



**ТЕЛЕКОММУНИКАЦИИ: СЕТИ И ТЕХНОЛОГИИ,**  
**АЛГЕБРАИЧЕСКОЕ КОДИРОВАНИЕ**  
**И БЕЗОПАСНОСТЬ ДАННЫХ**

**МАТЕРИАЛЫ МЕЖДУНАРОДНОГО НАУЧНО-ТЕХНИЧЕСКОГО СЕМИНАРА**

**(Минск, апрель – декабрь 2016 г.)**

**В двух частях**

**ЧАСТЬ 1**



Министерство образования Республики Беларусь  
Учреждение образования  
«Белорусский государственный университет  
информатики и радиоэлектроники»

**ТЕЛЕКОММУНИКАЦИИ: СЕТИ И ТЕХНОЛОГИИ,**  
**АЛГЕБРАИЧЕСКОЕ КОДИРОВАНИЕ**  
**И БЕЗОПАСНОСТЬ ДАННЫХ**

МАТЕРИАЛЫ МЕЖДУНАРОДНОГО НАУЧНО-ТЕХНИЧЕСКОГО СЕМИНАРА

(Минск, апрель – декабрь 2016 г.)

В двух частях

Часть 1

**TELECOMMUNICATIONS: NETWORKS AND TECHNOLOGIES,**  
**ALGEBRAIC CODING AND DATA SECURITY**

Минск БГУИР 2016

УДК 621.391:004.056.55  
ББК 32.811+32.973.26-018.2  
Т31

Руководитель семинара В. К. Конопелько

Редакционная коллегия:

М. Н. Бобов, А. А. Борискевич, В. Ф. Голиков, Л. Л. Ключев,  
В. А. Лабунов, Л. М. Лыньков, В. Ю. Цветков, Т. М. Гилицкая

Т31 **Телекоммуникации:** сети и технологии, алгебраическое кодирование и безопасность данных : материалы междунар. науч.-техн. семинара. В 2 ч. Ч. 1 (Минск, апрель – декабрь 2016 г.) = Telecommunications: Networks and Technologies, Algebraic Coding and Data Security. – Минск : БГУИР, 2016. – 88 с. : ил.  
ISBN 978-985-543-244-0 (ч. 1).

Сборник содержит статьи по телекоммуникациям: сетям телекоммуникаций и информационной безопасности в области научно-теоретических разработок и прикладных применений, алгебраическому кодированию и обработке изображений, программно-аппаратным и защитным средствам обеспечения в телекоммуникационных сетях.

Для научных сотрудников в области телекоммуникаций, преподавателей, аспирантов, магистрантов и студентов технических вузов.

*Научное издание*

Корректор *Т. В. Мироненко*

Ответственный за выпуск *В. К. Конопелько*

Компьютерный дизайн и верстка *Е. Г. Макейчик*

Подписано в печать 28.06.2016. Формат 60×84 1/8. Бумага офсетная. Гарнитура «Таймс». Отпечатано на ризографе. Усл. печ. л. 10,46. Уч.-изд. л. 7,3. Тираж 50 экз. Заказ 138.

Издатель и полиграфическое исполнение: учреждение образования «Белорусский государственный университет информатики и радиоэлектроники».

Свидетельство о государственной регистрации издателя, изготовителя, распространителя печатных изданий №1/238 от 24.03.2014,

№2/113 от 07.04.2014, №3/615 от 07.04.2014.

ЛП №02330/264 от 14.04.2014.

220013, Минск, П. Бровки, 6

ISBN 978-985-543-244-0 (ч. 1)  
ISBN 978-985-543-243-3

© УО «Белорусский государственный университет информатики и радиоэлектроники», 2016

## СОДЕРЖАНИЕ

<b>Шевчук О.Г.</b> Формирование геометрических дескрипторов линий и их идентификация .....	5
<b>Аль-Бахдили Х.К., Томилин В.В., Конопелько В.К., Аль-Заиди З.Х.М., Алкалби А.С.Дж</b> Вероятностное кодирование длин серий в сжатии полутоновых изображений .....	11
<b>Alzaki H.M.</b> Texture image segmentation based on classification, skeletonization and crossing of contour elements .....	18
<b>Макейчик Е.Г., Королёв А.И., Конопелько В.К.</b> Метод коррекции зависимых ошибок сверточными кодами с обнаружением некорректируемых пакетов .....	25
<b>Волков К.А., Карбалаи Салех Х.М., Парасочка А.В., Зайцев С.С., Романов Р.Р., Махмуд М.Б.М.</b> Моделирование движения беспилотного летательного аппарата с оптической навигационной системой .....	31
<b>Хоменок М.Ю.</b> Фурье-анализ произведения псевдослучайной последовательности косинусоидальных импульсов и ее задержанной копии .....	38
<b>Новицкий В.В., Цветков В.Ю. Аль-Гертани К.М.О., Аль-Аббуди А.А.Р.</b> Аппаратно-ориентированный алгоритм сжатия изображений на основе адаптивной иерархической кластеризации вейвлет-коэффициентов .....	43
<b>Садик Б.Дж., Бобов М.Н., Цветков В.Ю., Аль-Байати А.Э.К., Абдулхуссейн Х.М.А.</b> Сжатие мультиспектральных изображений на основе прогрессивного кодирования длин серий .....	51
<b>Панькова В.В., Саломатин С.Б.</b> Алгебро-геометрические коды проективных и аффинных пространств .....	56
<b>Almiahi O.M.</b> Exploration of stability for segmentation method based on wave region growing .....	63
<b>Волченков Д.В.</b> Эмуляция корпоративной IP-телефонии на основе OpenSource-приложений	70
<b>Кириллук Д.И., Кулаженко Ю.И.</b> Параметризация кривых линий на основе аппроксимации окружностями .....	76
<b>Аль-Машатт Е.А.А., Бойправ О.В., Лыньков Л.М., Мухуров Н.И.</b> Влияние формы отверстий фольгированных материалов на их характеристики отражения и передачи электромагнитного излучения .....	81

## CONTENTS

<b>Shevchuk O.G.</b> Formation of descriptors lines and their identification.....	5
<b>Al-Bahadily H.K., Tamin V.V., Kanapelka V.K., Al-Zaidi Z.H.M., Alkalbi A.S.J.</b> Probabilistic run length encoding for grayscale image compression.....	11
<b>Alzaki H.M.</b> Texture image segmentation based on classification, skeletonization and crossing of contour elements .....	18
<b>Makeichik E.G., Korolev A.I., Kanapelka V.K.</b> Method of error correction using dependent convolutional codes with the detected uncorrectable packages .....	25
<b>Volkov K.A., Karbalaie Saleh H.M., Parasochka A.V., Zaitsev S.S., Romanov R.R., Mahmood M.B.M.</b> Modeling of the movement of UAV with the optical navigation system.....	31
<b>Homenok M.J.</b> Fourier analysis of product for both a pseudorandom sequence of cosine pulses and its detained copy .....	38
<b>Navitski V.U., Tsviatkou V.Yu., Al-Gertani Q.M.O., Al-Abboodi A.A.R.</b> The hardware-oriented image compression algorithm based on adaptive hierarchical clustering of the wavelet coefficients ...	43
<b>Sadiq B.J., Bobov M.N., Tsviatkou V.Yu., Al-Baiati A.E.K., Abdulhussein H.M.A.</b> Compression of the multispectral images based on the progressive run length encoding.....	51
<b>Pankova V.V., Salomatin S.B.</b> Algebraic geometrical codes over projective and affine space .....	56
<b>Almiahi O.M.</b> Exploration of stability for segmentation method based on wave region growing.....	63
<b>Volchenkov D.V.</b> Emulation of corporate IP-telephony based on OpenSource applications .....	70
<b>Kirylyuk D.I., Kulazhenko Yu.I.</b> Parametrization of curves lines on images.....	76
<b>Almashat E.A.A., Boiprav O.V., Lynkou L.M., Mukhurov N.I.</b> Influence of the foil materials holes form on their electromagnetic radiation reflection and transmission characteristics .....	81

**ТЕЛЕКОММУНИКАЦИИ: СЕТИ И ТЕХНОЛОГИИ,  
АЛГЕБРАИЧЕСКОЕ КОДИРОВАНИЕ И БЕЗОПАСНОСТЬ ДАННЫХ**

УДК 539.216:546.824-31

**ФОРМИРОВАНИЕ ГЕОМЕТРИЧЕСКИХ ДЕСКРИПТОРОВ ЛИНИЙ И ИХ  
ИДЕНТИФИКАЦИЯ**

О.Г. ШЕВЧУК

*Белорусский государственный университет информатики и радиоэлектроники  
П. Бровки, 6, Минск, 220013, Беларусь**Поступила в редакцию 22 апреля 2016*

Предложен метод формирования дескриптора на основе гистограммы локальных ориентаций фрагментов выделенной линии, а также способ идентификации линий, используя предложенный метод. Проведен анализ разработанного метода при изменении яркости и контраста. Показано, что метод устойчив к изменению яркости.

*Ключевые слова:* детектирование линий, идентификация линий, дескриптор, гистограмма.

**Введение**

Основными задачами при сшивке изображений являются параметризация и идентификация ключевых объектов. В качестве ключевых объектов на изображении могут выступать: точка (пиксель), линия или сегмент изображения. Большинство методов идентификации и параметризации в качестве ключевых объектов используют точки (пиксели).

Параметризация детектированных ключевых объектов осуществляется с помощью дескрипторов. Наиболее распространенными дескрипторами являются методы, основанные на вычислении градиента яркости точки и ее окрестности, что позволяет идентифицировать области, края и углы. Такими методами являются SIFT[1], SURF[2], BRIEF[3], ORB[4], CLON[5, 6], DAISY[7] и др.

Однако данные методы неприменимы при сшивке изображений, которые имеют фрактальную или самоподобную структуру (море, пустыня, горы), а также в условиях резкой смены освещения (день и ночь), поэтому существует необходимость использования в качестве ключевых объектов выделенные линии, а в качестве дескрипторов – методы, основанные на расчете геометрических параметров детектированных линий таких изображений.

Цель работы – разработка метода формирования дескриптора на основе гистограммы локальных ориентаций фрагментов выделенной линии.

**Метод формирования дескриптора на основе построения гистограммы локальных ориентаций фрагментов выделенной линии**

Предложенный метод формирования дескриптора на основе построения гистограммы локальных ориентаций фрагментов выделенной линии, или DLOLF (Gistograma of local orientation of line fragments), в качестве ключевых элементов использует контурные или средние линии объектов. Обрабатываемая линия имеет две концевые точки, не имеет острых углов и должна быть предварительно нормализована по толщине методами, описанными в [8]. Для каждой выделенной линии метод формирует дескрипторы путем построения гистограмм локальных ориентаций их фрагментов.

На вход метода подается изображение  $I = \left\| i(y, x) \right\|_{(y=0, Y-1, x=0, X-1)}$ , где  $i(y, x) = 0..255$  – яркостное значение пикселя на изображении,  $Y, X$  – размеры изображения по вертикали и горизонтали. Блок-схема алгоритма представлена на рис. 1.

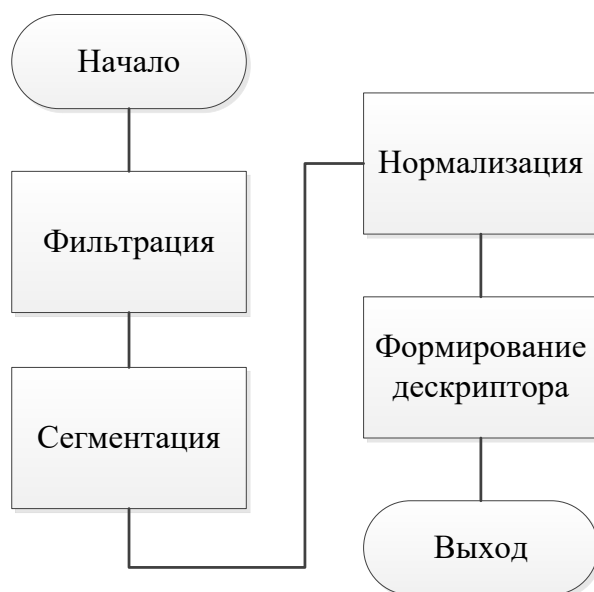


Рис. 1. Блок-схема метода DLOLF

Алгоритм работы метода состоит из следующих шагов.

Шаг 1. Формирование матрицы бинарных образов  $I_B = \left\| i_B(y, x) \right\|_{(y=0, Y-1, x=0, X-1)}$  для входного изображения  $I$  с помощью алгоритма контурной фильтрации Canny [9], где  $i_B(y, x) = 1$  для пикселя, принадлежащего контуру,  $i_B(y, x) = 0$  для фонового пикселя.

Шаг 2. Сегментация контурных линий. Каждому контурному пикселю  $i_B(y, x) = 1$  присваивается номер контура, к которому он принадлежит. В результате формируется матрица контуров  $S = \left\| s(n) \right\|_{n=1, N}$  и матрица количества концевых точек в каждом контуре  $K = \left\| k(n) \right\|_{n=0, N}$ , где  $s(n)$  – координаты контурных пикселей  $n$ -го контура, представленных в виде матриц  $X(n) = \left\| x(n, c) \right\|_{(c=0, C-1)}$ ,  $Y(n) = \left\| y(n, c) \right\|_{(c=0, C-1)}$ ,  $k(n)$  – количество концевых точек для  $n$ -го контура,  $N$  – количество найденных контуров,  $C$  – количество пикселей в  $n$ -м контуре.

Шаг 3. Нормализация выделенных контуров по толщине. Нормализация контуров осуществляется с помощью метода нормализации контурных линий по толщине [8]. В процессе нормализации из матриц координат контуров  $X(n)$ ,  $Y(n)$  удаляются пиксели, которые визуально и физически делают линию толще. В результате формируются контуры  $s(n)$  толщиной в один пиксель.

Шаг 4. Анализ контуров. Сперва производится анализ концевых точек контуров  $k(n)$ . Если  $k(n) = 2$  – принимается решение, что контур  $s(n)$  является линией и осуществляется расчет форм-фактора  $f$  [10].

Шаг 5. Формирование дескриптора.

5.1. Вычисление локальных ориентаций каждого пикселя линии, кроме концевых. Локальная ориентация каждого пикселя линии, за исключением концевых точек, определяется путем установления соответствия окрестности точки линии размером  $3 \times 3$  пикселя одной из 12 масок, представленных на рис. 2.

5.2. Вычисление локальной ориентации концевых точек. Для определения локальной ориентации концевых точек рассчитывается ориентация выделенной линии, используя следующее выражение:

$$O_L = \frac{y_2 - y_1}{x_2 - x_1},$$

где  $y_1, x_1$  и  $y_2, x_2$  – координаты концевых точек линии.

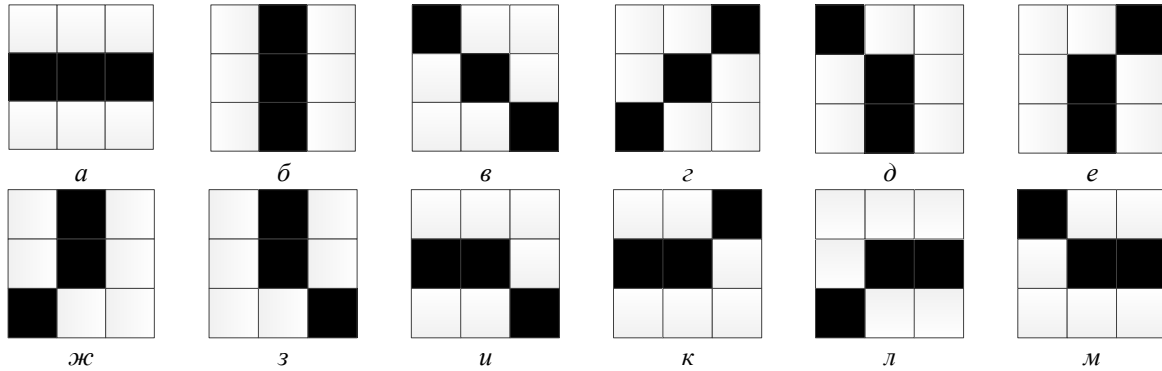


Рис. 2. Маски определения локальной ориентации пикселя: а-м – маски 1-12 соответственно

5.3. Гистограмма дескриптора формируется путем подсчета количества совпадений для каждой из масок и состоит, соответственно, из 12 элементов.

Шаг 6. Окончание алгоритма. В результате работы метода для каждой выделенной линии формируется дескриптор, состоящий из вектора, являющегося гистограммой возможных локальных ориентаций каждого пикселя ключевой линии, длиной в 12 элементов.

Идентификация линий происходит в 2 этапа.

1. Оценка разницы значения форм-фактора [10] линий  $\Delta f$ , используя выражение:  $\Delta f \geq 0,03$ , где  $\Delta f = |f_1 - f_2|$  – разница форм-факторов,  $f_1$  и  $f_2$  – значения форм-факторов выделенных линий первого и второго изображений соответственно, вычисленных на этапе детектирования.

Зависимость изменения значения форм-фактора от угла поворота для различных линий представлена на рис. 3.

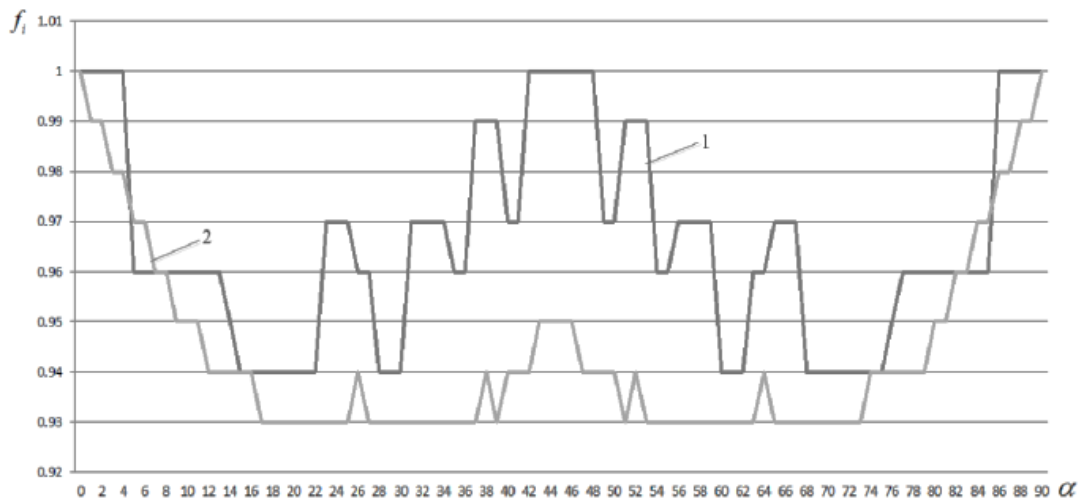


Рис. 3. значение форм-фактора в зависимости от угла поворота:

1 – выделенная линия длиной 15 пикселей; 2 – выделенная линия длиной 101 пиксель

2. Если отклонение  $\Delta f$  удовлетворяет заданному условию (линии похожи), то происходит оценка гистограмм локальных ориентаций (рис. 4). При совпадении этих гистограмм линии считаются соответствующими.



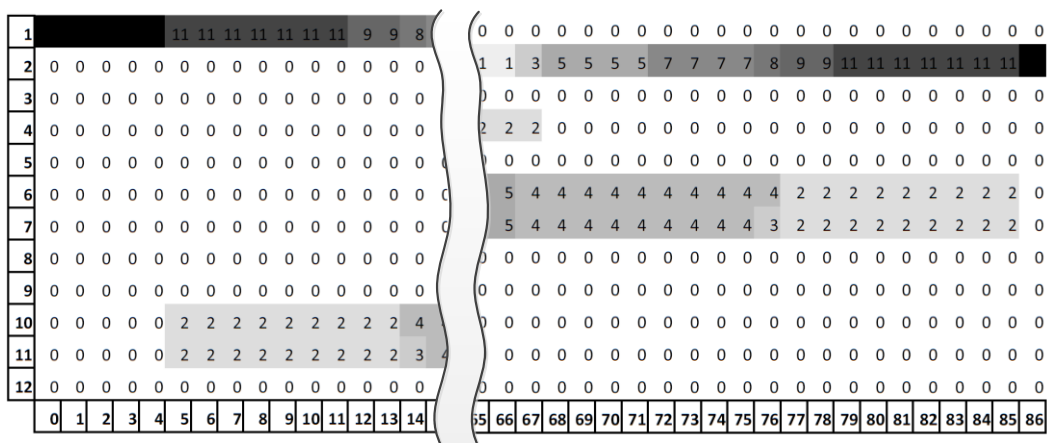


Рис. 4. Характеристика изменения гистограммы локальных ориентаций линии длиной 15 пикселей

### Оценка эффективности геометрического детектора линий и их идентификации

Разработанный метод реализован на языке программирования C++ с использованием библиотеки OpenCV. Эксперимент проведен на ЭВМ со следующими техническими характеристиками: процессор – Intel(R) Core(TM) i5-2320 CPU 3,0 ГГц; ОЗУ – 4 Гб; тип системы – 64-разрядная операционная система, процессор x64; операционная система – Windows 7. Для анализа использовалось изображение размером 600×600 пикселей (рис. 5). Изображения были изменены по яркости и контрасту, используя графический редактор.



Рис. 5. Исходное тестовое изображение

Оценка стабильности детектирования линий рассчитана, используя выражение  $S = \frac{k_i}{k_0} \times 100\%$ , где  $k_0$  – количество выделенных линий на исходном изображении;  $k_i$  – количество выделенных линий на измененном изображении;  $i = \overline{-50..50}$  – значение контраста или яркости относительно исходного тестового изображения.

Оценка стабильности идентификации линий рассчитана при помощи выражения  $S_I = \frac{d_i}{k_i} \times 100\%$ , где  $d_i$  – количество идентифицированных линий на измененном изображении относительно исходного изображения;  $k_i$  – количество выделенных линий на измененном изображении;  $i = \overline{-50..50}$  – значение контраста или яркости относительно исходного тестового изображения.

Оценка стабильности детектирования линий на изображении при изменении яркости и контраста представлена на рис. 6.

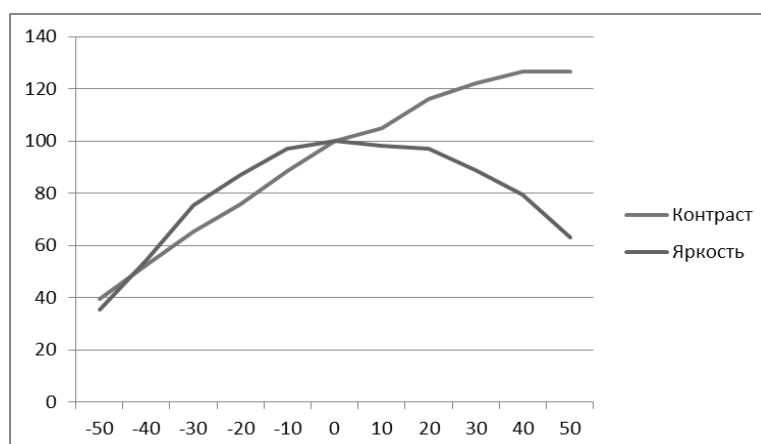


Рис. 6. График стабильности детектирования линий

Из рис. 6 видно, что при увеличении контраста количество детектированных линий увеличивается до 126 %, а при изменении яркости уменьшается до 35 %.

На рис. 7 представлена оценка стабильности идентификации линии исходного тестового изображения на измененном.

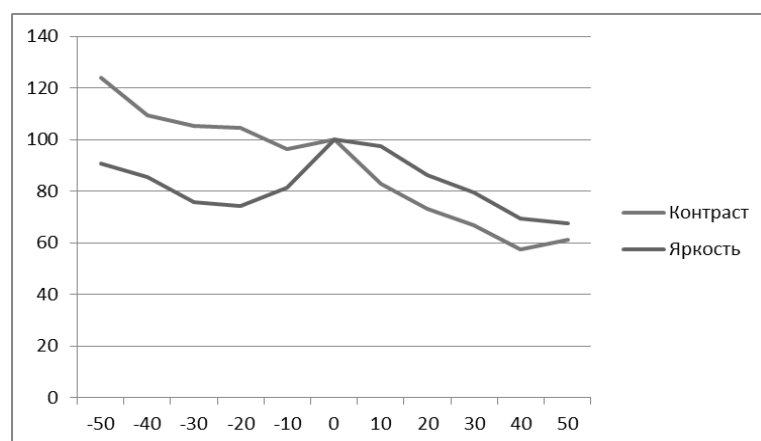


Рис. 7. График стабильности идентификации линий

Из рис. 7 видно, что предложенный метод идентификаций линий стабильнее при изменении яркости, чем при изменении контраста изображения. Также стоит отметить, что при изменении контраста от  $-50$  до  $-20$  значение стабильности  $S_i$  превосходит 100 % – это связано с тем, что одна и та же линия на изменном изображении может соответствовать нескольким линиям на исходном. Такая ошибка детектирования связана с маленьким размером линии (до 11 пикселей) и совпадением геометрических характеристик различных линий. Такие линии нуждаются в более детальном рассмотрении: уменьшении порога форм-фактора, дополнительном сравнении ориентаций линий, уменьшении порога различия дескрипторов.

### Заключение

Разработан метод формирования дескриптора на основе гистограммы локальных ориентаций фрагментов выделенной линии. Произведена оценка стабильности детектирования линий предложенным методом при изменении яркости и контраста изображения. Показано, что при увеличении контраста количество детектированных линий увеличивается до 126 %, а при изменении яркости уменьшается до 35 %. Оценка стабильности идентификации линий исходного изображения на измененном показала, что при уменьшении контраста возрастает ошибка идентификации.

# FORMATION OF DESCRIPTORS LINES AND THEIR IDENTIFICATION

O.G. SHEVCHUK

## Abstract

Method of forming a descriptor based on the histogram of local orientations of the fragments dedicated line, as well as methods for identifying lines using the proposed method is presented. The analysis of this method when the brightness and contrast are changing is conducted. It is shown that the method is resistant to change in brightness.

*Keywords:* detection line, identification line, descriptor, histogram.

## Список литературы

1. *Lowe D.* // International Journal of Computer Vision. 2004. Vol. 60. №2. P. 91-110.
2. *Bay H.* // Proc. of 9th European Conference on Computer Vision. 2006. P. 404-410.
3. *Calonder M., Lepetit V., Strecha Ch. et. al* // ECCV'10 Proceedings of the 11th European conference on Computer vision: Part IV P. 778-792.
4. *Rublee E., Rabaud V., Konolige K. et. al* // In Proceedings of the International Conference on Computer Vision (CVPR'11). 2011. P. 2564-2571.
5. *Kalal Z., Matas J., Mikolajczyk K.* // ICPR'10. 2010. P. 2756-2759.
6. *Jolliffe I.T.* Principal Component Analysis. Springer, 2002.
8. *Шевчук О.Г., Кирилюк Д.И., Макейчик Е.Г. и др.* // Докл. БГУИР. 2015. №7(93). С. 51-57.
9. *Sanny J.A.* // IEEE Transactions On Pattern Analysis And Machine Intelligence. 1986. Vol. 8. P. 679-698.
10. *Бородина О.Г., Цветков В.Ю.* // Известия СПбГЭТУ «ЛЭТИ». 2015. №1. С. 41-45.

УДК 004.932.72

## ВЕРОЯТНОСТНОЕ КОДИРОВАНИЕ ДЛИН СЕРИЙ В СЖАТИИ ПОЛУТОНОВЫХ ИЗОБРАЖЕНИЙ

Х.К. АЛЬ-БАХДИЛИ, В.В. ТОМИЛИН, В.К. КОНОПЕЛЬКО,  
З.Х.М. АЛЬ-ЗАЙДИ, А.С.Дж. АЛКАЛБИ

Белорусский государственный университет информатики и радиоэлектроники  
П. Бровка, 6, Минск, 220013, Беларусь

Поступила в редакцию 16 мая 2016

Предложены модифицированные алгоритмы кодирования длин серий для сжатия полутоновых изображений без потерь и с потерями, отличающиеся от базового алгоритма учетом вероятности повтора яркостных значений пикселей изображений и обеспечивающие за счет этого уменьшение размера кодированных данных.

*Ключевые слова:* сжатие изображений, кодирование длин серий.

### Введение

Для сжатия изображений с потерями и без потерь в настоящее время широко используются алгоритмы JPEG [1] и JPEG 2000 [2], алгоритмы LZW (Lempel-Ziv-Welch) (используемый в стандартах GIF (Graphics Interchange Format), TIFF (Tagged Image File Format) [3, 4]), Deflate и LZMA (используемые в архиваторах Zip, 7-zip [5]), PPM и LZSS (используемые в архиваторе Rar [5]). Данные алгоритмы имеют достаточно высокую вычислительную сложность. В тех случаях, когда временные и вычислительные ресурсы ограничены, необходимо использовать более простые алгоритмы эффективного кодирования. К ним относится алгоритм кодирования длин серий RLE (Run-Length Encoding) [6]. Он может использоваться для сжатия без потерь изображений с малым числом резких перепадов яркости (мультипликационных, медицинских, квантованных и сегментированных). В сочетании с предварительным квантованием изображений, алгоритм RLE может использоваться для сжатия с потерями. Недостатком данного алгоритма является отсутствие учета вероятности повтора значений пикселей изображений.

Целью работы является разработка модифицированных алгоритмов кодирования длин серий, основанных на учете вероятности повтора значений пикселей для сжатия изображений без потерь и с потерями.

### Алгоритм RLE

Алгоритм RLE основан на подсчете числа повторов значений следующих друг за другом символов (рис. 1), может иметь блочную и поточную реализацию.

Блочная реализация алгоритма RLE предполагает предварительное накопление кодируемых данных для их анализа и выбора параметров кодирования. Анализ производится на основе таблицы  $I/N$ -длин серий, в которой одному или нескольким одинаковым по значению и следующим друг за другом символам  $I$  ставится в соответствие их количество  $N$  (табл. 1).

Таблица 1. Длины серий

$I$	$i(0)$	$i(1)$	...	$i(s)$
$N$	$n(0)$	$n(1)$	...	$n(s)$

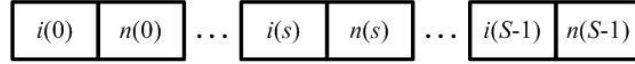


Рис. 1. Кодирование длин серий на основе алгоритма RLE (структура закодированных данных)

На основе таблицы длин серий определяются битовая глубина  $BD_I$  изображения и битовая глубина значений длин серий  $BD_N$  с помощью выражений:

$$BD_I = \left\lceil \log_2 \left( \max(i(s))_{(s=0, S-1)} \right) \right\rceil, \quad (1)$$

$$BD_N = \left\lceil \log_2 \left( \max(n(s))_{(s=0, S-1)} \right) \right\rceil, \quad (2)$$

где  $i(s)$  – значение кодируемого  $s$ -го символа из таблицы длин серий;  $n(s)$  – число повторов  $s$ -го символа (длина серии);  $S$  – число кодируемых символов (число строк в табл. 1).

Затем осуществляется кодирование длин серий в результате формирования на выходе кодера пар  $\{i(s), n(s)\}$ . Структура закодированных данных представлена на рис. 1. Блочная реализация алгоритма RLE позволяет получить минимальный объем кода. Ее недостатком является задержка при кодировании, необходимая для накопления данных и построения таблицы длин серий.

При поточной реализации алгоритма RLE нет необходимости строить таблицу длин серий, а подсчет числа символов может осуществляться по мере их поступления. Это позволяет существенно повысить скорость кодирования. Структуры алгоритма и закодированных данных при поточной реализации алгоритма RLE такие же, как при блочной. Значения битовой глубины  $BD_I$  изображения и битовой глубины значений длин серий  $BD_N$  выбираются независимо от поступающих данных. Они могут быть выбраны с избытком или недостатком (при переполнении  $n(s)$  формируется новая серия  $\{i(s), n(s)\}$ ). В некоторых случаях это может приводить к увеличению объема кода, что является недостатком поточной реализации алгоритма RLE. Улучшить характеристики алгоритма RLE при поточной реализации можно за счет формирования динамической таблицы длин серий и периодического обновления параметров кодирования.

При кодировании изображений размер  $R_{I/N}$  (бит) кода, коэффициент сжатия  $CR$  и вычислительная сложность  $C_{I/N}$  для алгоритма RLE определяются с помощью выражений:

$$R_{I/N} = S(BD_I + BD_N), \quad (3)$$

$$CR = 8YX / R_{I/N}, \quad (4)$$

$$C_{I/N} = YX + 4S, \quad (5)$$

где  $YX$  – размер изображения, определяющий число операций на буферизацию и формирование таблицы  $I/N$  размером  $S$  записей;  $4S$  – дополнительные операции на поиск максимального значения по  $I$  ( $S$  операций), поиск максимального значения по  $N$  ( $S$  операций), кодирование и передачу ( $2S$  операций).

### Алгоритм кодирования длин серий $I2/B/N$

Предлагается модификация  $I2/B/N$  алгоритма кодирования длин серий RLE для сжатия полутоновых изображений, основанная на учете вероятности повтора значений пикселей в строках. В процессе выполнения алгоритма строится таблица  $N/P$  вероятностей повторов (табл. 2), в которой каждому значению длины серии  $n$  ставится число таких серий  $p_n(n)$ .

Таблица 2. Вероятности повторов

$N$	1	2	...	$n$
$P$	$p_n(1)$	$p_n(2)$	...	$p_n(n)$

Для изображений характерно постепенное убывание функции  $p_n(n)$  при увеличении значения  $n$ .

При кодировании длины серии по алгоритму  $I2/B/N$  сначала формируется символ  $I$ . Затем, если символ  $I$  повторяется, то формируется бит  $b1(s) = 1$ , иначе  $b1(s) = 0$  (первый символ повтора). Если символ  $I$  повторяется снова, то формируется бит  $b2(s) = 1$ , иначе  $b2(s) = 0$  (второй символ повтора). Если символ  $I$  повторяется снова, то формируется символ  $n(s)$ , учитывающий число повторов (изначально  $n(s) = 0$ , если символ  $I$  повторяется снова, то  $n(s) > 0$ ). В результате могут формироваться серии вида:

$$\{i(s), b1(s) = 0\}, \{i(s), b1(s) = 1, b2(s) = 0\}, \{i(s), b1(s) = 1, b2(s) = 1, n(s)\}.$$

Структуры алгоритма  $I2/B/N$  кодирования длин серий и закодированных им данных приведены на рис. 2.

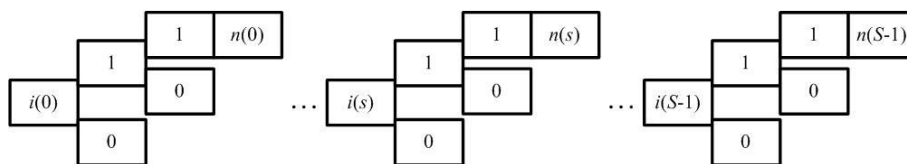


Рис. 2. Модифицированное  $I2/B/N$  кодирование длин серий (структура закодированных данных)

Алгоритм  $I2/B/N$  кодирования длин серий так же, как алгоритм RLE, может иметь блочную и поточную реализации.

Размер  $R_{I2/B/N}$  (бит) кода для алгоритма  $I2/B/N$  определяется с помощью выражения

$$R_{I2/B/N} = S(BD_I + 1) + \sum_{s=0}^{S-1} b1(s) + BD_N \sum_{s=0}^{S-1} b2(s). \quad (6)$$

Для определения вычислительной сложности используется выражение

$$C_{I2/B/N} = YX + 4S + \sum_{s=0}^{S-1} b1(s) + \sum_{s=0}^{S-1} b2(s). \quad (7)$$

Выражение (7) учитывает  $YX$  операций на буферизацию и формирование таблицы  $I/N$ -длин серий размером  $S$  записей,  $S$  операций на поиск максимального значения по  $I$ ,  $S$  операций на поиск максимального значения по  $N$ ,  $2S + \sum_{s=0}^{S-1} b1(s) + \sum_{s=0}^{S-1} b2(s)$  операций на кодирование и передачу.

### Алгоритм кодирования длин серий $I3/B/N$

Предлагается модификация  $I3/B/N$  алгоритма кодирования длин серий RLE для сжатия полутонных изображений, основанная на учете вероятности повтора значений пикселей в строках. Алгоритм  $I3/B/N$  отличается от алгоритма  $I2/B/N$  использованием дополнительного символа  $b3(s)$ , который принимает значение 1, если символ  $I$  повторяется в третий раз подряд, и принимает значение 0, если третий повтор символа  $I$  отсутствует. Структура закодированных данных в соответствии с алгоритмом  $I3/B/N$  приведена на рис. 3.

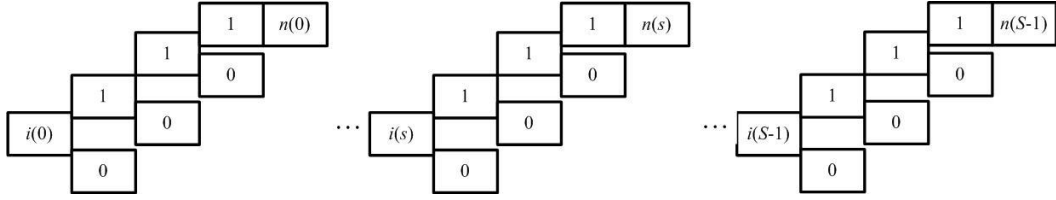


Рис. 3. Структура закодированных по алгоритму  $I3/B/N$  данных

Размер  $R_{I3/B/N}$  (бит) кода для алгоритма  $I3/B/N$  определяется с помощью выражения

$$R_{I3/B/N} = S(BD_I + 1) + \sum_{s=0}^{S-1} b1(s) + \sum_{s=0}^{S-1} b2(s) + BD_N \sum_{s=0}^{S-1} b3(s). \quad (8)$$

Вычислительная сложность алгоритма  $I3/B/N$  оценивается с помощью выражения

$$C_{I3/B/N} = YX + 4S + \sum_{s=0}^{S-1} b1(s) + \sum_{s=0}^{S-1} b2(s) + \sum_{s=0}^{S-1} b3(s). \quad (9)$$

Выражение (9) учитывает  $YX$  операций на буферизацию и формирование таблицы  $I/N$  -длин серий размером  $S$  записей,  $S$  операций на поиск максимального значения по  $I$ ,  $S$  операций на поиск максимального значения по  $N$ ,  $2S + \sum_{s=0}^{S-1} b1(s) + \sum_{s=0}^{S-1} b2(s) + \sum_{s=0}^{S-1} b3(s)$  операций на кодирование и передачу.

### Квантование изображений для сжатия с потерями в пространственной области

Методы сжатия изображений с потерями основаны на эффективном кодировании коэффициентов преобразования с их предварительным квантованием. Во многих методах именно квантователь вносит потери и определяет, в основном, коэффициент сжатия. Кодирование коэффициентов преобразования позволяет достичь наибольших коэффициентов сжатия за счет концентрации основной энергии в относительно небольшом числе значимых коэффициентов преобразования. Однако само преобразование требует существенных вычислительных ресурсов и временных затрат. Поэтому актуальной задачей является разработка алгоритма сжатия изображений с потерями, использующего квантование значений пикселей изображения и их последующее эффективное кодирование.

Предлагается алгоритм построчного двухпорогового квантования значений пикселей полутонового изображения, ориентированный на последующее эффективное кодирование длин серий. В алгоритме используется оценка разности значений центрального пикселя и соседних с ним пикселей в окрестности Мура по отношению к двум порогам  $\Delta_B$  (определяет условия разделения значений яркости соседних пикселей по соседним уровням квантования) и  $\Delta_S$  (определяет условия приведения значения пикселя к несоответствующему ему уровню квантования из-за локального характера отличия яркости этого пикселя от яркости окрестных пикселей), связанным соотношением

$$\Delta_B = K \Delta_S,$$

где  $K \geq 1$  – балансный коэффициент, определяющий степень влияния значения второго порога на результаты квантования (если значение  $K$  стремится к  $\infty$ , то результаты квантования стремятся к результатам однопорогового алгоритма).

Если модуль минимальной разности значения центрального пикселя окрестности Мура со значениями квантованных левого и трех верхних пикселей меньше порога  $\Delta_B$ , то значение центрального пикселя окрестности Мура меняется на значение ближайшего к нему по яркости квантованного пикселя. Если данное условие не выполняется, то для оценки используется порог  $\Delta_S$ .

Определяется квантованный пиксель окрестности Мура (левый или один из трех верхних), ближайший по значению к центральному пикселю. Вычисляются модули разностей значений

этого пикселя со всеми значениями окрестных пикселей в окрестности Мура. Если все эти разности меньше порога  $\Delta_s$ , то центральному пикселю окрестности Мура присваивается значение ближайшего ему по значению квантованного пикселя в окрестности Мура. Если данное условие не выполняется, то центральный пиксель окрестности Мура сохраняет свое значение.

### Оценка эффективности вероятностного кодирования длин серий для сжатия полутоновых изображений без потерь

Для тестовых изображений, представленных на рис. 4, в табл. 3 приведены размеры кода, полученные для алгоритма RLE, алгоритмов из табл. 3, а также алгоритма Хаффмана, архиваторов Zip и Rar, где  $X$  – обозначение алгоритма в табл. 3.

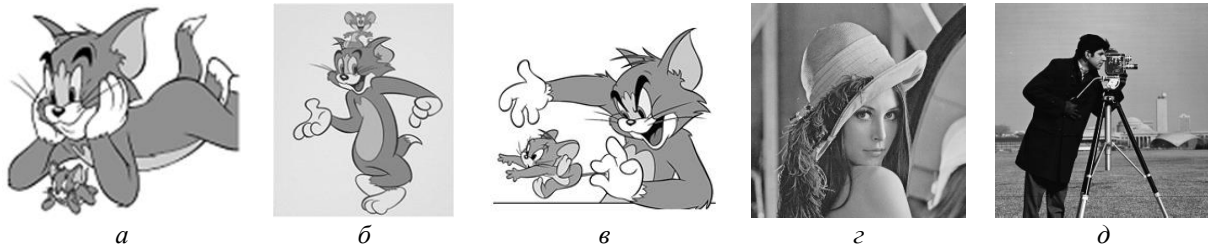


Рис. 4. Тестовые полутоновые изображения: а – M1 (128×128 пикселей); б – M2 (256×256 пикселей); в – M3 (512×512 пикселей); г – Lena (256×256 пикселей); д – cameraman (256×256 пикселей)

Из табл. 3 следует, что минимальный размер кода обеспечивают алгоритмы Rar и Zip. По сравнению с ними алгоритм  $I/3B/N$  показывает примерно на 26 % худший результат, но превосходит на 57 % и 8 % алгоритмы RLE и Хаффмана соответственно. При этом алгоритмы RLE,  $I/2B/N$ ,  $I/3B/N$  обеспечивают по сравнению с алгоритмами Rar и Zip уменьшение времени кодирования в 31,2 и 53,0 раз соответственно при размере изображения 256×256 пикселей, в 18,5 и 16,6 раз соответственно при размере изображения 1024×1024 пикселей, в 10,0 и 13,5 раз соответственно при размере изображения 8192×8192 пикселей (рис. 5).

Таблица 3. Размеры кода при сжатии тестовых изображений без потерь

Алгоритм	Размер кода (байт) для изображений			Алгоритм	Размер кода (байт) для изображений		
	M1	M2	M3		M1	M2	M3
RLE	18620	77896	242604	$2I/S/N$	34484	135364	261921
$I/S/N$	18739	72412	143162	$2I/B/N$	32599	127870	222123
$I/B/N$	21504	91397	303235	$2I/B/S/N$	32826	128820	225640
$I/B/S/N$	20957	81407	165901	$2I/2B/N$	32713	128345	223002
$I/2B/N$	14694	57583	106871	$2I/2B/S/N$	32825	128753	225460
$I/2B/S/N$	17689	68649	118177	$2I/B/2N(L/R)$	30046	118028	208692
$I/3B/N$	14577	56701	103512	Zip	13070	44603	76470
$I/3B/S/N$	17687	68581	117997	Rar	12505	43062	75887
$2I/N$	30714	120377	208857	Хаффмана	14361	58281	113091

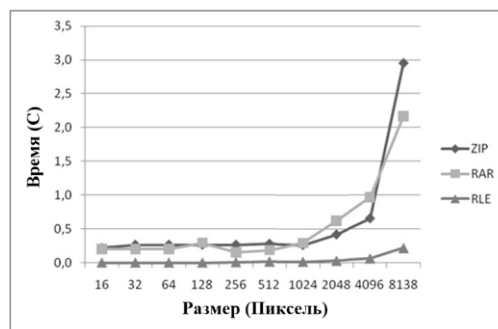


Рис. 5. Зависимости времени кодирования изображений от их размера для алгоритмов Zip, Rar, RLE и его предложенных модификаций в операционной системе Windows 7



## Оценка эффективности вероятностного кодирования длин серий для сжатия полутоновых изображений с потерями

На рис. 6 приведены тестовые изображения, для которых, а также для тестовых изображений, приведенных на рис. 4,б и 4,в, произведен анализ эффективности алгоритмов кодирования RLE и его модификаций при сжатии с потерями по отношению к алгоритмам Zip, Rar, Хаффмана, JPEG и JPEG2000 (рис. 6).

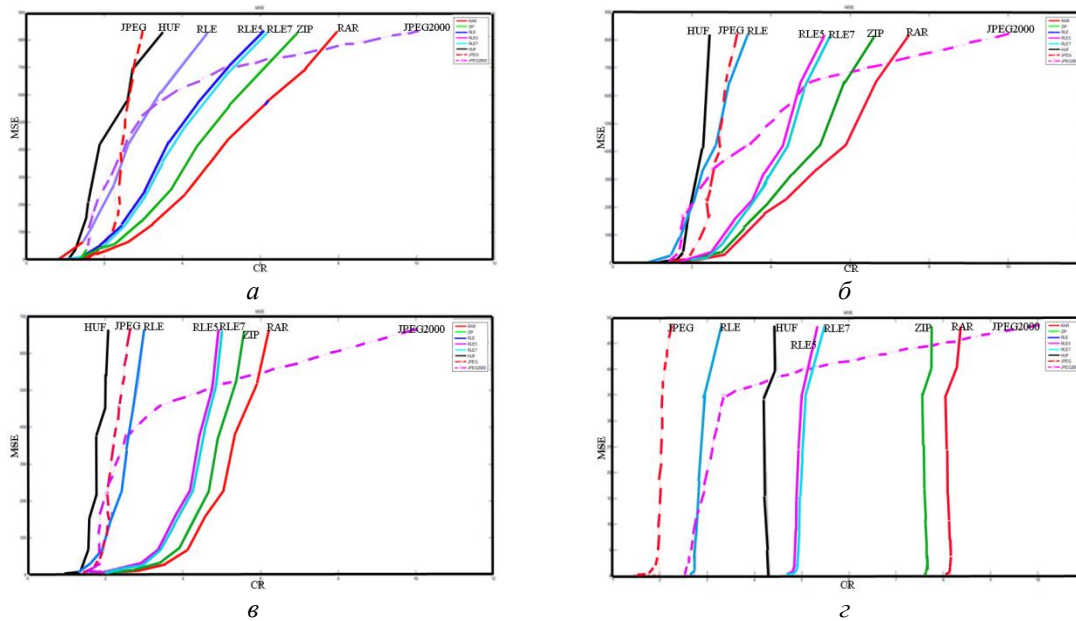


Рис. 6. Зависимости среднеквадратической ошибки сжатия (MSE) тестовых полутоновых изображений от коэффициента сжатия (CR): *a* – для тестового изображения Lena; *б* – для тестового изображения cameraman; *в* – для тестового изображения M2; *г* – для тестового изображения M3

Для сжатия с потерями на основе алгоритма RLE и его модификаций *I2/B/N* и *I3/B/N*, алгоритмов Zip, Rar и Хаффмана использован алгоритм двухпорогового построчного квантования изображений, рассмотренный выше. Как следует из рис. 6, предложенные алгоритмы вероятностного кодирования длин серий обеспечивают в 1,1-1,6 раза и 1,2-1,5 раза меньший коэффициент сжатия с потерями (в зависимости от типа изображения) по сравнению с алгоритмами Rar и Zip соответственно. При этом они превосходят базовый алгоритм RLE в 1,4-2 раза и алгоритм Хаффмана в 1,2-2,3 раза. При сжатии изображений в 2-5 раз предложенные алгоритмы обеспечивают уменьшение среднеквадратической ошибки в 3-10 раз в зависимости от типа изображения по сравнению с алгоритмом JPEG 2000. При одинаковой среднеквадратической ошибке предложенные алгоритмы проигрывают алгоритму JPEG в коэффициенте сжатия в 1,2-1,8 раза.

В табл. 4 приведено время сжатия с потерями тестовых изображений с помощью различных алгоритмов. Из табл. 4 следует, что предложенные алгоритмы вероятностного кодирования длин серий обеспечивают уменьшение времени сжатия в 1,7-6 раз по сравнению с RLE; в 170-1750 раз по сравнению с RAR; в 230-2200 раз по сравнению с ZIP; в 50-400 раз по сравнению с JPEG; в 70-670 раз по сравнению с JPEG 2000.

Таблица. 4. Время сжатия с потерями тестовых изображений

Алгоритм	Время сжатия изображений, с			
	M2	M3	Lena	cameraman
RLE	0,000000855	0,000000855	0,00000241	0,000000855
<i>I2/B/N</i>	0,000000285	0,000000285	0,00000144	0,000000285
<i>I3/B/N</i>	0,000000285	0,00000142	0,000000962	0,000000285
RAR	0,258	0,249	0,219	0,300
ZIP	0,300	0,324	0,322	0,682
JPEG	0,080	0,073	0,200	0,116
JPEG 2000	0,100	0,102	0,182	0,190

## Заклучение

Предложены алгоритмы вероятностного кодирования длин серий для сжатия полутоновых изображений без потерь и с потерями, отличающиеся от базового алгоритма RLE учетом вероятности повтора значений пикселей в строках изображения. Показано, что при сжатии без потерь предложенные алгоритмы обеспечивают уменьшение размера закодированных изображений на 57 % и 8 % по сравнению с алгоритмами RLE и Хаффмана соответственно; уменьшение времени кодирования в 10-31 раз и в 13-53 раза при изменении размера изображений от 256×256 пикселей до 8192×8192 пикселей по сравнению с алгоритмами Rar и Zip соответственно. При сжатии с потерями предложенные алгоритмы обеспечивают повышение коэффициента сжатия изображений до 2 раз по сравнению с алгоритмом RLE, до 2,3 раза по сравнению с алгоритмом Хаффмана, уменьшение среднеквадратической ошибки до 10 раз по сравнению с алгоритмом JPEG 2000 при коэффициенте сжатия до 2-5 раз. Установлено, что предложенные алгоритмы обеспечивают уменьшение времени кодирования до 6 раз по сравнению с RLE, 1750 раз по сравнению с RAR, до 2200 раз по сравнению с ZIP, до 400 раз по сравнению с JPEG, до 670 раз по сравнению с JPEG 2000.

## PROBABILISTIC RUN LENGTH ENCODING FOR GRAYSCALE IMAGE COMPRESSION

H.K. AL-BAHADILY, V.V. TAMILIN, V.K. KANAPELKA,  
Z.H.M. AL-ZAIDI, A.S.J. ALKALBI

### Abstract

New modified RLE algorithms to compress grayscale images with lossy and lossless compression, depending on the probability of repetition of pixels in the image and the pixel values to reduce the size of the encoded data.

*Keywords:* image compression, run length encoding.

### Список литературы

1. *Pennebaker W.B.* JPEG Still Image Compression Standard. New York, 1993.
2. *Ebrahimi T.* // Proc. of the SPIE. San Diego, July-August 2000. Vol. 4115. P. 446-454.
3. *Миано Дж.* Форматы и алгоритмы сжатия изображений в действии. М., 2003.
4. *Сэлмон Д.* Сжатие данных, изображений и звука. М., 2004.
5. *Ватолин Д., Ратушняк А., Смирнов М. и др.* Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. М., 2003.
6. *Golomb S.W.* // IEEE Transactions on Information Theory. 1966. P. 399-401.

UDC 621.391

**TEXTURE IMAGE SEGMENTATION BASED ON CLASSIFICATION,  
SKELETONIZATION AND CROSSING OF CONTOUR ELEMENTS**

H.M. ALZAKI

*Belarusian State University of Informatics and Radio Electronics  
P. Brovki, 6, Minsk, 220013, Belarus**Submitted 23 May 2016*

A method for texture image segmentation based on classification, skeletonization and crossing of contour elements is presented. The essence of the method consists in the contouring of the image, determining the position of contour elements in the image of different types (points, lines, and shapes) converting closely spaced similar contour elements into binary regions objects, binary coding mutual position obtained areal objects within the boundaries of the image segmentation are resulting in code matrix.

*Keywords:* texture image segmentation, classification, contour elements.

**Introduction**

Texture segmentation is among the most challenging problems in image segmentation. The problem already begins with the definition of textures. The human eye can easily recognize different textures. Fig. 1 shows different images. Image segmentation techniques can be classified into two broad groups – (1) region-based, and (2) contour-based approaches [1]. First group approaches try to find partitions of the image pixels into sets according to coherent image properties such as brightness, color and texture. Second group approaches start with a first stage of edge detection, followed by a linking process that seeks to exploit curvilinear continuity. These two approaches need not be different from each other. Boundaries of regions can be defined to be contours. If one enforces closure in a contour-based framework [2, 3] then one can get regions from a contour-based approach. In contour-based approaches, often the first step of edge detection is done locally. Subsequently efforts are made to improve results by a global linking process that seeks to exploit curvilinear continuity. A criticism of this approach is that the edge or no edge decision is made prematurely. To detect extended contours of very low contrast, a very low threshold has to be set for the edge detector (Fig. 1). Different methods used for texture analysis and texture segmentation, one of them – the energy map, which are widely used in texture analysis and can be used for texture segmentation. Their disadvantage is the high computational complexity of the simple texture and separation with a high degree of homogeneity.

In this paper, it is proposed to use contour elements for texture analysis. However these approaches are based on a small set of contour primitives and sufficiently rough statistical evaluation, which leads to significant errors separation of complex textures.

Contour analysis (e.g. edge detection) may be adequate for untextured images, but in a textured region it results in a meaningless tangled web of contours. Think for instance of what an edge detector would return on the bean region in (Fig. 2). The traditional “solution” for this problem in edge detection is to use a high threshold so as to minimize the number of edges found in the texture area. This is obviously a non-solution such an approach means that low-contrast extended contours will be missed as well.

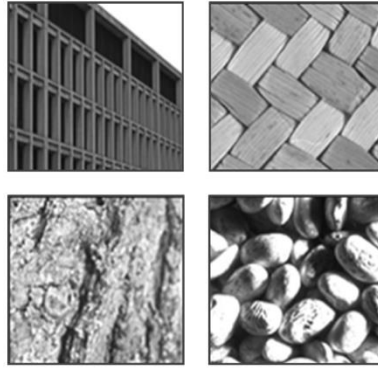


Fig. 1. Some challenging images for a segmentation algorithm. The goal is to develop a single grouping procedure which can deal with all these types of images

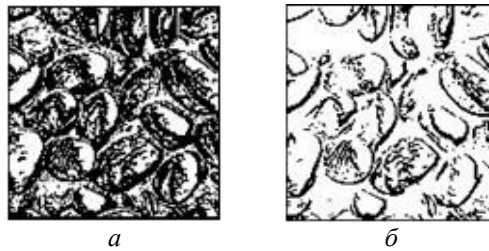


Fig. 2. Demonstration of texture as a problem for the contour process. Each image shows the edges found with a Prewitt filter and Roberts filters for the penguin image using different thresholds:  
*a* – low threshold; *b* – high threshold

### Texture image segmentation based on energy map

Widely spread for texture analysis is the method of energy maps [4, 5], based on the image filtering, weighting and summing the results. The essence of the methods to use filters with smooth changing of brightness, smooth contour lines, broken contour lines, contour points and small spots with constant or slowly changing brightness; resulting in the formation of binary matrices, single elements that indicate the presence of corresponding positions of certain objects; weighting obtained matrices and their summing element-wise, resulting in an energy map (Fig. 3). Image texture has a number of perceived qualities which play an important role in describing texture. Laws [5] identified the following properties as playing an important role in describing texture: uniformity, density, coarseness, roughness, regularity, linearity, directionality, direction, frequency, and phase. Laws texture energy measures determine texture properties by assessing Average Gray Level, Edges, Spots, Ripples and Waves in texture. The measures are derived from three simple vectors.  $L3 = (1, 2, 3)$  which represents averaging;  $E3 = (-1, 0, 1)$  calculating first difference (edges); and  $S3 = (-1, 2, -1)$  corresponding to the second difference (spots). After convolution of these vectors with themselves and each other, five vectors result:

- Level  $L5 = [1, 4, 6, 4, 1]$
- Edge  $E5 = [-1, -2, 0, 2, 1]$
- Spots  $S5 = [-1, 0, 2, 0, -1]$
- Ripples  $R5 = [1, -4, 6, -4, 1]$
- Waves  $W5 = [-1, 2, 0, -2, -1]$

Energy map contains mainly smoothly changing in the brightness, which can be separated by using known image segmentation methods, such as region growing [6, 7]. To improve the texture segmentation (to eliminate the influence on the result of high-frequency noise and joining smaller closely objects to the brightness of large objects) before applying the method of region growing the energy map is processing with low-pass filtered using a Gaussian filter.

The method of energy map is not strictly defines the types of filters that can be used to select image elements, which in theory makes the method suitable for the determination of arbitrary texture. However, combining the results of filtering in the energy map can lead to a segmentation fault, especially for complex textures, due to the lack of ranking results by relevance filtering. Eliminating this

disadvantage is possible due to: a) the use of contour elements for image texture analysis as the most homogeneous and therefore equivalent to merge the results; b) a clearly classification of contour elements of the structure to form a universal device for describing both simple and complex textures; c) logically combining for the results of filtering.

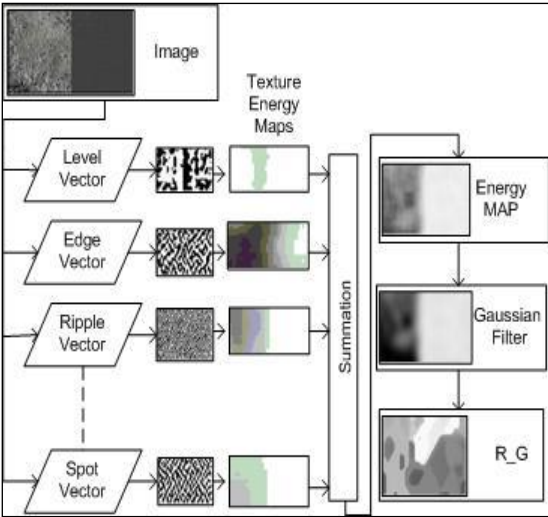


Fig. 3. Texture image segmentation based on the energy map

**Texture image segmentation based on classification, skeletonization and crossing of contour elements**

A method for texture image segmentation based on classification and skeletonization and crossing of contour elements. The essence of the method consists the contouring of the image, determining the position of contour elements of the image (points, lines, and shapes) of various types, transformation closely spaced similar contour elements into binary area objects, binary coding mutual position obtained polygon objects within the boundaries of the original image segmentation are resulting as matrix code (Fig. 4).

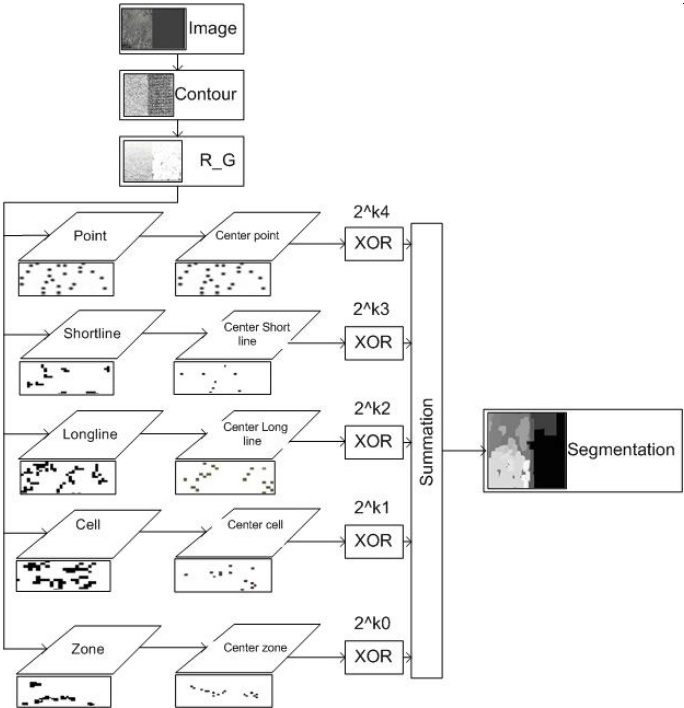


Fig. 4. Algorithm of texture image segmentation based on classification, skeletonization and crossing of contour elements

Algorithm of texture image segmentation based on classification, skeletonization and crossing of contour elements consists of the following steps.

Step 1. Contour filtering for the image. In this step may be any filter. In order to achieve high speed processing at a sufficiently high quality selection of contour lines is suggested to use the results of the union of the two loops filter: Roberts [8] and Prewitt [9], allocating brightness drops diagonally, horizontally and vertically (Fig. 5). The result is a binary matrix in which the elements correspond to single isolated contour points forming lines of varying difficulty and length, smaller area features.

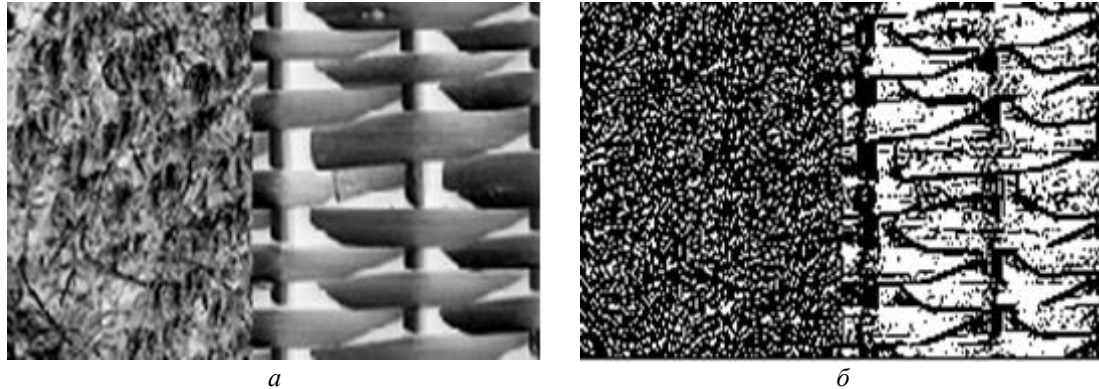


Fig. 5. Contour filtering for the image: *a* – original image; *b* – image contour by Roberts and Prewitt filters

Step 2. Segmentation of contour element. To perform this step, the algorithm may be any segmentation method. Most effective in this case is the method of region growing (Fig. 5), which provides high speed and accuracy. As a result of this step is formed by matrix, in which background elements corresponds to zero segment number, and other elements.

Step 3. Classification of the contour elements. At this step of the algorithm are allocated points, consisting of one or two pixels; short lines formed by two or three pixels; long line having two end points and formed by four or more pixels (limiting the number of pixels in the line length can be determined by processing the histogram of lengths of lines based on the size of the processed image) (Fig. 6); «Cell», formed by the intersection of several contour lines; «Spot» a small area objects with no endpoints; large area objects. All identified contour elements except large area objects are discussed later as a texture image. As a result of this step, each contour elements having it is number, is associated with the identifier (Descriptor), of the plurality of identifiers, relating it to a certain class. In the Table are shown the identification parameters used for the labeling of contour element are shown. Each row of Table is histogram of the number of neighboring contour pixels of different contour elements. Contour element «Point» (row 1) does not have end points and contains less than 5 neighbors contour pixels. Contour element «Short direct line» (row 2) comprises two endpoints and a small number neighbors contour pixels. Contour element «Short curvy line» (row 3) has two endpoints with a number of neighboring contour pixels. Contour element «Spiral» (row 4) is shorter curvy line with sharp bends, which has two end points and points with more than two neighbors contour pixels in places bends. Contour element «Cell» (row 5) has three or more end points and a plurality of contour pixels with a large number of neighbors. Contour element «Spot» (row 6) does not have end points, but contains a large number of contour pixels and number of neighbors more 4 pixels. Contour element «long direct line» (row 7) has two end points and a large number neighbors contour pixels.

Step 4. Classe the contour elements. A result of this step of the algorithm is formed by a plurality of bit planes, the unit elements each of which indicates the position of the respective contour elements of a certain class.

Step 5. Discretization of contour elements. At this step the contour elements replacing each bit plane a plurality of equidistant points. Each point and short line is assigned one point. Long lines are replaced by series points. «Cell» and «spots» are assigned to the grid points. The distance between the points in the series and the mesh is approximately the same. This makes it possible to align the input contour elements of different types to the energy generated at the next step of the integral of the binary image (Fig. 7).The result is a plurality point bit planes.

**Identification parameters and identifiers contour elements**

Classes of Contour elements	Identification parameters								Number of end points
	Number of neighbors contour pixels								
	1	2	3	4	5	6	7	8	
Point	Less than 5 neighbors of contour pixels								
Short direct line	2	2	1	0	0	0	0	0	2
Short curvy line	2	1	0	2	3	0	1	0	2
Spiral	2	0	0	7	3	1	4	2	2
Cell	4	1	2	10	4	8	3	9	4
Spot	0	0	9	1	3	10	5	12	0
Long direct line	2	14	5	1	0	1	0	0	2

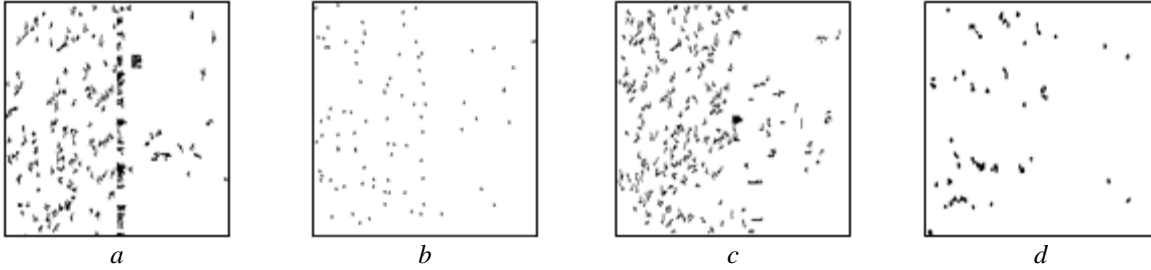


Fig. 6. Classification of the contour elements according to number of neighbors of contour pixels and end points: *a* – «Cell»; *b* – «Point»; *c* – «Long line»; *d* – «Spot»

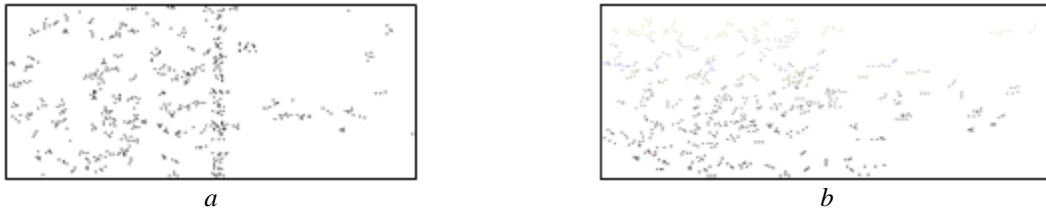


Fig. 7. Discretization of contour elements: *a* – cell is broken and replaced to number of contour pixels; *b* – long line is broken and replaced to number of contour pixels

Step 6. Determine histogram of the size of mask. As a result for this step it is found the histogram that starts with mask  $3 \times 3$  and this size are increasing by  $2(5 \times 5, 7 \times 7, 9 \times 9, \dots)$  until it finds neighbor for this contour element and finely result will be histogram and according for this histogram it will be found the size of the mask which will be used for next step.

Step 7. Segmentation each of contour element. As a result of this step is a bit plane in which each contour element have number according to the size of the mask in the above step. For this step it can be used any effective and fast segmentation method for this step, the method of region growing is used.

Step 8. Checking and comparison of the results of (Step 5) and (Step 7). According to this step to fill and return each contour element after (Step 5) but in this case according to the size of mask and the pixel that will be in the region of the mask will take the same number in the matrix of (step 7) as shown in the (Fig. 8).

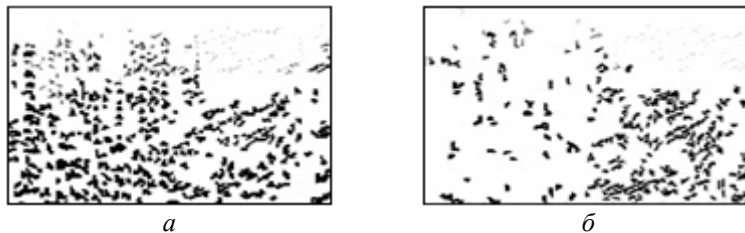


Fig. 8. Replace contour element according to compere (Step 5 and 7): *a* – cell with new numbers according to the size of mask; *b* – long line with new numbers according to the size of mask

Step 9. Flooding each of contour pixel. Replace each contour pixel by the value of the contour element. In this step each region has new number and it will indicate to texture region as shown in the (Fig. 9).



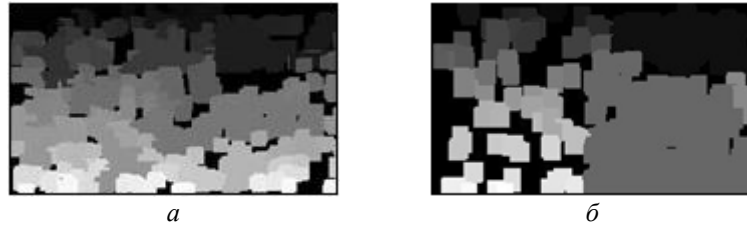


Fig. 9. Flooding each pixel according to the size of the mask:  $a$  – cell after flooding;  $b$  – long line after flooding

Step 10. Logical combination of classes. A result of this step is performed by the construction of the resulting integrated image obtained a result of the concatenation of the previous step integrated binary images. In the simple case, the number of bit planes in the resulting integral image replace with the initial binary number of integral images. In the simple case, the number of bit planes in the resulting integral image coincides with the initial binary number of integral images. If necessary, the logical association of contour elements of certain classes ( points, short lines, cells and spots) before the formation of the resulting integrated image is a logical addition of the corresponding binary integral images as shown in the (Fig. 10).



Fig. 10. Logical combination of classes

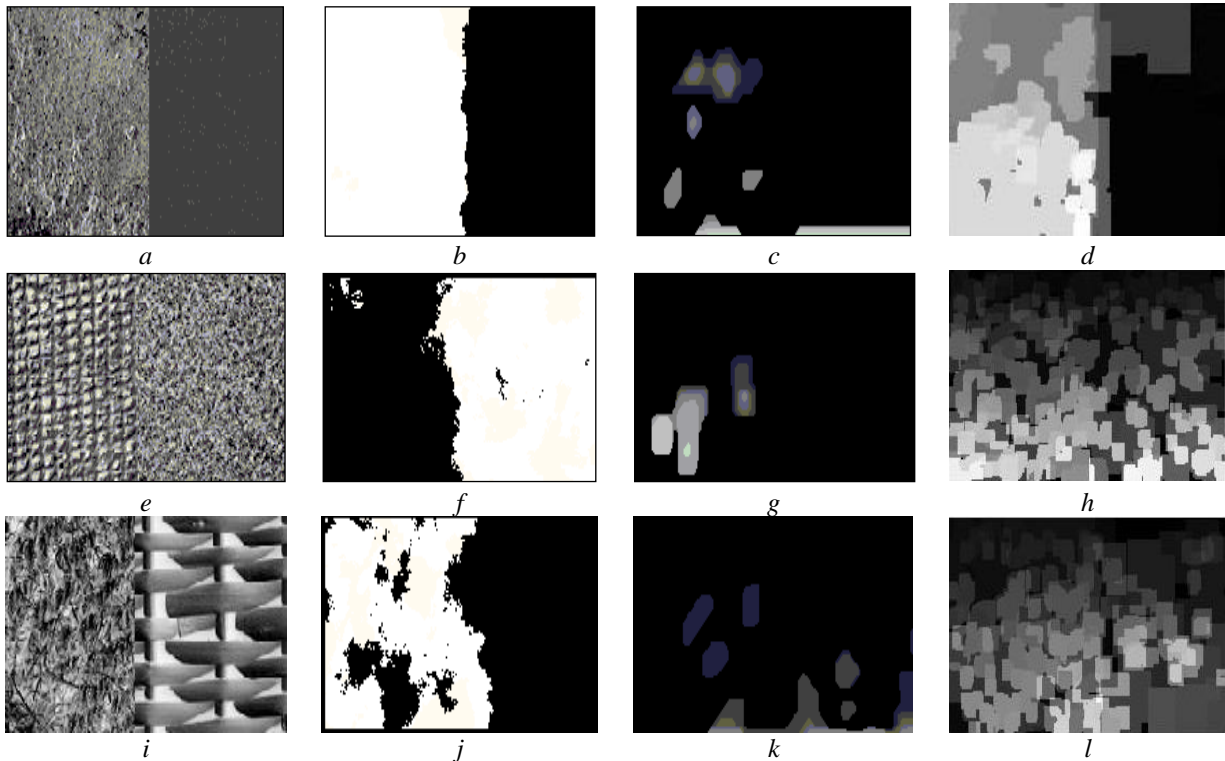


Fig. 11. Testes and results of image texture segmentation:  $a$  – Test 1;  $b$  – Test 1 Texture images segmentation based on classification of contour elements and logical addition of classes;  $c$  – Test 1 segmentation by energy map;  $d$  – Test 1 Segmentation by proposed method;  $e$  – Test 2;  $f$  – Test 2 Texture images segmentation based on classification of contour elements and logical addition of classes;  $g$  – Test 2 segmentation by energy map;  $h$  – Test 2 segmentation by proposed method;  $i$  – Test 3;  $j$  – Test 3 Texture images segmentation based on classification of contour elements and logical addition of classes;  $k$  – Test 3 segmentation by energy map;  $l$  – Test 3 segmentation by proposed method



## Conclusion

A texture image segmentation based on classification, skeletonization and crossing of contour elements determine the position of contour elements in the image of different types (points, lines, and shapes) and determine the distance between each contour pixel according to the distance determine the size of the mask which will be used to flood each of contour elements and get different texture region. The proposed method compared with the method based on the energy map which depends on the summation for result of different filters and with the method of texture images segmentation based on classification of contour elements and logical addition of classes which depends on the summation for different binary plans. The proposed method provides less segmentation error comparing with the method based on the energy map and the method of texture images segmentation based on classification of contour elements and logical addition of classes. For comparing test images, published in the test data bases Brodatz and UIUCTex, were used.

## References

1. *Ashraf M., Safari B.D., Zaki N.* // International Journal of Computer Science & Information Technology (IJCSIT). 2011. Vol. 3. №5. P. 99-106.
2. *Elder J., Zucker S.* // Conf. Computer Vision, Cambridge. England. 2005. Vol. I. P. 399-412.
3. *Jacobs D.* // IEEE Trans. Pattern Anal. Mach. Intell. 1996. Vol. 18. №1. P. 23-37.
4. *Lee D-Ch., Shchenk T.* // A Collection of Papers Presented At the XVII Congress of ISPRS. 1992. №48. P. 75-80.
5. *Ertuğrul Ö.* // International Journal of Intelligent Information Systems. 2014. Vol. 3. №2. P. 13-18.
6. *Kumar M., Mehta K. K.* // International Journal of Computer Technology and Applications. 2011. Vol. 2. №4. P. 855-859.
7. *Sharma Ritu, Sharma Rajesh* // International Journal of Innovative Research in Computer and Communication Engineering. 2014. Vol. 2. P. 5686-5692.
8. *Muthukrishnan R., Radha M.* // International Journal of Computer Science & Information Technology (IJCSIT). 2011. Vol. 3. №6. P. 259-267.
9. *Shrivakshan G., Chandrasekar C.* // IJCSI International Journal of Computer Science Issues. 2012. Vol. 9. №1. P. 269-276.

УДК 621.391.14

## МЕТОД КОРРЕКЦИИ ЗАВИСИМЫХ ОШИБОК СВЕРТОЧНЫМИ КОДАМИ С ОБНАРУЖЕНИЕМ НЕКОРРЕКТИРУЕМЫХ ПАКЕТОВ

Е.Г. МАКЕЙЧИК, А.И. КОРОЛЁВ, В.К. КОНОПЕЛЬКО

Белорусский государственный университет информатики и радиоэлектроники  
П. Бровка, 6, Минск, 220013, Беларусь

Поступила в редакцию 22 апреля 2016

Рассматривается метод коррекции зависимых ошибок сверточными кодами с обнаружением некорректируемых пакетов ошибок. Проводится сравнительный анализ эффективности предложенного метода коррекции зависимых ошибок с методом двухканального независимого кодирования информации и известными диффузными самоортогональными сверточными кодами (ДФССК) при равной скорости передачи кодов.

*Ключевые слова:* самоортогональный сверточный код, пакет ошибок, перемежитель, деперемежитель, синдром, синдромная последовательность.

### Метод и устройство коррекции зависимых ошибок сверточными кодами

В соответствии с [1-3] коррекция зависимых или группирующихся ошибок является актуальной задачей современных цифровых систем передачи информации. В [1, 2] рассматриваются каналные кодеки, корректирующие пакеты ошибок разной кратности или длины, но не обнаруживающие некорректируемые пакеты ошибок. Ниже рассматриваются метод и устройство коррекции зависимых ошибок с обнаружением некорректируемых пакетных ошибок.

Метод коррекции зависимых ошибок и обнаружение некорректируемых пакетов ошибок основан на использовании в канальном кодере соответствующего перемежителя кодовых символов и формирователя дополнительных проверочной и синдромной последовательностей, которые формируются из перемеженных кодовых символов. На рис. 1 представлена обобщенная структурная схема канального кодера, реализующего предложенный метод кодирования информации самоортогональным сверточным кодом (ССК) с алгоритмом порогового декодирования.



Рис. 1. Обобщенная структурная схема канального кодера:  $MS_k$  – канальный мультиплексор; ФДПП – формирователь дополнительной проверочной последовательности

Кодирование информационных символов осуществляется следующим образом. С выхода канального кодера кодовые символы поступают на вход перемежителя и после перемежения параллельными потоками поступают одновременно на соответствующие входы канального мультиплексора ( $MS_k$ ) и формирователя дополнительной проверочной последовательности (ФДПП). Параметры перемежителя выбираются исходя из заданной кратности корректируемых зависимых ошибок (пакета ошибок). ФДПП содержит  $I$  ( $I \gg 1$ ) параллельных регистров сдвига

и  $I(I \gg 1)$  многоходовых сумматоров по модулю два. Количество ячеек памяти в каждом регистре сдвига равно кратности корректируемого пакета ошибок ( $t_{к.п.}$ ).

При  $I = n_a \geq t_{к.п.}$  ( $n_a$  – длина кодового ограничения ССК) выходы всех ячеек памяти каждого регистра сдвига заводятся на соответствующий многоходовой сумматор по модулю два. Из перемеженных кодовых символов формируются символы дополнительной проверочной последовательности  $w = w_1, w_2, \dots, w_{n_a}$ . Формирование проверочных символов выполняется в соответствии со следующим алгоритмом [3, 4]:

$$\begin{aligned} w_1 &= \alpha_{1.1} \oplus \alpha_{1.2} \oplus \dots \oplus \alpha_{1,n_a} \\ w_2 &= \alpha_{2.1} \oplus \alpha_{2.2} \oplus \dots \oplus \alpha_{2,n_a} \\ &\vdots \qquad \qquad \qquad \vdots \\ w_{n_a} &= \alpha_{na.1} \oplus \alpha_{na.2} \oplus \dots \oplus \alpha_{na,n_a} \end{aligned} \quad (1)$$

Из (1) следует, что проверочный символ  $w_1$  формируется путем суммирования по модулю два всех первых символов  $I$  кодовых ограничений  $n_a$ , входящих в первый регистр сдвига ФДПП. Второй проверочный символ  $w_2$  формируется путем суммирования по модулю два всех вторых символов  $I$  кодовых ограничений  $n_a$ , входящих во второй регистр сдвига ФДПП. Аналогичным образом формируются проверочные символы  $w_3, w_4, \dots, w_{n_a}$ .

Сформированные дополнительные проверочные символы  $w = w_1, w_2, \dots, w_{n_a}$  поступают на соответствующие входы канального мультиплексора и передаются в канал связи вслед за кодовыми символами. В результате объединения перемеженных кодовых символов и дополнительных проверочных символов длина кодового ограничения ССК становится равной  $n'_a = n_a + w$  символов. Принятые кодовые символы поступают на вход канального декодера, обобщенная структурная схема которого представлена на рис. 2. В схеме используются следующие обозначения: КРИ-1/ $I$  – коммутатор распределения информации на  $I$  подпотоков; ФДППиСС – формирователь дополнительной проверочной последовательности и синдромных символов; MS<sub>к</sub> – канальный мультиплексор; MS<sub>сс</sub> – мультиплексор синдромных символов; Сч.НСС – счетчик ненулевых синдромных символов; ДШПО – дешифратор пакетных ошибок; ФИА – формирователь интервала анализа.

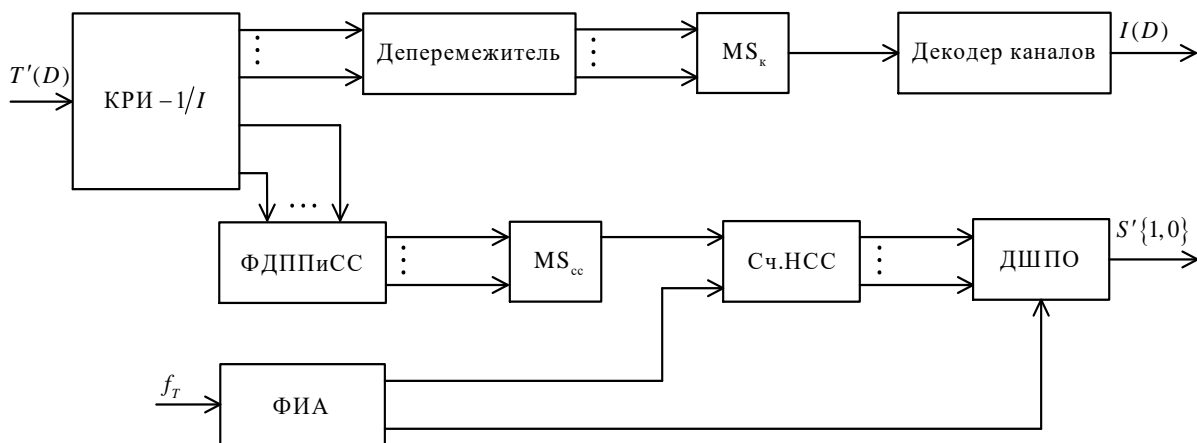


Рис. 2. Обобщенная структурная схема канального декодера

Декодирование кодовых символов производится следующим образом. В канальном декодере принятые кодовые символы  $n'_a$  после КРИ-1/ $I$  поступают одновременно на соответствующие входы дешеремежителя и формирователя дополнительной проверочной последовательности  $w' = w'_1, w'_2, \dots, w'_{na}$ . Из принятых проверочных символов ( $w$ ) и вновь сформирован-

ных ( $w'$ ) формируются символы синдромной последовательности (СП) по следующему правилу [1-3]:

$$C = C_1, C_2, \dots, C_{n_a} = (w_1 \oplus w'_1), (w_2 \oplus w'_2), \dots, (w_{n_a} \oplus w'_{n_a}). \quad (2)$$

Сформированные символы СП преобразуются из параллельного кода в последовательный и поступает на счетчик ненулевых синдромных символов (Сч.НСС), выходы которого подключены к дешифратору пакетных ошибок (ДШПО). Принятие решения о состоянии канала связи (о длине корректируемых пакетов ошибок) принимается по окончании интервала анализа, формируемого соответствующим формирователем интервала анализа (ФИА). Интервал анализа выбирается (устанавливается) равным  $n'_a = n_a + w$  тактов. Если количество ненулевых синдромных символов, сформированных на основе дополнительных проверочных символов, превысит порог принятия решения  $\eta$ , то с выхода дешифратора выдается сигнал  $S$  с уровнем логической единицы, что будет свидетельствовать о приеме некорректируемого пакета ошибок, а в противном случае формируется сигнал  $S$  с уровнем логического нуля. Величина порога  $\eta$  выбирается исходя из максимальной кратности одиночного пакета ошибок  $t_n \leq I$  бит корректируемого ССК после депережежения кодовых символов.

Эффективность обнаружения некорректируемых пакетов ошибок в значительной степени зависит от правильного выбора величины порога  $\eta$ , количества используемых порогов в ДШПО, параметров ССК и перемежителя-деперемежителя, а также от закона распределения длин пакетов ошибок и заданной достоверности передачи информации [1-4].

### **Оценка эффективности метода коррекции зависимых ошибок с обнаружением некорректируемых пакетов ошибок**

Экспериментально установлено [4], что вероятность обнаружения некорректируемых пакетов ошибок увеличивается, если в ДШПО используется два порога принятия решения, а именно:  $\eta_1$  – соответствует обнаружению корректируемого пакета ошибок,  $\eta_2$  ( $\eta_2 > \eta_1$ ) – соответствует обнаружению некорректируемого пакета ошибок, а если порог находится в пределах  $\eta_1 < \eta < \eta_2$ , то состояние канала связи считается неопределенным и получателю информации может выдаваться сигнал «Z».

Оптимальные значения порогов  $\eta_1$  и  $\eta_2$  можно получить, исходя из экспериментального определения распределения длин пакетов ошибок в канале связи, либо теоретическим путем, исходя из модели канала связи.

**Пример 1.** Предположим, что распределение длин  $t_n$  пакетов ошибок в канале связи подчиняется геометрическому закону ( $P(t_n) = (1 - \delta) \cdot \delta^t$ ), а вероятности появления в кодовой последовательности ненулевых символов  $P(1)$  и нулевых символов  $P(0)$  равны, т.е.  $P(1) = P(0) = 0,5$ . Тогда вероятность того, что количество ненулевых символов СП или число позиций, в которых различаются символы дополнительных проверочных последовательностей, превысит порог будет равна [5]:

$$P_l = \sum_{k=L_{n_a}+1}^{t_n} (k^{t_n}) \cdot 1/2^{t_n}. \quad (3)$$

Так как длина пакета  $t_n$  в общем случае должна быть больше величины порога  $\eta_1$ , определяемой числом ненулевых символов, то формула (3) вероятности появления пакета ошибок фиксированной длины ( $t_n = \text{const}$ ) при условии, что число ненулевых символов больше  $\eta_1$ , будет иметь вид

$$P_l = P(t_n) \cdot P_1 = (1 - \nabla) \nabla^{t_n} \sum_{k=\eta_1+1}^{t_n} (k^{t_n}) \cdot 1/2^{t_n}. \quad (4)$$

Тогда вероятность ошибки первого ряда ( $\alpha$ ) в выборе величины порога  $\eta_1$  можно определить, просуммировав выражение (4) по  $t_n$ , т.е.

$$\alpha = \sum_{k=\eta_1+1}^{t_n-1} (1-\nabla)\nabla^{t_n-k} \sum_{k=\eta_1+1}^{t_n} (k^{t_n}) \cdot 1/2^{t_n}. \quad (5)$$

Вероятность ошибки второго рода ( $\beta$ ) или вероятность необнаружения пакета длины  $t_n$  соответственно будет равна

$$\beta = \sum_{k=t_n}^{\infty} (1-\nabla)\nabla^{t_n-k} \sum_{k=0}^{\infty} (k^{t_n}) \cdot 1/2^{t_n}. \quad (6)$$

Следовательно, если известен параметр  $\nabla$  (величина допустимого риска принятия решения), то с допустимой точностью можно определить необходимое значение  $\eta_1$ .

**Пример 2.** Предположим, что экспериментальное распределение длин пакетов ошибок в канале связи описывается моделью Мертца [5]: пакет ошибок содержит 50 символов, а длительность защитных интервалов между пакетами составляет  $L_{\text{защ.}} = 1,6 \cdot 10^4$  символов. Ошибки в пакетах, кроме первого и последнего, распределены случайным образом с вероятностью ошибочного символа  $P = 0,5$ .

Выбираем параметры ССК с алгоритмом порогового декодирования и перемежителя. Скорость передачи ССК определяется допустимой избыточностью передаваемой информации. Пусть  $R = 4/5$ ,  $J = 2$  – число ортогональных проверок,  $n_a = 25$  символов – длина кодового ограничения. Параметры перемежителя принимаем равными: тип перемежителя – треугольный,  $I = n_a = 25$ ,  $\varepsilon = 2$  такта – начальная задержка во втором регистре сдвига перемежителя,  $H = \varepsilon \cdot I \cdot n_a = 2 \cdot 25 \cdot 25 = 1250$  кодовых символов – интервал перемежения, на котором может возникнуть пакет ошибок кратностью  $t_n \leq 50$  символов. Так как  $H = 1250 \ll L_{\text{защ.}} = 1,6 \cdot 10^4$  символов, то каналный кодек будет обеспечивать  $\delta = L_{\text{защ.}}/H = 1,6 \cdot 10^4/1250 = 1,3$  раза повышение достоверности передачи информации.

Выбор значений порогов  $\eta_1$  и  $\eta_2$  производится исходя из следующего: при возникновении на интервале  $H = 1250$  кодовых символов пакета ошибок длины  $t_n = 50$  символов,  $t_n/2$  символов будут ошибочными. Следовательно,  $C/2 = t_n/2 = 50/2 = 25$  синдромных ошибочных символов будут ненулевыми. С учетом дисперсии распределения ошибочных символов в пакетах ошибок, которую для данного случая можно принять равной  $\delta^2 = t_n/4$ , значение порога  $\eta_1$  выбирается равным  $\eta_1 = t_n/2 - 3, \delta^2 = 18$ . Значение второго порога принимается равным  $\eta_2 = 25$ .

Таким образом, если за время анализа  $l_{\text{ан.}} = t_n = 50$  тактов количество ненулевых символов не превысит значения  $\eta_1 = 18$ , то это будет свидетельствовать о «хорошем» состоянии канала связи, т.е. длина пакета ошибок не превысит расчетную ( $t_n = 50$  символов). Если же количество ненулевых символов больше или равно  $\eta_2 = 25$ , то длина пакета ошибок в канале связи будет больше расчетной, что свидетельствует об увеличении ошибок в канале связи. При количестве ненулевых символов СП в пределах  $18 < \eta < 25$ , канал связи находится в неопределенном состоянии, т.е. когда в канале связи возможны как пакетные, так и независимые ошибки, или многократные пакеты ошибок.

В табл. 1 и 2 приведены результаты сравнительного анализа эффективности предложенного метода коррекции зависимых ошибок с методом  $N(N = 2)$  – канального независимого кодирования информации и известными диффузными ССК (ДФССК). В качестве критериев оценки эффективности выбраны два параметра канального кодера:  $L_{\text{защ.}}$  – длина защитного интервала в кодовых символах и  $\theta$  – количество ячеек памяти регистров сдвига перемежителя, дepe-

ремешителя и канальных кодеков,  $t_n$  – кратность независимых ошибок,  $J$  – число ортогональных проверок кода.

Таблица 1. Результаты сравнительного анализа эффективности методов коррекции зависимых ошибок (пакетов ошибок)

Скорость передачи кода $R$	Диффузный ССК					Канальный кодек, реализующий предложенный метод кодирования информации						
	$t_n$ бит	$t_n$ бит	$n_a$ бит	$L_{\text{защ.}}$ бит	$\theta$ я.п.	$t_n$ бит	$I$	$\varepsilon$	$n$ бит	$H$ бит	$L_{\text{защ.}}$ бит	$\theta$ я.п.
2/3	3	2	84	81	135	9	9	1	9	81	72	82
	7	3	261	254	400	12	12	2	24	288	276	298
	12	4	513	501	840	42	21	2	42	882	840	905
3/4	6	2	312	306	462	8	8	2	16	138	120	130
	9	3	592	583	882	16	16	1	16	256	240	258
4/5	7	2	605	398	840	9	9	8	27	325	216	344
7/8	–	–	–	–	–	25	25	1	25	625	600	638

Таблица 2. Результаты сравнительного анализа эффективности методов коррекции зависимых ошибок (пакетов ошибок)

Скорость передачи кода $R$	Двухканальное независимое кодирование информации					Канальный кодек, реализующий предложенный метод кодирования информации						
	$t_n$ бит	$J$	$n_a$ бит	$L_{\text{защ.}}$ бит	$\theta$ я.п.	$t_n$ бит	$I$	$\varepsilon$	$n$ бит	$H$ бит	$L_{\text{защ.}}$ бит	$\theta$ я.п.
2/3	6	4	84	81	135	9	9	1	9	81	72	82
	10	4	261	254	400	12	12	2	24	288	276	298
	16	8	513	501	840	42	21	2	42	882	840	905
3/4	8	4	312	306	462	8	8	2	16	138	120	130
	12	4	592	583	882	16	16	1	16	256	240	258
4/5	9	4	605	398	840	9	9	8	27	325	216	344
7/8	25	4	–	–	–	25	25	1	25	625	600	638

### Заключение

Из результатов анализа, приведенных в табл. 1 и 2, следует, что предложенный метод коррекции зависимых ошибок и обнаружение некорректируемых пакетов ошибок эффективнее в 1,2-2,0 раза применения известных ДФССК и в 2,3-2,6 раза эффективнее двухканального ( $N = 2$ ) независимого кодирования информации как по кратности (длине) корректируемых зависимых ошибок, так и по количеству ячеек памяти канального кодера. Кроме того, предложенный метод коррекции ошибок, обеспечивает обнаружение некорректируемых пакетов ошибок а также осуществляет непрерывный контроль качества канала связи.

# METHOD OF ERROR CORRECTION USING DEPENDENT CONVOLUTIONAL CODES WITH THE DETECTED UNCORRECTABLE PACKAGES

E.G. MAKEICHIK, A.I. KOROLEV, V.K. KANAPELKA

## Abstract

The dependence of the error correction method of convolutional codes with detection of uncorrectable errors packages is considered. A comparative analysis of the effectiveness of the proposed dependent error correction method with the method of two-channel independent encoding information and known diffuse self-orthogonal convolutional codes (DFSSK) codes with equal rate is given.

*Keywords:* self-orthogonal convolutional code, burst-error, interleaver, deinterleaver, syndrome, syndromic sequence.

## Список литературы

1. Блейхут Р. Теория и практика кодов, контролирующих ошибки. М., 1986.
2. Кларк Дж. мл., Кейн Дж. Кодирование с исправлением ошибок в системах цифровой связи. М., 1987.
3. Конопелько В.К., Липницкий В.А., Дворников В.Д. и др. Теория прикладного кодирования. Минск, 2004.
4. Королёв А.И. Модифицированные алгоритмы порогового декодирования сверточных кодов. Минск, 1997.
5. Rodgers W.E., Lackey R.B. // IEEE Trans. Inform. Theory. 1980. Vol. IT-26, №3. P. 354-359.

УДК 004.932.2

## МОДЕЛИРОВАНИЕ ДВИЖЕНИЯ БЕСПИЛОТНОГО ЛЕТАТЕЛЬНОГО АППАРАТА С ОПТИЧЕСКОЙ НАВИГАЦИОННОЙ СИСТЕМОЙ

К.А. ВОЛКОВ, Х.М. КАРБАЛАИ САЛЕХ, А.В. ПАРАСОЧКА,  
С.С. ЗАЙЦЕВ, Р.Р. РОМАНОВ, М.Б.М. МАХММУД

*Белорусский государственный университет информатики и радиоэлектроники  
П. Бровки, 6, Минск, 220013, Беларусь*

*Поступила в редакцию 30 мая 2016*

Произведен анализ параметров движения беспилотного летательного аппарата для моделирования полета с оптической навигационной системой. Рассмотрена задача формирования изображений бортовых камер. Предложена система моделирования движения беспилотного летательного аппарата с оптической навигационной системой. Обозначены возможные направления исследований с использованием предложенной системы моделирования.

*Ключевые слова:* оптическая навигация, моделирование движения, беспилотный летательный аппарат.

### Введение

Разработка систем управления с использованием автономных пассивных оптических навигационных систем является крайне актуальной, поскольку независимость от глобальных систем позиционирования и дистанционного управления оператором в реальном масштабе времени позволяет существенно расширить область применения беспилотного летательного аппарата (БЛА) и круг решаемых задач [1-3]. Основной проблемой при реализации систем оптической навигации является обеспечение надежности их функционирования на различных типах местности, в различных климатических и метеоусловиях, что требует проведения значительного числа экспериментальных исследований. Таким образом, проведение натурных исследований разрабатываемых алгоритмов требует значительных временных и финансовых затрат, а получаемые результаты могут быть непрезентативны из-за невозможности обеспечения всего комплекса экспериментов и проблем организационного, технического и погодного характера. Альтернативой натурным исследованиям на начальных этапах разработки является компьютерное моделирование полета БЛА с использованием имеющихся аэрофотоснимков земной поверхности, сделанных в различных климатических условиях и для различных типов местности [4, 5]. Такой подход позволяет существенно сократить затраты на исследовательскую и экспериментальную работу при создании эффективных алгоритмов оптической навигации БЛА [6, 7].

В работе приводится анализ параметров движения БЛА, рассматривается задача формирования изображения бортовых камер, предлагается система моделирования движения БЛА с оптической навигационной системой.

### Анализ условий моделирования движения БЛА

Рассмотрим движение БЛА по траектории без скольжения при постоянной высоте и в отсутствие ветра. Данный подход не обеспечивает высокоточные оценки координат БЛА в каждой точке траектории ввиду имеющихся допущений, однако позволяет легко получить оценочные значения параметров движения. Такой подход оправдан, поскольку за стабилизацию и кор-



ректировку движения на коротких интервалах пути отвечает автопилот с инерциальной системой навигации [8, 9], алгоритмы функционирования которого хорошо известны и выходят за рамки данной статьи.

Пусть  $v$  – скорость движения БЛА относительно земли (м/с),  $g$  – ускорение свободного падения (м/с<sup>2</sup>),  $\varphi$  – угол крена самолета (°). Тогда радиус разворота, выполняемого с постоянной скоростью может, быть оценен по формуле

$$R = \frac{v^2}{g \tan\left(\varphi \frac{\pi}{180}\right)} \text{ (м)},$$

где  $\tan(\cdot)$  – функция тангенса угла, заданного в радианах.

Тогда центростремительное ускорение может быть определено как

$$a_{ц} = \frac{v^2}{R} = g \tan\left(\varphi \frac{\pi}{180}\right) \text{ (м/с}^2\text{)}.$$

Время выполнения разворота может быть рассчитано, соответственно, как

$$t = \frac{2\pi R}{v} = \frac{2\pi v}{g \tan\left(\varphi \frac{\pi}{180}\right)} \text{ (с)}.$$

Таким образом, угол поворота БЛА за некоторое время  $\Delta t$  может быть рассчитан по формуле

$$\alpha = 360^\circ \frac{\Delta t}{t} = 360^\circ \frac{\Delta t \operatorname{tg} \tan\left(\varphi \frac{\pi}{180}\right)}{2\pi v} = 180 \frac{\Delta t \operatorname{tg} \tan\left(\varphi \frac{\pi}{180}\right)}{\pi v} \text{ (}^\circ\text{)}.$$

Для осуществления прикидочного расчета радиусов и времени поворота БЛА используем характерные значения скорости и углов крена легких БЛА малого и среднего радиуса действия. Примем допустимый диапазон скоростей БЛА  $v = [10, 40]$  м/с, а диапазон углов крена –  $\varphi = [10, 30]$  °. Тогда согласно представленным выше формулам, имеем следующие значения, приведенные в таблице:

**Оценочные значения параметров движения БЛА**

$v$ , м/с	$\varphi$ , °	$R$ , м	$a_{ц}$ , м/с <sup>2</sup>	$t$ , с	$\alpha$ для $\Delta t = 1$ с
10	10	58	1,7	36	9,9
20	10	231	1,7	73	5,0
30	10	520	1,7	109	3,3
40	10	925	1,7	145	2,5
10	20	28	3,6	18	20,5
20	20	112	3,6	35	10,2
30	20	252	3,6	52	6,8
40	20	448	3,6	70	5,1
10	30	18	5,7	11	32,5
20	30	71	5,7	22	16,2
30	30	159	5,7	33	10,8
40	30	282	5,7	44	8,1

Полученные значения позволяют определить частоту пересчета траектории движения модели порядка 10-30 Гц, чтобы обеспечить смещение фотоизображения земной поверхности для каждой точки пересчета в пределах 1-10 м по прямой и 0,1-3° по углу.

## Формирование фотоизображения земной поверхности

В настоящее время для моделирования движения БЛА с оптической навигационной системой в открытом доступе присутствуют аэрофото- и космоснимки земной поверхности высокого разрешения трех картографических сервисов: Bing Maps, Google Maps и Yandex Maps [4, 5]. В данных сервисах используется геодезическая система WGS 84, в которой Земля считается эллипсоидом с экваториальным радиусом  $R_A = 6378137$  м и полярным радиусом  $R_B = 6356752$  м. Отображение точек земной поверхности на плоскую карту осуществляется с использованием равноугольной проекции Меркатора, в которой все параллели и меридианы представлены в виде прямых линий, причем линии меридианов являются равноотстоящими (рис. 1). Для проецирования в Bing Maps, Google Maps применяется сферическая проекция Меркатора (земная поверхность аппроксимируется сфероидом и проецируется на цилиндрическую поверхность карты), а для Yandex Maps – эллиптическая (поверхность аппроксимируется эллипсоидом). Важными особенностями проекции Меркатора является сохранение углов и невозможность отображения полюсов.

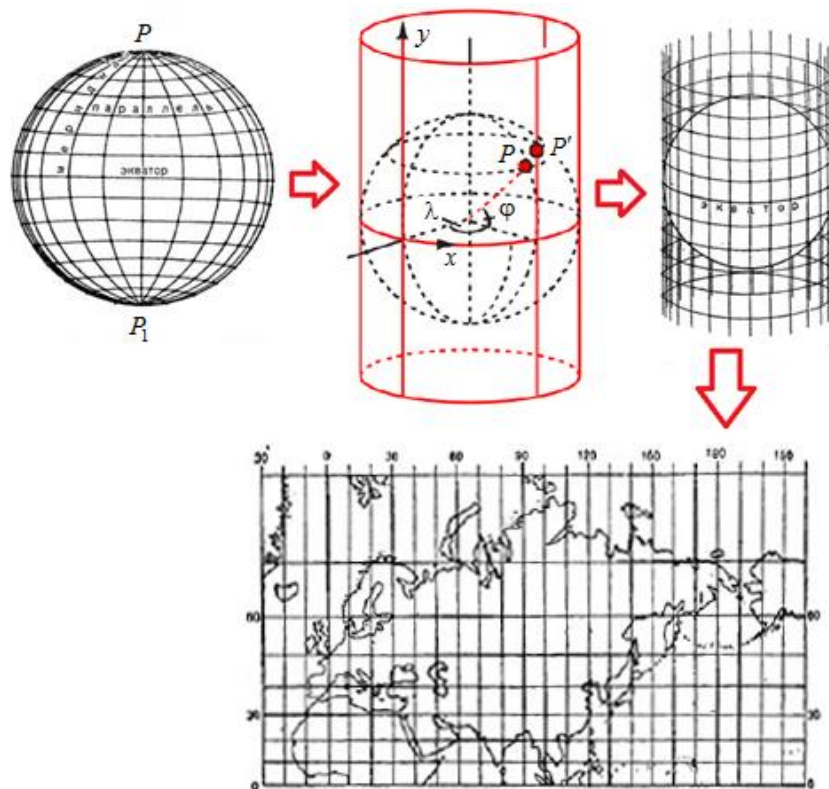


Рис. 1. Преобразование изображения земной поверхности в равноугольной проекции Меркатора

Для быстрого отображения произвольных участков земной поверхности картографические сервисы обеспечивают возможность доступа к ее отдельным квадратным фрагментам (тайлам) с размером  $256 \times 256$  пикселей в проекции Меркатора в требуемом масштабе. При этом уровни масштаба организованы ступенчато, так что на первом уровне присутствует полное изображение Земли на одном тайле, а для каждого следующего уровня масштаб увеличивается в 2 раза.

Формулы для прямого и обратного пересчета из геодезических координат в координаты тайлов имеют следующий вид. Преобразование широты  $\varphi$  и долготы  $\lambda$  в координаты  $X_C$  и  $Y_C$  сетки тайлов для сферической проекции производится по формулам

$$X_C = \frac{\lambda + 180}{360} 2^s,$$

$$Y_C = 0,5 \left[ 1 - \frac{\ln \left( \tan \left( \varphi \frac{\pi}{180} \right) + \frac{1}{\cos \left( \varphi \frac{\pi}{180} \right)} \right)}{\pi} \right] 2^S.$$

Преобразование широты  $\varphi$  и долготы  $\lambda$  в координаты  $X_{\vartheta}$  и  $Y_{\vartheta}$  сетки тайлов для эллиптической проекции производится по формулам

$$J = \frac{\sqrt{R_A^2 - R_B^2}}{R_A},$$

$$X_{\vartheta} = \frac{\lambda + 180}{360} 2^S,$$

$$Y_{\vartheta} = 2^{S-1} - \frac{2^{S-2}}{\pi} \left( \ln \frac{1 + \sin \left( \varphi \frac{\pi}{180} \right)}{1 - \sin \left( \varphi \frac{\pi}{180} \right)} - J \ln \frac{1 + J \sin \left( \varphi \frac{\pi}{180} \right)}{1 - J \sin \left( \varphi \frac{\pi}{180} \right)} \right).$$

Обратное преобразование из координат сетки тайлов для сферической проекции в геодезические координаты производится по формулам

$$\varphi = \arctan \left( \sin h \left( \pi - 2\pi \frac{Y_C}{2^S} \right) \right) \frac{180}{\pi},$$

$$\lambda = 360 \frac{X_C}{2^S} - 180$$

где  $S$  – уровень масштаба ( $S \geq 1$ ),  $\sin(\cdot)$  и  $\cos(\cdot)$  – функции синуса и косинуса угла, заданного в радианах,  $\arctan(\cdot)$  – функция арктангенса,  $\sin h(\cdot)$  – функция гиперболического синуса,  $\ln(\cdot)$  – функция натурального логарифма.

С учетом приведенных выше формул, линейный размер пикселя на фотоизображении может быть определен как

$$d = \frac{4\pi R_A \cos \left( \varphi \frac{\pi}{180} \right)}{2^S 256} = 2^{-6-S} \pi R_A \cos \left( \varphi \frac{\pi}{180} \right) \text{ (м/пиксель)}.$$

### Система моделирования движения БЛА

Для моделирования движения БЛА была спроектирована и реализована программная система (рис. 2), состоящая из двух подсистем: А – моделирования движения и Б – навигации и управления. За синхронизацию работы подсистем отвечает Блок 1, формирующий команды для выполнения потактового расчета. По сигналам Блока 1 производится расчет положения БЛА в пространстве Блоком 2, расчет координат целей Блоком 7 и формирование команд на органы управления Блоком 8.

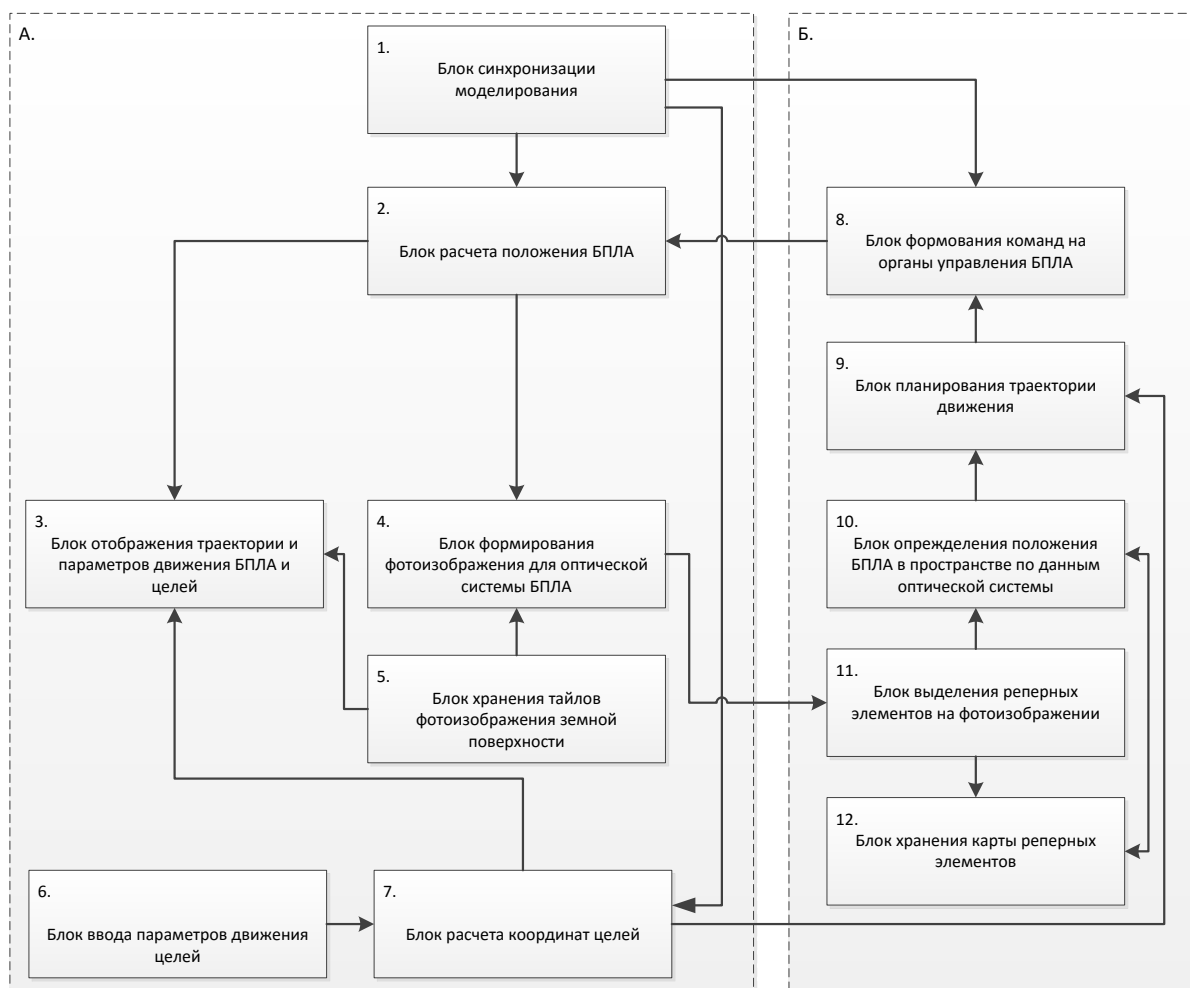


Рис. 2. Система моделирования движения БЛА с оптической навигацией

Подсистема А моделирования движения функционирует следующим образом.

На каждом такте моделирования информация о текущем положении БЛА в пространстве из Блока 2 поступает в Блок 3 для визуализации пользователю на карте, и в Блок 4 для формирования актуального фотоизображения от оптической системы БЛА (рис. 3). Тайлы цифровой карты земной поверхности в Блоки 3 и 4 поступают из Блока 5, ответственного за их хранение.

Ввод параметров движения целей оператором осуществляется посредством Блока 6, а дальнейший расчет их координат осуществляется Блоком 7. Информация об актуальном положении целей выводится оператору на карте посредством Блока 3 (совместно с информацией о положении и траектории БЛА).

Подсистема Б навигации и управления включает в себя Блоки 8-12, которые соответствуют блокам системы управления реального БЛА, но поддерживают работу в режиме потактового моделирования.

Сформированные данные от оптической системы из Блока 4 поступают в Блок 11, который осуществляет выделение реперных элементов на изображении [6, 7]. Дескрипторы реперных элементов из Блока 11 поступают в Блок 10, куда также поступает информация из Блока 12, где содержится карта реперных элементов с геодезической привязкой координат. Блок 10 рассчитывает текущее положение БЛА, сравнивая найденные реперные элементы с имеющейся цифровой картой, и передает координаты в Блок 9. Также вычисляются координаты реперов, которые отсутствуют на имеющейся карте, и передаются в Блок 12 для ее пополнения. Блок 9 производит расчет требуемого вектора движения с учетом текущих координат БЛА и полученных из Блока 7 координат целей. Найденный вектор передается в Блок 8, который формирует команды на органы управления, которые далее передаются в Блок 2.

После выполнения всех расчетов в подсистемах А и Б Блок 1 генерирует сигнал на проведение очередного такта моделирования, в котором в качестве начальных значений используются результаты предыдущего такта.

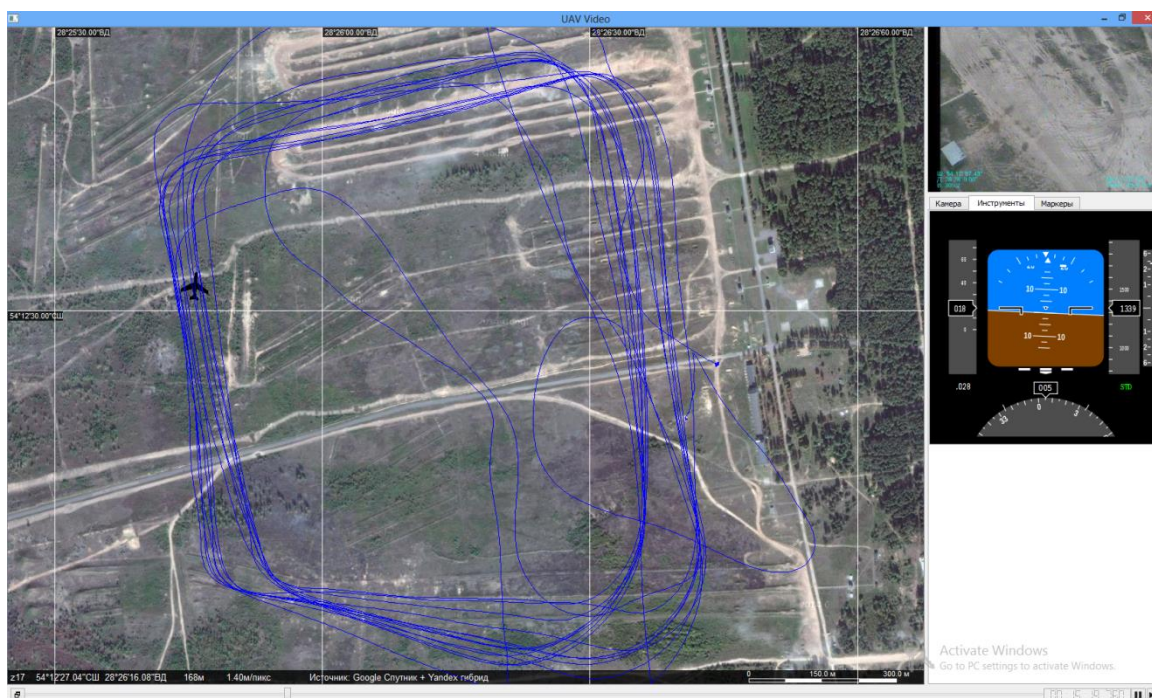


Рис. 3. Интерфейс системы моделирования движения БЛА с оптической навигационной системой

### Применение системы моделирования движения БЛА

Описанная программная система используется для анализа эффективности различных алгоритмов и форматов хранения данных для системы оптической навигации БЛА. В частности, проводятся исследования в следующих областях:

- оценка эффективности алгоритмов коррекции яркости/контраста/цветовой гаммы изображения;
- оценка надежности и быстродействия алгоритмов выделения реперных элементов на изображении;
- оценка эффективности способов хранения и алгоритмов поиска дескрипторов реперных элементов в базе данных;
- оценка надежности и быстродействия алгоритмов определения положения БЛА в пространстве с использованием цифровой карты реперных элементов;
- оценка надежности и быстродействия алгоритмов определения положения объектов в пространстве относительно БЛА;
- оценка оптимальности и быстродействия алгоритмов планирования траектории движения БЛА при сопровождении подвижных целей;
- оценка оптимальности и быстродействия алгоритмов планирования траектории движения БЛА при движении по заданному маршруту;
- оценка эффективности алгоритмов планирования траектории движения БЛА в пространстве с запретными зонами;
- оценка применимости различных стратегий управления для БЛА самолетного и вертолетного типов.

На рис. 3 приводится пользовательский интерфейс системы моделирования движения БЛА, позволяющий отображать траекторию и параметры полета, а также видео с оптической системы. Отображение всей информации производится в реальном масштабе времени, также существует возможность ретроспективного просмотра.

## Заключение

В работе произведен анализ параметров движения БЛА для моделирования полета с оптической навигационной системой. Рассмотрена задача формирования изображений бортовых камер для исследования алгоритмов оптической навигации и планирования траектории движения. Предложена система моделирования движения БЛА с оптической навигационной системой и выполнена ее программная реализация. Обозначены возможные направления исследований с использованием предложенной системы моделирования.

## MODELING OF THE MOVEMENT OF UAV WITH THE OPTICAL NAVIGATION SYSTEM

K.A. VOLKOV, H.M. KARBALAEI SALEH, A.V. PARASOCHKA,  
S.S. ZAITSEV, R.R. ROMANOV, M.B.M MAHMOOD

### Abstract

The analysis of the movement of unmanned aerial vehicle (UAV) parameters for modeling of flight with the optical navigation system is made. The problem of image formation of dash cameras is considered. Modeling of the movement of UAV with the optical navigation system is offered. The possible directions of researches with use of the offered system of modeling are designated.

*Keywords:* optical navigation, motion simulation, unmanned aerial vehicle.

### Список литературы

1. *Arkin R.C.* Behavior-based Robotics. Cambridge, 1998.
2. *Beard R.W.* // IEEE Transactions on Robotics and Automation. 2002. Vol. 18. P. 911-922.
3. *Pachter M., Ceccarelli N., Chandler P.R.* // Proceedings of the IEEE Conference on Decision and Control. New Orleans, 2007.
4. *Якубайлик О.Э., Попов В.Г.* // Вычислительные технологии, 2009. Т. 14. №6. С. 116-126.
5. *Жилнев М.Ю.* // Геоматика. 2009. №3. С. 56-64.
6. *Szeliski R.* Computer Vision: Algorithms and Applications Springer. New York, 2010.
7. *Форсайт Д., Понс Ж.* Компьютерное зрение. Современный подход. М., 2004.
8. *Андреевский Б.Р., Фрадков А.Л.* Современные направления синтеза систем автоматического управления летательными аппаратами. СПб, 2004.
9. *Богословский С.В., Дорофеев А.Д.* Динамика полетов летательных аппаратов. СПб, 2002.

УДК 621.391.7:512.772

## ФУРЬЕ-АНАЛИЗ ПРОИЗВЕДЕНИЯ ПСЕВДОСЛУЧАЙНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ КОСИНУСОИДАЛЬНЫХ ИМПУЛЬСОВ И ЕЕ ЗАДЕРЖАННОЙ КОПИИ

М.Ю. ХОМЕНОК

Белорусский государственный университет информатики и радиоэлектроники  
П. Бровка, 6, Минск, 220013, Беларусь

Поступила в редакцию 14 июня 2016

Приведены основные результаты анализа энергетического спектра произведения фазоманипулированного сигнала на полутактовой поднесущей и ее задержанной копии в соответствии с материалами депонированной рукописи [1].

*Ключевые слова:* ряд Фурье, энергетический спектр, корреляционный анализ, псевдослучайная последовательность, фазоманипулированный сигнал.

### Введение

В работе [2] выполнен анализ энергетического спектра произведения задержанных друг относительно друга двух копий псевдослучайной последовательности прямоугольных импульсов (ПСП ПИ), а в [3] – для сигнала с фазовой манипуляцией на  $180^\circ$  по закону ПСП.

Отличительной особенностью энергетического спектра в обоих случаях является наличие гармоник на тактовой частоте. Поэтому данный тип функционального преобразования находит применение для формирования опорного колебания, когерентного с тактовой частотой информационной последовательности [2].

Как следует из [1], в режиме детектирования с преобразованием частоты процесс на выходе преобразователя соответствует фазоманипулированному сигналу (ФМнС) на полутактовой поднесущей, который в режиме синхронизма соответствует псевдослучайной последовательности косинусоидальных импульсов (ПСП КИ), спектральные свойства которой представляют практический интерес также в системах передачи на основе модуляции минимальным сдвигом.

### Энергетический спектр произведения псевдослучайной последовательности косинусоидальных импульсов и ее задержанной копии

С целью анализа энергетического спектра определим амплитудный спектр произведения сигнала с манипуляцией фазы на  $180^\circ$  на полутактовой поднесущей по закону ПСП и его задержанной во времени копии.

Представим ФМнС в виде

$$S(t) = \sqrt{2}S_o A(t) \cos(2\pi f_o t + \varphi_o), \quad (1)$$

где  $S_o$  – амплитуда ФМнС;  $A(t) \in \pm 1$  – информационная последовательность;  $f_o$ ,  $\varphi_o$  – соответственно, частота и начальная фаза несущего колебания ФМнС.

Следовательно, произведение ФМнС и его задержанной во времени копии определится как

$$S'(t) = S(t) S(t - \tau) = S_o^2 A(t) A(t - \tau) \left[ \cos(2\pi f_o \tau) + \cos(2\pi 2f_o t - 2\pi f_o \tau + 2\phi_o) \right]. \quad (2)$$

Если  $S_o A(t)$  – псевдослучайная последовательность прямоугольных импульсов (ПСП ПИ), то энергетический спектр произведения ПСП ПИ и ее задержанной во времени копии

$$S_o^2 A(t) A(t - \tau) \quad (3)$$

в интервале  $|\tau| < \tau_{\text{и}}$  задается следующим выражением [2]:

$$N(f, \tau) = S_o^4 \left[ 1 - \frac{|\tau|}{\tau_{\text{и}}} \left( 1 + \frac{1}{N} \right) \right]^2 \delta(f) + \sum_{n=1}^{\infty} \frac{2S_o^4}{(\pi n)^2} \sin^2 \left( \frac{\pi n \tau}{\tau_{\text{и}}} \right) \delta \left( f - \frac{n}{\tau_{\text{и}}} \right) + \sum_{n=1}^{\infty} \frac{2S_o^4 (N+1)}{(\pi n)^2} \sin^2 \left( \frac{\pi n \tau}{N\tau_{\text{и}}} \right) \delta \left( f - \frac{n}{N\tau_{\text{и}}} \right), \quad (4)$$

где  $N$  – число элементарных символов в ПСП;  $\tau_{\text{и}}$  – длительность элементарного символа;  $\tau$  – время задержки;  $\delta(f)$  – дельта-функция;  $N(f, \tau)$  – энергетический спектр сигнала (3).

Соответственно, амплитудный спектр  $G_a(f, \tau)$  произведения ПСП ПИ и ее задержанной во времени копии равен

$$G_a(f, \tau) = S_o^2 \left[ 1 - \frac{|\tau|}{\tau_{\text{и}}} \left( 1 + \frac{1}{N} \right) \right] \delta(f) + \sum_{n=1}^{\infty} \frac{2S_o^2}{\pi n} \left| \sin \left( \frac{\pi n \tau}{\tau_{\text{и}}} \right) \right| \delta \left( f - \frac{n}{\tau_{\text{и}}} \right) + \sum_{n=1}^{\infty} \frac{2S_o^2 \sqrt{N+1}}{\pi n} \left| \sin \left( \frac{\pi n \tau}{N\tau_{\text{и}}} \right) \right| \delta \left( f - \frac{n}{N\tau_{\text{и}}} \right). \quad (5)$$

Поэтому амплитудный спектр сигнала в (2) определится в соответствии с выражением

$$S_o^2 \left[ 1 - \frac{|\tau|}{\tau_{\text{и}}} \left( 1 + \frac{1}{N} \right) \right] \left[ \delta(f) \cos(2\pi f_o \tau) + \delta(f - 2f_o) \right] + \sum_{n=1}^{\infty} \frac{2S_o^2}{\pi n} \left| \sin \left( \frac{\pi n \tau}{\tau_{\text{и}}} \right) \right| \left[ \delta \left( f - \frac{n}{\tau_{\text{и}}} \right) \cos(2\pi f_o \tau) + 1/2 \delta \left( f - 2f_o \pm \frac{n}{\tau_{\text{и}}} \right) \right] + \sum_{n=1}^{\infty} \frac{2S_o^2 \sqrt{N+1}}{\pi n} \left| \sin \left( \frac{\pi n \tau}{N\tau_{\text{и}}} \right) \right| \left[ \delta \left( f - \frac{n}{N\tau_{\text{и}}} \right) \cos(2\pi f_o \tau) + 1/2 \delta \left( f - 2f_o \pm \frac{n}{N\tau_{\text{и}}} \right) \right]. \quad (6)$$

Если  $f_o = 1/2 f_T$ , а  $\phi_o = 1/2 \phi_T$ , то сигнал, определенный в (1), имеет смысл ФМНС на полутактовой поднесущей. В этом случае, как показывает анализ слагаемого

$$\frac{S_o^2}{\pi n} \left| \sin \left( \frac{\pi n \tau}{\tau_{\text{и}}} \right) \right| \delta \left( f - 2f_o \pm \frac{n}{\tau_{\text{и}}} \right),$$

в (6) при  $n=1$  и  $\tau = 1/2 \tau_{\text{и}}$  в спектре сигнала (2), имеют место биения между гармоникой спектра сигнала (3) на тактовой частоте и несущим колебанием сигнала (2) также на тактовой частоте, т.е.

$$A_T(\tau) \sin(2\pi f_T t + \phi_T) \cos(2\pi 2f_o t - 2\pi f_o \tau + 2\phi_o),$$

где  $A_T(\tau)$ ,  $\phi_T$  – соответственно, амплитуда и начальная фаза гармоники спектра сигнала (3) на тактовой частоте, определяемой значащими моментами ПСП.



В результате этих биений в спектре сигнала (2) возникает составляющая, равная

$$1/2 A_T(\tau) \sin[2\pi(f_T - 2f_o)t + 2\pi f_o \tau + \varphi_T - 2\varphi_o] \quad (7)$$

и соответствующая при  $f_o = 1/2 f_T$  и  $\tau = 1/2 \tau_{и}$  дискриминационной характеристике фазовой автоподстройки:

$$1/2 A_T(\tau_{и}/2) \sin(\pi/2 + \varphi_T - 2\varphi_o), \quad \text{где } 1/2 A_T(\tau_{и}/2) = S_o^2/\pi.$$

Другой особенностью спектра сигнала (2) является составляющая при  $f_o \neq 1/2 f_T$ , определяемая выражением

$$S_o^2 [1 - 1/2(1 + 1/N)] \cos(2\pi f_o \tau),$$

которое при  $f_o = 1/2 f_T + \Delta f$ ,  $\tau = 1/2 \tau_{и}$  приводится к виду

$$1/2 S_o^2 \cos[(1/2 f_T + \Delta f) \tau_{и}] = 1/2 S_o^2 \cos(\pi/2 + \pi \Delta f \tau_{и}) = -A_n \sin(\pi \Delta f \tau_{и}), \quad (8)$$

где  $A_n = 1/2 S_o^2$  – постоянная составляющая сигнала (3). Равенство (8) соответствует дискриминационной характеристике частотной автоподстройки с полосой захвата  $\Delta f = \pm 1/2 f_T$ , реализуемой в схеме многочастотного дискриминатора.

При  $f_o = 1/2 f_T$ ,  $\varphi_o = 1/2 \varphi_T$ ,  $\tau = 1/2 \tau_{и}$  выражение (4) определяет амплитудный спектр произведения смещенных во времени на полтакта друг относительно друга копий последовательности косинусоидальных импульсов (КИ), образованной произведением ПСП ПИ и полутактовой поднесущей, синхронной с ее значащими моментами. Этот спектр определяется выражением

$$\begin{aligned} & \frac{S_o^2}{\pi} \delta(f) + 1/2 S_o^2 \left[ 1 - 1/2 \left( 1 + \frac{1}{N} \right) \right] \delta(f \pm f_T) + \sum_{\substack{n=-\infty \\ n \neq 0}}^{\infty} \frac{2n S_o^2}{\pi(4n^2 - 1)} \left| \sin \left( \frac{\pi(2n-1)}{2} \right) \right| \delta \left( f - \frac{2n}{\tau_{и}} \right) \\ & + \sum_{n=-\infty}^{\infty} \frac{S_o^2 \sqrt{(N+1)}}{2\pi n} \left| \sin \left( \frac{\pi n}{2N} \right) \right| \delta \left( f \pm f_T - \frac{n}{N\tau_{и}} \right). \end{aligned} \quad (9)$$

Как следует из (9) и как это показано на рисунке, в этом случае спектр содержит постоянную составляющую и гармоники, наиболее интенсивные из которых расположены на частотах  $f_T$  и  $2nf_T$ , где  $n = 1, 2, 3, \dots$ .

Для получения произведения копий ПСП КИ, задержанных друг относительно друга на полтакта, необходимо функцию (9), определяющую спектр сигнала (2) при указанном выше условии, умножить на периодическую последовательность прямоугольных импульсов (ПП ПИ) тактовой частоты и единичной амплитуды, ряд Фурье для которой имеет вид:

$$A_{10}(t) = \sum_{n=1}^{\infty} \frac{4(-1)^{n-1}}{\pi(2n-1)} \sin \left[ \frac{2\pi}{\tau_{и}} (2n-1)t + \varphi_T \right]. \quad (10)$$

Это эквивалентно умножению сигнала (2) при  $f_o = 1/2 f_T$ ,  $\varphi_o = 1/2 \varphi_T$ , на ПП ПИ полутактовой частоты. Тогда гармоническая составляющая на тактовой частоте в спектре произведения ПСП КИ с ее задержанной во времени на  $\tau_{и}/2$  копии определится путем биений гармоник спектра (9) на частотах  $2n/\tau_{и}$  с гармониками ПП ПИ (10) на частотах  $(2n-1)/\tau_{и}$ ,  $n = 1, 2, 3, \dots$ , и произведением постоянной составляющей спектра (9) на гармонику тактовой частоты ПП ПИ (10).

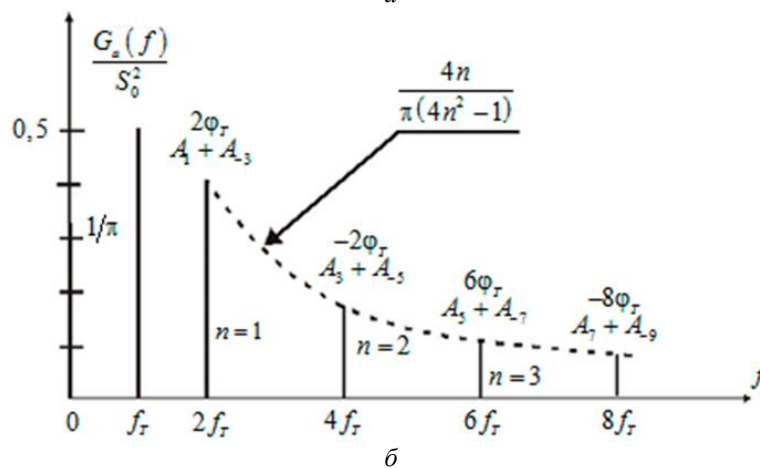
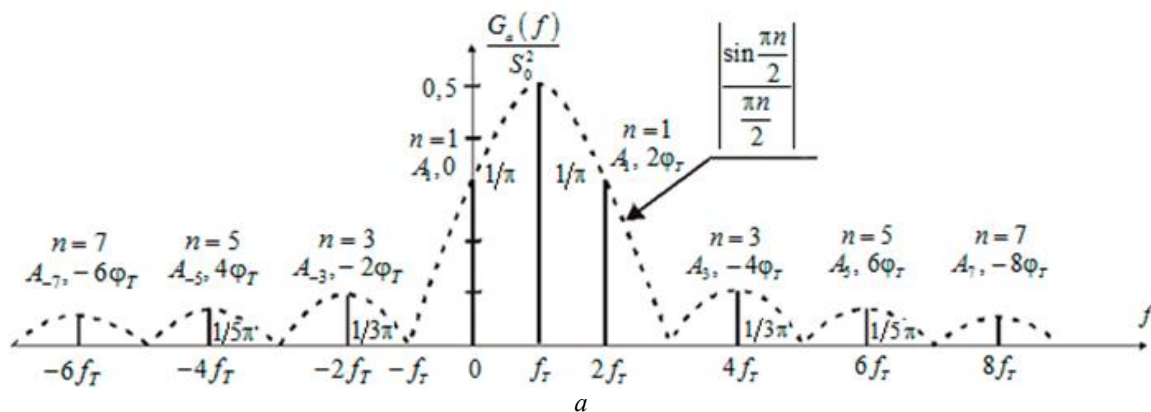
Таким образом, с учетом того, что амплитуды гармоник в (9) на частотах  $2n/\tau_{и}$  модулированы по знаку в соответствии с  $(-1)^{n-1}$ , амплитуда гармоники на тактовой частоте в спектре произведения копий ПСП КИ, задержанных друг относительно друга на полтакта, равна

$$\frac{4S_o^2}{\pi^2} \left[ 1 + 4 \sum_{n=1}^{\infty} \frac{n}{(4n^2 - 1)^2} \right] = \frac{6S_o^2}{\pi^2}, \text{ где } \sum_{n=1}^{\infty} \frac{n}{(4n^2 - 1)^2} = \frac{1}{8}.$$

При этом постоянная составляющая определится биениями гармоник на тактовой частоте сигналов (2) и (10) и окажется равной  $S_o^2/\pi$ .

В таблице для сравнения приведены значения постоянной составляющей и амплитуд гармоник на тактовой частоте в спектре произведения задержанных на пол такта друг относительно друга копий ПСП ПИ и ПСП КИ.

Составляющая спектра	ПСП ПИ	ПСП КИ
Постоянная составляющая	$1/2 S_o^2$	$S_o^2/\pi$
Гармоника на тактовой частоте	$2 S_o^2/\pi$	$6 S_o^2/\pi^2$



Дискретные составляющие амплитудного спектра перемножения двух смещенных на полтакта друг относительно друга копий последовательности косинусоидальных импульсов:  
 а – математический спектр; б – физический спектр

### Заключение

Анализ дискретных компонент спектра (6) на нулевой и тактовой частотах, определяемых соответственно выражениями (7) и (8), для сигнала (2) в режиме отсутствия синхронизма между значащими моментами манипулирующей последовательности и поднесущей показывает возможность реализации совместной схемы частотно-фазовой автоподстройки на основе схемы многочастотного дискриминатора [1]. Это дает дополнительные возможности для оптимизации точности слежения и динамики работы системы синхронизации.

В системах тактовой синхронизации, использующих данный тип нелинейного преобразования для косинусоидальной формы несущего импульса, энергетические потери по мощности, как следует из сравнения результатов, приведенных в таблице, составляют ~0,4 дБ.

Снижение энергетика на гармонике тактовой частоты в  $\pi^2/9$  раз, обусловлено гладкой формой косинусоидального типа несущего импульса и, как результат, следует уменьшение крутизны его фронта по отношению к импульсу прямоугольной формы.

## **FOURIER ANALYSIS OF PRODUCT FOR BOTH A PSEUDORANDOM SEQUENCE OF COSINE PULSES AND ITS DETAINED COPY**

M.J. HOMENOK

### **Abstract**

The article reflects the main results of energy spectrum analysis of the phase manipulated signal on the subcarrier of half frequency clock signal in compliance with the deposit manuscript [1].

*Keywords:* Fourier series, power spectrum, correlation analysis, pseudorandom sequence, phase-shift keyed signal.

### **Список литературы**

1. Хоменок М.Ю. // Деп. в БелНИИНТИ №805Бе-Д83. 1983.
2. Стилкер Дж. Цифровая спутниковая связь. М., 1979.
3. Чердынцев В.А., Ходасевич Р.Г. // Радиотехника и электроника. 1975. Т. 20. С. 427-431.

УДК 621.391

## АППАРАТНО-ОРИЕНТИРОВАННЫЙ АЛГОРИТМ СЖАТИЯ ИЗОБРАЖЕНИЙ НА ОСНОВЕ АДАПТИВНОЙ ИЕРАРХИЧЕСКОЙ КЛАСТЕРИЗАЦИИ ВЕЙВЛЕТ-КОЭФФИЦИЕНТОВ

В.В. НОВИЦКИЙ, В.Ю. ЦВЕТКОВ, К.М.О. АЛЬ-ГЕРТАНИ, А.А.Р. АЛЬ-АББУДИ

*Белорусский государственный университет информатики и радиоэлектроники  
П. Бровка, 6, Минск, 220013, Беларусь*

*Поступила в редакцию 10 мая 2016*

Дано строгое математическое описание нового алгоритма вейвлет-сжатия изображений, обладающего низкой вычислительной сложностью и высокой степенью параллелизма. Он предназначен для аппаратной реализации в ПЛИС, которая будет сжимать поток аэрокосмических изображений на борту спутника дистанционного зондирования Земли. В качестве прототипов были выбраны алгоритмы SPECK и МЕСТ. Численное моделирование на тестовых аэрокосмических изображениях показало хорошие результаты.

*Ключевые слова:* вейвлет-сжатие, спутник дистанционного зондирования Земли, ПЛИС.

### Введение

Актуальность проблемы разработки новых алгоритмов сжатия обусловлена быстрым ростом количества информации, которое необходимо передавать по различным сетям и относительно медленным ростом пропускной способности каналов, на которых строятся данные сети. Частным случаем сетей с существенно ограниченной пропускной способностью каналов является система дистанционного зондирования земли (ДЗЗ), где самым медленным является канал между спутником ДЗЗ и пунктом приема целевой информации (ЦИ). Отсюда вытекает задача разработки алгоритма вейвлет-сжатия с высокой степенью параллелизма и низкой вычислительной сложностью, который хорошо подходит для реализации в кристалле ПЛИС и в то же время сопоставим по качеству восстановленных изображений с существующими алгоритмами.

### Математическое описание нового алгоритма вейвлет-сжатия

*Формирование масштабируемых вейвлет-структур.* Наиболее эффективные алгоритмы сжатия изображений [1–5] работают в области дискретного вейвлет-преобразования. С его помощью осуществляется формирование матрицы вейвлет-коэффициентов  $C = c(i, j)_{(i=0, \sqrt{N}-1, j=0, \sqrt{N}-1)}$  исходного полутонового изображения  $I$ . Двумерное дискретное вейвлет-преобразование, формирующее пространственно-частотное представление исходного изображения  $I = I(i, j)_{(i=0, \sqrt{N}-1, j=0, \sqrt{N}-1)}$  можно представить в виде

$$W(I, J) = C, \quad (1)$$

где  $J$  – заданная глубина вейвлет-разложения изображения  $I$ ;  $0 < J \leq J_{\max}$ ;  $J_{\max} = \log_2(\sqrt{N})$  – максимально возможное количество уровней разложения;  $N$  – число вейвлет-коэффициентов исходного изображения  $I$ .

*Математическое описание алгоритма НВСТ (Hardware Block Cluster Tree).* Для формирования прогрессивного выходного потока (масштабируемого кода) необходимо кодировать матрицу вейвлет-коэффициентов  $C$  в определенном порядке (начиная со старшей битовой плоскости и пространственно-частотного поддиапазона  $L$ ). В связи с этим предлагается алгоритм быстрого кодирования вейвлет-коэффициентов исходного полутонового изображения, допускающий множественное распараллеливание и предполагающий аппаратную реализацию с низким количеством вычислительных ресурсов. Данный алгоритм основан на иерархической кластеризации сегмента битовой плоскости фиксированного размера и построении оптимального кода на основе оценки компактности кластеризованного сегмента. В качестве прототипов были выбраны алгоритмы SPECK [4] и МЕСТ [6]. Алгоритм НВСТ состоит из следующих шагов.

Шаг 1. Сканирование матрицы вейвлет-коэффициентов  $C = c(i, j)_{(i=0, \overline{\sqrt{N}-1}, j=0, \overline{\sqrt{N}-1})}$  размерностью  $\sqrt{N} \times \sqrt{N}$  в одномерный массив  $C_Z = c_Z(i)_{(i=0, \overline{N-1})}$  по Z-развертке (рис. 1):

1.1 Формирование двумерного массива координат Z-развертки  $Z = z(i, j)_{(i=0, \overline{1}, j=0, \overline{N-1})}$  (пример на рис. 2);

1.2 Инициализация массива  $C_Z$  элементами матрицы  $C$  по массиву координат  $Z$ :

$$\begin{cases} C_Z = c_Z(i)_{(i=0, \overline{N-1})}, \\ c_Z(i) = c(z(0, i), z(1, i)). \end{cases} \quad (2)$$

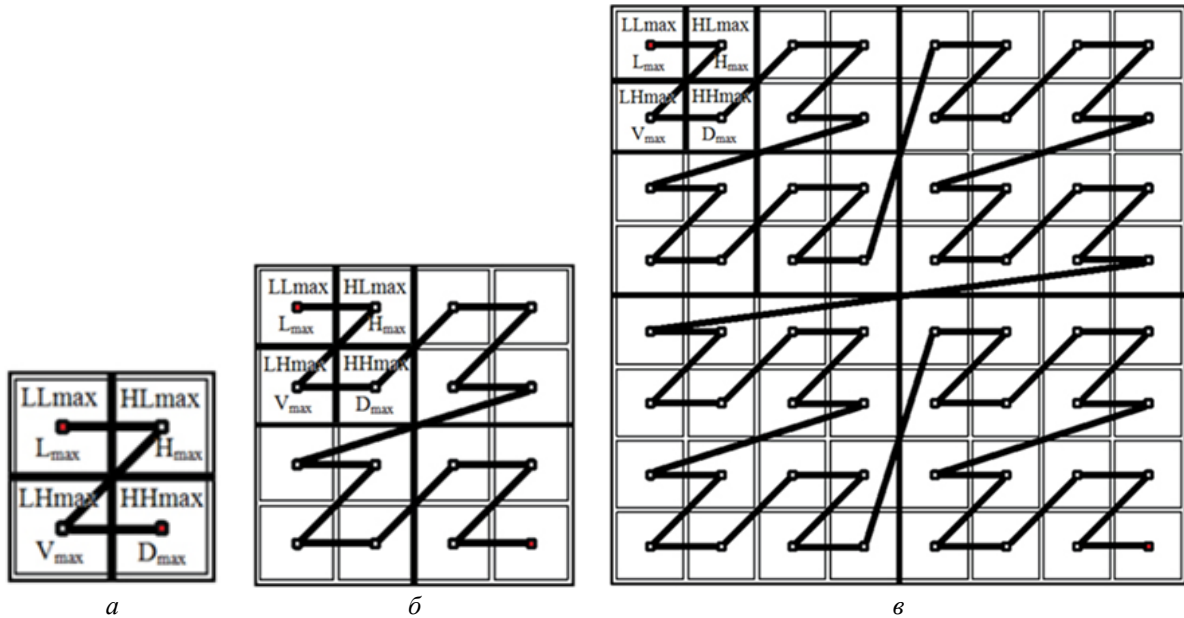


Рис. 1. Три масштаба Z-развертки:  $a - (m = 1)$ ;  $b - (m = 2)$ ;  $v - (m = 3)$

Сканирование в таком порядке позволяет сохранить пространственное масштабирование выходного потока и существенно упрощает дальнейшее кодирование матрицы вейвлет-коэффициентов.

Шаг 2. Выбор длины кодового блока  $N_{CB}$  (сегмента битовой плоскости массива  $C_Z$ ), причем длина массива  $C_Z$  должна быть кратна  $N_{CB}$  и  $\log_2 N_{CB}$  должен быть целым числом не меньше трех. Оптимальным считается  $N_{CB} = 1024$ . Исходя из выбранного размера кодового блока, рассчитывается максимальное число уровней кластеризации

$$L_M = \log_4 N_{CB}. \quad (3)$$

В процессе кодирования используется два значения уровней кластеризации  $\{1, L_M\}$ . Таким образом выбор  $N_{CB}$  влияет на быстродействие, степень сжатия и вычислительную сложность алгоритма НВСТ. Как правило, чем больше  $N_{CB}$ , тем выше степень сжатия, но тем больше аппаратных ресурсов требует алгоритм для своей реализации.

	Текущий индекс	(Y,X)-координаты массива
LL2	0	(0,0)
HL2	1	(0,1)
LH2	2	(1,0)
HH2	3	(1,1)
	4	(0,2)
	5	(0,3)
HL1	6	(1,2)
	7	(1,3)
	8	(2,0)
LH1	9	(3,0)
	10	(2,1)
	11	(3,1)
	12	(2,2)
HH1	13	(2,3)
	14	(3,2)
	15	(3,3)

Рис. 2. Пример массива формирования массива координат Z-развертки для  $m = 2$

Шаг 3. Задание максимально допустимого количества бит  $B_{MAX}$  для выходного прогрессивного потока. При сжатии без потерь и с потерями  $B_{MAX}$  определяется с помощью соотношения

$$B_{MAX} = \left( \frac{N_{CB} \times (N_{CB} + 2)}{N} \times B_{PP} + 8 \right) / C_R^*, \quad (4)$$

где  $B_{PP}$  – число бит на пиксель исходного изображения;  $C_R^*$  – заданный коэффициент сжатия. В случае сжатия без потерь  $C_R^* = 1$ .

Шаг 4. Инициализация массива  $M_{SP}$  значимых вейвлет-коэффициентов (маски)

$$\begin{cases} M_{SP} = m_{SP}(i)_{(i=0, \overline{N-1})}, \\ m_{SP}(i) = 0, \end{cases} \quad (5)$$

где  $m_{SP}(i) \in \{0,1\}$ ;  $i$  – номер вейвлет-коэффициента  $c_Z(i)$ .

Шаг 5. Определение начального значения порога  $T$  значимости вейвлет-коэффициентов

$$T = 2^n, \quad (6)$$

где  $n = n_{MAX}$ ,  $0$  – номер битовой плоскости массива вейвлет-коэффициентов  $C_Z = c_Z(i)_{(i=0, \overline{N-1})}$ ,

$c_Z(i) = \text{sign}(c_Z(i)) \times \left( \sum_{n=0}^{n_{MAX}} b_n 2^n \right)$  – значение  $i$ -го вейвлет-коэффициента;  $b_n \in \{0,1\}$  –  $n$ -й бит  $c_Z(i)$ .

Шаг 6. Инициализация массива  $M_{PS}$  (в который будет записываться выходной прогрессивный поток алгоритма НВСТ) и текущего числа бит  $B_{PS}$ :

$$\begin{cases} M_{PS} = m_{PS}(b)_{(b=0, \overline{B_{MAX}-1})}, \\ m_{PS}(i) = \begin{cases} n_{MAX_i} & \text{при } 0 \leq i < 8, \\ 0 & \text{при } 8 \leq i < B_{MAX}, \end{cases} \end{cases} \quad (7)$$

$$B_{PS} = 8, \quad (8)$$

где  $m_{PS}(b) \in \{0,1\}$ ;  $b$  – номер элемента (бита) массива  $M_{PS}$ ;  $n_{MAX_i}$  –  $i$ -й бит номера старшей битовой плоскости, который записывается в начало потока.

Шаг 7. Инициализация указателя  $P_{CB}$  на текущий сегмент кодируемой битовой плоскости и номера текущей битовой плоскости

$$P_{CB} = 0, \quad (9)$$

$$n = n_{MAX}. \quad (10)$$

Кодирование начинается со старшей битовой плоскости и нулевого сегмента. Количество сегментов (кодowych блоков) битовой плоскости определяется частным  $N / N_{CB}$ .

Шаг 8. Начало цикла кодирования по алгоритму НВСТ. Инициализация массива кодового блока  $M_{CB}$  битами вейвлет-коэффициентов в текущем сегменте  $n$ -й битовой плоскости

$$\begin{cases} M_{CB} = m_{CB}(i)_{(i=0, \overline{N_{CB}-1})}, \\ m_{CB}(i) = \begin{cases} 1 & \text{при } n \leq \log_2 |c_Z(P_{CB} + i)| \leq n + 1, \\ 0 & \text{иначе,} \end{cases} \end{cases} \quad (11)$$

где  $m_{CB}(i) \in \{0,1\}$ .

Шаг 9. Инициализация  $L_M$ -уровневого массива  $M_{CL}$  кластеризации сегмента  $n$ -й битовой плоскости

$$\begin{cases} M_{CL} = m_{CL}(i)_{(i=0, \overline{S_{LM}-1})}, \\ m_{CL}(i) = \begin{cases} m_{CB}(i) & \text{при } 0 \leq i < N_{CB}, \\ 0 & \text{при } N_{CB} \leq i < S_{LM}, \end{cases} \end{cases} \quad (12)$$

где  $m_{CL}(i) \in \{0,1\}$ ;  $S_{LM} = (4N_{CB} - N_{LM}) / 3$  – общее число исходных и кластеризованных элементов массива  $M_{CL}$  на уровнях  $0, L_M$ ;  $N_{LM}$  – число кластерных деревьев на  $L_M$ -м уровне кластеризации

$$N_{LM} = N_{CB} / N_C^{L_M}, \quad (13)$$

где  $N_C$  – количество элементов в кластере (для алгоритма НВСТ равно 4).

Шаг 10. Инициализация коэффициентов оценки компактности для уровней кластеризации  $L=1$  и  $L=L_M$ :

$$\{C_{L_1}, C_{L_M}\} = 0. \quad (14)$$

Шаг 11.  $L_M$ -уровневая иерархическая кластеризация  $M_{CL}$ . Значения элементов массива  $M_{CL}$  на  $L$  уровне определяются с помощью соотношения

$$m_{CL}(S(L) + i) = \text{Bin} \left( \sum_{j=0}^3 m_{CL} \left( S(L) - \frac{N_{CB}}{4^{L-1}} + 4i + j \right) \right), \quad (15)$$

где  $L = \overline{1, L_M}$ ;  $0 \leq i < \frac{N_{CB}}{4^L}$ ;  $S(L) + i$  – абсолютный номер элемента массива  $M_{CL}$  на  $L$ -м уровне кластеризации;  $i$  – номер кластера на  $(L-1)$ -м уровне или относительный номер элемента на текущем  $L$ -м уровне кластеризации;  $S(L) = \sum_{l=0}^{L-1} \left( \frac{N_{CB}}{4^l} \right)$  – общее количество элементов  $M_{CL}$ , принадлежащих уровням кластеризации  $[0, L-1]$ ;  $j$  – номер элемента в пределах кластера,  $\text{Vin}(x) = \begin{cases} 1 & \text{при } |x| > 0, \\ 0 & \text{при } x = 0, \end{cases}$  – пороговая функция. Конечным результатом шага является массив

$M_{CL}$  вплоть до  $L_M$ -го уровня, элементы которого определяются соотношением

$$m_{CL}(i) = \begin{cases} m_{CB}(i) & \text{при } 0 \leq i < N_{CB}, \\ m_{CL}(S(L) + i) & \text{при } N_{CB} \leq i < S_{L_M}. \end{cases} \quad (16)$$

Шаг 12. Оценка компактности кодирования для уровней кластеризации  $L=1$  и  $L=L_M$ :

$$C_{L_1} = \left\{ \sum_{j=0}^{\frac{N_{CB}-1}{4}} m_{CL}(S(1) + j) \right\} \times 4 + \frac{N_{CB}}{4}, \quad (17)$$

$$C_{L_M} = \sum_{l=1}^{L_M} \left\{ \left( \sum_{j=0}^{\frac{N_{CB}-1}{4^l}} m_{CL}(S(l) + j) \right) \times 4 \right\}, \quad (18)$$

$$C_{\min} = \min(C_L, C_{L_M}, N_{CB}), \quad (19)$$

где  $C_{\min}$  – минимально возможное число бит для кодирования заданного сегмента  $M_{CB}$  алгоритмом НВСТ.

Шаг 13. Проверка выполнения условия  $C_{\min} = C_{L_M}$ . В случае истины оптимальным является код, получаемый в результате кластеризации с числом уровней  $L=L_M$ . Перед началом кодирования в выходной поток необходимо поместить флаг, который будет указывать декодеру количество уровней кластеризации:

$$\begin{cases} m_{PS}(B_{PS}) = 0, \\ m_{PS}(B_{PS} + 1) = \begin{cases} 0 & \text{при } C_{L_M} = 0, \\ 1 & \text{иначе,} \end{cases} \\ B_{PS} = B_{PS} + 2. \end{cases} \quad (20)$$

Флаг «00» сигнализирует о том, что в текущем сегменте нет значимых вейвлет-коэффициентов (все биты сегмента равны нулю). После записи флага идет формирование и запись в выходной поток кода, компактно описывающего текущий сегмент  $n$ -й битовой плоскости:

$$m_{PS}(B_{PS}) = \begin{cases} m_{CL}(S(L-1) + j) & \text{при } m_{CL}\left(S(L) + \frac{j}{4}\right) > 0, \\ B_{PS} = B_{PS} + 1 & \text{при } m_{CL}\left(S(L) + \frac{j}{4}\right) > 0, \end{cases} \quad (21)$$

где  $L = \overline{1, L_M}$ ;  $0 \leq j < \frac{N_{CB}}{4^{L-1}}$ ;  $S(L) + \frac{j}{4}$  – абсолютный номер вершины текущего кластерного дерева.



Шаг 14. Проверка выполнения условия  $C_{\min} = C_{L_1}$ . В случае истины оптимальным является код, получаемый в результате кластеризации с числом уровней  $L = L_1$ . Перед началом кодирования в выходной поток необходимо поместить флаг, который будет указывать декодеру количество уровней кластеризации

$$\begin{cases} m_{PS}(B_{PS}) = 1, \\ m_{PS}(B_{PS} + 1) = 0, \\ B_{PS} = B_{PS} + 2. \end{cases} \quad (22)$$

После записи флага идет формирование и запись в выходной поток кода, компактно описывающего текущий сегмент  $n$ -й битовой плоскости:

$$m_{PS}(B_{PS}) = \begin{cases} m_{CL}(S(1) + j) \text{ при } 0 \leq j < \frac{N_{CB}}{4}, \\ B_{PS} = B_{PS} + 1, \end{cases} \quad (23)$$

$$m_{PS}(B_{PS}) = \begin{cases} m_{CL}(S(0) + j) \text{ при } m_{CL}\left(S(1) + \frac{j}{4}\right) > 0, \\ B_{PS} = B_{PS} + 1 \text{ при } m_{CL}\left(S(1) + \frac{j}{4}\right) > 0, \end{cases} \quad (24)$$

где  $0 \leq j < N_{CB}$ ;  $S(1) + \frac{j}{4}$  – абсолютный номер вершины текущего кластерного дерева.

Шаг 15. Проверка выполнения условия  $C_{\min} = N_{CB}$ . В случае истины текущий сегмент  $n$ -й битовой плоскости является шумом, и оптимальным вариантом будет его непосредственная передача в выходном потоке после записи соответствующего флага:

$$\begin{cases} m_{PS}(B_{PS}) = 1, \\ m_{PS}(B_{PS} + 1) = 1, \\ B_{PS} = B_{PS} + 2. \end{cases} \quad (25)$$

$$m_{PS}(B_{PS}) = \begin{cases} m_{CL}(S(0) + j) \text{ при } 0 \leq j < N_{CB}, \\ B_{PS} = B_{PS} + 1. \end{cases} \quad (26)$$

Шаг 16. Формирование информации о знаках значимых в текущем сегменте  $n$ -й битовой плоскости вейвлет-коэффициентов и запись их в выходной поток:

$$\begin{cases} m_{PS}(B_{PS}) = \begin{cases} 1 \text{ при } m_{CB}(i) > 0 \text{ и } m_{SP}(P_{CB} + i) = 0 \text{ и } c_Z(P_{CB} + i) \geq 0, \\ 0 \text{ при } m_{CB}(i) > 0 \text{ и } m_{SP}(P_{CB} + i) = 0 \text{ и } c_Z(P_{CB} + i) < 0, \end{cases} \\ B_{PS} = B_{PS} + 1, \end{cases} \quad (27)$$

где  $0 \leq i < N_{CB}$ .

Шаг 17. Запись информации о значимости вейвлет-коэффициентов текущего сегмента  $n$ -й битовой плоскости в массив  $M_{SP}$ . В данном массиве также хранится информация о значимости вейвлет-коэффициентов всех предыдущих битовых плоскостей:

$$m_{SP}(i) = m_{CB}(P_{CB} + i) \text{ при } m_{CB}(P_{CB} + i) > 0, \quad (28)$$

где  $0 \leq i < N_{CB}$ .

Шаг 18. Окончание цикла кодирования вейвлет-коэффициентов и завершающий шаг алгоритма. Выполнение инкремента указателя на текущий сегмент  $n$ -й битовой плоскости:

$$P_{CB} = P_{CB} + N_{CB} \cdot \quad (29)$$

Проверка на конец битовой плоскости  $P_{CB} = N_{CB}$ . В случае истины декремент номера текущей битовой плоскости

$$n = n - 1, P_{CB} = 0, \quad (30)$$

иначе – переход к шагу 8. Проверка  $n$  на равенство нулю и завершение алгоритма НВСТ, иначе – переход к шагу 8.

### Результаты и их обсуждение

В таблице представлены численные характеристики алгоритмов JPEG2000 [5], МЕСТ [6] и НВСТ для известного тестового изображения «Lena».

Численные результаты алгоритмов для изображения «Lena»

Алгоритм	Режим сжатия	Без потерь		Потери, 2 bpp		Потери, 1 bpp		Потери, 0.5 bpp	
JPEG2000	Семейство вейвлетов	CDF 5/3		CDF 9/7		CDF 9/7		CDF 9/7	
	Схема, уровни ДВП	Пирамида, 5		Пирамида, 5		Пирамида, 5		Пирамида, 5	
	CR	1,861		4,032		8,033		16,072	
	PSNR, дБ	$\infty$		<b>43,364</b>		<b>39,290</b>		<b>36,289</b>	
МЕСТ	Семейство вейвлетов	CDF 5/3		CDF 9/7		CDF 9/7		CDF 9/7	
	Схема, уровни ДВП	Пирамида, 7		Пирамида, 7		Пирамида, 7		Пирамида, 7	
	CR	1,757		3,998		7,995		15,981	
	PSNR, дБ	$\infty$		<b>42,856</b>		<b>38,602</b>		<b>35,242</b>	
НВСТ	Семейство вейвлетов	CDF 5/3		CDF 9/7		CDF 9/7		CDF 9/7	
	Схема, уровни ДВП	Пир. 4	Клас. 4	Пир. 4	Клас. 4	Пир. 4	Клас. 4	Пир. 4	Клас. 4
	CR	1,791	1,750	3,998	3,999	7,999	7,993	15,972	15,994
	PSNR, дБ	$\infty$	$\infty$	<b>43,565</b>	<b>43,111</b>	<b>38,945</b>	<b>38,665</b>	<b>35,655</b>	<b>35,404</b>
	Семейство вейвлетов	CDF 5/3		CDF 5/3	Haar	CDF 5/3	Haar	CDF 5/3	Haar
	Схема, уровни ДВП	Пирамида, 3		Пир. 3	Пир. 4	Пир. 3	Пир. 4	Пир. 3	Пир. 4
	CR	1,780		3,999	3,999	7,993	7,998	15,999	15,988
	PSNR, дБ	$\infty$		<b>41,875</b>	<b>41,128</b>	<b>37,539</b>	<b>36,107</b>	<b>33,431</b>	<b>32,360</b>

При вычислении численных результатов в алгоритме НВСТ использовались различные вариации вейвлет-декомпозиции исходного изображения, такие как: количество уровней декомпозиции, схема декомпозиции и семейство вейвлетов. Это делалось с учетом того факта, что аппаратная реализация, к примеру, декомпозиции по классической схеме требует меньше ресурсов и времени, чем пирамидальная схема. В итоге можно сделать вывод, что алгоритм НВСТ выигрывает 0,6 дБ у алгоритма МЕСТ и проигрывает 1 дБ алгоритму JPEG2000 (кодэк Jasper) независимо от типа изображения.

### Заключение

Разработан простой и эффективный метод вейвлет-сжатия НВСТ, который обладает низкой вычислительной сложностью и высокой степенью параллелизма, что делает его хорошо пригодным для реализации в кристалле ПЛИС. Численное моделирование для тестовых изображений показало, что НВСТ выигрывает у прототипа МЕСТ порядка 0,6 дБ по метрике PSNR при сжатии с потерями и проигрывает 1 дБ JPEG2000, который считается лучшим алгоритмом вейвлет-сжатия изображений с потерями, но обладает высокой вычислительной сложностью.

# THE HARDWARE-ORIENTED IMAGE COMPRESSION ALGORITHM BASED ON ADAPTIVE HIERARCHICAL CLUSTERING OF THE WAVELET COEFFICIENTS

V.U. NAVITSKI, V.Yu. TSVIATKOU, Q.M.O. AL-GERTANI, A.A.R. AL-ABBOODI

## Abstract

The new algorithm of image wavelet-compression has been mathematically described, it has low-complexity and high degree of parallelism. It is destined to FPGA hardware implementation, which will compress a stream of aerospace on the board of ERS satellite. There are two algorithms, such as SPECK and MECT, were chosen as prototypes. Numeral modeling with test aerospace images has shown good results.

*Keywords:* wavelet compression, Earth remote sensing satellite, FPGA.

## Список литературы

1. *Shapiro J.M.* // IEEE Trans. Signal Processing. 1993. №41. P. 3445-3462.
2. *Said A.A., Pearlman W.A.* // IEEE Trans. On Circuits and Systems for Video Technology. 1996. Vol. 6. P. 243-250.
3. *Taubman D.S.* // IEEE Transactions on Image Processing. 2000. Vol. 9. №7. P. 1158-1170.
4. *Islam A., Pearlman W.A.* // ISO/IEC/JTC1/SC29. 1998. №873. P. 312-326.
5. *Taubman D.S., Marcellin M.W.* // JPEG2000: Image Compression, Fundamentals, Standards, and Practice. Berlin, 2002.
6. *Борискевич А.А., Цветков В.Ю.* // Докл. НАН Беларуси. 2009. Т. 53. №3. С. 38-48.

УДК 621.391

## СЖАТИЕ МУЛЬТИСПЕКТРАЛЬНЫХ ИЗОБРАЖЕНИЙ НА ОСНОВЕ ПРОГРЕССИВНОГО КОДИРОВАНИЯ ДЛИН СЕРИЙ

Б.Дж. САДИК, М.Н. БОБОВ, В.Ю. ЦВЕТКОВ,  
А.Э.К. АЛЬ-БАЙАТИ, Х.М.А. АБДУЛХУССЕЙН

*Белорусский государственный университет информатики и радиоэлектроники  
П. Бровка, 6, Минск, 220013, Беларусь*

*Поступила в редакцию 29 мая 2016*

Разработаны алгоритм и структура кодека сжатия мультиспектральных изображений, основанные на прогрессивном кодировании длин серий канальных разностей. Показана эффективность предложенных решений по сравнению с канальным кодированием длин серий.

*Ключевые слова:* мультиспектральное изображение, сжатие изображений, кодирование длин серий.

### Введение

Передача и хранение мультиспектральных изображений (МСИ) в системах дистанционного зондирования Земли (ДЗЗ) является серьезной проблемой из-за высокой скорости их формирования (около 40 Гбит/с). Решение данной проблемы требует согласования скорости формирования МСИ не только с пропускной способностью радиоканала передачи (ограничена в настоящее время примерно 10 Гбит/с) и емкостью памяти (10 минут записи МСИ требуют около 3 Тбайт памяти), но также и со скоростью кодирования. Самым быстрым алгоритмом кодирования является RLE (Run-Length Encoding) [1]. Его основной недостаток состоит в относительно низком коэффициенте сжатия изображений по сравнению с другими алгоритмами [2-4]. Это связано с учетом только локальной пространственной корреляции кодируемых значений, которая в изображениях ДЗЗ относительно слабая из-за сильной дисперсии значений младших разрядов пикселей. Улучшить данную ситуацию можно двумя способами: а) за счет кодирования разностей спектральных каналов МСИ; б) за счет прогрессивного кодирования битовых плоскостей.

Целью работы является разработка алгоритма и структуры кодека сжатия МСИ на основе прогрессивного кодирования длин серий.

### Алгоритм сжатия МСИ на основе прогрессивного кодирования длин серий канальных разностей

Предлагается алгоритм сжатия МСИ на основе прогрессивного кодирования длин серий канальных разностей.

Алгоритм кодирования состоит из следующих шагов.

Шаг 1. Прогрессивное кодирование опорного спектрального канала МСИ на основе RLE.

1.1. Кодирование битовых плоскостей опорного спектрального канала МСИ с помощью алгоритма RLE. Для каждой битовой плоскости опорного спектрального канала формируется RLE-код, состоящий из значения (0/1) первого пикселя плоскости и длин чередующихся серий одинаковых значений остальных пикселей.

1.2. Вычисление коэффициента сжатия для каждой битовой плоскости опорного спектрального канала МСИ.

1.3. Формирование компактного описания битовых плоскостей опорного спектрального канала МСИ, начиная от старшей плоскости к младшей. Если коэффициент сжатия битовой плоскости больше единицы, то в компактное описание включается единичный флаг компрессии и RLE-код битовой плоскости. Если коэффициент сжатия битовой плоскости меньше единицы, то в компактное описание включается нулевой флаг компрессии и сама битовая плоскость.

Шаг 2. Вычисление разностей соседних спектральных каналов МСИ, начиная с опорного спектрального канала. Для каждой пары соседних спектральных каналов МСИ формируется разностный спектральный канал в результате их попиксельного вычитания.

Шаг 3. Прогрессивное кодирование разностных спектральных каналов.

3.1. Кодирование битовых плоскостей разностных спектральных каналов с помощью алгоритма RLE. Для каждой битовой плоскости (включая знаковую битовую плоскость) каждого разностного спектрального канала, начиная с разности опорного и соседнего с ним спектральных каналов, формируется RLE-код, состоящий из значения (0/1) первого пикселя плоскости и длин чередующихся серий одинаковых значений остальных пикселей.

3.2. Вычисление коэффициента сжатия для каждой битовой плоскости каждого разностного спектрального канала.

3.3. Формирование компактного описания битовых плоскостей для каждого разностного спектрального канала, начиная от старшей (знаковой) плоскости к младшей, начиная от разности опорного и соседнего с ним спектральных каналов. Если коэффициент сжатия битовой плоскости больше единицы, то в компактное описание включается единичный флаг компрессии и RLE-код битовой плоскости. Если коэффициент сжатия битовой плоскости меньше единицы, то в компактное описание включается нулевой флаг компрессии и сама битовая плоскость.

Алгоритм декодирования состоит из следующих шагов (предполагается, что декодеру известно число, размеры по вертикали и горизонтали, разрядность, порядок передачи спектральных каналов).

Шаг 1. Прогрессивное декодирование опорного спектрального канала МСИ. Декодирование компактного описания битовых плоскостей опорного спектрального канала МСИ осуществляется, начиная от старшей плоскости к младшей. Сначала проверяется значение флага компрессии битовой плоскости. Если он равен единице, то битовая плоскость восстанавливается в результате декодирования ее RLE-кода. Если флаг компрессии битовой плоскости равен нулю, то битовая плоскость восстанавливается без декодирования непосредственно по компактному описанию. В результате формируется восстановленный опорный спектральный канал МСИ.

Шаг 2. Прогрессивное декодирование разностных спектральных каналов. Декодирование компактного описания битовых плоскостей каждого разностного спектрального канала осуществляется, начиная от старшей (знаковой) плоскости к младшей и от разности опорного и соседнего с ним спектральных каналов. Сначала проверяется значение флага компрессии битовой плоскости. Если он равен единице, то битовая плоскость восстанавливается в результате декодирования ее RLE-кода. Если флаг компрессии битовой плоскости равен нулю, то битовая плоскость восстанавливается без декодирования непосредственно по компактному описанию. В результате формируются восстановленные разности соседних спектральных каналов.

Шаг 3. Восстановление значений спектральных каналов на основе разностей соседних спектральных каналов МСИ, начиная с соседнего с опорным (уже восстановленным) спектрального канала. Каждая разность соседних спектральных каналов вычитается от соответствующего восстановленного спектрального канала. В результате восстанавливаются значения пикселей всех спектральных каналов МСИ.

### **Оценка эффективности алгоритмов сжатия МСИ на основе кодирования длин серий**

Для оценки эффективности алгоритмов сжатия МСИ на основе кодирования длин серий использованы 64-х каналные 16-ти разрядные МСИ размером 2048x614 пикселей (исходный размер каждого канала МСИ составляет 2514944 байт, размер всего МСИ составляет 160956416 байт).

В табл. 1 приведены коэффициенты сжатия битовых плоскостей опорного спектрального канала МСИ (размер каждой битовой плоскости составляет 157184 байт) на основе прогрессивного (поразрядного) поканального кодирования длин серий. Из табл. 1 следует, что алгоритм RLE обеспечивает сжатие только 6-ти старших битовых плоскостей. Младшие 10 битовых плоскостей должны входить в компактное описание без сжатия. Объем компактного описания опорного спектрального канала составляет при этом 1842576 байт, а коэффициент сжатия – 1,36 раза.

Таблица 1. Коэффициенты сжатия опорного спектрального канала МСИ при прогрессивном поканальном кодировании длин серий

Битовые плоскости опорного (24-го) спектрального канала	Размер битовых плоскостей после кодирования, байт	Коэффициент сжатия
16	6	26197,3
16_15	6	26197,3
16_14	6	26197,3
16_13	2812	55,9
16_12	127979	1,2
16_11	139927	1,1
16_10	212099	0,7
16_9	342175	0,5
16_8	442095	0,4
16_7	374402	0,4
16_6	391120	0,4
16_5	392893	0,4
16_4	392110	0,4
16_3	392750	0,4
16_2	385835	0,4
16_1	392010	0,4

В табл. 2 приведены коэффициенты сжатия битовых плоскостей разностного спектрального канала (для 24 и 25 спектральных каналов, размер каждой битовой плоскости составляет 157184 байт, старшая битовая плоскость – знаковая) на основе прогрессивного кодирования длин серий канальных разностей.

Таблица 2. Коэффициенты сжатия разностного спектрального канала при прогрессивном кодировании длин серий канальных разностей

Битовые плоскости 24-го спектрального канала	Размер битовых плоскостей после кодирования, байт	Коэффициент сжатия
16	1832	85,8
16_15	6	26197,3
16_14	6	26197,3
16_13	6	26197,3
16_12	6	26197,3
16_11	6	26197,3
16_10	6	26197,3
16_9	6	26197,3
16_8	395	397,9
16_7	81432	1,9
16_6	381171	0,4
16_5	434428	0,4
16_4	392198	0,4
16_3	392974	0,4
16_2	392076	0,4
16_1	393035	0,4

Из табл. 2 следует, что алгоритм RLE обеспечивает сжатие 10-ти старших битовых плоскостей. Младшие 6 битовых плоскостей должны входить в компактное описание без сжатия. Это значительно лучше по сравнению с прогрессивным поканальным кодированием длин серий

(табл. 1). Объем компактного описания разностного спектрального канала составляет при этом 1026805 байт, а коэффициент сжатия – 2,45 раза. Это в 1,8 раза выше по сравнению с прогрессивным канальным кодированием длин серий.

### Синтез структур кодеров сжатия МСИ на основе прогрессивного кодирования длин серий канальных разностей

Предложенный алгоритм может быть реализован с помощью схем двухуровневого последовательного или блочного прогрессивного кодирования канальных разностей.

На рис. 1 представлена структура последовательного двухуровневого прогрессивного кодера канальных разностей. В данном кодере осуществляется независимое кодирование опорного спектрального канала (например, 1-го), вычисление поразрядных канальных разностей для соседних спектральных каналов, выделение группы старших битовых плоскостей (плоскости 16-7 для рассматриваемых типов МСИ) и кодирование этих плоскостей с помощью алгоритма RLE, исключение из процесса кодирования длин серий младших битовых плоскостей полученных разностей (плоскости 6-1 для рассматриваемых типов МСИ). Достоинством данного кодера является получение наибольших коэффициентов сжатия МСИ за счет использования корреляции соседних спектральных каналов. Основные недостатки: отсутствие возможности произвольного доступа к спектральным каналам (для декодирования любого канала необходимо декодировать все предыдущие каналы); размножение ошибки (в случае искажения любого бита при передаче RLE-кода какого-либо спектрального канала теряется возможность восстановления всех последующих спектральных каналов).

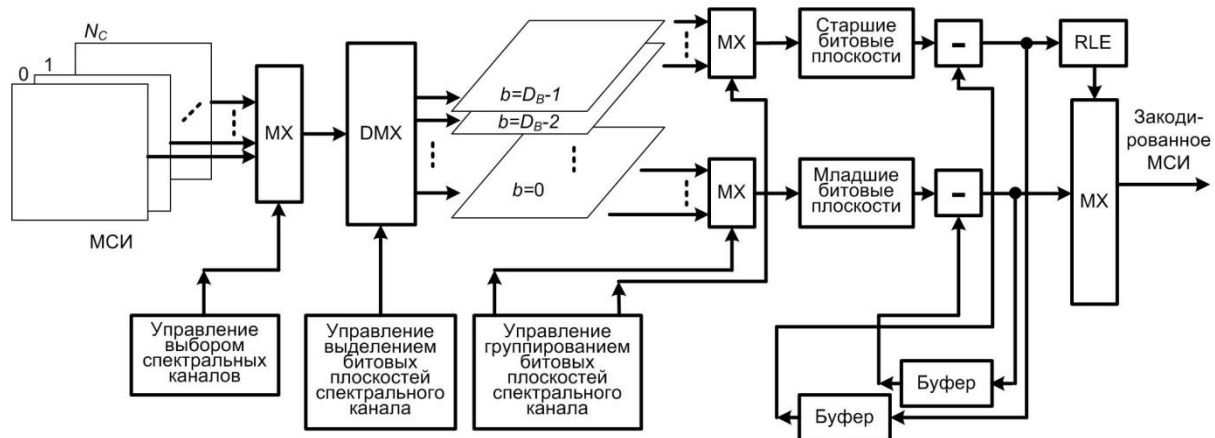


Рис. 1. Структурная схема последовательного двухуровневого прогрессивного кодера канальных разностей

На рис. 2 представлена структура блочного двухуровневого прогрессивного кодера канальных разностей. В данном кодере осуществляется независимое кодирование опорного спектрального канала (например, 1-го), вычисление поразрядных канальных разностей для некоторого числа спектральных каналов, отстоящих друг от друга на заданный интервал (например, 10), формирование блоков спектральных каналов с выбранным интервалом (равным 10-ти в рассматриваемом примере), вычисление поразрядных канальных разностей для спектральных каналов в пределах каждого блока относительно первого спектрального канала блока, выделение в каждом блоке группы старших битовых плоскостей (плоскости 16-7 для рассматриваемых типов МСИ) и кодирование этих плоскостей с помощью алгоритма RLE, исключение в каждом блоке из процесса кодирования младших битовых плоскостей полученных разностей (плоскости 6-1 для рассматриваемых типов МСИ). Достоинствами данного кодера является уменьшение времени произвольного доступа за счет последовательного декодирования только в пределах блока, размножение ошибки в пределах блока. Основной недостаток: уменьшение коэффициента сжатия МСИ из-за использования корреляции не соседних, а удаленных друг от друга спектральных каналов при формировании блоков.

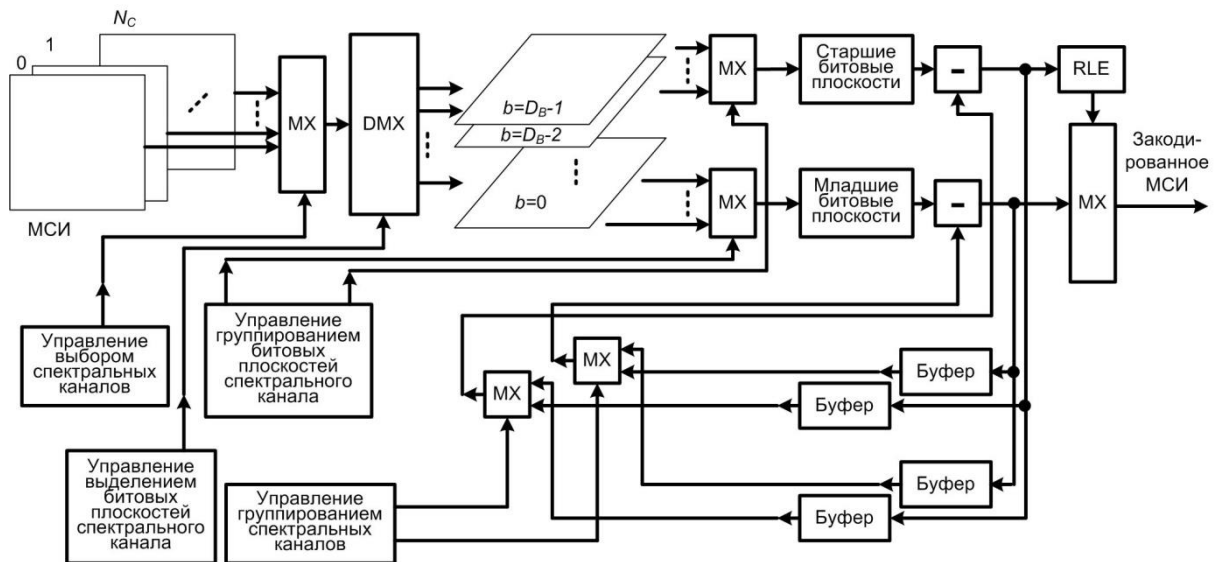


Рис. 2. Структурная схема блочного двухуровневого прогрессивного кодера каналов разностей

### Заключение

Разработаны блок-схемы алгоритмов и структурные схемы компрессии и декомпрессии мультиспектральных изображений на основе кодирования длин серий каналов разностей (алгоритм RLE). Произведена оценка эффективности прогрессивного кодирования каналов и каналов разностей мультиспектральных изображений с помощью алгоритма RLE. Установлено, что коэффициент сжатия разностного спектрального канала составляет 2,45 раза, что в 1,8 раза выше по сравнению с прогрессивным канальным кодированием длин серий.

## COMPRESSION OF THE MULTISPECTRAL IMAGES BASED ON THE PROGRESSIVE RUN LENGTH ENCODING

B.J. SADIQ, M.N. BOBOV, V.YU. TSVIATKOU,  
A.E.K. AL-BAIATI, H.M.A. ABDULHUSSEIN

### Abstract

The algorithm and the structure of the codec compression of multispectral images based on the progressive run length encoding of the channel differences are proposed. The efficiency of the proposed solutions compared to channel coding of run lengths is shown.

*Keywords:* multispectral image, image compression, run length encoding.

### Список литературы

1. Golomb S.W. // IEEE Transactions on Information Theory. 1966. P. 399-401.
2. Pennebaker W.B. JPEG Still Image Compression Standard. New York, 1993.
3. Ebrahimi T. // Proc. of the SPIE. San Diego, July-August 2000. Vol. 4115. P. 446-454.
4. Ватолин Д., Ратушняк А., Смирнов М. и др. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. М., 2003.



УДК 621.391.26

## АЛГЕБРО-ГЕОМЕТРИЧЕСКИЕ КОДЫ ПРОЕКТИВНЫХ И АФФИННЫХ ПРОСТРАНСТВ

В.В. ПАНЬКОВА, С.Б. САЛОМАТИН

Белорусский государственный университет информатики и радиоэлектроники  
П. Бровка, 6, Минск, 220013, Беларусь

Поступила в редакцию 20 апреля 2016

Рассмотрены методы построения алгебро-геометрических кодов над конечными полями с нахождением всех аффинных точек и точек в бесконечности, которые определяются в результате применения стремящегося к нулю неприводимого сглаживания аффинной кривой. Проведен сравнительный анализ параметров алгебро-геометрических кодов различного вида.

*Ключевые слова:* алгебро-геометрический код, проективные и аффинные кривые.

### Введение

Конструкция алгебро-геометрических (АГ) кодов требует множества точек, которые удовлетворяют неприводимым аффинным кривым и множеству рациональных функций на этих кривых. Проективное пространство является  $(n+1)$ -мерным, элементы в этом пространстве определяются над некоторым конечным полем. Такие элементы называются проективными точками и определяются как [1]  $(c_1, c_2, c_3, \dots, c_n, c_{n+1})$ , где  $c_i, i = 0, 1, \dots, n, n+1$  – элементы конечного поля.

Аффинное пространство является  $n$ -мерным, а элементы, определяемые в пространстве над некоторым конечным полем, называются аффинными точками и записываются в виде  $(c_1, c_2, \dots, c_n, 1)$ .

Если точки задаются в форме  $(c_1, c_2, \dots, c_n, 0)$ , то пространство называется гиперплоскостью в бесконечности [1-4]. Все точки, расположенные в гиперплоскости в бесконечности, называются точками в бесконечности. Для конструкции АГ-кодов представляет интерес нахождение всех аффинных точек и точек в бесконечности, которые определяются в результате применения стремящегося к нулю неприводимого сглаживания аффинной кривой.

### Проективные и аффинные кривые

Проективная кривая является  $(n+1)$ -мерной кривой, определенной проективными точками, что позволяет получать  $n$ -мерные аффинные кривые в  $(n+1)$  различных координатных системах.

Для 3-мерной проективной кривой  $C(x, y, z)$  существуют три аффинные кривые  $C(x, y, 1)$ ,  $C(x, 1, z)$  и  $C(1, y, z)$ .

Алгебро-геометрические коды используют кривые, полученные путем неприводимого аффинного сглаживания. Например, кривая  $C(x, y) = x^2 + y^2$  не является неприводимой над полем  $GF(2)$ , поскольку

$$(x + y)(x + y) = (x^2 + xy + xy + y^2) = |xy + xy = 0| = x^2 + y^2.$$

С другой стороны кривая  $C(x, y) = x^3 + y^3$  неприводима над  $GF(2)$ , но не является таковой над полем  $GF(3)$ , поскольку

$$(x + y)(x + y)(x + y) = (x^2 + 2xy + y^2)(x + y) = x^3 + x^2y + 2x^2y + 2xy^2 + xy^2 + y^3 = \\ = x^3 + 3x^2y + 3xy^2 + y^3 \equiv x^3 + y^3 \pmod{3}.$$

Точка на кривой называется неособой, если все частные производные кривой не сходятся к этой точке. Если все точки кривой являются неособыми, то кривая носит название неособой или гладкой [1, 2]. Важным является класс кривых Эрмита, определенных над квадратичным конечным полем  $GF(w^2)$ . Проективная кривая Эрмита определяется как

$$C(x, y, z) = x^{w+1} + y^w z + yz^w. \quad (1)$$

Кривая Эрмита является гладкой. Действительно, три производные имеют вид

$$\frac{\partial C(x, y, z)}{\partial x} = (w+1)x^w = x^w, \quad \frac{\partial C(x, y, z)}{\partial y} = wy^{w-1}z + z^w = z^w,$$

$$\frac{\partial C(x, y, z)}{\partial z} = y^w + wyz^{w-1} = y^w.$$

Признаком особой точки служат точки, имеющие производные, стремящиеся к  $(0, 0, 0)$ . Но в рассматриваемом случае такие точки в проективном пространстве отсутствуют, следовательно, точки являются неособыми, а кривая – гладкой.

### Нахождение точек на аффинной кривой

Проективные точки, которые удовлетворяют условию для проективной кривой  $C(x, y, z) = 0$ , представляются в виде формы  $(\alpha, \beta, \delta)$ , где  $\alpha, \beta, \delta$  являются элементами конечного поля. Для построения АГ-кода используются только аффинные точки вида  $(\alpha, \beta, 1)$ ,  $z = 1$ , а точки в бесконечности определяются формой  $(\alpha, \beta, 0)$ ,  $z = 0$ . Для нахождения всех аффинных точек проективная кривая отображается в кривые аффинных компонент. Для каждой аффинной кривой находят точки  $(\alpha, \beta, 1)$  и  $(\alpha, \beta, 0)$ .

Аффинная форма кривой Эрмита (1) в системе координат вида  $(x - y)$  с  $z = 1$ , имеет вид

$$C(x, y, z) = x^{w+1} + y^w + y. \quad (2)$$

В  $(x - z)$  системе координат,  $y = 1$ , кривая определяется как

$$C(x, 1, z) = x^{w+1} + z + z^w.$$

В  $(y - z)$  системе координат,  $x = 1$ , имеем

$$C(1, y, z) = 1 + y^w z + yz^w.$$

Как известно, кривая Эрмита имеет  $w^3 + 1$  точек. Для поля  $GF(9)$  получаем  $2^3 + 1 = 8 + 1 = 9$  точек и кривые вида:

$$C(x, y, z) = x^3 + y^2 z + yz^2.$$

Аффинная кривая в  $(x - y)$  системе имеет вид  $C(x, y) = x^3 + y^2 + y$ .

Аффинные точки, удовлетворяющие условию  $C(x, y, 1) = 0$ , находятся подстановкой в последнее выражение всех возможных значений переменных  $x$  и  $y$  при  $(z = 1)$  в поле  $GF(2^2)$ , где  $\alpha^2 = \alpha + 1$ . В результате получаем:

$$C(0,0,1) = 0^3 + 0^2 + 0 = 0, \quad C(0,1,1) = 0^3 + 1^2 + 1 = 0,$$

$$C(0,\alpha,1) = 0^3 + \alpha^2 + \alpha = \alpha(1 + \alpha) = \alpha^3 = 1 \neq 0,$$

$$C(0,\alpha^2,1) = 0^3 + \alpha^4 + \alpha^2 = \alpha^2(1 + \alpha^2) = \alpha^2\alpha = 1 \neq 0,$$

$$C(1,0,1) = 1^3 + 0^2 + 0 = 1 \neq 0, \quad C(1,1,1) = 1^3 + 1^2 + 1 = 1 \neq 0,$$

$$C(1,\alpha,1) = 1^3 + \alpha^2 + \alpha = 1 + 1 = 0,$$

$$C(1,\alpha^2,1) = 1^3 + \alpha^4 + \alpha^2 = 1 + \alpha + \alpha^2 = 1 + 1 = 0,$$

$$C(\alpha,0,1) = \alpha^3 + 0^2 + 0 = \alpha^3 = 1 \neq 0, \quad C(\alpha,1,1) = \alpha^3 + 1^2 + 1 = \alpha^3 = 1 \neq 0,$$

$$C(\alpha,\alpha,1) = \alpha^3 + \alpha^2 + \alpha = 1 + 1 = 0,$$

$$C(\alpha,\alpha^2,1) = \alpha^3 + \alpha^4 + \alpha^2 = \alpha^2(1 + \alpha + \alpha^2) = 0,$$

$$C(\alpha^2,0,1) = \alpha^6 + 0^2 + 0 = 1 \neq 0, \quad C(\alpha^2,1,1) = \alpha^6 + 1^2 + 1 = 1 \neq 0,$$

$$C(\alpha^2,\alpha,1) = \alpha^6 + \alpha^2 + \alpha = \alpha^6 + 1 = 1 + 1 = 0,$$

$$C(\alpha^2,\alpha^2,1) = \alpha^6 + \alpha^4 + \alpha^2 = \alpha^6 + \alpha + \alpha^2 = 1 + 1 = 0.$$

Аффинные точки, удовлетворяющие условию  $C(x, y, 1) = 0$ , приведены в табл. 1.

Таблица 1. Восемь проективных точек кривой Эрмита в  $(x - y)$  системе

$P_1 = (0, 0, 1)$	$P_2 = (0, 1, 1)$	$P_3 = (1, \alpha, 1)$	$P_4 = (1, \alpha^2, 1)$
$P_5 = (\alpha, \alpha, 1)$	$P_6 = (\alpha, \alpha^2, 1)$	$P_7 = (\alpha^2, \alpha, 1)$	$P_8 = (\alpha^2, \alpha^2, 1)$

Аналогичным образом можно найти точки для двух других кривых

$$C(x, 1, z) = x^3 + z + z^2 \text{ и } C(1, y, z) = 1 + y^2z + yz^2.$$

Восемь проективных точек, удовлетворяющих условию  $C(x, 1, z) = 0$ , приведены в табл. 2. Точка  $P_2$  является аффинной, она имеет форму  $(x, y, 1)$  и также присутствует в табл. 2. Совпадают точка в бесконечности и точка  $P_1 = (0, 1, 0)$ .

Таблица 2. Восемь проективных точек кривой Эрмита в  $(x - z)$  системе

$P_1 = (0, 1, 0)$	$P_2 = (0, 1, 1)$	$P_3 = (1, 1, \alpha)$	$P_4 = (1, 1, \alpha^2)$
$P_5 = (1, \alpha, 1)$	$P_6 = (1, \alpha, \alpha^2)$	$P_7 = (1, \alpha^2, 1)$	$P_8 = (1, \alpha^2, \alpha)$

Четыре проективные точки, удовлетворяющие условию  $C(1, y, z) = 0$ , приведены в табл. 3. В этом случае только точки  $P_1$  и  $P_3$  являются аффинными, так как они имеют форму  $(x, y, 1)$  и также присутствуют в табл. 1.

Таблица 3. Четыре проективные точки кривой Эрмита в  $(y - z)$  системе

$P_1 = (1, \alpha, 1)$	$P_2 = (0, \alpha, \alpha^2)$	$P_3 = (1, \alpha^2, 1)$	$P_4 = (1, \alpha^2, \alpha)$
------------------------	-------------------------------	--------------------------	-------------------------------

Следовательно, проективная кривая Эрмита имеет восемь аффинных точек и одну точку в бесконечности  $Q = (0, 1, 0)$ .

Коды, получаемые с помощью кривых с одной точкой в бесконечности, называются АГ-кодами с одной точкой или кодами Гоппы [3]. Примерами кривых с одной точкой в бесконечности служат эллиптические и гиперэллиптические кривые.

### Граничные соотношения

Верхняя граница, определяющая количество точек  $N$ , включая точку в бесконечности для кривых над полем  $GF(q)$ , носит название границы Хассе-Вейля и определяется как [1]:

$$|N| \leq (m-1)(m-2)\sqrt{q} + 1 + q,$$

где  $m$  степень кривой.

Например, для рассмотренного выше примера кривой Эрмита степени  $m=3$  и  $q=4$ , верхняя граница равна  $|N| \leq (3-1)(3-2)\sqrt{4} + 1 + 4$ ,  $|N| \leq 9$ .

Максимальное количество точек кривой степени  $m=3$ , определенной над  $GF(2^2)$ , равно 9. Для кривой Эрмита из примера было получено восемь аффинных точек и одна точка в бесконечности, что в сумме дает девять точек. Таким образом, кривая Эрмита лежит на границе Хассе-Вейля и является *максимальной* кривой. Такие кривые позволяют строить длинные коды.

Другой тип максимальных кривых дают эллиптические кривые над конечными полями. Например, эллиптическая кривая [2, 4]

$$C(x, y) = x^3 + x^2 + x + y^2 + y + 1$$

является максимальной кривой над полем  $GF(2^4)$ . Степень кривой  $m=3$ , число ее точек – 25.

Кривая Эрмита над полем  $GF(2^4)$  вида  $C(x, y) = x^5 + y^4 + y$  имеет степень  $m=5$  и определяет 65 точек.

Коды, построенные над эллиптическими кривыми значительно короче, чем коды, построенные с помощью кривых Эрмита. Но коды, построенные по эллиптическим кривым, являются более длинными по сравнению с кодами Рида-Соломона, конструкция которых определяется аффинной линией  $y=0$  и имеет степень  $m=1$ . Для конечного поля  $GF(q)$  граница Хассе-Вейля упрощается и имеет вид  $|N| \leq q+1$ . Так, для  $GF(2^4)$  наиболее длинный код Рида-Соломона имеет  $16+1=17$  точек, что меньше, чем у кода, построенного по эллиптической кривой.

### Рациональные функции на кривых

Конструкции порождающих матриц АГ-кодов основываются на рациональных функциях соответствующих кривых. Каждая рациональная функция оценивается в каждой из  $n$  аффинных точек, формируя строку порождающей матрицы, где  $n$  – длина кода. Рациональную функцию  $f(x, y, z)$  можно рассматривать как частное от деления двух функций  $g(x, y, z)$  и  $h(x, y, z)$ ,

имеющих одинаковые степени  $f(x, y, z) = \frac{g(x, y, z)}{h(x, y, z)}$ . Разные формы представления функции

позволяют более полно понять поведение кривых.

Например, функция  $f(x, y) = \frac{x}{y+1}$ , определенная над кубической кривой

$C(x, y) = x^3 + y^3 + 1 = 0$  и полем  $GF(2^2)$ , может быть записана в виде

$$x^3 = y^3 + 1, \quad x = \frac{y^3 + 1}{x^2} = \frac{(y+1)((y^2 + y + 1))}{x^2}, \quad \frac{x}{y+1} = \frac{y^2 + y + 1}{x^2}.$$

Кубическая кривая имеет 6 точек:  $P_1 = (0,1)$ ,  $P_2 = (0,\alpha)$ ,  $P_3 = (0,\alpha^2)$ ,  $P_4 = (1,0)$ ,  $P_5 = (\alpha,0)$ ,  $P_6 = (\alpha^2,0)$ . В этом случае функция  $f(x,y) = \frac{x}{y+1}$  определена для всех точек на кубической

кривой, за исключением точки  $P_1 = (0,1)$ , так как  $f(0,1) = \frac{0}{1+1} = \frac{0}{0}$ .

Если функция не определяется на кривой, то она имеет в исследуемой точке ноль порядка 1 и полюс порядка 1.

Функция  $f(x,y) = \frac{y^2 + y + 1}{x^2}$  в точке  $(0,1)$  дает результат  $f(0,1) = \frac{1^2 + 1 + 1}{0^2} = \frac{1}{0}$ , не имеет нулей, но имеет полюс порядка 2.

Порядок функции обозначается как  $v(f(x,y,z))$  и определяется путем суммирования порядков нулей и полюсов. Конструкция АГ-кодов на основе рациональных функций должна иметь полюс в точке бесконечности, но не должна иметь полюсов в любых других аффинных точках.

Последовательность рациональных функций на проективной линии имеет полюс только в точке бесконечности. Поэтому существует  $q$  аффинных точек и одна точка в бесконечности. Последовательность рациональных функций, имеющих полюс в точке бесконечности  $Q = (1,0,0)$ , может быть определена следующим образом. Последовательность рациональных

функций вида  $\left\{ \frac{x^i}{y^i} \right\}$ ,  $i > 0$  имеет полюс порядка  $i$  в  $Q$ , но не имеет полюсов в других аффинных точках. Рассматриваемый случай показывает, что число точек на проективной линии не может превышать кардинальность поля линии.

Коды, конструируемые из линии  $y = 0$ , известны как коды Рида-Соломона и представляют собой простейший вид алгебро-геометрических кодов. Малое количество точек на проективной линии показывает, почему коды Рида-Соломона имеют короткие длины, не превышающие размеры конечных полей.

### Рациональные функции на кривой Эрмита

Последовательность рациональных функций на проективной кривой Эрмита может быть сформирована для элементов  $\frac{x}{z}$  и  $\frac{y}{z}$ . Элемент  $\frac{x}{z}$  можно переписать в виде выражений

$$x^{w+1} = y^w z + yz^w, \quad \frac{x}{z} = \frac{y^w + yz^{w-1}}{x^w},$$

которые имеют порядок  $w$  в точке бесконечности  $Q = (0,1,0)$ .

Аналогично, для элемента  $\frac{y}{z}$  можно записать

$$x^{w+1} = y^w z + yz^w, \quad \frac{y^w z + yz^w}{x^{w+1}} = 1, \quad \frac{1}{z} = \frac{y^w + yz^{w-1}}{x^{w+1}}, \quad \frac{y}{z} = \frac{y^{w+1} + y^2 z^{w-1}}{x^{w+1}}.$$

Элемент имеет порядок  $w+1$  в точке бесконечности  $Q = (0,1,0)$ . Поэтому кривая Эрмита над полем  $GF(2^2)$  имеет  $r = 2$ ,  $v\left(\frac{x}{z}\right) = 2$ ,  $v\left(\frac{y}{z}\right) = 3$ .

Другие рациональные функции на кривой формируются путем комбинации произведения различных степеней  $\frac{x}{z}$  и  $\frac{y}{z}$ , и сложения их порядков. Например, произведение  $\frac{x}{z} \cdot \frac{x}{z} = \frac{x^2}{z^2}$  име-

ет порядок равный  $2 + 2 = 4$ . Произведение  $\frac{x}{z} \cdot \frac{y}{z} = \frac{xy}{z^2}$  имеет порядок  $2 + 3 = 5$ , а произведение

$\frac{y}{z} \cdot \frac{y}{z} = \frac{y^2}{z^2}$  имеет порядок  $3 + 3 = 6$ . В общем случае порядок рациональной функции  $\frac{x^i y^j}{z^{i+j}}$  на кривой Эрмита определяется выражением

$$v\left(\frac{x^i y^j}{z^{i+j}}\right) = iw + j(w + 1).$$

Такая последовательность рациональных функций обозначается как  $L(G)$ , где  $G$  является дивизором кривой. Дивизор кривой определяется как целое значение каждой точки кривой. АГ-код имеет два дивизора  $D$  и  $G$ . Дивизор  $D$  определяется значением  $D(P) = 1$  для каждой аффинной точки и вычисляется через сумму всех аффинных точек

$$D = \sum_{i=1}^n D(P_i)P_i = \sum_{i=1}^n P_i.$$

Сумма  $\sum_{i=1}^n D(P_i)$  называется степенью дивизора  $D$ ,  $d(D)$ .

Аналогично дивизор  $G$  определяется как целое значение  $G(Q)$  для каждой точки в бесконечности  $Q$  и вычисляется как сумма всех точек в бесконечности. Для кривых с одной точкой в бесконечности, дивизор  $G$  оценивается через произведение точки в бесконечности и степени  $G$ ,  $d(G)$ :

$$G = \sum_i G(Q_i)Q_i = d(G)Q.$$

Пространство  $L(G)$  содержит рациональные функции порядка  $d(G)$ . Для примера, рассмотренного выше, имеем, если  $G = 7Q$  и  $d(G) = 7$ , тогда

$$L(7Q) = \left\{ 1, \frac{x}{z}, \frac{y}{z}, \frac{x^2}{z^2}, \frac{xy}{z^2}, \frac{y^2}{z^2}, \frac{x^3}{z^3}, \frac{x^2 y}{z^3} \right\}.$$

Заметим, что порядки  $\frac{x^3}{z^3}$  и  $\frac{y^2}{z^2}$  совпадают и равны 6. В этом случае все функции, степени  $x$  в которых больше или равны 3, могут быть исключены из рассмотрения. Также можно исключить функции со степенями  $y$  большими 1. Поэтому, можно говорить только о семи функциях в  $L(G)$ :

$$L(7Q) = \left\{ 1, \frac{x}{z}, \frac{y}{z}, \frac{x^2}{z^2}, \frac{xy}{z^2}, \frac{y^2}{z^2}, \frac{x^2 y}{z^3} \right\}.$$

Теорема Римана-Роха используется для вычисления количества рациональных функций в  $L(G)$  с учетом порядков и следовательно определяет размерность и минимальное расстояние кода. Число рациональных функций в  $L(G)$  называется размерностью  $G$ ,  $l(G)$ . Теорема утверждает, что существуют неотрицательные числа  $\gamma$ , для которых  $l(G) - d(G) = 1 - \gamma$ , при условии, что  $d(G) > 2\gamma - 2$ . Неотрицательное число  $\gamma$  называется *родом кривой* и определяется как

$$\gamma = \frac{(m-1)(m-2)}{2},$$

где  $m$  – степень кривой.

Род кривой играет важную роль в оценке размерности параметров кода.

Пример. Число рациональных функций кривой Эрмита нулевого порядка меньше или равно 21 с точкой в бесконечности  $Q = (0,1,0)$ . Действительно, рациональные функции  $\frac{x}{z}$  и  $\frac{y}{z}$  имеют порядки соответственно  $w = 4$  и  $w + 1 = 5$ .

Поэтому последовательность функций имеет вид

$$L(21Q) = \left\{ 1, \frac{x}{z}, \frac{y}{z}, \frac{x^2}{z^2}, \frac{xy}{z^2}, \frac{y^2}{z^2}, \frac{x^3}{z^3}, \frac{x^2y}{z^3}, \frac{xy^2}{z^3}, \frac{y^3}{z^3}, \frac{x^4}{z^4}, \frac{x^3y}{z^4}, \frac{x^2y^2}{z^4}, \frac{xy^3}{z^4}, \frac{y^4}{z^4}, \frac{x^4y}{z^5} \right\}.$$

Все функции, содержащие степени  $x$ , большие, чем 4, должны быть исключены из рассмотрения как функции, порядки которых дублируют существующие порядки. Так, в последовательности отсутствует функция  $\frac{x^5}{z^5}$ , порядок которой такой же как порядок  $v\left(\frac{x^4}{z^4}\right)$ . Множество порядков полученных функций имеет вид

$$\{1, 4, 5, 8, 9, 10, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21\}.$$

Соответственно получаем 16 функций в  $L(G)$ . Такое множество порядков известно, как множество *непустых* порядков. Порядки 0, 2, 3, 6, 7 и 11 отсутствуют в полученном множестве. Такие порядки называют *пустыми*. Число пустых порядков равно  $\gamma$ .

Степень кривой Эрмита равна  $m = 5$ , род кривой равен  $\gamma = 6$  и  $d(G) = 21$ . Далее, из того факта, что  $2\gamma - 2 = 10$  меньше, чем  $d(G)$ , следует возможность применения теоремы Римана-Роха  $l(G) = d(G) + 1 - \gamma = 21 + 1 - 6 = 16$ , что совпадает с числом функций в  $L(G)$ .

### Заключение

Рассмотрены методы построения алгебро-геометрических кодовых структур с использованием проективных и аффинных пространств. Сравнительный анализ показывает, что наибольшим разнообразием обладают коды, построенные на основе кривых Эрмита.

## ALGEBRAIC GEOMETRICAL CODES OVER PROJECTIVE AND AFFINE SPACE

V.V. PANKOVA, S.B. SALOMATIN

### Abstract

Methods of constructing of algebraic-geometric codes over finite fields with finding of all affine points and points at infinity, which are defined by the application of irreducible smoothing of an affine curve, which aspires to zero, are considered. The comparative analysis of the parameters of algebraic-geometric codes of various type is carried out.

*Keywords:* algebraic-geometric code, affine and projective curves.

### Список литературы

1. Carrasco R.A., Martin J. Non-binary error control coding for wireless communication and data storage. John Wiley & Sons, Ltd. 2008.
2. Lin S., Costello D.J. Error Control Coding. New Jersey, 2004.
3. Влэдуц С.Г., Ногин Д.Ю., Цфасман М.А. Алгебро-геометрические коды. Основные понятия. М., 2003.
4. Chaoping Xing, San Ling // IEEE Transactions on Information Theory. 2000. Vol 46(4). P. 1527-1532.

UDC 621.391

## EXPLORATION OF STABILITY FOR SEGMENTATION METHOD BASED ON WAVE REGION GROWING

O.M. ALMIAHI

*Belarusian State University of Informatics and Radio Electronics  
P. Brovki, 6, Minsk, 220013, Belarus*

*Submitted 17 May 2016*

A method of progressive segmentation of gray scale images based on wave quasi parallel region growing is proposed. In contrast to known methods of segmentation of the proposed method allows to divide the area with smoothing drops of brightness and adapt to the constraints time of segmentation.

*Keywords: image segmentation, region growing.*

### Introduction

The Image segmentation is widely applied in solving various tasks of video information processing. Image segmentation is defined as the process of partitioning the digital image into different sub regions of homogeneity. Each of the pixels in a region is similar with respect to some characteristic or computed property, such as color, intensity, or texture. The result of image segmentation is a collection of segments which combine to form the entire image [1].

The efficiency of image segmentation depends on standard conditions, such as the kind of feature information used for grouping, the credibility of feature extraction and merging of feature information [2, 3]. In some cases, the segmentation time may be limited. It is also possible the task of image segmentation with smoothing drops of brightness. The known segmentation methods based on the formation of regions using a watershed [4], quantization of an histogram [5], split and merge regions using Quadra-tree [6, 7], region growing [8, 9], are not effective under these conditions. Segmentation using the watershed does not provide excretion for the smooth drops of brightness of images. Segmentation based on quantization of an histogram does not provide an accurate separation of regions because of assigning the same segment number with the same brightness. Methods based on the separation and merging regions using Quadra-tree and based on growing regions [10, 11] can accurately image segmentation, but also does not allow to find the boundaries of the regions on the smooth drops of brightness, which results in a segmentation fault. In addition, all the considered methods do not provide the adaptation to a limitation on the time of segmentation. In this regard, is relevant task of developing a method of image segmentation, which takes into account these shortcomings.

The aim is to develop a method of image segmentation, which allows to separate the regions with smooth drops of brightness and adapt to the limitations of time of segmentation.

### Method of image segmentation based on the wave region growing

For the progressive segmentation of halftone images, a method based on the wave quasi-parallel growing regions is proposed. The essence of the method is a quasi-parallel region growing around the chosen initial points of growth that provides automatic separation of the regions with smooth drop of brightness, that the known methods segmenting with errors.

Algorithm of Image segmentation based on the wave region growing consists of the following steps.



Step 1. Initialization. In this step is carried out buffering of image  $I = \parallel i(y, x) \parallel_{(y=0, Y-1, x=0, X-1)}$  which have size  $YX$  pixels and threshold detection  $\Delta_G = f(I)$ , determines the conditions of accession of the pixel to the grown of the region. To select the threshold  $\Delta_G$  can be used the histogram of first derivatives of pixel values in rows and columns. Forming a zero matrix of segmentation  $SM = \parallel sm(y, x) \parallel_{(y=0, Y-1, x=0, X-1)}$ , elements which determines as following equation:

$$sm(y, x) \Leftarrow 0, \quad (1)$$

where  $y = \overline{0, Y-1}$ ,  $x = \overline{0, X-1}$ .

The counter of segmented values  $C_{SM}$  is assigned a value 0:  $C_{SM} \Leftarrow 0$ . Initialized zero stack of collision  $CM_Y = \parallel cm_Y(z) \parallel_{(z=0, Z_{CM}-1)}$  and  $CM_X = \parallel cm_X(z) \parallel_{(z=0, Z_{CM}-1)}$ , where  $cm_Y(z) \Leftarrow 0$ ,  $cm_X(z) \Leftarrow 0$ , where  $z = \overline{0, Z_{CM}-1}$ ,  $Z_{CM}$  – maximum number of elements in the stack. Pointer stack of collision stopping on 0,  $z = 0$ .

Step 2. Start cycle of segmentation, choosing initial points for growth of regions from the stack. For example local maximum of histogram pixels values of image, which correspond to the zero values in the matrix of segmentation can be used. Let matrices  $BM_Y = \parallel bm_Y(k) \parallel_{(k=0, K_{BM}-1)}$ ,  $BM_X = \parallel bm_X(k) \parallel_{(k=0, K_{BM}-1)}$  contain coordinates of initial points of growth, where  $K_{BM}$  – number of initial points of growth.

Step 3. Initialization stacks  $\{GM_Y(k)\}_{(p=0, K_{BM}-1)}$ ,  $\{GM_X(k)\}_{(p=0, K_{BM}-1)}$  for growth of regions,  $GM_Y(k) = \parallel gm_Y(k, p(k)) \parallel_{(p(k)=0, P_{GM}(k)-1)}$ ,  $GM_X(k) = \parallel gm_X(k, p(k)) \parallel_{(p(k)=0, P_{GM}(k)-1)}$ .

The pointer of every stack of growth of region is set to 1:  $p(k) \Leftarrow 1$  where  $k = \overline{0, K_{BM}-1}$ . The counter  $C_{SM}$  of segmented value is assigned to the number of initial points of growth:  $C_{SM} \Leftarrow K_{BM}$ . Elements of matrix of segmentation, the coordinates which correspond to the coordinates of the initial points of growth are assigned numbers of segments (each element is assigned a number of previously unused):

$$sm(bm_Y(k), bm_X(k)) \Leftarrow (k+1), \quad (2)$$

where  $k = \overline{0, K_{BM}-1}$ .

Step 4. Initialization counter of cycles of busting regions are grown (set to zero),  $k \Leftarrow 0$ .

Step 5. Start cycle of busting grown of regions, from stack of the growth of the regions, the number which corresponds to the counter value of cycle busting of grown regions, are extracted coordinates of the current grown pixel, which indicates a value corresponding to the stack pointer. The value of the stack pointer decreases by one:

$$y_B \Leftarrow gm_Y(k, p(k)), \quad (3)$$

$$x_B \Leftarrow gm_X(k, p(k)), \quad (4)$$

$$p(k) \Leftarrow p(k) - 1. \quad (5)$$

Step 6. Initialization the counter of surrounding pixels  $l \Leftarrow 0$ .

Step 7. Start the analysis cycle of the surrounding pixels.

Step 8. Checking to zero the value of element of the matrix of segmentation, the coordinates which correspond to the found coordinates  $(y_A, x_A)$  of the surrounding pixel. If this value is not equal to zero ( $sm(y_A, x_A) \neq 0$ ), then go to the step 13, else ( $sm(y_A, x_A) = 0$ ) – go to the next step.

Step 9. The absolute value of the difference values of the current grown pixel  $i(y_B, x_B)$  and the current surrounding pixel  $i(y_A, x_A)$  is compared with a given threshold  $\Delta_G$ . If the absolute value of the difference is less than the threshold ( $|i(y_B, x_B) - i(y_A, x_A)| < \Delta_G$  the surrounding pixels must be joined to the region), then go to the next step, otherwise ( $|i(y_B, x_B) - i(y_A, x_A)| \geq \Delta_G$ ) go to step 16.

Step 10. Elements of the matrix of segmentation, which coordinates correspond to the coordinates  $(y_A, x_A)$  of the current surrounding pixel, assigned a value of element of segmentation matrix, which coordinates are correspond to the coordinates  $(y_B, x_B)$  of the current grown pixel

$$sm(y_A, x_A) \Leftarrow sm(y_B, x_B). \quad (6)$$

Step 11. Pointer the current stack of the growth region is incremented by one. In the current stack of the growth region are brought coordinates  $(y_A, x_A)$  of the current surrounding pixel:

$$gm_y(k, p(k)) \Leftarrow y_A, \quad (7)$$

$$gm_x(k, p(k)) \Leftarrow x_A, \quad (8)$$

$$p(k) \Leftarrow p(k) + 1. \quad (9)$$

Step 12. Value counter  $C_{SM}$  of segmented values is incremented by 1:  $C_{SM} \Leftarrow C_{SM} + 1$  – go to the step 16.

Step 13. Checking the collision. Compares the value of elements of segmentation, the coordinates which correspond to the coordinates  $(y_A, x_A)$  of the current surrounding pixel and  $(y_B, x_B)$  the current grown pixel. If these values are equal ( $sm(y_A, x_A) = sm(y_B, x_B)$ ), then go to the step 16, else ( $sm(y_A, x_A) \neq sm(y_B, x_B)$ ) – go to the next step.

Step 14. The absolute value of the difference the pixel values of the image, coordinates which correspond to the coordinates  $(y_B, x_B)$  of the current grown pixel and  $(y_A, x_A)$  the current surrounding pixel are compared with the given threshold  $\Delta_G$ . The absolute value of the difference of values of pixels in the image, the coordinates which correspond to the coordinates  $(bm_y(k), bm_x(k))$  and  $(bm_y(s), bm_x(s))$  of the initial points of growth for the considered segments  $k$  and  $s$ , are also compared with given threshold  $\Delta_G$ . If the absolute value of the difference less than the threshold ( $|i(y_B, x_B) - i(y_A, x_A)| < \Delta_G$ )  $\vee$  ( $|i(bm_y(k), bm_x(k)) - i(bm_y(s), bm_x(s))| < \Delta_G$ ) – surrounding pixel must be joined to the current region, but already joining to other region – having allocation of collision), then go to the next step, else – go to the step 16.

Step 15. The current element of the stack of collisions are assigned to coordinates of the current surrounding pixel and the current growing pixel, the stack pointer of collisions twice incremented by one:

$$cm_y(z) \Leftarrow y_B, cm_x(z) \Leftarrow x_B, z \Leftarrow z + 1, \quad (10)$$

$$cm_y(z) \Leftarrow y_A, cm_x(z) \Leftarrow x_B, z \Leftarrow z + 1. \quad (11)$$

Step 16. Ending cycle of analysis of the surrounding pixels. Counter surrounding pixels is incremented by 1:  $l \Leftarrow l + 1$ . Checking inequality the counter value of the surrounding pixels and the

number of neighboring pixels (8 pixels). If the counter of the surrounding pixels is less than 8 ( $l < 8$ ), then skip to step 7, else ( $l = 8$ ) – out of the cycle.

Step 17. Ending cycle busting growing of regions. The counter cycle busting of growing regions increased by one:  $k \leftarrow k + 1$ . Checking inequality the counter value of the cycle busting of growing regions and number of initial growth points. If the counter of cycle busting growing regions less than number  $K_{BM}$  of initial growth points. If the counter of cycle busting growing regions less than number of initial points of growth ( $k < K_{BM}$ ), then skip to step 5, ( $k \geq K_{BM}$ ) – out of the cycle.

Step 18. Ending the cycle of segmentation. If the counter  $C_{SM}$  of segmented values equal to number pixels of image ( $C_{SM} = YX$ ), then out of the cycle (formed an intermediate matrix of segmentation), else ( $C_{SM} \neq YX$ ) – go to step 2.

Step 19. Initialized resulting matrix  $SM_R = \left\| sm_R(y, x) \right\|_{(y=0, Y-1, x=0, X-1)}$  of segmentation. Values matrix of segmentation are brought in result matrix of segmentation:

$$sm_R(y, x) \leftarrow sm(y, x), \quad (12)$$

where  $y = \overline{0, Y-1}$ ,  $x = \overline{0, X-1}$ .

Step 20. Checking the stack pointer of collision as 0. If the pointer equal to zero  $z = 0$ , then skip and out from the algorithm, else ( $z \neq 0$ ) – go to the next step.

Step 21. Resolving collisions. In stack of collisions are found associated segment numbers, which are assigned new numbers. These numbers are brought in result matrix of segmentation. The stack pointer of collision is decreases by the corresponding number. When the stack pointer of collision equal to zero, skip for out of algorithm.

As a result of executing this algorithm is formed matrix of segmentation, value of each element which indicates a segment number, which belongs to the segmented pixel of image with corresponding to coordinates. With each cycle of busting growing of regions, the size of the segments gradually increasing, in what appears the progressive character of segmentation implemented by the proposed method. In the case of interruption (e.g. when searching for small objects) algorithm part of the image will remain un-segmented, however, will find all the dominant centers of regions and these regions will be uniformly segmented.

### Evaluating the effectiveness of segmentation algorithms

The stability of the number of segments to change the image forming conditions is the main advantage of the proposed method of segmentation. The proposed method is implemented in MATLAB 2012. The experiment was performed on a computer with the following specifications: Processor – AMD A8-5600U APU – 3.60 GHz; RAM – 4 GB; type of system – 32-bit operating system Windows 7.

In Fig. 1 the results of segmentation for standard test image France 2048 using suggested method and method of segmentation based on region growing are shown.

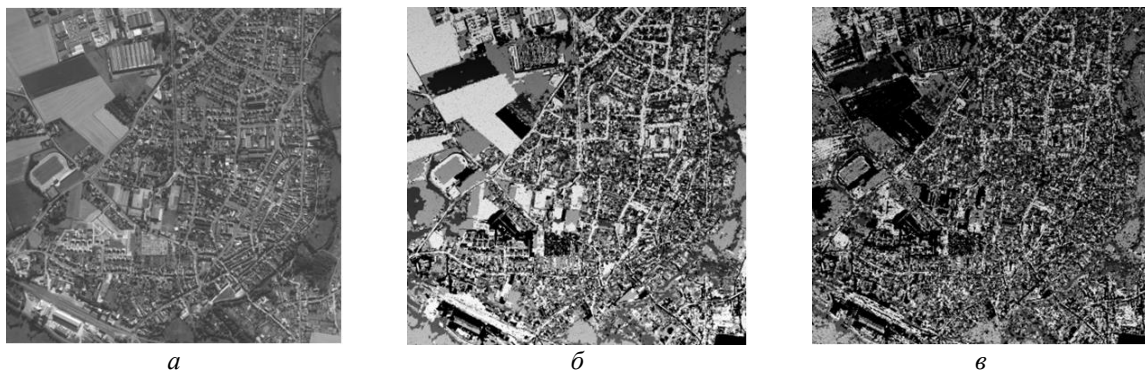


Fig. 1. Results of segmentation for standard test image France:  $a$  – test image;  $b$  – result for proposed method;  $c$ ) region growing

An assessment of the effectiveness of the proposed method of progressive segmentation based on wave quasi parallel of region growing and known methods of segmentation based on region growing, split and merge areas, watershed. As performance indicators used the time of segmentation, the stability of borders and the number of segments.

In table. presents the results of evaluation of the time of segmentation for considered methods. From table. Follows, that the proposed method wins in the speed of segmentation to 4 times and 40 times in comparison with the segmentation methods based on region growing and split and merge regions with an image size of  $128 \times 128$  pixels, losing in speed of segmentation to 5 ones when size  $512 \times 512$ ; 2,5 ones when size  $1024 \times 1024$  with method of segmentation based on region growing.

Segmentation time for test images, s

Methods of segmentation	Time of Segmentation for tested images			
	Boat 128	Lena 256	Barbara 512	France 8
Suggested method	<b>0,111867</b>	<b>0,7586</b>	<b>5,8029</b>	<b>88,6354</b>
Region growing	<b>0,439241</b>	<b>0,5269</b>	<b>0,8005</b>	<b>35,9598</b>
Splitting and merging regions	<b>4,3641</b>	–	–	–

In Fig. 2 shows the dependence of the areas of segments by changing the brightness, contrast, and image rotation, characterizing sustainability boundaries of segments. Stability is evaluated by the area ratio of segments for the base image to the area of the segments for the modified image, subject to change brightness, contrast and angle of rotation. From Fig. 3 follows, that suggested method losing in the stability of areas of segments to 7; 6; 6 ones when changing brightness, contrast and angle of rotation respectively in comparing with method of segmentation based on region growing. suggested method by comparing with the segmentation method based on splitting and merging regions losing in the stability of areas of segments to 3; 3,5 ones when changing brightness and contrast, but wins in the stability of areas of segments to 2,7 ones when changing the angle of rotation, results below is calculated on standard test images Lena  $128 \times 128$  and water  $128 \times 128$  respectively.

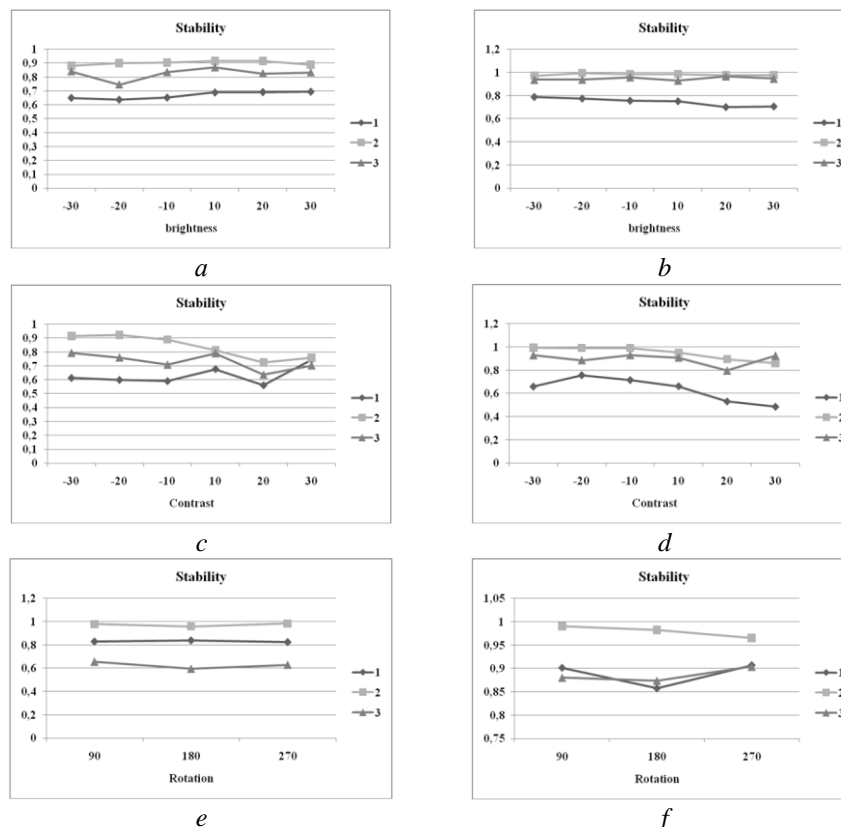


Fig. 2. Dependence areas of segments from changing forming of image, Lena and water (1 – suggested method; 2 – region growing; 3 – splitting and merging regions): a, b – when changing the brightness; c, d – when changing the contrast; e, f – when changing the angle of rotation

In Fig. 3 it is shown the dependence number of segments from changing the brightness, contrast, and image rotation, characterizing sustainability results of segments. Stability is evaluated by ratio of the number of segments for the base image to the number of segments for the modified image, subject to change brightness, contrast and angle of rotation. From Fig. 4 follows, that proposed method winning in stability of number of segments to 3,6; 4,5 and 3 ones when changing in brightness, contrast and angle of rotation in comparing with method of segmentation based on region growing. suggested method by comparing with the segmentation method based on splitting and merging regions winning in the stability of number of segments to 4,5; 4,4 and 7 ones when changing in brightness, contrast and angle of rotation of image respectively.

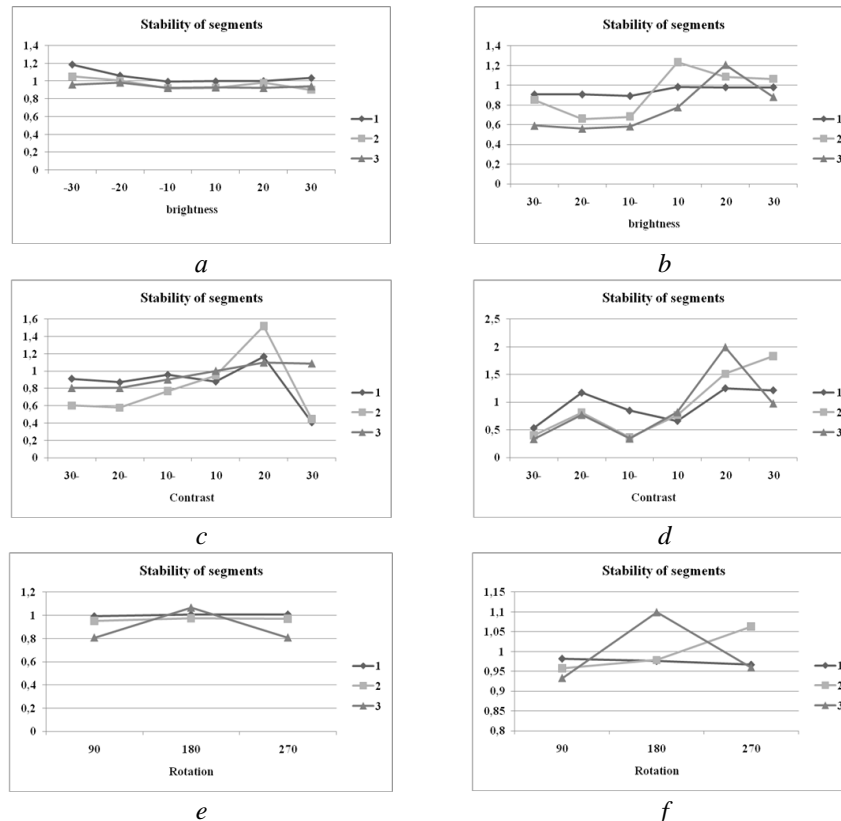


Fig. 3. Dependence number of segments from changing forming of image, Lena and water (1 – suggested method; 2 – region growing; 3 – splitting and merging regions): *a, b* – when changing the brightness; *c, d* – when changing the contrast; *e, f* – when changing the angle of rotation

### Summary

For the progressive segmentation of halftone images proposed a method based on the wave quasi parallel region growing. The essence of the method is a quasi-parallel growing areas around the chosen initial points of growth, that provides automatic separation of areas with a smooth drop of brightness, which known methods segmenting with errors. Showing, that proposed method provides increased sensitivity of segmentation to drops in brightness 1,6 % in comparing with method of segmentation based on region growing and 10,9 % in comparing with method of splitting and merging of regions. Established, that the proposed method losing in the stability areas of segments to 7; 6 and 6 ones when changing brightness, contrast and angle of rotation respectively in comparing with method of segmentation based on region growing. By comparing with segmentation method of splitting and merging regions, the proposed method losing in the stability of areas of segments to 3; 3,5 ones when changing brightness and contrast, but wins in the stability of areas of segments to 2,7 ones when changing the angle of rotation. Proposed method winning in stability of number of segments to 3,6; 4,5 and 3 ones when changing in brightness, contrast and angle of rotation in comparing with method of segmentation based on region growing. Suggested method by comparing with the segmentation method based on splitting and merging regions winning in the stability of number of segments to 4,5;

4,4 and 7 ones when changing in brightness, contrast and angle of rotation of image respectively. Established, that the proposed method winning in the speed of segmentation to 4 and 40 ones in comparing with method of segmentation based on region growing and method of splitting and merging regions when size of image 128×128 pixels.

### References

1. *Lalitha M., Kiruthiga M., Loganathan C.* // International Journal of Science and Research (IJSR). 2013. Vol. 2. №2. P. 348-358.
2. *Hyunki R., Haengsuk L.* // Proceedings of the 7th International Conference on Information Security and Assurance. 2013. Vol. 21. P. 185-189.
3. *Dass R., Priyanka, Devi S.* // International Journal of Electronics & Communication Technology (IJECT). 2012. Vol. 3. P. 66-70.
4. *Seal A., Das A., Sen P.* // International Journal of Computer Science and Information Technologies (IJCSIT). 2015. Vol. 6. P. 2295-2297.
5. *Chang J.H., Fan K.Ch., Chang Y.L.* // Image and Vision Computing. 2002. Vol. 20. P. 203-216.
6. *Muhsin Z.F., Rehman A., Altameem A. et. al* // The Imaging Science Journal. 2014. Vol. 62. №1. P. 56-62.
7. *Xiaolin Wu.* // IEEE Transactions On Pattern Analysis And Machine Intelligence. 1993. Vol. 15. P. 808-815.
8. *Singh K.K., Singh A.* // International Journal of Computer Science Issues. 2010. Vol. 7. №5. P. 414-417.
9. *Shih F.Y., Cheng S.* // Image and Vision Computing. 2005. №23. P. 877-886.
10. *Sharma Ritu, Sharma Rajesh* // International Journal of Innovative Research in Computer and Communication Engineering. 2014. Vol. 2. I. 9. P. 5686-5692.
11. *Mohd Saad N., Abu-Bakar S.A.R., Muda Sobri et. al* // Proceedings of the International MultiConference of Engineers and Computer Scientists. Hong Kong, 14-16 March 2012. P. 674-677.

УДК 621.391

## ЭМУЛЯЦИЯ КОРПОРАТИВНОЙ IP-ТЕЛЕФОНИИ НА ОСНОВЕ OPENSOURCE-ПРИЛОЖЕНИЙ

Д.В. ВОЛЧЕНКОВ

*Белорусский государственный университет информатики и радиоэлектроники  
П. Бровка, 6, Минск, 220013, Беларусь*

*Поступила в редакцию 25 апреля 2016*

Рассмотрены характерные особенности и проблемы межсетевого взаимодействия терминальных VoIP-устройств с виртуальными цифровыми АТС, произведена проверка возможности диагностики и отладки их работы в сети в результате анализа серверных лог-файлов и трафика в хостовой ОС. Установлены условия стабильной работы OpenSource-приложений и систем, на основе которых может производиться развертывание цифровых АТС в виртуальной среде, управляемой гипервизором.

*Ключевые слова:* IMS, OpenSource, АТС, виртуальная система, гипервизор, диагностика, сеть, анализ трафика.

### Введение

Для современных телефонных сетей характерен практически полностью завершённый переход к пакетному методу передачи голосовых данных. Несмотря на многочисленные преимущества IP-систем, следующие из объединения трафика в универсальных конвергентных сетях, включающих в себя телефонию, Интернет и телевидение на базе Triple Play подхода, для них также характерны специфические проблемы, связанные с необходимостью тонкой настройки на сетевом оборудовании QoS-параметров и оптимизации маршрутов для передачи голосового трафика. QoS учитывает такие характеристики потока, как джиттер, задержку, потери пакетов.

В конвергентных сетях также характерно применение унифицированного телекоммуникационного оборудования, в котором основные функции разделения и обработки трафика различных типов выполняет в большей степени программный, чем аппаратный комплекс. Чрезвычайно распространёнными являются виртуальные решения: как коммерческие SaaS, так и локальные, представляющие собой производительные аппаратные платформы со специфическими интерфейсами, например, для подключения BRI или E-карт.

Системы IMS предъявляют такие же базовые требования к сетевой инфраструктуре, как и обычные компьютерные сети. Множество специфических протоколов IMS, таких как SIP, RTP, SDP работают на прикладном уровне и подразумевают существование в сети поддержки базовой адресации, маршрутизации, а также систем хранения данных и протоколов доступа к файлам, например, для взаимодействия терминальных телефонных устройств с сервером, для получения информации о назначенном номере, линиях, а также голосовых шлюзах.

При отладке работоспособности современных телекоммуникационных систем важную роль имеет способ снятия и анализа трафика, проходящего в сети. Существует множество средств захвата трафика – от отдельных программно-аппаратных комплексов, до пользовательских диагностических утилит, позволяющих с достаточной точностью и подробностью показывать пользователю процессы взаимодействия устройств в сети через по пакетный анализ трейсов. Такие утилиты находят широкое применение в компаниях, не специализирующихся на те-

лекommunikационных услугах, однако использующих VoIP-системы для внутренних нужд и требующих достаточного уровня качества голосовых сервисов.

На любом предприятии себестоимость товара или услуги формируется с учетом затрат на его производство, в том числе косвенных, поэтому компании в странах со строгим законодательством относительно лицензионного ПО стараются с целью снижения затрат на ИТ-инфраструктуру в производстве использовать ПО, распространяемое под лицензией Open-Source.

Целью работы является эмуляция сетевой инфраструктуры предприятия с собственной IP телефонной системой с использованием OpenSource-приложений в качестве серверной платформы на базе гипервизора, а также бесплатных диагностических средств анализа трафика для изучения технологий межсетевого взаимодействия IMS различных масштабов.

### Выбор гипервизора, гостевой системы и IP АТС

В качестве тестовой топологии для простоты развертывания АТС и вспомогательной инфраструктуры выбрана схема с сервером, коммутатором, а также двумя IP-телефонами (рис. 1).



Рис. 1. Топология сетевой инфраструктуры

В качестве сервера использован ПК со следующими характеристиками: процессор Intel Core 2 Duo 2,1 ГГц, 1 Гб ОЗУ. Коммутатор – D-Link DES3526, телефоны Cisco 7962G. Все настройки проводились на стороне сервера, коммутатор и телефоны выступали в качестве пассивных устройств.

Ввиду распространенности ОС Windows 7 на ПК она использовалась как базовая хостовая система, на которую устанавливался бесплатный гипервизор 2 типа VirtualBox (рис. 2).

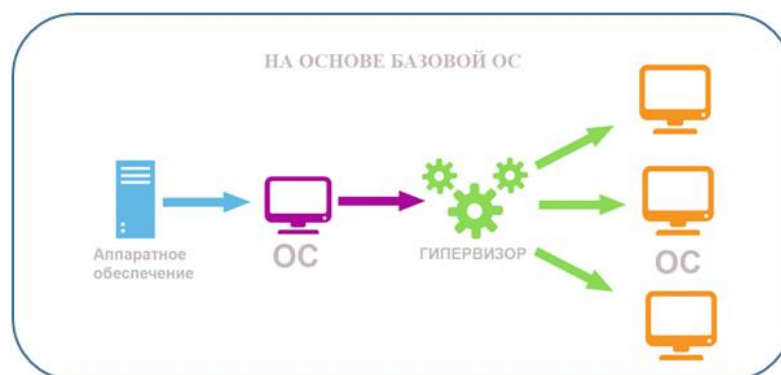


Рис. 2. Схема установки гостевой виртуальной ОС с серверной частью VoIP системы

Учитывая распространенность в промышленных решениях предустановленных и готовых к настройке вспомогательных для IP-телефонии пакетов с DHCP, TFTP, NTP для работы в сетевых окружениях в качестве дистрибутива выбран CentOS Linux. Из схожих соображений в качестве цифровой АТС выбрана Asterisk с возможностью графического конфигурирования FreePBX (рис. 3).



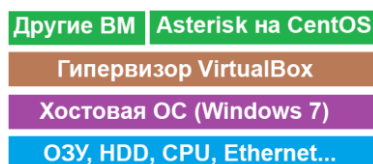


Рис. 3. Архитектура сервера

### Создание сервера и конфигурационных файлов для IP-телефонов

В гипервизоре VirtualBox существует несколько вариантов подключения гостевых ОС к локальной сети: через NAT-трансляцию адреса физического адаптера хостовой машины или через взаимодействие типа «мост». Первый способ не эффективен для приложений реального времени ввиду задержек и трудностей в трансляции адресов и портов для голосового RTP-трафика, приводящим к потере голосовых портов при трансляциях. Поэтому выбран режим моста, в котором виртуальный сервер посредством гипервизора имеет прозрачное подключение к коммутатору через Ethernet-порт хостовой машины, что исключило необходимость в NAT (рис. 4).



Рис. 4. Подключение сервера к локальной сети

В качестве интерфейса для захвата трафика выбрал Ethernet-порт ПК с виртуальным сервером (т.к. сервер в исследуемой топологии агрегировал как сигнальную, так и голосовую составляющую соединения между телефонами).

Учитывая алгоритм загрузки телефонов, включающий в себя последовательное обнаружение DHCP-сервера DISCOVER-пакетом со специфической опцией 150, означающей запрос у сервера опционального адреса TFTP-сервера, получение адреса и шлюза, и последующее обращение к TFTP-серверу по полученному из DHCP-адресу с запросом файлов конфигурации, была проведена настройка соответствующих серверных служб в гостевой ОС CentOS.

Произведено конфигурирование DHCP-сервера «dhcpd» с помощью следующих команд:

```
[root@server ~]# nano /etc/sysconfig/dhcpd
DHCPDARGS=eth0
[root@server ~]# nano /etc/dhcp/dhcpd.conf
option domain-name «labvoice»;
authoritative;
option domain-name-servers 192.168.1.111;
option voip-tftp-servers code 150 = ip-address;
option voip-tftp-servers 192.168.1.111;
subnet 192.168.1.0 netmask 255.255.255.0
{range dynamic-bootp 192.168.1.10 192.168.1.250;
option broadcast-address 192.168.1.255;
option routers 192.168.1.111;}
[root@server ~]# service dhcpd start
```

В результате на сервере интерфейс eth0 стал работать в режиме DHCP-сервера, назначая по запросу телефонов при загрузке IP-адрес, сообщая адрес хранилища файлов конфигурации, а также других параметров (шлюз и сервер DNS), используемых во всех прикладных окружениях.

Также произведена настройка TFTP-сервера с помощью следующих команд.

```
[root@server ~]# chkconfig xinetd on
[root@server ~]# service xinetd start
[root@server ~]# nano /etc/xinetd.d/tftp
server_args = -c -s /tftpboot -v -v -v
disable = no
[root@server ~]# mkdir /tftpboot
[root@server ~]# chmod 777 /tftpboot
[root@server ~]# service xinetd restart
```

Таким образом, создается и настраивается активный процесс TFTP-сервера «tftpd», указывающего по запросу телефонов в сети на созданную директорию /tftpboot, в которой располагаются файлы конфигурации телефонов.

Телефонные конфигурации представляют собой неизменяемые файлы прошивки, предоставляемой производителем, а также создаваемые пользователем XML-файлы, в имени которых располагается уникальный для каждого телефона MAC-адрес, по которому каждое устройство определяет следующие предназначенные для него настройки:

<preferredCodec>g711alaw</preferredCodec> – используемый голосовой кодек (можно несколько);

<phoneLabel>Sofia</phoneLabel> – имя, отображаемое на экране телефона;

<name>6580</name> – номер, созданный для телефона на АТС;

<line button=«1»> – конфигурация линии и привязка ее к кнопке;

<featureLabel>line 1</featureLabel> – имя линии, отображаемое на экране телефона;

<proxy>USECALLMANAGER</proxy>;

<port>5060</port>;

<dialTemplate>dialplan.xml</dialTemplate> – файл с информацией о маршрутизации звонков, располагающийся на сервере.

Файл dialplan.xml сконфигурирован таким образом, что после 3 секунд отсутствия набора набранная комбинация для маршрутизации передается на сервер с АТС:

```
<DIALTEMPLATE>
```

```
<TEMPLATE MATCH=«*» Timeout=«3»/>
```

```
</DIALTEMPLATE>
```

После полной настройки сервера и создания на нем необходимых файлов для загрузки телефонов создаются терминальные расширения в консоли Asterisk FreePBX для двух телефонов с указаниями номеров и имен пользователей (рис. 5).

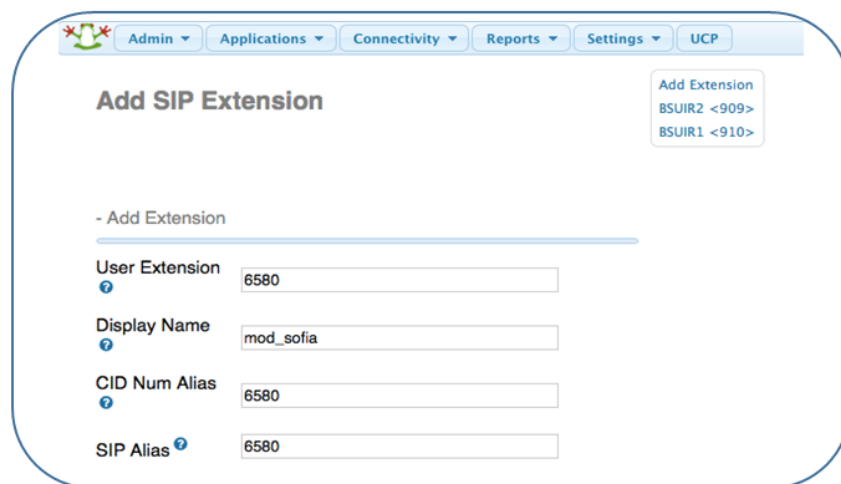


Рис. 5. Пример создания одного из телефонных расширений в Asterisk FreePBX

## Отладка процедуры загрузки телефонов и совершения вызова

Так как процессы в CentOS настроены таким образом, чтобы записывать события в лог-файл, их можно проанализировать с помощью утилит cat и grep, отфильтровав вывод по имени процесса [root@server ~]# cat /var/log/messages | grep -e 'tftpd|dhcpd' (рис. 6).

```
Apr 13 20:09:59 localhost dhcpd: DHCPDISCOVER from 50:1c:bf:fc:86:30 via eth0
Apr 13 20:10:00 localhost dhcpd: DHCPOFFER on 172.22.1.12 to 50:1c:bf:fc:86:30 (SEP501CBFFC8630) via eth0
Apr 13 20:10:00 localhost dhcpd: DHCPREQUEST for 172.22.1.12 (172.22.1.1) from 50:1c:bf:fc:86:30 (SEP501CBFFC8630) via eth0
Apr 13 20:10:00 localhost dhcpd: DHCPACK on 172.22.1.12 to 50:1c:bf:fc:86:30 (SEP501CBFFC8630) via eth0
Apr 13 20:10:03 localhost in.tftpd[3267]: RRQ from 172.22.1.12 filename term62.default.loads
Apr 13 20:10:04 localhost in.tftpd[3268]: RRQ from 172.22.1.12 filename jar42sip.9-4-2TH1-1.sbn
Apr 13 20:10:14 localhost in.tftpd[3292]: RRQ from 172.22.1.12 filename cnu42.9-4-2TH1-1.sbn
Apr 13 20:10:18 localhost in.tftpd[3316]: RRQ from 172.22.1.12 filename apps42.9-4-2TH1-1.sbn
Apr 13 20:10:43 localhost in.tftpd[3317]: RRQ from 172.22.1.12 filename dsp42.9-4-2TH1-1.sbn
Apr 13 20:10:46 localhost in.tftpd[3318]: RRQ from 172.22.1.12 filename cvm42sip.9-4-2TH1-1.sbn
Apr 13 20:11:47 localhost dhcpd: DHCPREQUEST for 172.22.1.12 from 50:1c:bf:fc:86:30 (SEP501CBFFC8630) via eth0
Apr 13 20:11:47 localhost dhcpd: DHCPACK on 172.22.1.12 to 50:1c:bf:fc:86:30 (SEP501CBFFC8630) via eth0
```

Рис. 6. Вывод команды анализа лог-файла, демонстрирующий процесс DHCP DORA и загрузку файлов конфигурации

Из вывода (рис. 6) видно, как происходит процесс загрузки телефона и какие файлы на сервере запрашиваются для его работы. Таким образом, в случае проблем в локальной сети анализ лог-файлов сервера может дать исчерпывающую отладочную информацию об ошибках.

Для анализа трафика, проходящего в тестовой сети, на хостовую ОС Windows 7 установлена OpenSource-утилита Wireshark. Так как это приложение работает напрямую с сетевыми интерфейсами, трафик, который обрабатывает сервер посредством гипервизора доступен для записи и анализа в реальном времени.

При совершении тестового звонка собирается отладочная информация о сигнальных пакетах SIP (рис. 7), а также захватывается RTP-трафик со всеми характеристиками установленного соединения, такими как используемый кодек, частота дискретизации (если переменная – то средняя и отклонения), джиттер, потери пакетов, задержка. Кроме того, Wireshark позволяет выводить данные о временных флуктуациях в доставке пакетов в графических гистограммах и воспроизводить разговор для диагностики (рис. 8).

No.	Time	Source	Destination	Protocol	Info
3	0..	192.168.1.111	192.168.1.12	SIP/SDP	Request: INVITE sip:6580@192.168.1.12;transport=udp
4	0..	192.168.1.12	192.168.1.111	SIP	Status: 100 Trying
5	0..	192.168.1.12	192.168.1.111	SIP	Status: 180 Ringing
6	3..	192.168.1.12	192.168.1.111	SIP/SDP	Status: 200 OK
7	3..	192.168.1.111	192.168.1.12	SIP	Request: ACK sip:6580@192.168.1.12:5060;transport=udp
1904	2..	192.168.1.12	192.168.1.111	SIP	Request: BYE sip:mod_sofia@192.168.1.111:5060
1905	2..	192.168.1.111	192.168.1.12	SIP	Status: 200 OK

Рис. 7. Сигнальные пакеты в сети, отфильтрованные в выводе Wireshark в реальном времени по ключевому слову «sip»

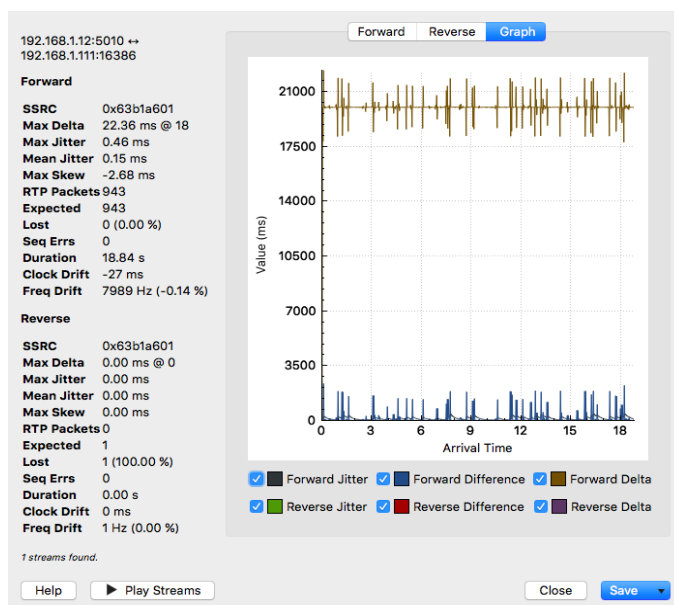


Рис. 8. Анализ прошедшей голосовой сессии средствами Wireshark

## Заключение

Произведена эмуляция виртуальной IMS АТС, позволяющей исследовать протоколы цифровой телефонии SIP и RTP, логику взаимодействия IP-телефонов в локальных сетях. Проведен анализ сигнальной информации и структуры пакетов, экспериментально определены условия стабильной работы АТС Asterisk в виртуальной среде, а также возможность захвата трейсов трафика гостевых ОС на уровне хостовой системы. На стороне сервера установлена возможность использования лог-файлов для отладки межсетевое взаимодействия вспомогательных для IMS-системы протоколов.

## EMULATION OF CORPORATE IP-TELEPHONY BASED ON OPENSOURCE APPLICATIONS

D.V. VOLCHENKOV

### Abstract

Overview of features and issues of VoIP terminals and virtual digital ATE, their internetworking, conducted debug possibility tests of IMS via server logs and data traffic dumps analyzing under host OS is done. Work stability of OpenSource packets and systems for ATE deploying in virtual digital environment under hypervisor control is adjusted.

*Keywords:* IMS, OpenSource, ATS, virtual system, hypervisor, diagnostics, network, traffic analysis.

### Список литературы

1. *Borocznyk T.* CentOS Bible. Hoboken, 2009.
2. *Madsen L., Bryant R., Meggelen J.V.* Asterisk: The Definitive Guide. Sebastopol, 2013.
3. Voip-info: Asterisk phone Cisco 79x1 XML configuration files for SIP. [Электронный ресурс]. Режим доступа: <http://www.voip-info.org/wiki/view/Asterisk+phone+cisco+79x1+xml+configuration+files+for+SIP>. Дата доступа: 22.04.2016.
4. Wireshark: RTP statistics. [Электронный ресурс]. Режим доступа: [https://wiki.wireshark.org/RTP\\_statistics](https://wiki.wireshark.org/RTP_statistics). Дата доступа: 20.05.2016.
5. Wireshark: UDP packets' jitter and delay. [Электронный ресурс]. Режим доступа: <https://ask.wireshark.org/questions/12837/udp-packets-jitter-and-delay>. Дата доступа: 22.04.2016.

УДК 528.854.2

## ПАРАМЕТРИЗАЦИЯ КРИВЫХ ЛИНИЙ НА ОСНОВЕ АППРОКСИМАЦИИ ОКРУЖНОСТЯМИ

Д.И. КИРИЛЮК, Ю.И. КУЛАЖЕНКО

*Белорусский государственный университет информатики и радиоэлектроники  
П. Бровка, 6, Минск, 220013, Беларусь*

*Поступила в редакцию 30 мая 2016*

Приведены методы геометрической параметризации линий. Проведен теоретический анализ этих методов в рамках задачи мобильных систем наблюдения.

*Ключевые слова:* параметризация линий, кривые на изображении.

### Введение

В настоящее время в связи с развитием мобильных систем наблюдения специального назначения стоит актуальная задача – обработка изображений в реальном масштабе времени. Для ее решения широко используются методы, которые основаны на распределении градиента яркости в окрестностях реперных точек. Однако в условиях проекционных искажений и шумовых помех эффективность градиентного подхода снижается. Устранение данного недостатка возможно за счет применения геометрического подхода. Геометрические методы ранее широко применялись в задачах для обработки изображений, имеющих синтетический характер (например, печатные платы, детали конструкций, САПР). Поэтому актуальной задачей в настоящее время является модернизация существующих геометрических методов и создание новых методов для обработки изображений, имеющих естественный характер (например, спутниковые, ландшафтные). Целью настоящей работы является анализ методов геометрической параметризации линий на изображениях.

### Методы параметризации кривых линий

В рамках таких задач к дескрипторам линий предъявляются следующие требования: устойчивость к проективным преобразованиям, устойчивость к шуму, высокая скорость формирования, произвольность формы кривой.

Известны следующие способы параметризации произвольных кривых линий.

1. Диаметр.
2. Эксцентриситет.
3. Форм-фактор.
4. Статистические характеристики.
5. Кривизна.
6. Аппроксимация аналитическими кривыми.
7. Аппроксимация ломаными.
8. Аппроксимация сплайнами.
9. Кодирование цепными кодами.
10. Сигнатуры.
11. Фурье-параметризация.
12. Преобразование Хафа.
13. Меры симметрий.

Кратко приведем описание этих методов.

Простейшим случаем произвольной линии является прямая. Примером исследований прямых линий посвящены работы [1, 2]. Важным численным показателем кривой  $C$  является диаметр, который определяется формулой

$$Diam(C) = \max_{i,j} [D(p_i, p_j)],$$

где  $D$  – мера расстояния,  $p_i$  и  $p_j$  – точки, принадлежащие кривой.

Этот параметр является важным в случаях, когда кривая образует замкнутый контур или близкий к замкнутому контуру, т.е. когда значение диаметра меньше расстояния от точки начала кривой до точки конца. Развитием этого численного показателя может служить дескриптор, состоящий из значения самого диаметра и направление отрезка, соединяющего две экстремальные точки, которые и определяют диаметр (этот отрезок называют большой осью). Аналогично определяется малая ось, как отрезок перпендикулярный большой оси и имеющий такую минимальную длину, что проведенный через концы обеих осей прямоугольник полностью содержит в себе всю границу (рис. 1).



Рис. 1. Диаметр, большая и малая оси

Еще одним численным параметром, характеризующим кривую, является эксцентриситет, т.е. отношение большой оси к малой [3].

Для параметризации формы кривой используют численный параметр – форм-фактор, который определяется формулой

$$f(n) = r(n)/s(n),$$

где  $r(n)$  – расстояние между концевыми точками линии,  $s(n)$  – некоторая мера расстояния, характеризующая длину линии [4]. Форм-фактор в совокупности с направлением отрезка, соединяющего начало и конец кривой линии, может являться полезным дескриптором.

Форму кривых линий можно количественно описывать с помощью статистических характеристик, таких как математическое ожидание, дисперсия, моменты более высокого порядка и другие [5].

Дескрипторы, являющиеся численными характеристиками формы линии, т.е. диаметра, эксцентриситета, форм-фактора, некоторые статистические характеристики являются вычислительно простыми, но слабо характеризующими форму линии, поэтому их использование целесообразно для предварительного сбора информации. После предварительного вычисления таких параметров следует принимать решение о необходимости дальнейшего вычисления более сложных дескрипторов.

Для построения дескриптора произвольной линии можно использовать кривизну кривой, т.е. скорость изменения угла наклона. Однако в данном способе имеются сложности вычисления в дискретном случае из-за возникновения локальных зазубрин, хотя разность углов наклона соседних сегментов линии может быть полезна.

Использование аналитических кривых не является целесообразным, т.к. подобрать математическое представление (коэффициенты) кривой линии удастся лишь в сравнительно простых случаях, и как правило, эти случаи связаны с какими-то искусственными линиями.

Существуют подходы к построению дескриптора через аппроксимацию кривой ломаными линиями [6] (рис. 2). Такое описание является вычислительно простым, но неустойчивым.



Рис. 2. Аппроксимация с помощью ломаной

Методы аппроксимации кривой линии с помощью ломаной сильно зависят от начальной и конечной точек, от выбора шага дискретизации, поэтому являются крайне неустойчивыми.

Для параметризации кривой применяют кодирование цепными кодами [7]. В этом случае линия представляется в виде последовательности соединенных отрезков, для которых указаны длина и направление. Направление каждого отрезка кодируется в соответствии со схемой нумерации. В общем случае такой метод имеет два недостатка: получаемая цепочка кодов оказывается слишком длинной, и любые малые возмущения приводят к изменениям кодовой последовательности, которые не связаны с общей формой кривой.

В качестве развития этого подхода можно предложить метод, который каждой последовательности из трех пикселей ставит в соответствие номер фазовой ориентации. Всего для трех пикселей возможно 12 фазовых ориентаций. Анализ стабильности этого метода при повороте проводился следующим образом. Исходное изображение, содержащее кривую линию, поворачивалось на  $360^\circ$  с шагом  $5^\circ$ . В итоге было получено 72 дескриптора, между которыми установлено покоординатное соответствие. По каждому элементу дескрипторов вычислено значение дисперсии. Среднее значение дисперсии равно 3,46.

Сигнатуры – одномерные функции, взаимно-однозначно определяющие кривую линию. В данном способе важен выбор некоторой фиксированной точки (центра), относительно которого будет строиться функция [8]. Например, вычисляются углы между отрезками от выбранной центральной точки до точек кривой. Последовательность таких углов может служить дескриптором.

Фурье-дескриптор кривой линии позволяет свести задачу с двумерного случая к одномерному. Для этого последовательность координат соседних пикселей кривой представляется в виде последовательности комплексных чисел (одна координата будет действительной частью, а другая мнимой частью комплексного числа.) задается последовательность  $s(k) = x(k) + iy(k)$ , где  $x(k)$  – координата  $x$   $k$ -й точки линии,  $y(k)$  – координата  $y$   $k$ -й точки линии. Дискретное преобразование Фурье конечной последовательности  $s(k)$  задается уравнением

$$a(u) = \frac{1}{K} \sum_{k=0}^{K-1} s(k) e^{-2\pi i u k / K},$$

где  $K$  – количество пикселей, из которых состоит линия;  $u = 0, 1, 2, \dots, K - 1$ .

Комплексные коэффициенты  $a(u)$  и будут являться дескриптором. Обратное преобразование Фурье позволяет по этим коэффициентам восстановить исходную линию. Фурье-дескриптор устойчив к повороту и параллельному переносу с некоторой модернизацией [9]. Недостатком является то, что некоторые детали кривой при преобразованиях могут теряться.

Определяющим шагом в методах, использующих цепные коды и Фурье-дескрипторы, является выбор начальной точки, т.е. стабильность этих методов в первую очередь зависит от выбора точки отсчета. Если удастся повысить эффективность выбора такой точки, то использование таких методов будет целесообразным. Применение сигнатур для требуемых в работе условий предполагает согласованный выбор фиксированной точки и выбор способа задания меры относительно этой точки.

Если существуют замкнутые многоугольники, то для их характеристики можно использовать вычисления меры симметрий [10].

Аппроксимации кривых линий сплайнами посвящено много работ [11, 12], т.к. сплайны обладают хорошими аппроксимационными свойствами. Для реализации этого подхода кривая

линия должна обладать условием существования функции, что на практике не всегда случается. Использование сплайнов и мер симметрий в рамках поставленной задачи не имеет смысла, т.к. такие методы не обладают необходимой вычислительной сложностью, и как следствие, не позволяют обрабатывать данные в реальном времени.

Методы Хафа для параметризации произвольных кривых линий неприменимы ввиду высокой вычислительной сложности.

**Сравнительная характеристика методов геометрической параметризации линий**

Дескрипторы линий	Вычислительная сложность	Произвольность формы линии	Устойчивость к аффинным преобразованиям
диаметр	низкая	–	–
эксцентриситет	низкая	–	–
форм-фактор	низкая	+	–
статистические характеристики	низкая	+	–
кривизна	средняя	+	–
аппроксимация аналитическими кривыми	высокая	–	+
аппроксимация ломаными	низкая	+	–
аппроксимация сплайнами	высокая	–	+
кодирование цепными кодами	средняя	+	+
сигнатуры	средняя	+	+
Фурье-параметризация	средняя	+	+
преобразование Хафа	высокая	+	+

**Заключение**

Теоретический анализ показал, что для решения поставленной задачи, с учетом вышеуказанных требований, наиболее эффективны методы на основе Фурье-дескрипторов, сигнатур или цепных кодов.

**PARAMETRIZATION OF CURVES LINES ON IMAGES**

D.I. KIRYLUK, Yu.I. KULAZHENKO

**Abstract**

Methods of geometrical parametrization of lines are given. The theoretical analysis of these methods within a problem of mobile systems of supervision is carried out.

*Keywords:* parametrization of lines, curves on the image.



### Список литературы

1. Журавлёв А.А., Цветков В.Ю // Информатика. 2014. С. 85-96.
2. Запрягаев С.А., Сорокин А.И. // Прикладная информатика. 2009. №4(22). С. 76-81.
3. Freeman H., Shapira. R. // Comm ACM. 1975. №18. P. 409-413.
4. Бородина О.Г., Цветков В.Ю. // Известия СПбГЭТУ «ЛЭТИ». 2015. №1. С. 41-45.
5. Papoulis A. Probability, random variables, and stochastic processes. New York, 1991.
6. Loncaric J. // Progress in Systems and Control Theory. 1998. Vol. 24. P. 303-322.
7. Bribiesca E. Pattern Recognition. 1999. №32. P. 235-251.
8. Ballard D.H., Brown C.M. Computer Vision. New Jersey, 1982.
9. Reddy B.S., Chatterj B.N. // IEEE Transactions on Image Processing. 1996. Vol. 5. P. 1266-1271.
10. Тузиков А.В. Анализ симметричности и сравнение объектов на основе сложения Минковского. Минск, 1998.
11. Абламейко С.В., Васильев В.П. // Теория и методы автоматизации проектирования: научно-технический сборник. 1979. Вып. 2. С. 35-40.
12. Абламейко С.В., Васильев В.П. // Автоматизация проектирования технологических процессов: научно-технический сборник. 1980. Вып. 1. С. 25-31.

УДК 621.3.011.22

## ВЛИЯНИЕ ФОРМЫ ОТВЕРСТИЙ ФОЛЬГИРОВАННЫХ МАТЕРИАЛОВ НА ИХ ХАРАКТЕРИСТИКИ ОТРАЖЕНИЯ И ПЕРЕДАЧИ ЭЛЕКТРОМАГНИТНОГО ИЗЛУЧЕНИЯ

Е.А.А. АЛЬ-МАШАТТ, О.В. БОЙПРАВ, Л.М. ЛЫНЬКОВ, Н.И. МУХУРОВ

*Белорусский государственный университет информатики и радиоэлектроники  
П. Бровка, 6, Минск, 220113, Беларусь*

*Поступила в редакцию 20 мая 2016*

Представлены результаты исследования характеристик отражения и передачи электромагнитного излучения в диапазоне частот 0,7...17 ГГц фольгированных материалов на основе алюминия в зависимости от формы содержащихся в них отверстий (круглая, квадратная, прямоугольная). Обоснованы преимущества применения таких материалов в целях изготовления экранирующих электромагнитное излучение кожухов для средств вычислительной техники.

*Ключевые слова:* алюминий, коэффициенты отражения и передачи электромагнитного излучения, фольгированный материал.

### Введение

Снижение уровня побочного электромагнитного излучения (ЭМИ) средств вычислительной техники может быть обеспечено путем экранирования этих средств, для чего применяются материалы, уменьшающие напряженность электрической и магнитной составляющих такого излучения в диапазоне частот от десятков мегагерц до десятков гигагерц. На основе этих материалов создаются конструкции, предназначенные для монтажа на стены помещений, в которых располагаются экранируемые средства вычислительной техники. В случае если создание таких конструкций является невозможным ввиду материальных и/или временных ограничений, то решение задачи снижения уровня побочного ЭМИ средств вычислительной техники реализуется путем изготовления специальных экранирующих кожухов для них. Одно из требований, предъявляемых к таким кожухам, – воздухопроницаемость. В работе [1] предложено для получения конструкций электромагнитных экранов, характеризующихся свойством воздухопроницаемости и низкими значениями коэффициента передачи ЭМИ, использовать фольгированные материалы на основе алюминия, содержащие отверстия. В работах [2, 3] установлено, что от геометрических параметров материалов, используемых для экранирования ЭМИ, зависят закономерности их взаимодействия с ЭМИ радиочастотного диапазона длин волн за счет так называемых краевых эффектов. Цель настоящей работы заключалась в исследовании влияния формы отверстий фольгированных материалов на основе алюминия на значения их коэффициентов отражения и передачи ЭМИ в диапазоне частот 0,7...17 ГГц.

### Методика проведения эксперимента

В работе исследованы три типа образцов фольгированных материалов на основе алюминия. Образец каждого из типов характеризовался наличием отверстий определенной геометрической формы. Отверстия в образце типа 1 представляли собой квадраты, размер сторон которых составлял 10 мм, отверстия образца типа 2 – круги с диаметром 10 мм, отверстия образца

типа 3 – равнобедренные прямоугольные треугольники, размер проведенной к гипотенузе высоты которых равен 10 мм.

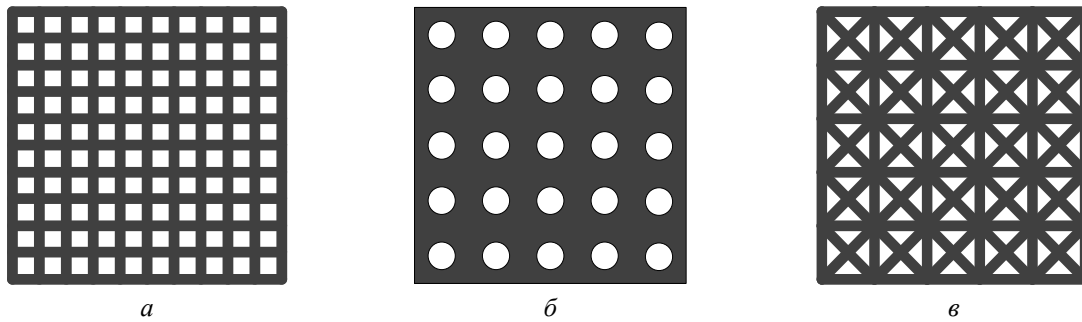


Рис. 1. Схематические изображения внешнего вида исследованных образцов фольгированных материалов на основе алюминия: *a* – образец типа 1; *б* – образец типа 2; *в* – образец типа 3

Измерение значений коэффициентов отражения и передачи ЭМИ в диапазоне частот 0,7...17 ГГц исследованных образцов фольгированных материалов на основе алюминия выполнялись в соответствии с методикой [4].

### Результаты и их обсуждение

На основе результатов проведенных измерений построены частотные зависимости коэффициентов отражения и передачи ЭМИ в диапазоне 0,7...17 ГГц, которые представлены на рис. 2, 3 (кривые 1 соответствуют указанным зависимостям для исследованного образца фольгированного материала типа 1; кривые 2 и 3 – для исследованных образцов фольгированных материалов типов 2 и 3 соответственно).

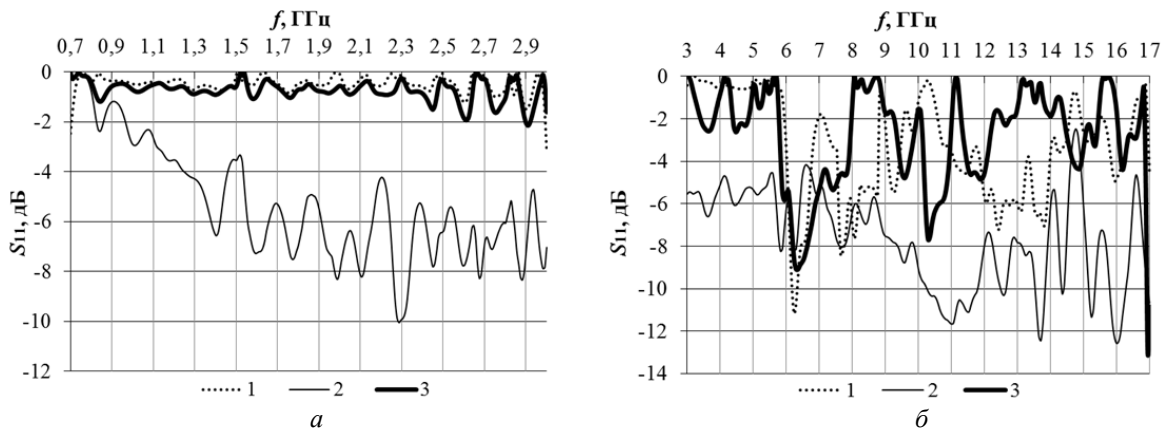


Рис. 2. Частотные зависимости коэффициента отражения ЭМИ в диапазоне 0,7...3 ГГц (*a*) и 3...17 ГГц (*б*) исследованных образцов фольгированных материалов на основе алюминия

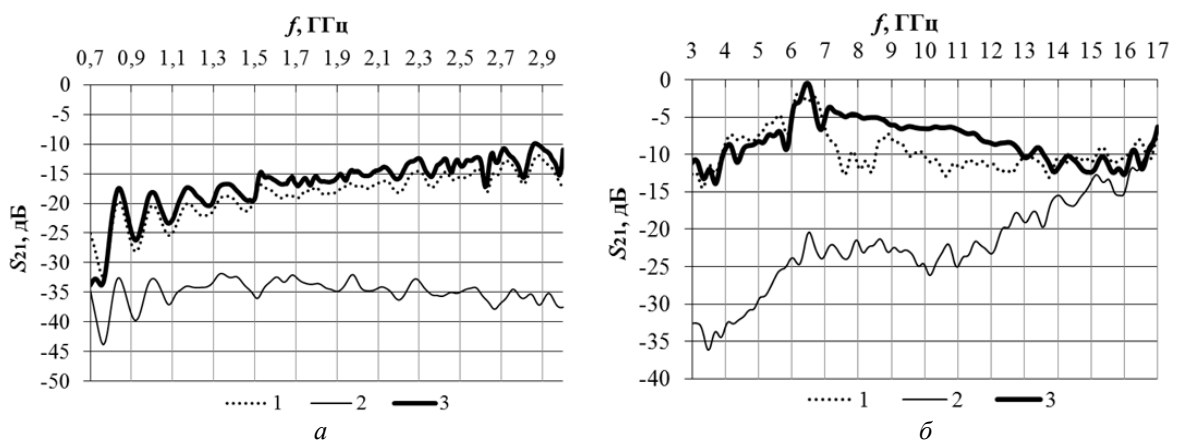


Рис. 3. Частотные зависимости коэффициента передачи ЭМИ в диапазоне 0,7...3 ГГц (*a*) и 3...17 ГГц (*б*) исследованных образцов фольгированных материалов на основе алюминия

Показано, что фольгированный материал на основе алюминия, содержащий отверстия в виде квадратов и треугольников, характеризуется значениями коэффициентов отражения и передачи ЭМИ в диапазоне частот 0,7...3 ГГц, равными соответственно  $-0,5...-2$  дБ и  $-10...-35$  дБ. В диапазоне частот 3...17 ГГц величины указанных параметров составляют  $-0,5...-11$  дБ и  $-1...-13$  дБ. Для фольгированного материала на основе алюминия, содержащего отверстия в виде кругов, величины коэффициентов отражения и передачи ЭМИ в диапазоне частот 0,7...3 ГГц составляют  $-0,5...-10$  дБ и  $-33...-43$  дБ, в диапазоне частот 3...17 ГГц –  $-4...-12,5$  дБ и  $-10...-37$  дБ.

Образец типа 2 характеризуется меньшими значениями коэффициента передачи ЭМИ в диапазоне частот 0,7...17 ГГц по сравнению с образцами типов 1 и 3 ввиду того, что площадь его отверстий равна  $0,785 \text{ см}^2$  и меньше, чем площади отверстий образцов типов 1 и 3, составляющие  $1 \text{ см}^2$ . Увеличение коэффициента передачи ЭМИ всех исследованных образцов с ростом частоты в диапазоне 3...17 ГГц обусловлено возрастанием значения, соответствующего разности размера отверстий и длины волны.

Значения коэффициента отражения ЭМИ в диапазоне частот 0,7...17 ГГц образцов типов 1 и 3 превышают в среднем на 6 дБ величины аналогичного параметра для образца типа 2. Это может быть связано с явлением интерференции электромагнитных волн, рассеиваемых на углах отверстий образцов типов 1 и 3 и характеризующихся одинаковой фазой [2, 3].

### **Заключение**

Установлено, что наименьшими значениями коэффициентов отражения и передачи ЭМИ в диапазоне частот 0,7...17 ГГц характеризуется фольгированный материал на основе алюминия, содержащий отверстия круглой формы. Величины указанных параметров соответственно составляют  $-0,5...-13$  дБ и  $-10...-43$  дБ. Изменение формы отверстий в таком материале с круглой на квадратную или треугольную приводит к увеличению на  $1...11$  дБ значений его коэффициента отражения ЭМИ в диапазоне частот 0,7...17 ГГц и на  $1...25$  дБ значений коэффициента передачи ЭМИ в указанном диапазоне частот. На основе полученных результатов можно сделать вывод о том, что для изготовления экранирующих ЭМИ кожухов для средств вычислительной техники наиболее целесообразно применять фольгированные материалы на основе алюминия, содержащие отверстия круглой формы. Оптимальный диаметр этих отверстий – 10 мм.

## **INFLUENCE OF THE FOIL MATERIALS HOLES FORM ON THEIR ELECTROMAGNETIC RADIATION REFLECTION AND TRANSMISSION CHARACTERISTICS**

E.A.A. ALMASHAT, O.V. BOIPRAV, L.M. LYNKOU, N.I. MUKHUROV

### **Abstract**

The research results of electromagnetic radiation reflection and transmission characteristics in frequency range 0,7...17 GHz of the alumina foil materials depending on the shape of holes contained therein are presented. The advantages of such materials utilization for production of electromagnetic radiation shielding cases for computer equipment are grounded.

*Keywords:* alumina, electromagnetic radiation reflection and transmission coefficients, foil material.

### Список литературы

1. *Лыньков Л.М., Борботько Т.В., Бойправ О.В. и др.* Устройство защиты человека от электромагнитного излучения радиоэлектронного оборудования / Патент РФ №10985.
2. *Уфимцев П.Я.* Теория дифракционных краевых волн в электродинамике. М., 2012.
3. *Уфимцев П.Я.* Метод краевых волн в физической теории дифракции. М., 1962.
4. *Бойправ О.В., Борботько Т.В., Лыньков Л.М. и др.* // Электронный журнал «Труды МАИ». 2013. №67. Режим доступа: <http://www.mai.ru/science/trudy/published.php?ID=41430>. Дата доступа: 31.05.2016.

## СВЕДЕНИЯ ОБ АВТОРАХ

1. Абдулхуссейн Хайдер М. Абдулхуссейн - магистрант кафедры сетей и устройств телекоммуникаций БГУИР
2. Алкалби Аммар Саттар Джаббар - магистрант кафедры сетей и устройств телекоммуникаций БГУИР
3. Аль-Аббуди Али Абдул Разактареш - магистрант кафедры сетей и устройств телекоммуникаций БГУИР
4. Аль-Байати Али Эмад Кадхум - магистрант кафедры сетей и устройств телекоммуникаций БГУИР
5. Аль-Бахдили Хасан Кассим - аспирант кафедры сетей и устройств телекоммуникаций БГУИР
6. Аль-Гергани Кутэйба Мудэхер Орайби - магистрант кафедры сетей и устройств телекоммуникаций БГУИР
7. Аль-Заиди Заид Хамид Мохсин - магистрант кафедры сетей и устройств телекоммуникаций БГУИР
8. Альзаки Хайдер Макки Хамид - аспирант кафедры сетей и устройств телекоммуникаций БГУИР
9. Аль-Машатт Емад Абдульхафид Альмухтар - аспирант кафедры защиты информации БГУИР
10. Альмияхи Осама Мажид Хилал - аспирант кафедры сетей и устройств телекоммуникаций БГУИР
11. Бобов Михаил Никитич - д.т.н., профессор кафедры сетей и устройств телекоммуникаций БГУИР
12. Бойправ Ольга Владимировна - ассистент кафедры защиты информации БГУИР
13. Волков Кирилл Аркадьевич - к.т.н., доцент кафедры сетей и устройств телекоммуникаций БГУИР
14. Волченков Даниил Владимирович - студент кафедры сетей и устройств телекоммуникаций БГУИР
15. Зайцев Станислав Сергеевич - магистрант кафедры сетей и устройств телекоммуникаций БГУИР
16. Карбалаи Салех Хоссейн Мохаммадэбрахим - магистрант кафедры сетей и устройств телекоммуникаций БГУИР
17. Кирилюк Денис Игоревич - аспирант кафедры сетей и устройств телекоммуникаций БГУИР

## СВЕДЕНИЯ ОБ АВТОРАХ

18. Конопелько Валерий Константинович - д.т.н., профессор, зав. кафедрой сетей и устройств телекоммуникаций БГУИР
19. Королёв Алексей Иванович - к.т.н., доцент кафедры сетей и устройств телекоммуникаций БГУИР
20. Кулаженко Юрий Иванович - д.ф.-м.н., доцент, начальник управления подготовки научных кадров высшей квалификации БГУИР
21. Макейчик Екатерина Геннадьевна - старший преподаватель кафедры сетей и устройств телекоммуникаций БГУИР
22. Махммуд Махммуд Басил Махммуд - магистрант кафедры сетей и устройств телекоммуникаций БГУИР
23. Мухуров Николай Иванович - д.т.н., профессор кафедры защиты информации БГУИР
24. Новицкий Виталий Владимирович - магистрант кафедры сетей и устройств телекоммуникаций БГУИР
25. Панькова Вероника Витальевна - аспирант кафедры информационных радиотехнологий БГУИР
26. Парасочка Андрей Владимирович - магистрант кафедры сетей и устройств телекоммуникаций БГУИР
27. Романов Роман Русланович - магистрант кафедры сетей и устройств телекоммуникаций БГУИР
28. Садик Бакир Джафар Садик - аспирант кафедры сетей и устройств телекоммуникаций БГУИР
29. Саломатин Сергей Борисович - к.т.н., доцент кафедры сетей и устройств телекоммуникаций БГУИР
30. Томилин Валерий Валерьевич - студент кафедры защиты информации БГУИР
31. Хоменок Михаил Юлианович - к.т.н., доцент кафедры сетей и устройств телекоммуникаций БГУИР
32. Цветков Виктор Юрьевич - к.т.н., доцент кафедры сетей и устройств телекоммуникаций БГУИР
33. Шевчук Оксана Геннадьевна - аспирант кафедры сетей и устройств телекоммуникаций БГУИР





**ДЛЯ ЗАПИСЕЙ**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---



ISBN 978-985-488-834-7



9 789854 888347