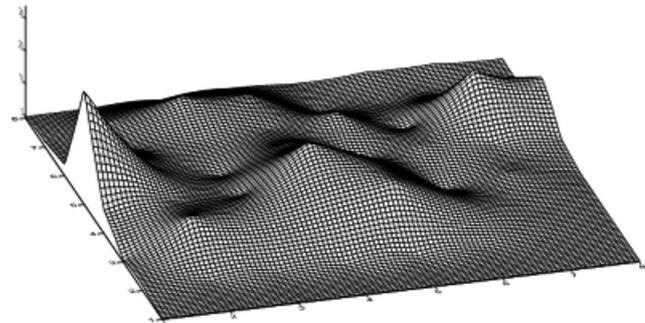


Рис. 3. Трехмерная гистограмма использования доменных блоков

Для лучшего отражения непрерывного изменения исследуемой величины на трехмерных поверхностях обычно применяются изолинии, которые отражают распределение этой величины на поверхности.

Если построить изолинии для поверхности $Z(x, y)$ и наложить на исходное изображение, то получим наглядное представление изменения самоподобия участков (рис. 4).

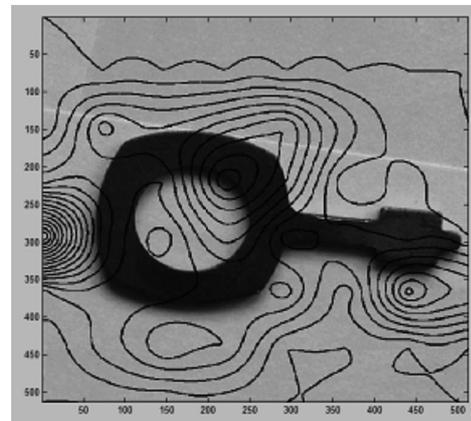


Рис. 4. Представление изменения самоподобия участков изображения

Распределение локальных свойств самоподобия внутри изображения лучше всего характеризовать параметрами представленной трехмерной гистограммы использования доменных блоков для восстановления изображения по его фрактальному коду.

Характер распределения самоподобия изображения

Представление изменения самоподобия участков изображения с помощью изолиний обладает наглядностью и позволяет качественно оценить распределение, но не позволяет получить количественную оценку. Для задач обработки изображений необходимо представить распределение самоподобия в виде характеристик, имеющих числовое или аналитическое выражение.

Для этого строится гистограмма использования доменных блоков $H^D(j)$ (рис. 5).

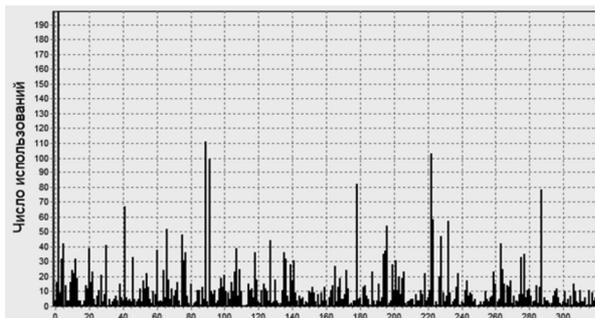


Рис. 5. Гистограмма использования доменных блоков во фрактальном коде изображения

Чем больше значение $H^D(j)$ для доменного блока D_j , тем более характерным является участок соответствующий доменному блоку для этого изображения. Следовательно, участки с наибольшим значением $H^D(j)$ можно использовать как шаблоны изображений в задаче распознавания и классификации изображений.

Если значения $H^D(j)$ упорядочить по убыванию, то в общем виде получится следующего вида кривая (рис. 6). График, представленный на рис. 6, отражает характер распределения самоподобия на изображении. Если аппроксимировать полученные значения, тогда можно получить аналитическое выражение, характеризующее распределение признаков самоподобия.

Наибольшее соответствие наблюдается при аппроксимации логарифмическим уравнением вида:

$$y(x) = k \ln(x) + b.$$

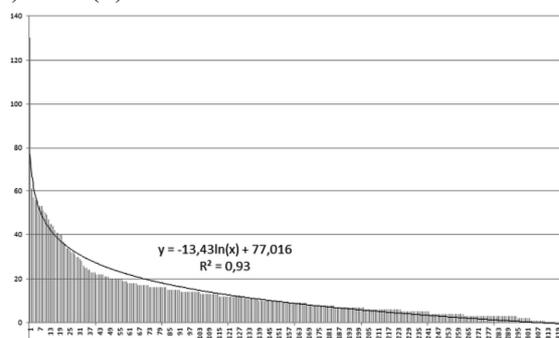


Рис. 6. Характер распределения самоподобия на изображении

Экспериментальные исследования

При использовании признака в задачах распознавания образов необходимо его инвариантное поведение относительно различных преобразований, например, таких как преобразование изменения яркости и поворота изображения. Согласно ранее проведенному исследованию, распределение самоподобия на изображении является инвариантным признаком [8].

Для классификации изображений по их характерным участкам были взяты текстурные изображения размерами 1800x1200 пикселей. Из каждого изображения было получено 5 тестовых изображений размерами 512x512 пикселей. Для каждого тестового изображения были сформированы характерные участки и рассчитаны коэффициенты уравнений, описывающих характер распределения самоподобия на изображениях. После чего было произведено их сравнение. Результаты расчета коэффициентов уравнений, описывающих распределение самоподобия на тестовых изображениях, приведены в табл. 1.

Из табл. 1 видно, что:

- коэффициенты уравнений распределения самоподобия изображений одного класса близки между собой;
- коэффициенты для изображений разных классов отличаются.

Также было выявлено, чем больше абсолютное значение коэффициента k , тем меньше изображение имеет характерных участков, но они выразительны в большей степени.

На рис. 7 приведены тестовые изображения с рассчитанными коэффициентами уравнений распределения самоподобия, по которому видно, что первые три изображения принадлежат одному классу, что подтверждается коэффициентами, а четвертое – принадлежит другому классу.

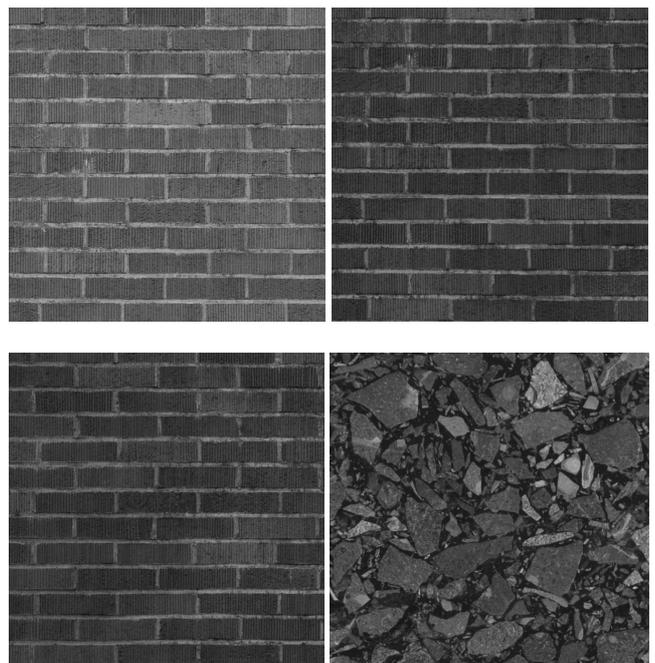


Рис. 7. Примеры тестовых изображений с рассчитанными коэффициентами уравнений распределения самоподобия

Таблица 1

Коэффициенты уравнений распределения самоподобия

	Класс а		Класс b		Класс с		Класс d		Класс е	
	к	б	к	б	к	б	к	б	к	б
Тест 1	-15,83	87,21	-19,10	99,35	-15,69	84,51	-15,05	80,50	-14,03	78,43
Тест 2	-16,13	87,46	-20,24	105,2	-18,18	98,27	-13,01	75,41	-13,61	75,36
Тест 3	-15,25	85,18	-18,38	95,92	-19,07	102,3	-14,43	81,14	-13,58	74,98
Тест 4	-15,52	85,20	-19,83	98,41	-17,43	93,24	-13,12	75,55	-12,97	73,48
Тест 5	-15,03	82,99	-20,40	106,5	-17,05	95,87	-13,41	77,62	-13,21	73,84
Среднее	-15,55	85,61	-19,59	101,1	-17,48	94,84	-13,80	78,04	-13,48	75,21
Дисп.	0,194	3,299	0,710	20,77	1,602	44,43	0,799	7,238	0,165	3,828

Заключение

Таким образом, предлагаются новые признаки изображений, характеризующие внутреннее распределение самоподобия и наиболее подобные участки изображения. Распределение самоподобия дает возможность использовать в качестве информативных признаков изображения наиболее характерные его участки, полученные на основе гистограммы $H^D(j)$, которые в большинстве случаев являются уникальными для каждого изображения или класса изображений, и коэффициенты уравнения k и b , описывающего характер распределения локальных признаков самоподобия.

В результате исследования было определено, что распределение самоподобия изображения инвариантно к различным преобразованиям. Коэффициенты уравнения, описывающего распределение самоподобия, близки для изображений одного класса, но различны для изображений разных классов.

Так как распределение самоподобия изображений разных классов описывается отличающимися уравнениями, то этот признак является классифицирующим.

Литература

1. Новейшие методы обработки изображений. / Под ред. А.А. Потапова – М.: ФИЗМАТЛИТ, 2008. – 496с. – ISBN 978-5-9221-0841-6.
2. Шелухин О.И., Осин А.В., Смольский С.М. Самоподобие и фракталы. Телекоммуникационные приложения. / Под ред. О.И. Шелухина. – М.: ФИЗМАТЛИТ, 2008. – 368с. – ISBN 978-5-9221-0949-9.
3. Методы компьютерной обработки изображений. / Под ред. В.А. Сойфера. - 2-е изд., испр. - М.: ФИЗМАТЛИТ, 2003. - 784с. - ISBN 5-9221-0270-2
4. Уэлстид С. Фракталы и вейвлеты для сжатия изображений в действии. Учебное пособие. - М.: Издательство Триумф, 2003 - 320 с.: ил.

5. Привезенцев Д.Г. «Модель цифрового изображения с использованием систем итерируемых функций» // Информационные технологии моделирования и управления. – 2010. – №6(65) – С. 761-769.

6. Привезенцев Д.Г., Жизняков А.Л. «Фрактальная модель цифрового изображения» // Алгоритмы, методы и системы обработки данных: сб. науч. тр. Издательско-полиграфический центр МИ ВЛГУ. – 2010. – Вып. 15. – с.147-152.

7. Жизняков А.Л. Формирование и анализ наборов признаков многомасштабных последовательностей цифровых изображений [Текст] // Программные продукты и системы. Изд. ЗАО НИИ «Центрпрограммсистем». – 2007. - №4.

8. Жизняков А.Л., Привезенцев Д.Г., Фомин А.А. Классификация изображений на основе локальных признаков самоподобия [Текст] // Ползуновский вестник. - 2011. - № 3, Ч. 1. - С. 12-14. ISSN 2072-8921.

USE OF CHARACTER OF DISTRIBUTION OF SELF-SIMILARITY AS THE SIGN OF THE DIGITAL IMAGE IN THE CLASSIFICATION TASK

Zhiznyakov A.L., Privezentsev D.G.

In article the description of property of self-similarity of the digital image is resulted. The new signs of images characterizing internal distribution of self-similarity and the most similar blocks of the image are offered. The algorithm of formation fractal signs of images is described: characteristic blocks of the image and character of distribution of self-similarity. Results of researches of possibility of use of distribution of self-similarity in the classification task of images are resulted.

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ НА СОВРЕМЕННЫХ ГРАФИЧЕСКИХ ПРОЦЕССОРАХ

Пантелеев А.Ю., аспирант Национального исследовательского ядерного университета «МИФИ», инженер по архитектуре в NVIDIA, e-mail: apantelev87@gmail.com

Ключевые слова: графический процессор, быстрое преобразование Фурье, свертка, корреляция, границы применимости.

Введение

Современные графические процессоры (Graphics Processing Unit, GPU) обладают значительно большей вычислительной мощностью, чем любые другие программируемые устройства общего назначения. Разница в пиковой производительности с мощными центральными процессорами (Central Processing Unit, CPU) персональных компьютеров достигает 20–50 раз, при том, что потребляемая мощность больше всего в 2 раза, а цена сопоставима. При этом их можно использовать для решения широкого круга задач, включая задачи цифровой обработки сигналов (DSP). В настоящее время существуют реализации самых различных алгоритмов для GPU, включая БПФ, алгоритмы линейной алгебры, включая операции над разреженными матрицами и векторами, всевозможные алгоритмы обработки изображений и видео, алгоритмы компьютерного зрения.

В данной статье рассмотрены особенности реализации базовых алгоритмов ЦОС (DSP) на GPU, поддерживающих устройства NVIDIA, как наиболее распространенных и имеющих наилучшие средства программирования и набор библиотек. При этом, те же самые принципы реализации алгоритмов можно использовать и для конкурирующего стандарта параллельного программирования – OpenCL. Он поддерживается как устройствами от NVIDIA, так и графическими процессорами от AMD, центральными процессорами с архитектурой x86 любых производителей и на некоторых других устройствах. Однако стоит учитывать, что хотя функционально программа, использующая OpenCL, будет работать на любых устройствах, поддерживающих этот стандарт, техники оптимизации программ существенно различаются, и поэтому программа, эффективно использующая CPU, может быть неоптимальной при исполнении на GPU, и наоборот.

Основные особенности выполнения программ на GPU

Модель исполнения программ на GPU и основные техники оптимизации хорошо описаны в соответствующих документах [1–4]. Наиболее важные в рамках данной статьи детали модели исполнения заключаются в следующем:

1. Программа выполняется на отдельном устройстве, а ее запуск в доступных в настоящее время поколе-

Рассмотрена реализация на графических процессорах основных алгоритмов цифровой обработки сигналов (ЦОС): быстрого преобразования Фурье (БПФ), свертки, корреляции, умножения матрицы на вектор. По каждому алгоритму обозначены факторы, влияющие на производительность; в большинстве случаев она ограничена пропускной способностью памяти, а не вычислительными возможностями устройств. Представлены примеры применения графических процессоров в системах ЦОС.

ниях GPU иницируется только центральным процессором. По сути, программа является процедурой – она имеет точку входа, входные и выходные данные и должна закончиться за ограниченное время. Это означает, что обработка данных на GPU носит блочный характер, там нельзя использовать обработку данных в режиме непрерывного потока или по прерыванию.

2. Вычисления с плавающей точкой одинарной точности производятся на GPU быстрее, чем целочисленные (особенно умножение). Конкретное соотношение пропускных способностей зависит от архитектуры устройства.

3. Устройство имеет высокоскоростной доступ только к своей внутренней памяти. Для того чтобы данные приложения были доступны устройству, их следует туда скопировать, а потом скопировать обратно результат. GPU может напрямую работать с данными из памяти CPU, но это слишком медленно: пропускная способность самого быстрого варианта шины, через которую происходит копирование данных (то есть, PCIe 3.0 x16), составляет 16 Гб/с в каждую сторону, тогда как пропускная способность внутренней динамической памяти устройства в настоящее время достигает 264 Гб/с (для видеокарты AMD Radeon HD 7970). Для сравнения, пропускная способность памяти CPU составляет до 20 Гб/с.

4. Программа запускается на так называемой сетке (grid) из блоков, каждый из которых содержит до 1024 параллельно работающих потоков (threads). Каждый поток выполняет одну и ту же программу, но может определять ветвления независимо от других потоков. Такая модель обычно называется SPMD (Single Program, Multiple Data). Потоки внутри блока могут синхронизировать свое выполнение друг с другом при помощи барьеров.

5. Устройство состоит из так называемых потоковых мультипроцессоров (SM, Streaming Multiprocessor) – эти блоки, по сути, являются векторными процессорами, выполняющими каждую инструкцию в программе синхронно для группы потоков. Такая группа называется «варп» (warp) в терминах NVIDIA, или «фронт волны»

(wavefront) в терминах AMD. Ветвления программы приводят к последовательному выполнению всех выполняющихся веток, поэтому являются нежелательными.

6. Память устройства имеет иерархическую структуру. Ближе всего к вычислительным устройствам расположена регистровая память, она же самая быстрая (порядка 0.5 – 2ТБ/с на SM). В этой памяти располагаются рабочие данные каждого потока; она не поддерживает произвольную адресацию, в отличие от всех остальных уровней памяти. Следующий уровень иерархии – разделяемая память (локальная в терминах OpenCL). Эта память доступна всем потокам в группе, в ней же располагается кэш L1 глобальных данных; ее пропускная способность составляет 256–512 ГБ/с на SM, а объем в сумме с кэшем L1 – 64 КБ на SM. Далее располагается кэш L2, когерентный между всеми мультипроцессорами, через который производятся все операции с глобальной памятью; его пропускная способность составляет 256–512 ГБ/с, а объем порядка 256–768 КБ. Последним уровнем иерархии является глобальная память большого объема (до 6 ГБ), производительность которой имеет порядок 140–260 ГБ/с.

7. При адресации глобальной памяти важно соблюдать правила параллельного доступа или слияния запросов (coalescing). Хотя с функциональной точки зрения поддерживается произвольный доступ к памяти, наилучшая производительность достигается, если потоки в варпе адресуют последовательно расположенные 4-байтные слова, а весь блок данных выровнен по границе 32 или 128 байт. Более детальные сведения об оптимизации доступа к глобальной памяти приведены в руководстве [2] и презентации [5].

8. Разделяемая память организована в виде множества банков шириной 4–8 байт с независимой адресацией. Последовательные адреса располагаются в соседних банках памяти, а адреса с определенным шагом – в соседних адресах одного банка памяти. Этот шаг определяется как произведение числа банков (например, 32) на ширину одного банка. Если один запрос чтения или записи из варпа затрагивает не более одного

адреса в каждом банке памяти, такой запрос выполняется с максимальной скоростью. Если же запрос затрагивает два и более адреса в любом банке памяти, то такой запрос выполняется в соответствующее число раз медленнее. Такая ситуация называется конфликтом банков.

9. Глобальная память имеет очень большую латентность доступа (порядка 400–1000 тактов), и для получения высокой производительности ее необходимо маскировать. Если программа интенсивно работает с памятью, то для этого необходимо генерировать много независимых запросов из каждого мультипроцессора одновременно. Для количественной оценки необходимого количества варпов или параллельных запросов в память из каждого варпа следует использовать закон Литтла, который гласит, что среднее количество запросов, находящихся в системе (памяти), равняется времени обработки одного запроса (то есть латентности), умноженному на скорость входа или выхода запросов в систему. Следует учитывать, что с увеличением загрузки системы памяти ее латентность возрастает – это связано с заполнением запросами внутренних очередей.

Сведения о системе памяти особенно важны при реализации алгоритмов DSP, так как производительность всех алгоритмов, представленных в этой статье, ограничена пропускной способностью памяти, а не вычислительными возможностями устройства. Исключение составляет работа с числами с плавающей точкой двойной точности на устройствах, которые заведомо не были спроектированы для высокопроизводительной обработки данных с плавающей точкой двойной точности, вроде NVIDIA GeForce GTX 680. В табл. 1. приведена справочная информация по нескольким распространенным моделям видеокарты вычислительных ускорителей последних поколений, которая помогает определить факторы, ограничивающие производительность. Некоторые данные о внутреннем устройстве GPU, которые могут быть важны для оптимизации производительности, производители не публикуют, но их можно выяснить при помощи направленных тестов – такие тесты описаны в работе [6].

Таблица 1.

Характеристики распространенных устройств от производителей NVIDIA и AMD последних поколений

Производитель Устройство	NVIDIA			AMD	
	Tesla C2075	GeForce GTX 580	GeForce GTX 680	Radeon HD 5870	Radeon HD 7970
Наименование архитектуры	Fermi	Fermi	Kepler	Cypress	Tahiti
Количество блоков SM	14	16	8	32	32
Размер варпа	32	32	32	64	64
Максимальное количество регистров на поток	63	63	63	128*4	256
Объем регистровой памяти на SM, КБ	128	128	256	256	256
Объем разделяемой памяти на SM, КБ	16/48	16/48	16/32/48	32	64
Число банков разделяемой памяти	32	32	32	32	32
Ширина банка разделяемой памяти, байт	4	4	4/8	4	4
Производительность FP32, GFLOPS	1030	1581	3090	2720	3789
Производительность FP64, GFLOPS	515	198	129	544	947
Пропускная способность глобальной памяти, ГБ/с	144	192	192	153	264
Объем глобальной памяти, ГБ	6	1.5	2	2	3
Поддержка ECC (коррекции ошибок в памяти)	Да	Нет	Нет	Нет	Нет
Потребляемая мощность, Вт	225	244	195	188	≈200
Энергоэффективность, FP32 GFLOPS/Вт	4.6	6.5	15.9	14.5	18.9

Быстрое преобразование Фурье

Самой известной, «классической» реализацией БПФ является алгоритм Кули-Тьюки, выполняемый по простому основанию (обычно 2). Он обладает низкой алгоритмической сложностью $O(N \log_2 N)$, высокой точностью и хорошо работает на скалярных процессорах или на системах, где производительность памяти относительно высока и доступ к памяти выполняется в случайном порядке (а не в блочном режиме). При реализации алгоритма Кули-Тьюки на GPU возникает естественный вопрос: как именно следует разделить вычисления по потокам, чтобы эффективно задействовать все вычислительные устройства. Проблема в том, что при любом разделении необходимо производить обмен данными между потоками при переходе от одного слоя алгоритма к другому, и для этого следует использовать разделяемую память. И вне зависимости от выбранного разделения доступ к разделяемой памяти будет осуществляться неэффективно из-за обилия конфликтов банков в каких-либо слоях алгоритма.

Чтобы уменьшить нагрузку на разделяемую память, следует модифицировать алгоритм. Достаточно заметить, что в каждом потоке можно производить отдельные БПФ над небольшим числом отсчетов – таким, чтобы все рабочие данные уместились в доступные регистры. Для вычислений с одинарной точностью максимальное количество точек по основанию 2 равняется 16, т.к. каждый отсчет является комплексным числом и занимает 2 регистра. Поскольку алгоритм БПФ является рекурсивным, т.е. обработка последовательности из N отсчетов включает в себя полную обработку двух подпоследовательностей из $N/2$ отсчетов и комбинацию результатов при помощи $N/2$ преобразований из 2 отсчетов. Вывод алгоритма БПФ для общего случая можно найти в [7].

Рекурсию можно использовать с любым разделяющим фактором (не только 2) на каждом шаге рекурсии. Например, последовательность из $N = 128$ отсчетов можно разделить на 16 подпоследовательностей по $N_1 = 8$ отсчетов, а затем на 8 подпоследовательностей по $N_2 = 16$ отсчетов. Допускаются и разложения более высоких порядков, например, последовательность из $N = 1024$ отсчетов можно разделить на $N_1 = 16$, $N_2 = 8$, $N_3 = 8$. Для $N = 128$ алгоритм выглядит следующим образом:

1. Запустить 16 потоков для обработки одной последовательности S .
2. Загрузить в память каждого потока k подпоследовательность $T_i[k] = S[16i + k]$, $0 \leq i < 8, 0 \leq k < 16$. Легко заметить, что соседние потоки адресуют последовательно расположенные элементы S , что обеспечивает эффективный доступ к глобальной памяти.
3. Произвести в каждом потоке БПФ-8 над последовательностью T_k .
4. Перераспределить данные между потоками с помощью разделяемой памяти, так что 8 потоков сформируют подпоследовательности $U_k[i] = T_i[k]$, т.е. произвести транспонирование матрицы данных. Для обеспече-

ния бесконфликтного доступа к разделяемой памяти строки матрицы в ней нужно располагать с шагом, на единицу большим, чем длина этих строк. Остальные 8 потоков не задействуются для оставшейся части алгоритма.

5. Умножить последовательности U_k на коэффициенты: $U'_k[i] = U_k[i] * e^{-2\pi j \frac{ki}{N}}$

6. Произвести в каждом потоке БПФ-16 над последовательностью U'_k .

7. Выгрузить результат вычисления из памяти потоков: $S[16i + k] = U'_k[i]$, $0 \leq i < 16, 0 \leq k < 8$.

Следует заметить, что элементарные преобразования (БПФ-8 и БПФ-16 в данном случае) производятся над отсчетами, имеющими естественный порядок, то есть эти алгоритмы нужно строить так, чтобы они производили бит-реверсную перестановку отсчетов. Однако поскольку они работают над данными, находящимися в неиндексируемой регистровой памяти, все циклы полностью разворачиваются до или во время компиляции, и бит-реверсная перестановка сводится к простому переименованию входных переменных алгоритма вместо перемещения данных.

Также следует заметить, что 16 потоков недостаточно даже для того, чтобы заполнить один варп и тем более для того, чтобы загрузить работой весь GPU. Эффективно использовать GPU можно, лишь запустив тысячи потоков одновременно, и поэтому стоит запускать обработку множества преобразований сразу. В простейшем случае такими преобразованиями могут быть строки или столбцы БПФ, проводимого над двумерным сигналом, например изображением. В более сложном случае это будут части преобразования последовательности большей длины, например, преобразование последовательности из 65,536 точек производится с помощью двух групп преобразований на 256 точек и транспонирования через глобальную память между ними.

Помимо отсчетов исходной последовательности, для преобразования Фурье нужны значения вращательных

коэффициентов $W_N^k = e^{-2\pi j \frac{k}{N}}$. Внутри элементарных преобразований их нужно относительно немного, и они одинаковые для всех потоков – их можно разместить как непосредственные константы в программе или как данные в памяти констант. Коэффициенты же, необходимые при переходе между направлениями преобразований (то есть, при транспонировании), различаются для каждого потока, т.к. в их выражение входит более одного индекса отсчета. Имеется два способа их получения. Первый состоит в том, чтобы прочитать таблицу коэффициентов из глобальной памяти. Но это увеличивает нагрузку на систему памяти не менее чем в 1.5 раза, что приведет к соответствующему замедлению алгоритма, т.к. его производительность обычно ограничена системой памяти. Второй способ заключается в расчете необходимых коэффициентов непосредственно в программе на GPU. Однако вычисление синуса и косинуса производится в несколько раз медленнее, чем сложение или умножение, и поэтому имеет смысл вычислить несколько основных

коэффициентов и умножить их на себя необходимое число раз для получения высших степеней. Следует помнить, что экспонента, пусть и комплексная, является степенной функцией, и поэтому выражение $W_N^{kn} = (W_N^k)^n$ справедливо для любых значений k и n .

Все вышеизложенные соображения можно найти в ранее опубликованных статьях об оптимизации БПФ на GPU, так как эта тема активно развивалась последние несколько лет. Следует отметить работы [8] и [9], в которых описан подобный алгоритм и сравнение его производительности с библиотекой CUFFT 1.1 на видеокарте GeForce 8800 GTX, а также с реализациями на CPU. Реализация БПФ на OpenCL с оптимизацией для графических процессоров от AMD описана в презентации [10].

Самым практичным способом вычисления БПФ на GPU является использование готовых библиотек. В случае использования CUDA достаточно остановиться на библиотеке CUFFT [11], которая входит в набор средств разработчика. Эта библиотека предоставляет интерфейс, похожий на известную библиотеку FFTW [12], и предоставляет возможность быстро выполнять одно- и многомерные преобразования практически любого количества точек (не только степени двойки), с одинарной и двойной точностью, включая преобразования с действительным входом или выходом. При использовании OpenCL имеются другие варианты, в частности, библиотека clAmdFft, входящая в состав набора AMDAPPML [13], а также более простой пример реализации тех же алгоритмов с открытым исходным кодом от компании Apple [14].

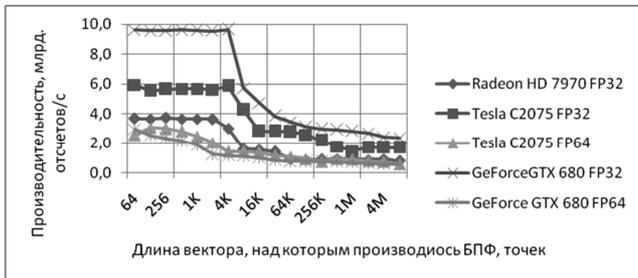


Рис. 1. Зависимость производительности БПФ от размерности преобразования

На рис. 1 представлен график реальной производительности БПФ на графических процессорах. Измерения проводились на устройствах NVIDIA Tesla C2075 и GeForce GTX 680 при помощи библиотеки CUFFT4.2, а также на видеокарте AMD Radeon HD 7970 при помощи библиотеки clAmdFft (к сожалению, на момент написания статьи последняя версия этой библиотеки, 1.8, не работает на GPU NVIDIA, поэтому произвести прямое сравнение качества кода не удастся). Чтобы эффективно загрузить работой устройства, во всех случаях проводилась группа преобразований общим объемом в 8 млн. точек. Как видно из графика, производительность приближается к 6 миллиардам отсчетов в секунду при выполнении вычислений одинарной точности (FP32) над последовательностями размерностью до 4096 точек включительно. При больших размерностях данные не помещаются в разделяемую память, и приходится про-

изводить два и более подхода, перезагружая данные через глобальную память, что приводит к кратному падению производительности. Несложно подсчитать, что в самых быстрых случаях производительность памяти GeForce GTX 680 находится на уровне 152 Гб/с (9.5×10⁹ отсчетов/с, 8 байтов на отсчет, чтение и запись), что составляет 79% от пропускной способности – очень хороший результат.

КИХ-фильтрация и корреляция

Результат работы фильтра с конечной импульсной характеристикой (КИХ) определяется сверткой, т.е. выражением $Y_k = \sum_i X_{k-i} W_i$, где X – входной вектор, а

W – вектор весовых коэффициентов или ядро свертки. Корреляция и свертка – две очень похожие операции, причем корреляция может быть выражена через свертку и наоборот, и поэтому в данном разделе они рассматриваются вместе. КИХ-фильтры, в отличие от БИХ, легко поддаются распараллеливанию, так как каждый выходной отсчет зависит только от входных отсчетов и коэффициентов фильтра, но не от других выходных отсчетов. Существует два способа разделить вычисления между потоками: первый – назначить каждому потоку по одному или несколько отсчетов исходной последовательности, второй – назначить каждому потоку по одному или несколько выходных отсчетов. Первый способ требует больше обращений к памяти для обмена промежуточными значениями отсчетов между потоками, и поэтому работает медленнее. Второй способ схематически представлен на рис. 2.

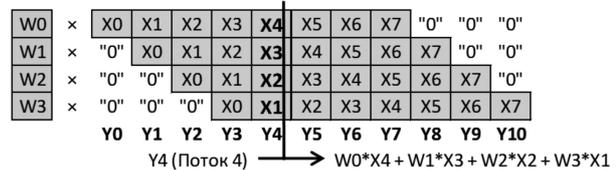


Рис. 2. Схематическое представление параллельной реализации КИХ-фильтра

Входные данные, как правило, размещаются в разделяемой памяти для обеспечения гарантированно быстрого доступа, и вектор X дополняется в начале и в конце небольшим количеством нулей или данных из соседних блоков сигнала. Если соседних блоков нет, то для уменьшения количества операций с нулями имеет смысл через каждые несколько отсчетов вектора W сдвигать окно вектора Y , с которым работают потоки, перемещая результаты через разделяемую память.

Фильтрация двумерных сигналов, т.е. изображений, выполняется похожим образом: фрагмент исходного изображения и матрица коэффициентов (ядро свертки) загружаются в разделяемую память, и каждый поток производит расчет одного или нескольких отсчетов результата. Однако в случае фильтрации изображений часто ядро свертки является разделяемым или сепарабельным, т.е. может быть получено с помощью произведения векторов горизонтального и вертикального ядер. Это свойство существенно снижает вычислительную сложность алгоритма – с квадратичной до линейной. Примером такого ядра является фильтр Гаусса (размы-

тие). Более детально реализация двумерного фильтра с разделяемым ядром описана в работе [15].

При больших размерах ядер, как одномерных, так и двумерных, для снижения объема вычислений применяется теорема о свертке, которая позволяет заменить множество скалярных произведений во временной области поэлементным умножением векторов в частотной области (и наоборот). Реализация фильтра в частотной области не представляет особых сложностей: достаточно произвести БПФ над входными данными и ядром, перемножить полученные спектры и произвести обратное БПФ для получения результата. При этом важно заранее достроить ядро и входной вектор нулями с любой стороны так, чтобы их размерность совпадала и равнялась ожидаемой размерности результата. Это связано с тем, что БПФ представляет входной сигнал как бесконечно повторяющуюся последовательность, и если размерность исходного сигнала не увеличить, то результатом будет циклическая свертка, а не линейная. Реализация двумерного фильтра при помощи БПФ из библиотеки CUFFT описана в работе [16].

Как отмечено ранее, производительность БПФ на GPU в большинстве случаев ограничена производительностью системы глобальной памяти. Это же верно и для реализации КИХ-фильтров через БПФ, а также при прямой фильтрации сигналов короткими ядрами. При увеличении размеров ядра производительность становится ограниченной разделяемой памятью, т.к. на каждые два чтения из разделяемой памяти приходится всего одна операция умножения с накоплением.

При реализации КИХ-фильтров через БПФ есть два способа повысить производительность по сравнению с «наивным» вариантом, описанным выше. Во-первых, ядро свертки чаще всего постоянно, и поэтому его спектр можно рассчитать один раз и сохранить в памяти GPU для всех последующих операций. Во-вторых, можно заметить, что расположение данных в памяти устройства на последней стадии прямого БПФ и первой стадии обратного БПФ, скорее всего, одинаковое, и поэтому можно обойтись без сохранения данных в глобальной памяти. То есть, произвести последнюю стадию прямого БПФ, умножение спектров и первую стадию обратного БПФ в одной программе на устройстве. Однако этот способ, хотя и позволяет повысить производительность, требует реализации собственного кода БПФ, так как существующие библиотеки не поддерживают такое «слияние» программ.

Из библиотек, реализующих фильтры и подобные операции, стоит отметить NPP от NVIDIA [17], в состав которой входит огромное количество функций для обработки сигналов и изображений, включая свертки, корреляции, гистограммы, геометрические преобразования изображений, операции дискретного косинусного преобразования для компрессии изображений, кластеризации и многое другое. Из открытых разработок можно назвать библиотеку GPUVSIP (Vector Signal Image Processing Library [18]), обладающую значительно более скромным функционалом.

Операции линейной алгебры

В обработке сигналов операции линейной алгебры находят довольно ограниченное применение. Наиболее распространенной операцией, вероятно, является умножение матрицы на вектор для расчета нейросетей и построения других алгоритмов. При умножении матрицы A размером $M \times K$ на вектор B из K элементов получается вектор C из M элементов, который определяется выражением

$$C_m = \sum_{i=1}^K A_{m,i} B_i.$$

Реализация этой операции на GPU не составляет сложности: достаточно рассчитывать каждым потоком один элемент вектора-результата, загружая данные матрицы по мере необходимости и расположив данные вектора или их фрагмент в разделяемой памяти. При этом, ввиду небольшого количества производимых вычислений, производительность будет ограничена системой глобальной памяти. Но такой подход хорошо работает лишь в тех случаях, когда вектор результата имеет большую размерность M – несколько тысяч элементов. И даже при таких размерах требуется читать несколько ячеек матрицы в каждом потоке одновременно, чтобы обеспечить эффективную загрузку глобальной памяти (это связано с ее латентностью и максимальным количеством активных потоков в GPU, см. закон Литтла). Более детальные сведения об оптимизации матрично-векторного умножения можно найти в работе [19].

При меньших значениях M , но больших значениях K имеется возможность разделить эту операцию по измерению K на несколько матрично-векторных произведений с последующим суммированием результатов. Если же оба измерения малы, то единственной возможностью эффективно загрузить GPU работой является выполнение множеств одинаковых операций параллельно. Например, в алгоритмах трехмерной графики очень часто применяются операции с векторами из 4 элементов и матрицами размерностью 4×4 – но они параллельно выполняются для каждой обрабатываемой вершины или каждого пикселя формируемого изображения, и каждый поток выполняет такую операцию последовательно.

Перемножение матриц также эффективно реализуется на GPU, но только при достаточно большой размерности результата или больших группах независимых задач. Производительность этого алгоритма уже может быть ограничена вычислительными возможностями устройства, а не системой памяти. Чтобы разгрузить систему памяти, применяется буферизация данных исходных матриц в разделяемой памяти и регистрах – то есть, используется так называемый register blocking – когда каждый поток производит расчет нескольких независимых элементов результата. Более подробно оптимизация перемножения матриц на GPU рассмотрена в работе [19].

Среди имеющихся реализаций операций линейной алгебры стоит отметить библиотеку CUBLAS от NVIDIA [21] и библиотеку clAmdBlas, входящую в состав AMDAPPML [13], а также библиотеку MAGMA (Matrix Al-

gebraon GPU and Multicore Architectures) [22] с открытым исходным кодом, поддерживающую как CUDA, так и OpenCL. Они реализуют разнообразные операции с плотными и разреженными векторами и матрицами. Однако большинство этих операций ориентированы только на большие задачи, и для обработки множества небольших задач более эффективным решением может быть создание специальной реализации.

Границы применимости GPU.

Сравнение с другими классами устройств

Обработка сигналов традиционно производится на устройствах, специально для этого предназначенных. Для законченных серийных устройств это обычно специализированные системы на кристалле (СНК), содержащие сложнофункциональные блоки, оптимизированные под выполнение конкретных алгоритмов. Часто применяются и менее специализированные DSP-процессоры, производительность которых варьируется в широких пределах и достигает единиц GFLOPS. В отдельных случаях применяются микросхемы программируемой логики (FPGA), позволяющие реализовать необходимый алгоритм аппаратно за умеренную цену и с высокой производительностью. Где же в этом разнообразии место для GPU?

Ответ на этот вопрос можно найти, рассмотрев сильные и слабые стороны GPU при обработке сигналов. К сильным сторонам относится высочайшая производительность, которую можно получить на одном чипе; относительно невысокая стоимость – менее 20,000 рублей за видеокарту потребительского класса; наличие удобных средств программирования и отладки, а также библиотек. К слабым сторонам – повышенные требования к размеру задач, необходимому для обеспечения эффективной работы; затрудненное применение во встраиваемых системах, ввиду форм-фактора и требований к наличию шины PCI-Express (PCIe) и драйверов под используемый процессор и ОС, а также необходимости отвода около 200 ватт тепла.

С учетом приведенных плюсов и минусов, можно назвать следующие возможности применения GPU для обработки сигналов.

1. Системы обработки данных с массивов радиоприемников. К примеру, в работе [23] описывается применение GPU для построения коррелятора сигналов с массива приемников интерферометра.

2. Кодирование и декодирование видео, обработка видео для улучшения качества – эти применения уже стали привычными для многих пользователей ПК. Стоит отметить проект SVP [24], который использует GPU для увеличения частоты кадров видео, воспроизводимого на ПК методом интерполяции движения в реальном масштабе времени. Однако для кодирования видео применение специальных аппаратных блоков, встроенных в последние модели GPU и CPU, является более эффективным решением, чем применение CUDA или OpenCL.

3. Обработка изображений: фильтры (размытие, увеличение резкости, поиск граней), построение гистограмм и автоматическое повышение контраста, дис-

кретное косинусное преобразование, геометрические преобразования, устранение размытости методом обратной свертки [25]. В современных графических пакетах, таких как Adobe Photoshop, некоторые алгоритмы уже реализованы с применением GPU.

4. Системы анализа видеопотоков, например, для идентификации номеров автомобилей или определения оставленных в общественном месте предметов. Популярная библиотека алгоритмов компьютерного зрения с открытым исходным кодом OpenCV [26] поддерживает ускорение работы при помощи GPU.

5. Ускорение работы прикладных пакетов программ, используемых при проектировании систем обработки сигналов. Существуют библиотеки для популярного программного комплекса MATLAB, перекладывающие многие расчетные задачи на GPU: Parallel Computing Toolbox от MathWorks [27] и Jacketot AccelerEyes [28].

На рис. 3 приведено количественное сравнение производительности GPU с другими классами устройств при выполнении БПФ-1024 в числах с плавающей точкой одинарной точности. Данные о производительности процессоров Texas Instruments (TI) и Analog Devices (ADI) получены с сайтов производителей, а о производительности системы ЭЛВИС «Мультикор» – из статьи [29]. Данные о специализированных БПФ-процессорах получены из руководства [30] (FPGA, использован самый быстрый из приведенных вариантов, работающих с данными с плавающей точкой) и статьи [31] (ASIC, работает с данными с фиксированной точкой). Производительность CPU Intel Xeon приведена на сайте библиотеки FFTW [12], а производительность GPU NVIDIA измерена автором данной статьи.

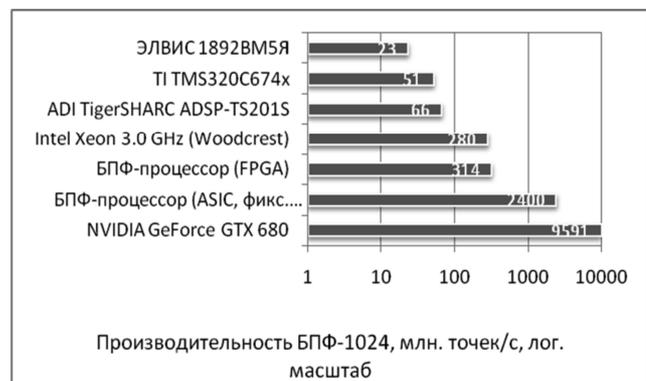


Рис. 3. Сравнение производительности БПФ-1024 на разных классах устройств

Сравнение энергоэффективности, то есть производительности в расчете на единицу потребляемой энергии, является значительно более сложной задачей. Для ее выполнения необходимо измерить мощность, потребляемую устройством при выполнении конкретной программы с определенными входными данными, так как каждая программа нагружает блоки устройства по-разному, а обработка случайных данных приводит к большему числу переключений логических элементов, чем обработка нулей и, как следствие, большей динамической потребляемой мощности. Поэтому сравним лишь теоретические значения с учетом указанных производителями значений максимальной производительности и

мощности. Значения для GPU приведены в табл. 1 и находятся на уровне 15 GFLOPS/Вт. У современных CPU этот показатель находится на уровне 0.5–0.7 GFLOPS/Вт, а у DSP-процессора ADI TigerSHARC ADSP-TS201S при 600 МГц – 1.03 GFLOPS/Вт.

Произвести количественное сравнение по таким важным показателям, как стоимость разработки и, особенно, удобство программирования, очень сложно. Качественное сравнение параметров различных классов устройств приведено в табл. 2.

Заключение

Графические процессоры с поддержкой вычислений общего назначения и иерархической системой произвольно адресуемой памяти появились в 2006 году и стремительно завоевали популярность в качестве универсальных вычислителей. В настоящее время они при-

меняются во многих отраслях науки и техники и позволяют получить ускорение в 10–100 раз при решении ряда тяжелых вычислительных задач по сравнению с центральными процессорами общего назначения. Для программирования этих устройств существуют специальные языки и средства отладки, а также качественные библиотеки, реализующие многие часто встречающиеся функции.

Рассмотренные в статье базовые алгоритмы используются в основе множества задач DSP. При реализации этих задач на GPU можно получить многократное ускорение по сравнению с большинством других устройств: CPU, FPGA, процессорами DSP. Во множество задач цифровой обработки сигналов, решаемых на GPU, входит обработка видео и звука, обработка изображений, анализ видеопотоков с камер наблюдения, анализ данных с массивов радиоприемников и ускорение работы систем разработки и прототипирования алгоритмов DSP.

Таблица 2

Сравнение основных характеристик различных классов устройств

Тип устройства	DSP процессоры	Спец. блокиFPGA	Спец. блокиASIC	CPUобщего назначения	GPU
Производительность	Средняя	Высокая	Очень высокая	Высокая	Очень высокая
Энергоэффективность	Высокая	Средняя	Очень высокая	Низкая	Очень высокая
Стоимость разработкипрограммы или блока	Низкая – Средняя	Высокая	Очень высокая	Низкая	Средняя
Стоимость устройства	Низкая	Низкая – Средняя	Зависит от объема производства	Средняя	Средняя
Удобство программирования	Среднее – Высокое	Специализированные блоки обычно не программируются		Высокое	Среднее – Высокое
Требования к вспомогательным устройствам	Минимальные	ПЗУ для конфигурации	Минимальные	Системная плата, ОЗУ, ПЗУ	CPU и пр., поддержка PCIe

Литература

- CUDA C Programming Guide // NVIDIA, 2012.
- CUDA C Best Practices Guide // NVIDIA, 2012.
- The OpenCL Specification, Version 1.2 // Khronos Group, 2011.
- AMD Accelerated Parallel Processing OpenCL Programming Guide // AMD, 2012.
- Micikevicius P. GPU Performance Analysis and Optimization // GPU Technology Conference 2012, NVIDIA.
- Wong H., Papadopoulou M. et. al. Demystifying GPU Microarchitecture through Microbenchmarking // IEEE International Symposium on Performance Analysis of Systems & Software (ISPASS), 2010.
- Burrus C.S. Fast Fourier Transforms//Houston, Texas: Connexions, 2008.
- Volkov V., Kazian B. Fitting FFT onto the G80 architecture. URL: http://www.cs.berkeley.edu/~kubitron/courses/cs258-S08/projects/reports/project6_report.pdf.
- Govindaraju N., Lloyd B. et al. High Performance Discrete Fourier Transforms on Graphics Processors // Proceedings of the 2008 ACM/IEEE conference on Supercomputing.
- Nukada A. Fast Fourier Transforms for AMD GPUs // AMD Fusion Developer Summit, 2011.
- Библиотека CUFFT. URL: <http://developer.nvidia.com/cufft>.
- Библиотека FFTW. URL: <http://www.fftw.org/>.
- Библиотека AMD APPML. URL: <http://developer.amd.com/libraries/appmathlibs/pages/default.aspx>.
- Apple OpenCL_FFT Sample. URL: http://developer.apple.com/library/mac/#samplecode/OpenCL_FFT/Introduction/Intro.html.
- Podlozhnyuk V. Image Convolution with CUDA// NVIDIA, 2007.
- Podlozhnyuk V. FFT-based 2D Convolution // NVIDIA, 2007.
- Библиотека NVIDIANPP. URL: <http://developer.nvidia.com/npp>.
- Библиотека GPU VSIPL. URL: <http://gpu-vsip1.gtri.gatech.edu>.
- Sorensen H. Auto-Tuning Dense Vector and Matrix-Vector Operations for Fermi GPUs. URL: http://gpulab.imm.dtu.dk/papers/Sorensen2012_PPAM.pdf.
- Volkov V., Demmel J. Benchmarking GPUs to Tune Dense Linear Algebra // International Conference for High Performance Computing, Networking, Storage and Analysis, 2008.
- Библиотека CUBLAS. URL: <http://developer.nvidia.com/cublas>.
- Библиотека MAGMA. URL: <http://icl.cs.utk.edu/magma/index.html>.
- Harris C., Haines K., Stavaley-Smith L. GPU Accelerated Radio Astronomy Signal Convolution // Experimental

Astronomy, Volume 22, Issue 1-2, pp. 129-141, 2008.

24. Smooth Video Project (SVP). URL: <http://www.svp-team.com/>.

25. Mazanec T., Hermanek A., Kamenicky J. Blind Image Deconvolution Algorithm on NVIDIA CUDA Platform // IEEE 13th International Symposium on Design and Diagnostics of Electronic Circuits and Systems (DDECS), 2010.

26. Библиотека OpenCV. URL: <http://opencv.willowgarage.com>.

27. Matlab Parallel Computing Toolbox. URL: <http://www.mathworks.com/products/parallel-computing/index.html>.

28. Библиотека AccelerEyes Jacket. URL: http://www.accelereyes.com/jacket_tour.

29. Солохина Т., Александров Ю. и др. Отечественные трехъядерные сигнальные микроконтроллеры с производительностью 1.5 GFLOPS // Электронные компоненты – №6 2006. С. 73–78.

30. Altera FFT MegaCore Function User Guide // Altera, 2011.

31. Song-Nien Tang, Jui-Wei Tsai, Tsin-Yuan Chang. A 2.4-GS/s FFT Processor for OFDM-Based WPAN Applications // IEEE Transactions on Circuits and Systems II: Ex-

press Briefs, Vol. 57, No. 6, 2010.

DIGITAL SIGNAL PROCESSING ON MBOODERN GPUS

Pantelev A.

Modern graphics processing units (GPUs) have much greater computational power than any other kinds of general purpose programmable devices, and have an affordable price at the same time. This paper discusses the implementation of the basic digital signal processing (DSP) algorithms on GPUs: fast Fourier transform (FFT), convolution and correlation, matrix-vector multiplication. References to the libraries that provide quality implementations of these and many other algorithms are made. For each algorithm, performance limiting factors are identified; in most cases, the performance is limited by memory bandwidth rather than compute capabilities of the devices. The performance of GPUs calculating FFT is up to 9 billion floating point samples per second, which exceeds the usual DSP processors by a factor greater than 100, and high-performance specialized hardware units by a factor of 2 to 20. The exemplar usages of GPUs in digital signal processing systems are shown.

15-я Международная конференция

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ЕЕ ПРИМЕНЕНИЕ

26 марта – 29 марта 2013 г., Москва, Россия

ИНФОРМАЦИОННОЕ ПИСЬМО

Уважаемый коллега!

Приглашаем Вас принять участие в работе 15-й Международной конференции «Цифровая обработка сигналов и ее применение- DSPA'2013», которая состоится в ИПУ РАН 26 марта – 29 марта 2013 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 ГРЧ
- 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)

МЕТОД ТЕКСТУРНОЙ СЕГМЕНТАЦИИ ИЗОБРАЖЕНИЙ НА ОСНОВЕ МАРКОВСКИХ СЛУЧАЙНЫХ ПОЛЕЙ

*Медведева Е.В., к.т.н., доцент кафедры радиоэлектронных средств Вятского государственного университета,
e-mail: EMedv@mail.ru*

*Курбатова Е.Е., аспирант кафедры радиоэлектронных средств Вятского государственного университета,
e-mail: kurbatovae@gmail.com*

Ключевые слова: текстурная сегментация, цифровые полутоновые изображения, марковский процесс, энтропия, оценка вероятности перехода.

Введение

В системах космического мониторинга поверхности Земли для решения задач анализа изображений широко применяется сегментация, которая позволяет оперативно выделить на изображениях интересные области. Часто естественные сцены, такие как леса, поля лишены существенных деталей на больших пространствах и не имеют четко выраженных границ. В этом случае необходимо использовать идентификацию областей изображений на основе анализа их текстуры.

В ряде работ [1] показано, что для описания текстуры могут быть использованы марковские случайные поля. Однако при обработке цифровых полутоновых изображений (ЦПИ) с числом уровней яркости 2^g возникает проблема хранения в памяти и оперирования с матрицами вероятностей переходов размерностью $2^g \times 2^g$. Такая обработка ЦПИ требует больших вычислительных затрат.

В данной работе предложен метод текстурной сегментации ЦПИ на основе марковских случайных полей, позволяющий при меньших вычислительных затратах эффективно выделить протяженные области с однородными статистическими характеристиками.

Метод текстурной сегментации

Пусть g -разрядное ЦПИ представлено набором g разрядных двоичных изображений (РДИ). Полагается, что l -е РДИ ($l = \overline{1, g}$) представляет собой марковское случайное поле с разделимой автокорреляционной функций вида [2]:

$$r_{i,j} = \sigma_\mu^2 \exp\{-\alpha_1 |f| - \alpha_2 |s|\},$$

где i, j – дискретные пространственные координаты по горизонтали и по вертикали; σ_μ^2 – дисперсия двумерного дискретного марковского процесса; α_1, α_2 – множители, зависящие от ширины спектральной плотности мощности случайных процессов по двум измерениям; f, s – шаг корреляции по горизонтали и вертикали.

Предложен метод текстурной сегментации цифровых полутоновых изображений (ЦПИ) на основе двумерных цепей Маркова. В качестве текстурного признака используется оценка вероятности перехода между элементами изображения. Метод эффективно выделяет текстурные области с разными статистическими характеристиками и позволяет снизить вычислительные затраты.

В этом случае РДИ можно представить суперпозицией двух одномерных цепей Маркова по горизонтали и вертикали с двумя равновероятными значениями $M_1^{(l)}$, $M_2^{(l)}$ и матрицами вероятностей переходов по горизонтали ${}^1\Pi = \left\| \pi_{ij}^{(l)} \right\|_{2 \times 2}$ и вертикали ${}^2\Pi = \left\| \pi_{ij}^{(l)} \right\|_{2 \times 2}$.

На рис. 1 представлено одностороннее марковское случайное поле ЦПИ, разделенное на области $F_i^{(l)}$ ($i = \overline{1, 4}$), элементы которых являются цепью Маркова различной размерности. На рис. 2 представлен фрагмент двумерного l -го РДИ, соответствующего марковскому случайному полю области $F_4^{(l)}$, где приняты обозначения: $v_1^{(l)} = \mu_{i,j-1}^{(l)}$; $v_2^{(l)} = \mu_{i-1,j}^{(l)}$; $v_3^{(l)} = \mu_{i,j}^{(l)}$; $v_4^{(l)} = \mu_{i-1,j-1}^{(l)}$.

Выделение протяженных областей с разной текстурой на изображении сводится к вычислению их статистических характеристик: оценок вероятностей переходов по горизонтали ${}^1\hat{\pi}_{ii}^{(l)}$ и вертикали ${}^2\hat{\pi}_{ii}^{(l)}$ - для областей $F_2^{(l)}$, $F_3^{(l)}$ и оценки ${}^1\hat{\pi}_{iii}^{(l)}$ вероятности переходов в двумерной цепи Маркова - для области $F_4^{(l)}$. Предполагается, что текстурный признак (вероятности перехода между элементами ЦПИ) внутри одной области изменяется незначительно, а на разных областях принимает существенно различные значения. Учитывая, что основные детальные области выражены на старших разрядах ЦПИ, для их выделения необходимо обработать РДИ старших разрядов ЦПИ с наиболее выраженными текстурными признаками.

Вычисление оценки ${}^1\hat{\pi}_{ii}^{(l)}$ вероятности перехода по горизонтали сводится к вычислению средней длины последовательности элементов одного знака [3]:

$${}^1\hat{\pi}_{ii}^{(l)} = 1 - \frac{2p_1^{(l)}}{\hat{\chi}^{(l,r)}}, \quad (2)$$

где $\hat{\chi}^{(l,r)}$ – оценка средней длины последовательности одинаковых элементов l -го разряда на r -м шаге уточнения оценки.

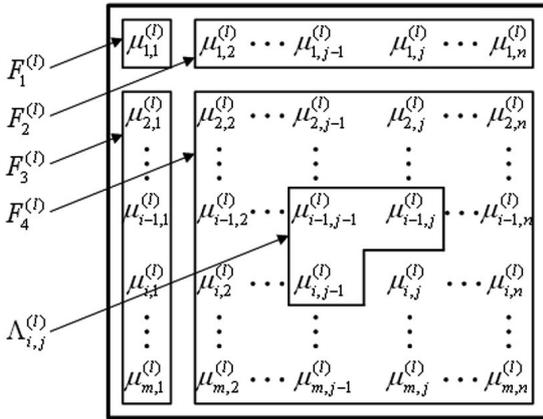


Рис. 1. Области одностороннего марковского случайного поля

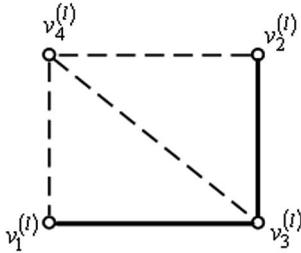


Рис. 2. Фрагмент области $F_4^{(l)}$

Для определения вероятности перехода $\pi_{iii}^{(l)}$ в двумерной цепи Маркова воспользуемся многомерной математической моделью изображения, синтезируемой в [4]. Вероятность состояния элемента $v_3^{(l)}$ области $F_2^{(l)}$ (рис. 1) определится энтропией элемента $v_3^{(l)}$ относительно состояний элементов ближайшей окрестности $\Lambda_{ij}^{(l)} = \{v_1^{(l)}, v_2^{(l)}\}$:

$$H(v_3^{(l)} | v_2^{(l)}, v_1^{(l)}) = -\log \frac{w(v_3^{(l)} | v_1^{(l)}) w(v_3^{(l)} | v_2^{(l)})}{w(v_3^{(l)} | v_2^{(l)}, v_1^{(l)})}, \quad (3)$$

где $w(v_3^{(l)} | v_1^{(l)})$, $w(v_3^{(l)} | v_2^{(l)})$ – одномерные плотности вероятностей перехода между соседними значениями; $w(v_3^{(l)} | v_1^{(l)}, v_2^{(l)})$ – плотность вероятности перехода в двумерной цепи Маркова.

Плотность вероятности перехода в двумерной двоичной цепи Маркова можно выразить в форме:

$$w(v_3^{(l)} | v_2^{(l)}, v_1^{(l)}) = \sum_{i,j,q=1}^2 \pi(v_3^{(l)} = M_i^{(l)} | v_1^{(l)} = M_j^{(l)}; v_2^{(l)} = M_q^{(l)}) \delta(v_1^{(l)} - M_j^{(l)}) \delta(v_2^{(l)} - M_q^{(l)}), \quad (4)$$

где $\delta(\cdot)$ - дельта функция.

На основании аргумента выражения (3) матрица $\Pi^{(l)}$ для различных сочетаний состояний элементов

окрестности $\Lambda_{ij}^{(l)}$ имеет вид:

$$\Pi^{(l)} = \begin{pmatrix} \pi_{iii}^{(l)} & \pi_{ijj}^{(l)} \\ \pi_{jii}^{(l)} & \pi_{jjj}^{(l)} \\ \pi_{jii}^{(l)} & \pi_{jjj}^{(l)} \\ \pi_{jii}^{(l)} & \pi_{jjj}^{(l)} \end{pmatrix} = \begin{pmatrix} \alpha_1^{(l)} & \alpha_1^{(l)} \\ \alpha_2^{(l)} & \alpha_2^{(l)} \\ \alpha_3^{(l)} & \alpha_3^{(l)} \\ \alpha_4^{(l)} & \alpha_4^{(l)} \end{pmatrix}, \quad (5)$$

$i, j = \overline{1, 2}; i \neq j, l = \overline{1, g}$,

элементы которой симметричны и удовлетворяют условию нормировки $\alpha_q + \alpha_q^{(l)} = 1$.

Элементы матрицы $\Pi^{(l)}$ (3) связаны с элементами матриц ${}^1\Pi^{(l)}$ и ${}^2\Pi^{(l)}$ следующими соотношениями

$$\begin{aligned} \alpha_1^{(l)} &= \pi_{iii}^{(l)} = \pi(v_3^{(l)} = M_1^{(l)} | v_1^{(l)} = M_1^{(l)}; v_2^{(l)} = M_1^{(l)}) = M_1^{(l)} = {}^1\pi_{ii}^{(l)} \quad {}^2\pi_{ii}^{(l)} / {}^3\pi_{ii}^{(l)}; \\ \alpha_4^{(l)} &= 1 - \alpha_1^{(l)}; \\ \alpha_2^{(l)} &= \pi_{jii}^{(l)} = \pi(v_3^{(l)} = M_1^{(l)} | v_1^{(l)} = M_1^{(l)}; v_2^{(l)} = M_2^{(l)}) = M_2^{(l)} = {}^1\pi_{ii}^{(l)} \quad {}^2\pi_{ij}^{(l)} / {}^3\pi_{ij}^{(l)}; \\ \alpha_3^{(l)} &= 1 - \alpha_2^{(l)}, \end{aligned} \quad (6)$$

где ${}^3\pi_{ij}^{(l)}$ – элемент матрицы вероятностей переходов ${}^3\Pi^{(l)} = {}^1\Pi^{(l)} \times {}^2\Pi^{(l)}$.

Начиная со второй строки РДИ, используя множество элементов $\psi = \{v_1^{(l)}, v_2^{(l)}, v_3^{(l)}, v_4^{(l)}\}$ и ранее вычисленную оценку ${}^1\hat{\pi}_{ii}^{(l)}$ вероятности переходов по горизонтали, вычисляются оценки ${}^2\hat{\pi}_{ii}^{(l)}$ и $\hat{\pi}_{iii}^{(l)}$.

Учитывая локальные изменения статистических характеристик на изображении, для их вычисления используется метод «сканирующего окна» [5]. Размер окна задается исходя из заданной точности оценки и минимальных вычислительных затрат.

При обработке первых строк и столбцов изображения размер окна будет увеличиваться по мере увеличения номера обрабатываемого элемента, пока не достигнет заданных размеров. Размер окна на каждом этапе обработки определяется из пары соотношений:

$$\begin{aligned} m &= \overline{1, (2i-1)}, \quad i = \overline{2, (M-1)/2}; \\ n &= \overline{1, (2j-1)}, \quad j = \overline{2, (N-1)/2}, \end{aligned} \quad (7)$$

где i, j – координаты обрабатываемого изображения, M, N – заданные ширина и высота сканирующего окна.

Усредняя оценки вероятностей переходов в пределах окна, вычисляется оценка средней вероятности перехода $\tilde{\pi}_{iii}^{(l)}$ для элемента, соответствующего центральному элементу окна:

$$\tilde{\pi}_{iii}^{(l,r,k)} = \frac{1}{m \cdot n} \sum_{r=1}^m \sum_{k=1}^n \hat{\pi}_{iii}^{(l,r,k)}. \quad (8)$$

На каждом последующем шаге окно перемещается на один элемент (слева-направо, сверху-вниз) и вычис-

ляется оценка средней вероятности перехода для каждого элемента РДИ.

Для выделения областей с разной текстурой вычисленная оценка $\tilde{\pi}_{iii}^{(l)}$ сравнивается с порогом. Пороговым значением H между двумя разными текстурными областями является оценка $\tilde{\pi}_{iii}^{(l)}$, которая для текстурных областей с разными вероятностными характеристиками может быть выбрана на основе анализа гистограмм изображения [6]. В случае, когда априорно известно, что изображение содержит области двух текстур с разными статистическими характеристиками, для их выделения будет достаточно двух меток (0 и 1). Всем элементам, для которых оценка $\tilde{\pi}_{iii}^{(l)}$ превышает порог, присваивается метка 1, остальным элементам – метка 0.

В случае, когда ЦПИ содержит несколько текстур, необходимо каждой из них присвоить свою метку. При этом используется несколько пороговых значений, соответствующих нескольким текстурам на изображении.

Результаты моделирования

Моделирование проводилось на искусственных и реальных изображениях. Формирование искусственных двоичных изображений осуществлялось с использованием двумерной математической модели изображения (3) и алгоритма, приведенного в [4].

Для оценки качества выделения текстурных областей вычислялось количество ошибочно сегментированных элементов (ESE), для определения которых сегментированное изображение сравнивалось с разметкой. Разметка – это изображение, используемое для формирования текстурного изображения, содержащее области с элементами одной яркости.

Качество выделения текстурных областей определяется размером скользящего окна. На рис. 3 представлен график зависимости количества ошибочно сегментированных элементов (ESE) от размера скользящего окна для разных значений вероятностей переходов, сегментируемых структур.

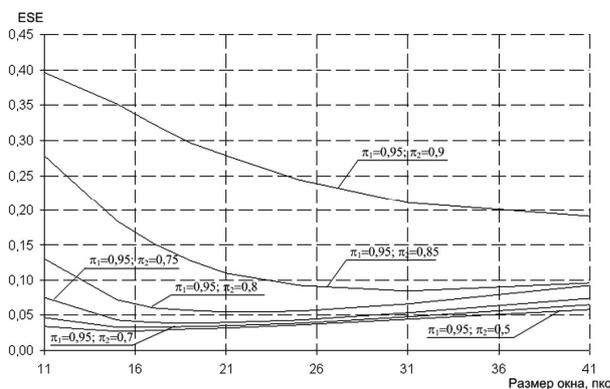


Рис. 3. Влияние размера окна на качество сегментации

Из приведенного графика следует, что размер окна зависит от вероятностей переходов текстур, которые присутствуют на изображении. Чем большие вероятности переходов имеют текстуры, тем больший размер окна требуется для их сегментации. Малые окна приво-

дят к значительной неоднородности сегментированных областей, но при этом позволяют точно локализовать границы областей.

Окна больших размеров обеспечивают однородность сегментированных областей, но при этом не удается точно определить границы областей. Приемлемым для большинства текстур и наиболее эффективным с точки зрения соотношения качество сегментации/время обработки является окно размером 21x21.

На рис. 4в приведен пример сегментации искусственного изображения размером 1024x1024 с четырьмя текстурными областями (рис. 4б), сгенерированного по разметке (рис. 4а), а также гистограмма текстурного признака изображения (рис. 4г). Изображение содержит области четырех текстур с матрицами вероятности переходов равными:

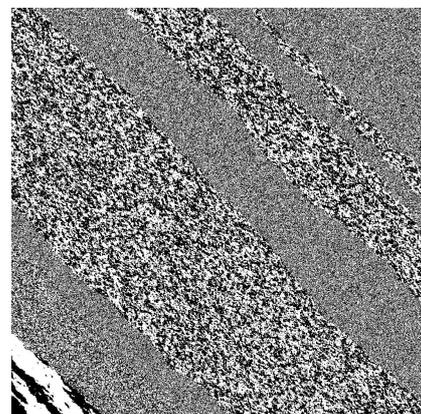
$${}^1\Pi = {}^2\Pi = \begin{bmatrix} 0,95 & 0,05 \\ 0,05 & 0,95 \end{bmatrix}, \quad {}^1\Pi = {}^2\Pi = \begin{bmatrix} 0,75 & 0,25 \\ 0,25 & 0,75 \end{bmatrix},$$

$${}^1\Pi = {}^2\Pi = \begin{bmatrix} 0,6 & 0,4 \\ 0,4 & 0,6 \end{bmatrix} \text{ и } {}^1\Pi = {}^2\Pi = \begin{bmatrix} 0,5 & 0,5 \\ 0,5 & 0,5 \end{bmatrix}.$$

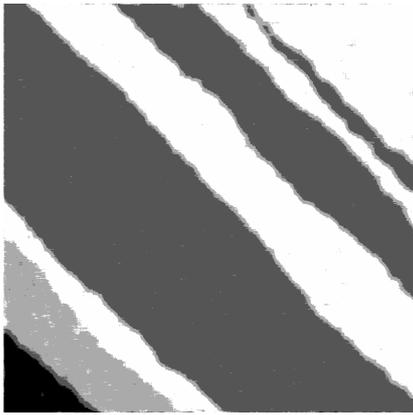
Количество пиков на гистограмме текстурного признака изображения соответствует количеству текстур на текстурном изображении. Пороговое значение выбирается как минимальное значение между двумя соседними пиками гистограммы. Выбрав порог по тестовому набору изображений с разными статистическими характеристиками, данное пороговое значение можно использовать для сегментации других изображений, содержащих области с такими же статистическими характеристиками.



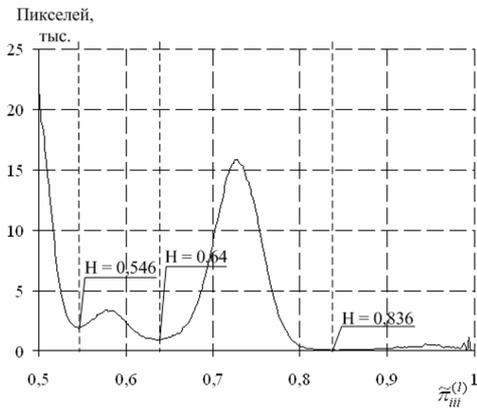
а)



б)



а)

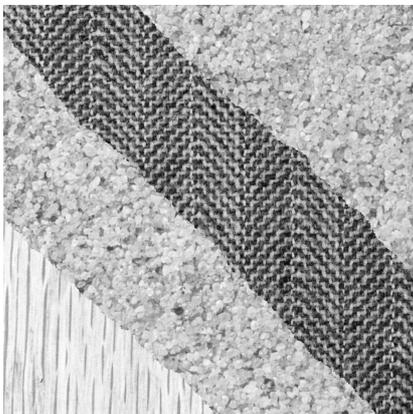


г)

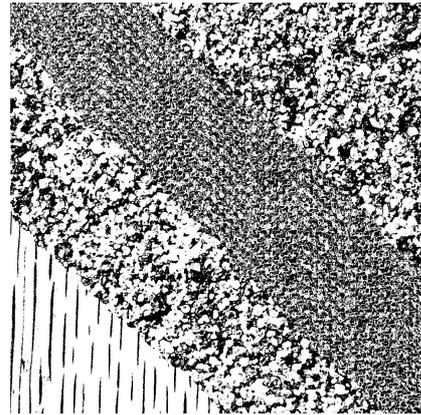
Рис. 4. Сегментация искусственных двоичных изображений:
а) разметка; б) текстурное РДИ;
в) сегментированное изображение;
г) гистограмма текстурного признака

Количество ошибочно сегментированных элементов в приведенном примере составляет 5,2%.

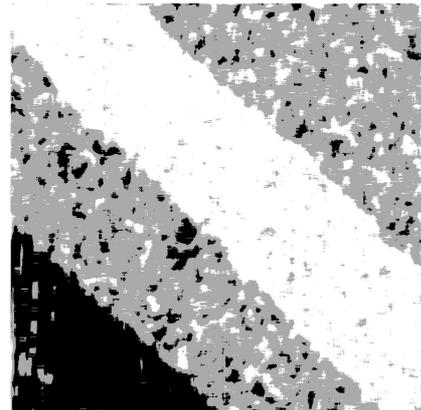
Для текстурной сегментации ЦПИ выбираются РДИ, на которых наиболее выражены текстурные области. На рис. 5 приведен пример сегментации изображения, полученного наложением естественных текстур на изображение разметки. Изображения естественных текстур получены из альбома текстур Бродатца [7], содержащий текстуры, встречающиеся в природе. Размер изображения 600х600. Сегментация осуществлялась по 7-му РДИ.



а)



б)



в)

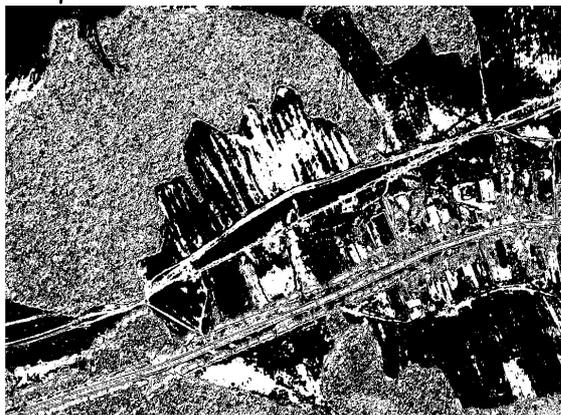
Рис. 5. Сегментация ЦПИ естественных текстур:
а) ЦПИ; б) РДИ седьмого разряда ЦПИ;
в) сегментированное изображение

Количество ошибочно сегментированных элементов (рис.4) составляет 12,7%. Качество сегментации естественных текстур несколько хуже, чем искусственных изображений. Это связано с неравномерностью освещения естественной сцены при съемке, что приводит к возникновению на изображении областей неоднородной яркости и, соответственно, к неточностям в результатах сегментации.

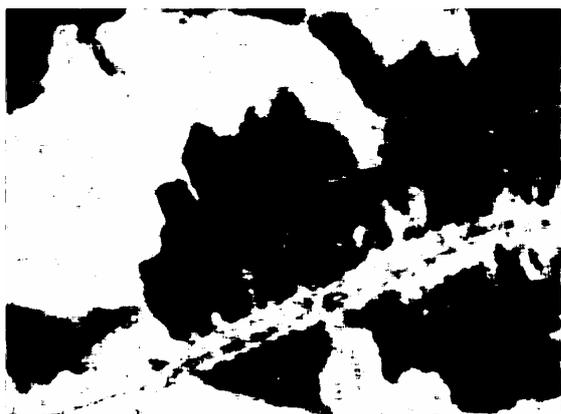
На рис. 6 приведен пример сегментации спутникового изображения (размер 1143х844), на котором эффективно выделена область леса. Сегментация осуществлялась по РДИ шестого разряда ЦПИ.



а)



б)



в)

Рис.6. Сегментация спутникового изображения:

а) ЦПИ; б) РДИ шестого разряда ЦПИ;

в) сегментированное изображение

Разработанный метод текстурной сегментации на основе марковских случайных полей эффективно выделяет протяженные области на ЦПИ с различной текстурой и позволяет снизить вычислительные ресурсы. Количество ошибочно сегментированных элементов составляет 2-14%. Разработанный метод может быть применен и для сегментации цветных изображений.

Литература

1. Ковтун И.В. Текстурная сегментация изображений на основе марковских случайных полей // Управляющие системы и машины. 2003. – № 4. – С. 46-55.
2. Хабиби, А. Двумерная байесовская оценка изображений / А. Хабиби // ТИИЭР. – 1972. - Т.60, №7. - С. 153-159.
3. Медведева Е.В. Метод вычисления информационных характеристик цифровых полутоновых изображений / Е. В. Медведева, Е.П. Петров // Инфокоммуникационные технологии, т.6, №3, 2008. - С.104-109.
4. Петров Е.П. Метод синтеза математических моделей видеоизображений на основе многомерных цепей Маркова / Е.П. Петров, Е. В. Медведева, А.П. Метелев // Нелинейный мир, № 4, 2011. - С.213-231.
5. Колодникова Н.В. Обзор текстурных признаков для задач распознавания образов / Н.В. Колодникова // Доклады ТУСУРа. Автоматизированные системы обработки информации, управления и проектирования. – 2004. – с. 113-124.
6. Шапиро Л. Компьютерное зрение / Л. Шапиро, Дж. Стокман; Пер. с англ. – М.: БИНОМ. Лаборатория знаний, 2006. – 752 с.
7. Brodatz Textures // Signal and Image Processing Institute USC – URL: <http://sipi.usc.edu/database/database.php?volume=textures>. – 24.02.2012.

METHOD OF TEXTURAL SEGMENTATION OF IMAGES BASED ON MARKOV CASUAL FIELDS

Medvedeva E.V., Kurbatova E.E.

The method of textural segmentation of digital halftone images (DHI) on the basis of two-dimension Markov chains is offered. The estimate of transition probability between elements of image is used as textural features. This method makes it possible to detect efficiently the extensive areas on the DHI with different textures and reduces computational resources.

Уважаемые коллеги!

Для тех, кто не успел оформить подписку на второе 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.dspra.ru.

Справки по телефонам: (495) 621-71-08, 621-06-10.

Факс: (495) 621-16-39. E-mail: rntores@mail.ru