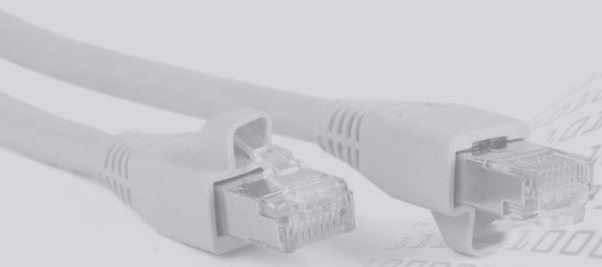
ТЕЛЕКОММУНИКАЦИИ: СЕТИ И ТЕХНОЛОГИИ,

АЛГЕБРАИЧЕСКОЕ КОДИРОВАНИЕ И БЕЗОПАСНОСТЬ ДАННЫХ

МАТЕРИАЛЫ МЕЖДУНАРОДНОГО НАУЧНО-ТЕХНИЧЕСКОГО СЕМИНАРА

(Минск, апрель – декабрь 2015 г.)



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 2015 г.)

TELECOMMUNICATIONS: NETWORKS AND TECHNOLOGIES, ALGEBRAIC CODING AND DATA SECURITY

Минск, 2015

УДК 621.391:004.056.55
ББК 32.811+32.973.26-018.2
Т31

Т31 **Телекоммуникации**: сети и технологии, алгебраическое кодирование и безопасность данных : материалы Международного научно-технического семинара (Минск, апрель – декабрь 2015 г.) Telecommunications: Networks and Technologies, Algebraic Coding and Data Security – Минск : БГУИР, 2014. – 92 с.

ISBN 978-985-488-834-7.

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 08.12.2012. Формат 60×84 ⅛. Бумага офсетная. Гарнитура «Таймс».
Отпечатано на ризографе. Усл. печ. л. 10,46. Уч.-изд. л. 8,9. Тираж 50 экз. Заказ 760.

Издатель и полиграфическое исполнение: учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
ЛИ №02330/264 от 14.04.2014. ЛП №02330/0494175 от 03.04.2009.
220013, Минск, П. Бровки, 6

ISBN 978-985-488-834-7

СОДЕРЖАНИЕ

CONTENTS

ТЕЛЕКОММУНИКАЦИИ: СЕТИ И ТЕХНОЛОГИИ, АЛГЕБРАИЧЕСКОЕ КОДИРОВАНИЕ И БЕЗОПАСНОСТЬ ДАННЫХ

УДК 539.216:546.824-31

ДЕТЕКТИРОВАНИЕ УГЛОВ КОНТУРНЫХ ЛИНИЙ НА ИЗОБРАЖЕНИИ С ПОМОЩЬЮ РАСШИРЯЮЩИХСЯ МАСОК

О.Г. ШЕВЧУК, А.В. КОСТУСЕВ, Е.Г. МАКЕЙЧИК

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

Поступила в редакцию 1 декабря 2015

Предложен метод детектирования углов контурных линий на основе расширяющихся масок для аэрокосмических изображений. Проведен анализ разработанного метода при изменении яркости, контраста и при повороте изображения. Показано, что метод устойчив к изменению яркости и повороту.

Ключевые слова: угловой детектор, аэрокосмические изображения, масочный анализ.

Введение

Нахождение ключевых точек на изображении является важной задачей в различных областях, таких как видеонаблюдение, параметризация и идентификация объектов, сшивка изображений и д.р. Ключевые точки должны быть [1]:

1) уникально определяемыми в некоторой окрестности и изображении в целом;

2) инвариантными к аффинным преобразованиям;

3) стабильными относительно шумов;

4) эффективными для параметризации и идентификации объектов в совокупности анализируемых изображений.

Самыми распространенными детекторами ключевых точек являются SIFT [2] и SURF [3]. Эти методы основаны на вычислении градиента яркости точки и ее окрестности, что позволяет детектировать области, края и углы.

Методы SIFT [2] и SURF [3] инвариантны к повороту, шумам и изменении позиции наблюдения. Однако данные методы неприменимы при сшивке аэрокосмических изображений, которые имеют фрактальную или себе подобную структуру (море, пустыня, горы), а также в условиях резкой смены освещения (день и ночь).

Для устранения данных недостатков необходимо в качестве ключевых точек использовать углы и линии. Цель работы – разработка метода детектирования углов контурных линий на основе расширяющихся масок для аэрокосмических изображений.

Теоритические сведения

В таких задачах, как разбиение контуров на отдельные прямые линии и сшивка изображений, в качестве ключевых точек используются углы.

Углы можно классифицировать следующим образом [4] (рис. 1):

1) L-образные (с различной величиной угла);

2) Y-образные (с различной величиной угла), частным случаем являются T-образные углы;

3) X-образные (с различной величиной углов).

Рис. 1. Виды углов на изображении

Существует ряд методов поиска углов на изображении: Харриса [5], Shi-Tomasi [6], FAST [7] и др. Угловые детекторы чаще всего работают с яркостной составляющей значения пикселя.

На вход такого детектора, работающего с яркостной составляющей, подается черно-белое изображение, в результате чего на выходе формируется список возможных углов со степенью подобия. Точки со степенью подобия больше порога определяются как углы, а меньше порога – отбрасываются.

Недостатком использования таких детекторов в приведенных выше задачах является то, что они обнаруживают область, в которой находится угловая точка, но не всегда точно определяют ее координаты, т.е. угловая точка в результате работы данных алгоритмов может находиться вне линии. Поэтому после применения таких детекторов необходимо уточнять местоположение каждого угла, например, с помощью маски.

Обнаружение углов с помощью масочного детектора осуществляется путем наложения заданной маски, шаблона, на бинарное изображение. Данный метод зависит от выбранной маски и чаще всего не эффективен при толщине линии в 2 и более пикселя. Размер использованной маски, чем толще линия – тем больше размер маски, влияет на вычислительную сложность детектора, а следовательно на время работы алгоритмов.

Для устранения существующих недостатков был предложен метод детектирования углов контурных линий на основе расширяющихся масок для аэрокосмического изображения с предварительной нормализацией контурных линии по толщине. Метод применяется только для выделенных контуров, не являющихся прямой линией. Свойства контура определяется по количеству концевых точек k, если $k = 2$ решение принимается на основе форм-фактора [8].

Метод масочного детектирования углов на изображении

Предложенный метод основан на последовательном наложении маски 3×3 пикселя на каждый пиксель выделенного контура, не являющегося прямой линией, кроме концевых точек контура. При неоднозначном решении для пикселя осуществляется дополнительная его проверка путем расширения маски до размера 5×5 пикселей.

На вход метода подается изображение $I = \left\| i(y, x) \right\|_{\left(y = \overline{0, Y-1},\, x = \overline{0, X-1} \right)}$, 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. 2.

Алгоритм работы метода состоит из следующих шагов.

Шаг 1. Формирование матрицы бинарных образов $I_B = \left\| i_B(y, x) \right\|_{\left(y = \overline{0, Y-1},\, x = \overline{0, X-1} \right)}$ 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 $S = \left\| s(n) \right\|_{n = \overline{1, N}}$ и матрица количества концевых точек в каждом контуре $K = \left\| k(n) \right\|_{n = \overline{0, N}}$, где $s(n)$ – координаты контурных пикселей n-го контура, представленных в виде матриц $X(n) = \left\| x(n, c) \right\|_{\left(c = \overline{0, C-1} \right)}$, $Y(n) = \left\| y(n, c) \right\|_{\left(c = \overline{0, C-1} \right)}$, $k(n)$ – количество концевых точек для n-го контура, N – количество найденных контуров, C – количество пикселей в n-м контуре.

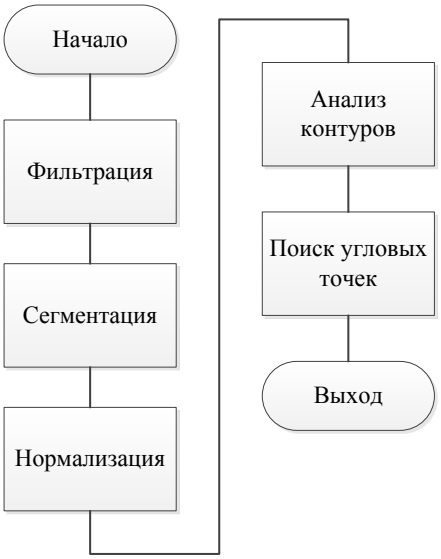


Рис. 2. Блок-схема углового масочного детектора

Шаг 3. Нормализация выделенных контуров по толщине. Нормализация контуров осуществляется с помощью метода нормализации контурных линий по толщине [10]. 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 [8] для проверки наличия кривизны контура $s(n)$.

Если $f=[0.8,1.2]$ – принимается решение, что контур $s(n)$ является прямой линий, следовательно, не имеет углов. Поэтому данный контур удаляется из матрицы S.

Шаг 5. Поиск угловых точек. В результате выполнения алгоритма формируются угловые матрицы $X_A=\|x(g,n)\|_{g=\overline{0,G}}$, $Y_A=\|y(g,n)\|_{g=\overline{0,G}}$, где $x(g,n)$, $y(g,n)$ – координаты угловой точки, g – порядковый номер координаты, G – количество найденных угловых точек, n – номер контура, к которому относится угловая точка.

5.1 Формирование маски $M_j=\|m_j(y,x)\|_{(y=\overline{0,2},x=\overline{0,2})}$ (рис. 3), где $j=\overline{1,16}$ – порядковый номер матрицы, $m_j(0,0)=\begin{cases}1, & \text{для } j=\{1,2,9\},\\ 0, & \text{для } j=\{\overline{3..8},\overline{10..16}\},\end{cases}$ $m_j(0,1)=\begin{cases}1, & \text{для } j=\{5,6,13,14\},\\ 0, & \text{для } j=\{\overline{1..4},\overline{7..12},15,16\},\end{cases}$

$m_j(0,2)=\begin{cases}1, & \text{для } j=\{1,4,11,16\},\\ 0, & \text{для } j=\{2,3,\overline{5..10},\overline{12..15},\},\end{cases}$ $m_j(1,0)=\begin{cases}1, & \text{для } j=\{6,7,11,12\},\\ 0, & \text{для } j=\{\overline{1..5},\overline{8..10},\overline{13..16}\},\end{cases}$ $m_j(1,1)=1,$

$m_j(1,2)=\begin{cases}1, & \text{для } j=\{5,\overline{8..10}\},\\ 0, & \text{для } j=\{\overline{1..4},6,7,\overline{11..16}\},\end{cases}$ $m_j(2,0)=\begin{cases}1, & \text{для } j=\{2,3,10,14\},\\ 0, & \text{для } j=\{1,\overline{4..9},\overline{11..13},15,16\},\end{cases}$

$m_j(2,1)=\begin{cases}1, & \text{для } j=\{7,8,15,16\},\\ 0, & \text{для } j=\{\overline{1..6},\overline{9..14}\},\end{cases}$ $m_j(2,1)=\begin{cases}1, & \text{для } j=\{3,4,12,13\},\\ 0, & \text{для } j=\{1,2,\overline{5..11},\overline{14..16}\}.\end{cases}$

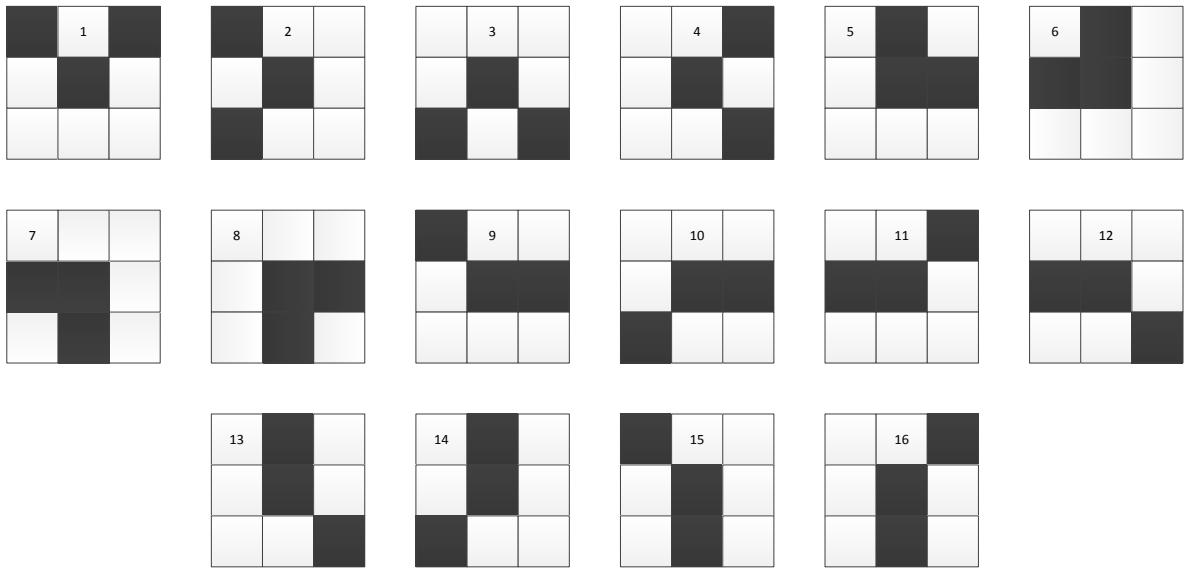


Рис. 3. Маски поиска угловых точек в контурах

5.2 Формирование угловых матриц Y_A, X_A. Окрестность каждого c-го пикселя контура $s(n)$ с координатами $y(n,c), x(n,c)$ матриц $Y(n), X(n)$ проверяется на соответствие одной из 16 угловых масок M_j.

Если окрестность пикселя с координатами $y(n,c), x(n,c)$ соответствует маскам $M_1..M_8$ – осуществляется переход на пункт 5.4. Если окрестность пикселя с координатами $y(n,c), x(n,c)$ соответствует маскам $M_9..M_{16}$ – осуществляется переход на пункт 5.3.

5.3 Уточнение. Для контурного пикселя с координатами $y(n,c), x(n,c)$, окрестность которых соответствует матрицам $M_9..M_{16}$, осуществляется уточнение путем расширения матриц до размера 5×5 пикселей, как показано на рис. 4. Если окрестность c-го пикселя с координатами $y(n,c), x(n,c)$ соответствует расширенной матрице (проверяется присутствие минимум одной контурной точки с каждой стороны в области, указанной серым цветом на рис. 4). Осуществляется переход на пункт 5.4.

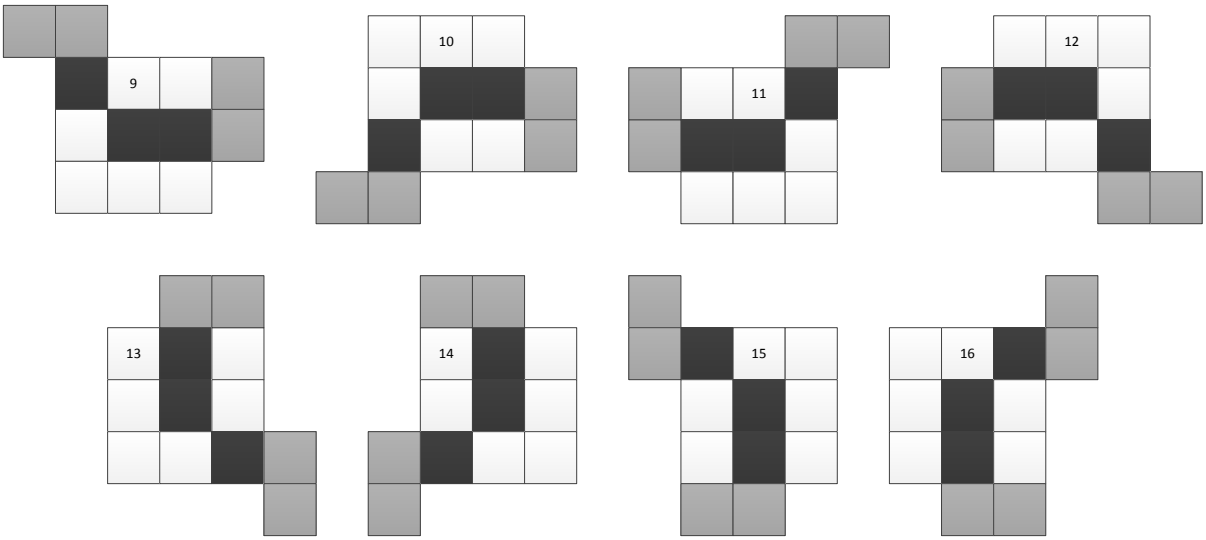


Рис. 4. Расширенные маски поиска угловых точек в контурах

5.4 Координаты пикселя $y(n,c), x(n,c)$, определенные как угловая точка, заносятся в матрицы Y_A, X_A соответственно.

В результате работы метода формируются матрицы координат Y_A, X_A, содержащие координаты угловых точек, которые в дальнейшем могут быть использованы для сшивки изображений.

Оценка эффективности детектора линий на основе расширяющихся масок

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 аэрокосмическое изображение размером 720×720 пикселей (рис. 5).

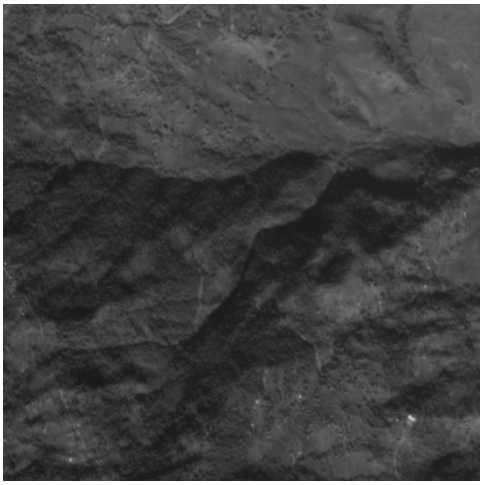
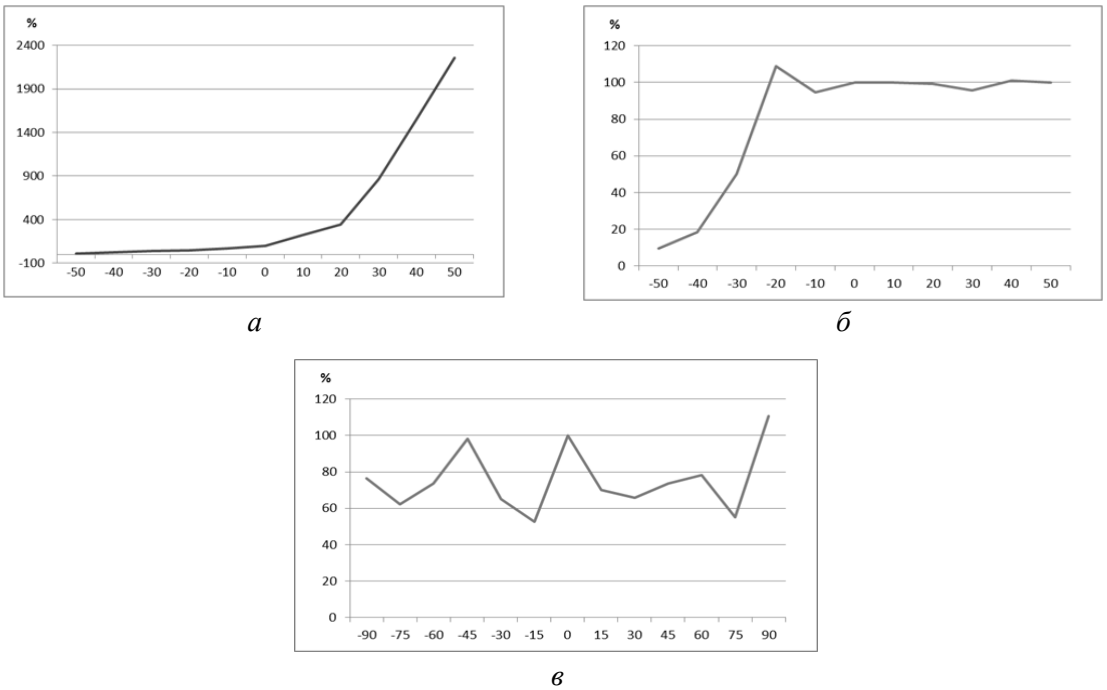


Рис. 5. Тестовое аэрокосмическое изображение

Для количественной оценки устойчивости произведено сопоставление числа детектированных углов, сохранивших свое местоположение после изменения яркости, контраста и поворота изображения, с числом углов, детектированных на исходном тестовом изображении. Результаты оценки приведены на рис. 6.



а *б* *в*

Рис. 6. График оценки устойчивости метода детектирования углов на основе масок: *а* – при изменении контраста изображения, *б* – при изменении яркости изображения, *в* – при повороте изображения

Из рис. 6 видно, что метод стабилен при изменении яркости от –20 % до 50 % относительно тестового изображения, однако теряет устойчивость при изменении яркости от –50 % до –30 % и при изменении контраста. При повороте изображения на угол $\alpha = -90 \ldots 90$ градусов количество детектированных углов изменяется от 60 % до 110 %.

Заключение

Разработан метод детектирования углов на основе расширяющихся масок для аэрокосмических изображений. Показано, что метод стабилен при изменении яркости от –20 % до 50 % относительно тестового изображения, а при повороте изображения на угол $\alpha = -90...90$ градусов количество детектированных углов изменяется от 60 % до 110 %. К недостаткам метода относится потеря устойчивости при изменении контраста.

DETECTION OF ANGLES CONTOUR LINES ON THE IMAGE USING EXPANDING MASKS

O.G. SHEVCHUK, A.V. KOSTUSEU, E.G. MAKEICHIK

Abstract

Method for detecting the angle of the contour lines on the basis of expanding masks for aerospace images is proposed. Analysis of this method when the brightness, contrast and image route are changed is held. It is shown that the method is resistant to changes in brightness and route.

Список литературы

1. *Rodehorst V., Koschan A.* // Proc. of 5th International Symposium Turkish-German Joint Geodetic Days TGJGD'06. 2006. P. 8.
2. *Lowe D.* // International Journal of Computer Vision. 2004. Vol. 60. №2. P. 91-110.
3. *Bay H.* // Proc. of 9th Euproean Conference on Computer Vision. 2006. P. 404-410.
4. *Xia G., Delon J., Gousseau Ya.* // International Journal of Computer Vision January. 2014. Vol. 106. P. 31-56
5. *Harris C., Stephens M.* // Proc. of 4th Alvey Vision Conference. 1988. P. 147-151.
6. *Shi J., Tomasi C.* // Proc. of 9th IEEE Conference on Computer Vision and Pattern Recognition. 1994. P. 593-600.
7. *Rosten E., Drummond T.* // Proc. of 9th Euproean Conference on Computer Vision. 2006. P. 430-443.
8. *Бородина О.Г., Цветков В.Ю.* // Известия СПбГЭТУ «ЛЭТИ». 2015. №1. С. 41-45.
9. *Canny J.A.* // IEEE Transactions On Pattern Analysis And Machine Intelligence. 1986. Vol. 8. P. 679-698.
10. *Шевчук О.Г., Кирилюк Д.И., Макейчик Е.Г. и др.* // Докл. БГУИР. 2015. №7(93). С. 51-57.

УДК 004.056.55

ОПРЕДЕЛЕНИЕ ВЕСА ОШИБКИ В ПРИМИТИВНЫХ БЧХ-КОДАХ

Н.В. СПИЧЕКОВА, В.А. ЛИПНИЦКИЙ

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

Военная академия Республики Беларусь
Минск-57, 220057, Беларусь

Поступила в редакцию 24 сентября 2015

Исследованы свойства синдромов ошибок весом 1-4 в примитивных БЧХ-кодах C_9, приведен алгоритм определения кратности возникшей ошибки по ее синдрому, показана связь между предлагаемым алгоритмом и методом определителей Блейхута.

Ключевые слова: помехоустойчивое кодирование, БЧХ-код, синдром ошибки, вес ошибки, метод определителей Блейхута.

Введение

Современные цифровые телекоммуникационные системы (ТКС), за исключением волоконно-оптических, функционируют с непременным кодированием информации помехоустойчивыми кодами. Это позволяет синхронно исправлять ошибки, возникающие в процессе передачи этой информации в каналах, как правило, насыщенных разного рода шумами и помехами.

Самый массовый вид ТКС – системы мобильной связи. На сегодняшний день они обеспечивают исправление двойных ошибок на блок передаваемой информации. Потребности увеличения информационных потоков и их скорости сталкиваются с необходимостью исправлять ошибки кратностью, большей двух. При любом методе коррекции ошибок – будь то метод алгебраических уравнений [1], перестановочные норменные методы [2], или иные – знание веса возникшей ошибки делают процедуру ее исправления более целенаправленной.

Единственным и исчерпывающим свидетельством наличия ошибки в принятом блоке-сообщении является ее синдром, непременно вычисляемый на входе каждого приемного устройства ТКС. Данная работа посвящена методикам определения кратности ошибки по ее синдрому в БЧХ-кодах, конструктивно рассчитанных на коррекцию 4-кратных ошибок.

Краткие сведения о БЧХ-кодах

Пусть C_9 – примитивный БЧХ-код длиной $n = 2^m - 1$ с проверочной матрицей $H = (\alpha^i, \alpha^{3i}, \alpha^{5i}, \alpha^{7i})^T$, двоичной, порядка $4m \times n$, для примитивного элемента α поля Галуа $GF(2^m)$ характеристики 2 из 2^m элементов [2, 3]. Такой код имеет конструктивное расстояние 9 (то есть минимальный вес ненулевых кодовых слов равен 9) и потому гарантированно исправляет ошибки весом ϖ, $1 \le \varpi \le 4$. Реальное минимальное расстояние d таких кодов может быть и большим. Так, у кода C_9 длиной 31 $d = 11$ ([3], с. 204).

Пусть $\bar{e}$ – вектор-ошибка, который наложился на очередной блок-сообщение $\bar{c}$ – двоичный вектор длиной n (т.е. с n координатами). Тогда на приемном конце ТКС будет принят вектор $\bar{x} = \bar{c} + \bar{e}$. О наличии ошибки в этом сообщении свидетельствует ее синдром

$S(\bar{x}) = H \cdot \bar{x}^T = S(\bar{e}) = H \cdot \bar{e}^T$, равный сумме столбцов матрицы H, номера которых совпадают с номерами ненулевых координат вектора $\bar{e}$. В силу структуры матрицы H синдром $S(\bar{e}) = H \cdot \bar{e}^T = (s_1, s_2, s_3, s_4)^T$, где компоненты синдрома s_1, s_2, s_3, s_4 являются некоторыми элементами поля Галуа $GF(2^m)$.

Синдромы всех ошибок весом ϖ, $1 \le \varpi \le 4$, попарно различны в коде C_9. Это обстоятельство является основой синдромных методов коррекции ошибок, например, метода алгебраических уравнений. Однако для высокоскоростных систем связи, в частности, для мобильных, такой метод не подходит – слишком медленный. Не спасает и метод Берлекэмпа-Месси [3, 4] решения уравнений степени 4 и выше. А если реальное минимальное расстояние больше конструктивного, то метод уравнений для коррекции ошибок кратностью, большей 4, и вовсе неприменим.

Теория норм синдромов (ТНС) в возникшей ситуации является единственным реальным и конструктивным средством коррекции многократных ошибок. Правда, адаптация ее к конкретным условиям требует отдельной серьезной работы. Первым и весьма желательным шагом здесь, как и в случае любого синдромного метода, является определение веса возникшей ошибки.

Свойства синдромов однократных и двукратных ошибок

Из сказанного выше следует, что вектор-ошибка $\bar{e}$ имеет вес 1 тогда и только тогда, когда ее синдром $S(\bar{e}) = H \cdot \bar{e}^T = (s_1, s_2, s_3, s_4)^T$ совпадает с одним из столбцов матрицы H, т.е. тогда и только тогда, когда

$$s_2 = s_1^3;\ s_3 = s_1^5;\ s_4 = s_1^7. \qquad (1)$$

Для двукратных ошибок имеет место следующая зависимость между компонентами их синдромов.

Теорема 1. *Пусть $\bar{e} = (i, j)$ – ошибка веса 2 на неизвестных позициях i и j. Пусть $S(\bar{e}) = (s_1, s_2, s_3, s_4)^T$ – ее синдром. Тогда $s_1 \neq 0$, а компоненты s_3 и s_4 алгебраически выражаются через первые две компоненты синдрома по формулам*:

$$\begin{aligned} s_3 &= s_1^5 + s_1^2 s_2 + \frac{s_2^2}{s_1}, \\ s_4 &= s_1^7 + s_1 s_2^2 + \frac{s_2^3}{s_1^2}. \end{aligned} \qquad (2)$$

Доказательство. Введем обозначения: $x = \alpha^{i-1}, y = \alpha^{j-1}$

Пусть $s_1 = 0$. Тогда $x + y = 0$. Отсюда $x = y$ и $i = j$. Однако это невозможно.

Для ошибки $\bar{e} = (i, j)$ выполняется: $x + y = s_1$, $x^3 + y^3 = s_2$, $x^5 + y^5 = s_3$, $x^7 + y^7 = s_4$. Подставляя выражения для компонент синдрома $S(\bar{e})$ в (2), получим:

$$x^5 + y^5 = (x+y)^5 + (x+y)^2 (x^3 + y^3) + \frac{(x^3 + y^3)^2}{x+y},$$

$$x^7 + y^7 = (x+y)^7 + (x+y)(x^3 + y^3)^2 + \frac{(x^3 + y^3)^3}{(x+y)^2}.$$

Преобразуем правые части полученных равенств:

$$(x+y)^5+(x+y)^2(x^3+y^3)+\frac{(x^3+y^3)^2}{x+y}=(x+y)^2\left((x+y)^3+(x^3+y^3)\right)+\frac{(x+y)^2(x^2+xy+y^2)^2}{x+y}=$$

$$=(x+y)^2(x^3+x^2y+xy^2+y^3+x^3+y^3)+(x+y)(x^2+xy+y^2)^2=$$

$$=(x+y)^2(x^2y+xy^2)+(x+y)(x^2+xy+y^2)^2=(x+y)\left((x+y)(x^2y+xy^2)+(x^2+xy+y^2)^2\right)=$$

$$=(x+y)(x^3y+x^2y^2+x^2y^2+xy^3+x^4+x^2y^2+y^4)=(x+y)(x^3y+xy^3+x^4+x^2y^2+y^4)=$$

$$=x^4y+x^2y^3+x^5+x^3y^2+xy^4+x^3y^2+xy^4+x^4y+x^2y^3+y^5=x^5+y^5,$$

$$(x+y)^7+(x+y)(x^3+y^3)^2+\frac{(x^3+y^3)^3}{(x+y)^2}=(x+y)^7+(x+y)^3(x^2+xy+y^2)^2+$$

$$+\frac{(x+y)^3(x^2+xy+y^2)^3}{(x+y)^2}=(x+y)^3\left((x+y)^4+(x^2+xy+y^2)^2\right)+(x+y)(x^2+xy+y^2)^3=$$

$$=(x+y)^3(x^4+y^4+x^4+x^2y^2+y^4)+(x+y)(x^2+xy+y^2)^3=(x+y)^3x^2y^2+(x+y)(x^2+xy+y^2)^3=$$

$$=(x+y)\left((x+y)^2x^2y^2+(x^2+xy+y^2)^2(x^2+xy+y^2)\right)=$$

$$=(x+y)\left((x^2+y^2)x^2y^2+(x^4+x^2+y^2+y^4)(x^2+xy+y^2)\right)=$$

$$=(x+y)(x^4y^2+x^2y^4+x^6+x^5y+x^4y^2+x^4y^2+x^3y^3+x^2y^4+x^2y^4+xy^5+y^6)=$$

$$=(x+y)(x^6+x^5y+x^4y^2+x^3y^3+x^2y^4+xy^5+y^6)=x^7+x^6y+x^5y^2+x^4y^3+x^3y^4+x^2y^5+xy^6+$$

$$+x^6y+x^5y^2+x^4y^3+x^3y^4+x^2y^5+xy^6+y^7=x^7+y^7.$$

Доказательство завершено.

Теорема 2. *Пусть* $S(\bar{e})=(s_1,s_2,s_3,s_4)^T$ *– синдром ошибки* $\bar{e}$ *веса* ϖ, $1\le\varpi\le4$, $s_1\neq0$, *и выполняется первое из соотношений* (2). *Тогда* $\varpi=1$ *или* $\varpi=2$.

Доказательство. Отметим, что если $\bar{e}$ – ошибка веса 1, то для ее синдрома выполняется (1). Подставив $s_2=s_1^3, s_3,=s_1^5, s_4=s_1^7$ в (2), получим тождества.

Предположим, что $\bar{e}=(i,j,k)$ – ошибка веса 3. Тогда $\alpha^{i-1}+\alpha^{j-1}+\alpha^{k-1}=s_1$, $\alpha^{3(i-1)}+\alpha^{3(j-1)}+\alpha^{3(k-1)}=s_3$, $\alpha^{5(i-1)}+\alpha^{5(j-1)}+\alpha^{5(k-1)}=s_3$. Введем обозначения: $x=\alpha^{i-1}$, $y=\alpha^{j-1}$, $z=\alpha^{k-1}$. Подставляя выражения для s_1,s_2,s_3 в первую формулу в (2), в новых переменных получим: $\frac{xyz(x+y)(x+z)(y+z)}{x+y+z}=0$. Из последнего равенства следует выполнение одного из следующих условий: $x=0$, $y=0$, $z=0$, $x=y$, $x=z$, $y=z$. Однако это невозможно.

Пусть теперь $\bar{e}=(i,j,k,u)$ – ошибка веса 4. Тогда: $\alpha^{i-1}+\alpha^{j-1}+\alpha^{k-1}+\alpha^{u-1}=s_1$, $\alpha^{3(i-1)}+\alpha^{3(j-1)}+\alpha^{3(k-1)}+\alpha^{3(u-1)}=s_2$, $\alpha^{5(i-1)}+\alpha^{5(j-1)}+\alpha^{5(ik-1)}+\alpha^{5(u-1)}=s_3$. Введем обозначения $x=\alpha^{i-1}$, $y=\alpha^{j-1}$, $z=\alpha^{k-1}$, $w=\alpha^{u-1}$. Подставляя выражения для s_1,s_2,s_3 в первую формулу в (2), в новых переменных получим: $\frac{(x+y)(x+z)(y+z)(x+w)(y+w)(z+w)}{x+y+z+w}=0$. Из последнего равенства следует выполнение одного из следующих условий: $x=y$, $x=z$, $y=z$, $x=w$, $y=w$, $z=w$. Однако это невозможно. Доказательство завершено.

Свойства синдромов трехкратных ошибок

Пусть $\sigma_1=x_1+x_2+x_3$, $\sigma_2=x_1x_2+x_1x_3+...+x_{n-1}x_n$, …, $\sigma_3=x_1x_2...x_n$ – элементарные симметрические многочлены от n переменных. Для степенных сумм $f_k=x_1^k+x_2^k+x_3^k$, $k=1,2,...,$ Ньютоном были установлены следующие реккурентные формулы:

$$f_k - f_{k-1}\sigma_1 + f_{k-2}\sigma_2 - ... + (-1)^{k-1} f_1\sigma_{k-1} + (-1)^k k\sigma_k = 0, k \le n, \quad (3)$$

$$f_k - f_{k-1}\sigma_1 + f_{k-2}\sigma_2 - ... + (-1)^n f_{k-n}\sigma_n = 0, k > n. \quad (4)$$

Очевидно,

$$f_1 = \sigma_1. \quad (5)$$

Рассмотрим формулы (3) и (4) в поле $GF(2^p)$ при $n = 3$. Если $k = 2$, то формула (3) имеет вид: $f_2 = f_1\sigma_1$, откуда $f_2 = \sigma_1^2$. При $k = 3$ из (3) следует: $f_3 + f_2\sigma_1 + f_1\sigma_2 + \sigma_3 = 0$. Подстановкой в это уравнение найденных выражений для f_1 и f_2 получаем:

$$\sigma_3 = f_3 + \sigma_1^3 + \sigma_1\sigma_2. \quad (6)$$

Если $k = 4$, то формула (4) имеет вид: $f_4 + f_3\sigma_1 + f_2\sigma_2 + f_1\sigma_3 = 0$. Подставляя в это уравнение выражения для f_1, f_2 и σ_3, получаем: $f_4 = \sigma_1^4$. При $k = 5$ из (4) следует: $f_5 + f_4\sigma_1 + f_3\sigma_2 + f_2\sigma_3 = 0$. Подстановкой в это уравнение найденных выражений для f_1, f_2, f_4 и σ_3, получаем: $\sigma_2(f_3 + \sigma_1^3) + f_5 + \sigma_1^2 f_3 = 0$. Если $f_3 + \sigma_1^3 \ne 0$, то из последнего равенства следует:

$$\sigma_2 = \frac{f_5 + \sigma_1^2 f_3}{f_3 + \sigma_1^3}. \quad (7)$$

Если $k = 6$, то формула (4) имеет вид: $f_6 + f_5\sigma_1 + f_4\sigma_2 + f_3\sigma_3 = 0$. Подставляя в это уравнение выражения для f_1, f_2, f_4, σ_2, σ_3, получаем: $f_6 = f_3^2$. При $k = 7$ из (4) следует: $f_7 + f_6\sigma_1 + f_5\sigma_2 + f_4\sigma_3 = 0$. Подстановкой в это уравнение найденных выражений для f_4, f_6 получаем:

$$f_7 = f_3^2\sigma_1 + f_5\sigma_2 + \sigma_1^4\sigma_3. \quad (8)$$

Теорема 3. *Пусть* $\bar{e} = (i, j, k)$ *– ошибка веса* 3 *на неизвестных позициях* i, j, k, $S(\bar{e}) = (s_1, s_2, s_3, s_4)^T$ *– ее синдром. Тогда* $s_1^3 + s_2 \ne 0$ *и*

$$s_4 = \sigma_1 s_2^2 + \sigma_2 s_3 + \sigma_1^4\sigma_3, \quad (9)$$

где

$$\sigma_1 = s_1, \sigma_2 = \frac{\sigma_1^2 s_2 + s_3}{\sigma_1^3 + s_2}, \sigma_3 = \sigma_1^3 + s_2 + \sigma_1\sigma_2. \quad (10)$$

Доказательство. Пусть для ошибки $\bar{e} = (i, j, k)$ веса 3 выполняется $s_1^3 + s_2 = 0$. Введем обозначения: $x = \alpha^{i-1}, y = \alpha^{j-1}, z = \alpha^{k-1}$. В новых переменных равенство $s_1^3 + s_2 = 0$ перепишется в виде: $(x + y + z)^3 + x^3 + y^3 + z^3 = 0$. Раскрывая скобки в левой части равенства, получим:

$$(x^2 + y^2 + z^2)(x + y + z) + x^3 + y^3 + z^3 = x^3 + x^2y + x^2z + xy^2 + y^3 + y^2z + xz^2 +$$
$$+ yz^2 + z^3 + x^3 + y^3 + z^3 = x^2y + x^2z + xy^2 + y^2z + xz^2 + yz^2 =$$
$$= x^2(y + z) + x(y^2 + z^2) + yz(y + z) = x^2(y + z) + x(y + z)^2 + yz(y + z) =$$
$$= (y + z)\left(x^2 + x(y + z) + yz\right) = (y + z)\left(x^2 + xy + xz + yz\right) = (y + z)\left(x(x + y) + z(x + y)\right) =$$
$$= (y + z)(x + y)\left(x + z\right).$$

Из равенства $(y + z)(x + y)\left(x + z\right) = 0$ следует выполнение одного из следующих условий: $x = y, x = z, y = z$. Однако это невозможно.

Для компонент синдрома $S(\bar{e})$ выполняется: $s_1 = x + y + z = f_1$, $s_2 = x^3 + y^3 + z^3 = f_3$, $s_3 = x^5 + y^5 + z^5 = f_5$, $s_4 = x^7 + y^7 + z^7 = f_7$, где f_1, f_3, f_5, f_7 – степенные суммы трех переменных x, y, z. Подставляя в (5)-(7) вместо f_1, f_3, f_5 соответственно s_1, s_2, s_3, получим (10). Подставляя в (8) вместо f_3, f_5, f_7 соответственно s_2, s_3, s_4, получим (9). Доказательство завершено.

Теорема 4. *Пусть* $S(\bar{e}) = (s_1, s_2, s_3, s_4)^T$ – *синдром ошибки* $\bar{e}$ *веса* ϖ, $1 \le \varpi \le 4$, $s_1^3 + s_2 \neq 0$, *и выполняется соотношение* (9), *где* $\sigma_1, \sigma_2, \sigma_3$ *определяются по формулам* (10). *Тогда* $\varpi = 2$ *или* $\varpi = 3$.

Доказательство. Из (1) следует, что для ошибки веса 1 выполняется $s_1^3 + s_2 = 0$.

Пусть $\bar{e} = (i, j)$ – ошибка веса 2. Введем обозначения: $x = \alpha^{i-1}, y = \alpha^{j-1}$. Тогда $x + y = s_1$, $x^3 + y^3 = s_2$, $x^5 + y^5 = s_3$, $x^7 + y^7 = s_4$. Подставив выражения для s_1, s_2, s_3, s_4 в (10), найдем: $\sigma_1 = x + y, \sigma_2 = xy, \sigma_3 = 0$. Подставляя полученные выражения в (9), получим: $s_4 = x^7 + y^7$ – тождество.

Если $\bar{e}$ – ошибка веса 3, то соотношение (4) выполняются в силу теоремы 4.

Пусть $\bar{e} = (i, j, k, u)$ – ошибка веса 4. Введем обозначения: $x = \alpha^{i-1}, y = \alpha^{j-1}, z = \alpha^{k-1}, v = \alpha^{u-1}$. Тогда $x + y + z + v = s_1$, $x^3 + y^3 + z^3 + v^3 = s_2$, $x^5 + y + z^5 + v^5 = s_3$, $x^7 + y^7 + z^7 + v^7 = s_4$. Подставляя выражения для s_1, s_2, s_3, s_4 в (10), найдем $\sigma_1, \sigma_2, \sigma_3$. Подставляя найденные выражения в (9), получим $\dfrac{xyzv(x+y)(x+z)(y+z)(x+v)(y+v)(z+v)}{s_1^3 + s_2} = 0$. Из последнего равенства следует выполнение одного из следующих условий: $x = 0$, $y = 0$, $z = 0$, $v = 0$, $x = y$, $x = z, y = z, x = v, y = v, z = v$. Однако это невозможно. Доказательство завершено.

Свойства синдромов с нулевыми компонентами

Теорема 5. *В коде* C_9 *не существует ошибки* $\bar{e}$ *веса* ϖ, $1 \le \varpi \le 4$, *синдром* $S(\bar{e})$ *которой был бы равен* $S(\bar{e}) = (0, 0, s_3, s_4)^T$ *для некоторых* $s_3, s_4 \in GF(2^m)$.

Доказательство. Пусть синдром $S(\bar{e})$ ошибки $\bar{e}$ веса ϖ, $1 \le \varpi \le 4$, равен $S(e) = (s_1, s_2, s_3, s_4)^T$.

При $\varpi = 1$ равенство $s_1 = 0$ невозможно. При $\varpi = 2$ равенство $s_1 = 0$ невозможно в силу теоремы 2. Если $\bar{e}$ – ошибка веса 3, для которой $s_1 = 0$, то в силу теоремы 4 $s_2 \neq 0$.

Пусть $\bar{e} = (i, j, k, u)$ – ошибка веса 4, для которой $s_1 = s_2 = 0$. Введем обозначения: $x = \alpha^{i-1}, y = \alpha^{j-1}, z = \alpha^{k-1}, v = \alpha^{u-1}$. Из равенств $s_1 = s_2 = 0$ следует: $x + y + z + v = 0$, $x^3 + y^3 + z^3 + v^3 = 0$. Из первого равенства следует: $v = x + y + z$. Подставляя это выражение во второе равенство, найдем: $x^3 + y^3 + z^3 + (x + y + z)^3 = 0$. Преобразуем левую часть полученного соотношения:

$$x^3 + y^3 + z^3 + (x + y + z)^3 = x^3 + y^3 + z^3 + (x + y + z)(x + y + z)^2 =$$
$$= x^3 + y^3 + z^3 + (x + y + z)(x^2 + y^2 + z^2) = x^3 + y^3 + z^3 + x^3 + xy^2 + xz^2 +$$
$$+ x^2 y + y^3 + yz^2 + x^2 z + y^2 z + z^3 = x(y^2 + z^2) + x^2(y + z) + yz(y + z) =$$
$$= x(y + z)^2 + x^2(y + z) + yz(y + z) = (y + z)(x(y + z) + x^2 + yz) =$$
$$= (y + z)(xy + xz + x^2 + yz) = (y + z)(xy + x^2 + xz + yz) =$$
$$= (y + z)(x(x + y) + z(x + y)) = (y + z)(x + y)(x + z).$$

Из условия $(y+z)(x+y)(x+z)=0$ следует выполнение одного из следующих равенств: $x=y, x=z, y=z.$ Однако это невозможно. Доказательство завершено.

Теорема 6. *Пусть* $S(\bar{e})=(0,s_2,s_3,s_4)^T$, $s_2\neq 0$, *– синдром ошибки* $\bar{e}$ *веса* ϖ, $1\le\varpi\le 4$. *Тогда* $\varpi=3$, *если выполняется соотношение* (9), *и* $\varpi=4$ *в противном случае.*

Доказательство. Из формулы (1) и теоремы 1 следует, что при $\varpi=1$ и $\varpi=2$ равенство $s_1=0$ невозможно. Из теорем 3 и 4 следует, что при $3\le\varpi\le 4$ равенство (9) имеет место лишь в случае, если $\varpi=3$. Доказательство завершено.

Теорема 7. *Пусть* $S(\bar{e})=(s_1,s_2,s_3,s_4)^T$, *где* $s_1^3+s_2=0$, *– синдром ошибки* $\bar{e}$ *веса* ϖ, $1\le\varpi\le 4$. *Тогда* $\varpi=1$, *если выполняется соотношение* (1), *и* $\varpi=4$ *в противном случае.*

Доказательство. Пусть $\varpi=2$ и $\bar{e}=(i,j)$. Тогда $\alpha^{i-1}+\alpha^{j-1}=s_1$, $\alpha^{3(i-1)}+\alpha^{3(j-1)}=s_2$. Равенство $s_1^3=s_2$ равносильно тому, что $\alpha^{i-1}\alpha^{j-1}(\alpha^{i-1}+\alpha^{j-1})=0$, т.е. $i=j$. Однако это невозможно.

В силу теоремы 3 равенство $s_1^3+s_2=0$ невозможно при $\varpi=3$. Следовательно, $\varpi=1$ или $\varpi=4$. Для завершения доказательства необходимо учесть, что (1) выполняются только при $\varpi=1$. Доказательство завершено.

Теоремы 1-7 служат обоснованием корректности следующего алгоритма определения в коде C_9 веса ϖ ошибки $\bar{e}$ по ее синдрому $S(\bar{e})=(s_1,s_2,s_3,s_4)^T$ в случае, когда $1\le\varpi\le 4$.

Алгоритм определения веса ϖ ошибки в коде C_9

1. Проверяем, выполняется ли равенства (1).
   1.1. Пусть (1) выполняется. Тогда $\varpi=1$ и алгоритм закончил работу.
   1.2. Пусть (1) не выполняется.
      1.2.1. Пусть $s_1\neq 0$. Проверяем, выполняется ли первое соотношение в (2).
         1.2.1.1. Пусть первое соотношение в (2) выполняется. Тогда $\varpi=2$ и алгоритм закончил работу.
         1.2.1.2. Пусть первое соотношение в (2) не выполняется.
            1.2.1.2.1. Пусть $s_1^3+s_2\neq 0$. Проверяем, выполняется ли соотношение (9).
               1.2.1.2.1.1. Пусть (9) выполняется. Тогда $\varpi=3$ и алгоритм закончил работу.
               1.2.1.2.1.2. Пусть (9) не выполняется. Тогда $\varpi=4$ и алгоритм закончил работу.
            1.2.1.2.2. Пусть $s_1^3+s_2=0$. Тогда $\varpi=4$ и алгоритм закончил работу.
      1.2.2. Пусть $s_1=0$. Проверяем, выполняется ли соотношение (9).
         1.2.2.1. Пусть (9) выполняется. Тогда $\varpi=3$ и алгоритм закончил работу.
         1.2.2.2. Пусть (9) не выполняется. Тогда $\varpi=4$ и алгоритм закончил работу.

Связь с методом определителей Блейхута

В [1] описан метод определения веса ошибки в БЧХ-коде на основании ее синдрома. Суть метода заключается в следующем.

Для БЧХ-кода C_{2t+1}, гарантированно исправляющего ошибки веса $\varpi, 1 \le \varpi \le t,$ и ошибки $\bar{e} = (i_1,...,i_\varpi)$ с синдромом $S(\bar{e}) = (s_1,\ldots,s_t)$ рассматривается матрица $M_\mu = \begin{pmatrix} S_1\ S_2 \ldots S_\mu \\ S_2\ S_3 \ldots S_{\mu+1} \\ \ldots \\ S_\mu\ S_{\mu+1} \ldots S_{2\mu-1} \end{pmatrix}$ порядка μ. Для двоичных БЧХ-кодов: $S_{2j} = S_j^2,\ S_{2j-1} = s_j,\ j \ge 1.$

Доказано (см. [1] с. 197), что $\det M_\mu \ne 0$, если μ равно весу ϖ произошедшей в действительности ошибки. Если же $\mu > \varpi$, то $\det M_\mu = 0$. При определении веса ϖ произошедшей в действительности ошибки поступают следующим образом. В качестве пробного значения берут $\varpi = t$ и вычисляют $\det M_\mu$. Если $\det M \ne 0$, то найдено правильное значение для ϖ. Если же $\det M_\mu = 0$, то уменьшают значение $\varpi = t$ на 1 и повторяют процедуру. Поступают таким образом до тех пор, пока не будет получен определитель, отличный от 0.

Рассмотрим код C_9. Предположим, что произошла ошибка $\bar{e} = (i_1,...,i_\varpi)$ веса $\varpi,$ $1 \le \varpi \le 4,$ с синдромом $s(\bar{e}) = (s_1, s_2, s_3, s_4)$. Тогда:

$$\det M_4 = \begin{vmatrix} s_1 & s_1^2 & s_2 & s_1^4 \\ s_1^2 & s_2 & s_1^4 & s_3 \\ s_2 & s_1^4 & s_3 & s_3^2 \\ s_1^4 & s_3 & s_3^2 & s_4 \end{vmatrix} =$$

$$= (s_1^6 + s_1^3 s_2 + s_2^2 + s_1 s_3)(s_1^{10} + s_1^7 s_2 + s_1 s_2^3 + s_1^5 s_3 + s_1^2 s_2 s_3 + s_3^2 + (s_1^3 + s_2) s_4);$$

$$\det M_3 = \begin{vmatrix} s_1 & s_1^2 & s_2 \\ s_1^2 & s_2 & s_1^4 \\ s_2 & s_1^4 & s_3 \end{vmatrix} = (s_1^3 + s_2)(s_1^6 + s_1^3 s_2 + s_2^2 + s_1 s_3);$$

$$\det M_2 = \begin{vmatrix} s_1 & s_1^2 \\ s_1^2 & s_2 \end{vmatrix} = s_1(s_1^3 + s_2);$$

$$\det M_1 = |s_1| = s_1.$$

В соответствии с методом определителей Блейхута:

- если выполняется (1), то $\det M_1 = s_1 \ne 0, \det M_2 = \det M_3 = \det M_4 = 0$ и $\varpi = 1$;

- если (1) не выполняется, $s_1 \ne 0$ и имеет место первое соотношение из (3), то $\det M_2 = s_1(s_1^3 + s_2) \ne 0, \det M_3 = \det M_4 = 0$ и $\varpi = 2$;

- если $s_1 \ne 0$, (1) и первое соотношение из (3) не выполняются, $s_1^3 + s_2 \ne 0$ и справедливо (4), тогда $\det M_3 = s_1(s_1^3 + s_2)(s_1^5 + s_1^2 s_2 + \frac{s_2^2}{s_1} + s_3) \ne 0, \det M_4 = 0$ и $\varpi = 3$;

- если $s_1 \ne 0$, (1) и первое соотношение из (3) не выполняются, $s_1^3 + s_2 \ne 0$ и равенство (4) не имеет места, то $\det M_4 = s_1(s_1^3 + s_2)(s_1^5 + s_1^2 s_2 + \frac{s_2^2}{s_1} + s_3)(\sigma_1 s_2^2 + \sigma_2 s_3 + \sigma_3 s_1^4 + s_4) \ne 0$, где $\sigma_1, \sigma_2, \sigma_3$ задаются формулами (5), и $\varpi = 4$;

- если $s_1 \ne 0$, (1) и первое соотношение из (3) не выполняются и $s_1^3 + s_2 = 0$, то $\det M_4 = s_1(s_1^5 + s_3) \ne 0$ и $\varpi = 4$;

- если (1) не выполняется, $s_1 = 0$ и справедливо (4), то $\det M_3 = s_2^3 \neq 0$, $\det M_4 = 0$ и $\varpi = 3$;

- если $s_1 = 0$ и (1) и (4) не выполняются, то $\det M_4 = s_2^2(s_3^2 + s_2 s_4) \neq 0$ и $\varpi = 4$.

Таким образом, алгоритм определения веса ошибки в коде C_9, изложенный в предыдущем пункте, эквивалентен методу определителей Блейхута.

Заключение

Определители Блейхута дают возможность определения кратности ошибки в коде. Но их применение требует введения дополнительных параметров и достаточно громоздкой процедуры вычисления самих определителей. Предложенный в работе алгоритм является прямым и непосредственным средством быстрого определения кратности ошибок ограниченного веса.

DETECTION OF ERROR WEIGHT IN THE PRIMITIVE BCH-CODE

N.V. SPICHEKOVA, V.A. LIPNITSKI

Abstract

The properties of the syndromes for errors of weight 1-4 in the primitive BCH-code C_9 are investigated. An algorithm to determine the multiplicity of an error based on its syndrome is described. The relationship between the proposed algorithm and Blahut's method of determinants is shown.

Список литературы

1. *Блейхут Р.* Теория и практика кодов, контролирующих ошибки. М., 1986.
2. *Липницкий В.А.*, *Конопелько В.К.* Норменное декодирование помехоустойчивых кодов и алгебраические уравнения. Мн., 2007.
3. *Мак-Вильямс Ф.Дж.*, *Вильямс Ф.Дж., Слоэн Н.Дж.А.* Теория кодов, исправляющих ошибки. М., 1979.
4. *Лидл Р., Нидеррайтер Г.* Конечные поля. М., 1988.

УДК 621.391

СЕГМЕНТАЦИЯ ТЕКСТУРНЫХ ИЗОБРАЖЕНИЙ НА ОСНОВЕ КЛАССИФИКАЦИИ КОНТУРНЫХ ЭЛЕМЕНТОВ И ЛОГИЧЕСКОГО СЛОЖЕНИЯ КЛАССОВ

Х.М. АЛЬЗАКИ, В.Ю. ЦВЕТКОВ, В.К. КОНОПЕЛЬКО

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

Поступила в редакцию 30 сентября 2015

Предложен метод сегментации текстурных изображений на основе классификации контурных элементов и логического сложения классов. Сущность метода состоит в контурной обработке исходного изображения, определении положения на изображении контурных элементов различного типа (точек, линий и фигур), преобразовании близко расположенных друг к другу однотипных контурных элементов в бинарные площадные объекты, двоичном кодировании взаимного расположения полученных площадных объектов в границах исходного изображения, сегментации полученной кодовой матрицы.

Ключевые слова: текстурная сегментация изображений, классификация контурных элементов.

Введение

В связи с повышением разрешения видеокамер [1], на формируемых с их помощью изображениях увеличивается количество текстурных областей. Из-за высокочастотного характера текстур, коэффициенты сжатия содержащих их изображений значительно меньше коэффициентов сжатия изображений, содержащих преимущественно однородные по яркости области и малое число резких перепадов яркости [2]. В связи с этим актуальной является проблема повышения эффективности сжатия текстурных изображений, решение которой требует, прежде всего, локализации текстурных областей. Это, в свою очередь, делает актуальной задачу текстурной сегментации, которая близка задаче текстурного анализа. Если в результате текстурного анализа необходимо определить тип (класс) текстуры или различить несколько текстур [3, 4], то в результате сегментации необходимо найти границу разделения текстурных областей [5]. Несмотря на данное различие, для текстурного анализа и текстурной сегментации используются общие методы [6-11]. Среди них – энергетические карты [11], которые широко используются в настоящее время в текстурном анализе и могут применяться для текстурной сегментации. Их недостатком является высокая вычислительная сложность и разделение простых текстур с высокой степенью однородности. В ряде работ [12] предлагается использовать контурные элементы изображений для текстурного анализа. Однако данные подходы основаны на небольшом наборе контурных примитивов и достаточно грубой статистической оценке, что приводит к значительным ошибкам разделения сложных текстур.

Целью работы является разработка метода текстурной сегментации изображений, имеющего относительно низкую вычислительную сложность и возможность управления чувствительностью и избирательностью для адаптации к сложным по характеру текстурам.

Текстурная сегментация изображений на основе энергетических карт

Широко распространение для текстурного анализа получил метод энергетических карт [11], основанный на фильтрации изображений, взвешивании и суммировании полученных ре-

зультатов. Сущность метода состоит в использовании фильтров для выделения крупных объектов с плавным изменением яркости, плавных контурных линий, ломаных контурных линий, контурных точек и небольших пятен с постоянной или плавно меняющейся яркостью; формировании в результате бинарные матриц, единичные элементы которых говорят о наличии в соответствующих позициях определенных объектов; взвешивании полученных матриц и их поэлементном суммировании, в результате чего формируется энергетическая карта (рис. 1).

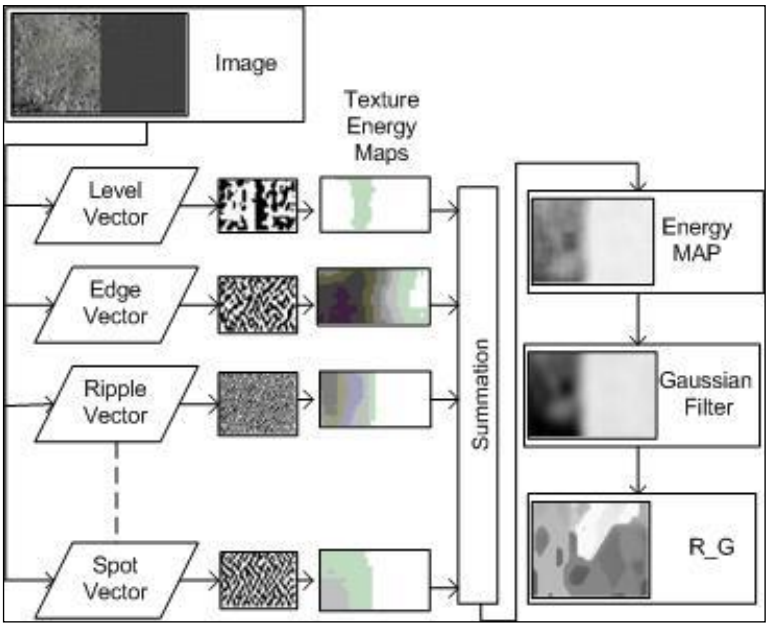


Рис. 1. Схема текстурной сегментации изображений на основе энергетических карт

Энергетическая карта содержит в основном области с плавным изменением яркости, которые могут быть разделены с помощью известных методов сегментации изображений, например выращивания областей [13]. Для улучшения качества текстурной сегментации (устранения влияния на результаты высокочастотного шума и присоединения мелких площадных объектов к близким по яркости крупным площадным объектам) перед применением метода выращивания областей энергетическая карта подвергается низкочастотной фильтрации с использованием фильтра Гаусса.

Метод энергетических карт не определяет строго типы фильтров, которые могут использоваться для выделения элементов изображений, что теоретически делает пригодным данный метод для определения произвольных текстур. Однако объединение результатов фильтрации в энергетическую карту может приводить к ошибкам сегментации, особенно сложных текстур, из-за отсутствия ранжирования результатов фильтрации по значимости. Устранение данного недостатка возможно за счет: а) использования контурных элементов изображений для текстурного анализа, как наиболее однородных и, следовательно, равнозначных для последующего объединения результатов; б) четкой классификации контурных элементов по структуре для формирования универсального аппарата для описания как простых, так и сложных текстур; в) логического объединения результатов фильтрации.

Метод текстурной сегментации изображений на основе классификации контурных элементов и логического сложения классов

Предлагается метод текстурной сегментации изображений на основе классификации контурных элементов и логического сложения классов. Сущность метода состоит в контурной обработке исходного изображения, определении положения на изображении контурных элементов (точек, линий и фигур) различного типа, преобразовании близко расположенных друг к другу однотипных контурных элементов в бинарные площадные объекты, двоичном кодировании взаимного расположения полученных площадных объектов в границах исходного изображения, сегментации полученной кодовой матрицы (рис. 2).

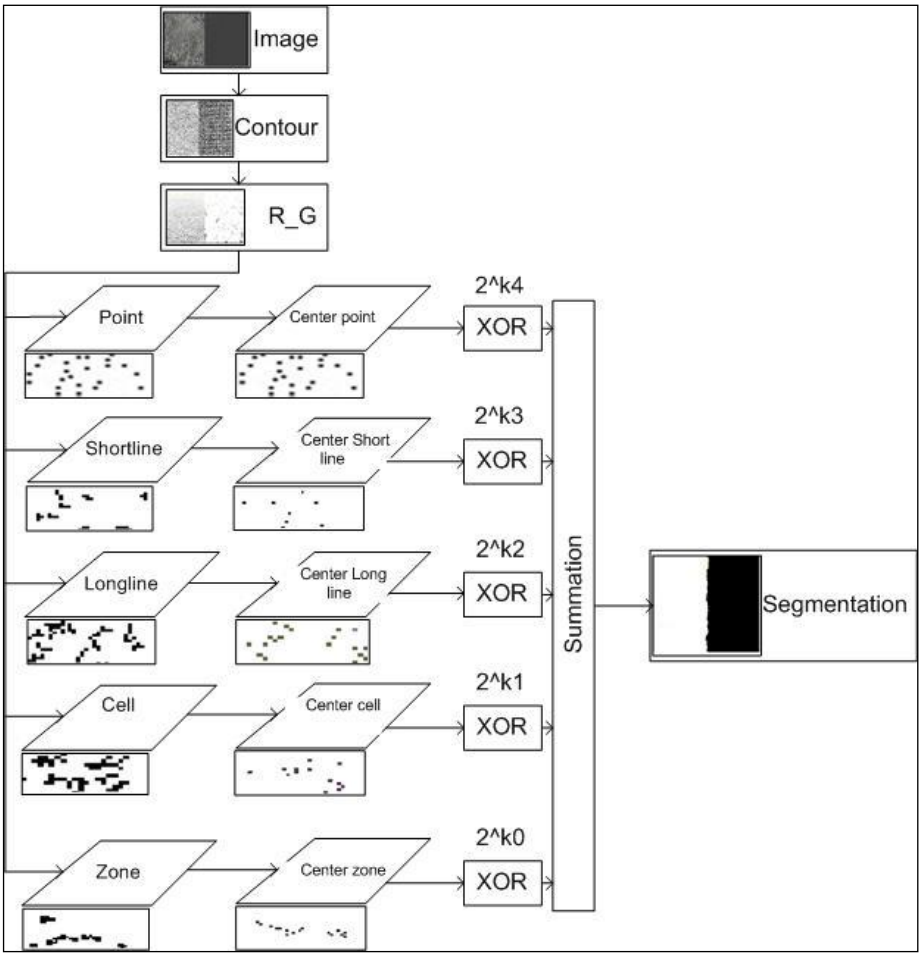


Рис. 2. Схема текстурной сегментации изображений на основе классификации контурных элементов и логического сложения классов

Алгоритм текстурной сегментации изображений на основе классификации контурных элементов и логического сложения классов состоит из следующих шагов.

Шаг 1. Контурная фильтрация изображения. На данном шаге могут использоваться любые фильтры. Для достижения высокой скорости обработки при достаточно высоком качестве выделения контурных линий предлагается использовать объединение результатов двух контурных фильтров: Робертса [14] и Превита [15], выделяющих перепады яркости по диагоналям, горизонтали и вертикали. В результате формируется бинарная матрица, в которой единичным элементам соответствуют изолированные контурные точки, образующие линии различной сложности и длины, небольшие площадные объекты.

Шаг 2. Сегментация контурных элементов изображения. Для выполнения данного шага алгоритма может использоваться любой метод сегментации. Наиболее эффективным в данном случае является метод выращивания областей, который обеспечивает высокое быстродействие и точность. В результате выполнения данного шага формируется матрица сегментации, в которой фоновым элементам соответствует нулевой номер сегмента, а остальным элементам – номера сегментов, которым они принадлежат.

Шаг 3. Классификация контурных элементов. На данном шаге алгоритма выделяются точки, состоящие из одного или двух пикселей; короткие линии, образованные двумя-тремя пикселями; длинные линии, имеющие две концевые точки и образованные четырьмя и более пикселями (предельное число пикселей в длиной линии может быть определено в результате обработки гистограммы длин линий с учетом размера обрабатываемого изображения); «кляксы», образованные пересечением нескольких контурных линий; «пятна» – небольшие площадные объекты, не имеющие концевых точек; большие площадные объекты. Все идентифицированные контурные элементы, кроме больших площадных объектов, рассматриваются далее в качестве образующих текстур. В результате выполнения данного шага каждому контурному

элементу, имеющему свой номер, ставится в соответствие идентификатор (дескриптор), относящий его к определенному классу. В табл. 1 приведены идентификационные параметры, используемые для классификации контурных элементов, и примеры идентификаторов для контурных элементов различных классов.

Таблица 1. **Идентификационные параметры и идентификаторы контурных элементов**

Класс контурного элемента	**Значения идентификационных параметров**								
Точка	Длина от 1 до 4 пикселей								
Линия (длина от 5 до 64 пикселей)	Концевая точка	1	2	3	4	5	6	7	8
	2	1	7	3	0	0	0	0	0
	3	2	3	6	0	0	0	0	0
	2	1	2	3	1	0	0	0	0
Клякса	2	1	1	4	4	1	0	0	0
	1	1	0	5	7	0	0	0	0

Шаг 4. Разделение контурных элементов по классам. В результате выполнения данного шага алгоритма формируется множество битовых плоскостей, единичные элементы в каждой из которых показывают положение соответствующих контурных элементов определенного класса.

Шаг 5. Дискретизация контурных элементов. На данном шаге выполняется замена контурных элементов каждой битовой плоскости множеством равноудаленных друг от друга точек. Каждой точке и короткой линии ставится в соответствие одна точка. Длинные линии заменяются сериями точек; «кляксам» и «пятнам» ставятся в соответствие сетки точек. Расстояние между точками в сериях и сетках примерно одинаково. Это позволяет выровнять вклад контурных элементов различного типа в энергию формируемого на следующем шаге бинарного интегрального изображения.

Шаг 6. Формирование бинарного интегрального изображения. В результате выполнения данного шага каждая битовая плоскость, содержащая серии и сетки точек, определяющих положение контурных элементов определенного класса, подвергается низкочастотной фильтрации и пороговой обработке, в результате чего серии и сетки точек преобразуются в площадные объекты. Размер фильтра согласуется с выбранным при дискретизации расстоянием между точками в сериях и сетках. Порог выбирается таким образом, чтобы исключить выделение одиночных и далеко расположенных друг от друга дискретов.

Шаг 7. Логическое объединение классов. В результате выполнения данного шага осуществляется построение результирующего интегрального изображения после конкатенации полученных на предыдущем шаге бинарных интегральных изображений. В простейшем случае число битовых плоскостей в результирующем интегральном изображении совпадает с числом исходных бинарных интегральных изображений. При необходимости логического объединения контурных элементов некоторых классов (например, точек и коротких линий или «клякс» и «пятен») перед формированием результирующего интегрального изображения осуществляется логическое сложение соответствующих бинарных интегральных изображений.

Шаг 8. Сегментация результирующего интегрального изображения. Для этого может использоваться любой метод сегментации по яркости без порога и предварительного квантования. Наиболее эффективный вариант – использование метода выращивания областей. Не сегментируются только нулевые элементы, соответствующие фону. Им в матрице сегментации также присваиваются нулевые значения.

В результате выполнения данного алгоритма формируется матрица сегментации, в которой каждому элементу ставится в соответствие ноль (если ему соответствует не текстурный пиксель) или номер текстурного сегмента, которому принадлежит соответствующий текстурный пиксель.

Оценка эффективности методов текстурной сегментации

Для эффективности точности методов текстурной сегментации использованы известные базы тестовых текстурных изображений Brodatz, UIUCTex. С помощью этих баз составле-

ны тестовые изображения, в которых левая и правая половины образованы изображениями из этих баз.

В качестве критерия эффективности методов сегментации использовано значение нормированной ошибки, вычисляемое в результате учета числа элементов матрицы сегментации, не попадающих на соответствующие сегменты эталонного изображения. Эталонное изображение состоит из двух сегментов одинакового размера, покрывающих тестовые изображения.

На рис. 3 приведены тестовые изображения и результаты их разделения с помощью предложенного метода и метода на основе энергетических карт. В табл. 2 представлены значения ошибки сегментации для рассматриваемых методов на тестовых изображениях, приведенных на рис. 3.

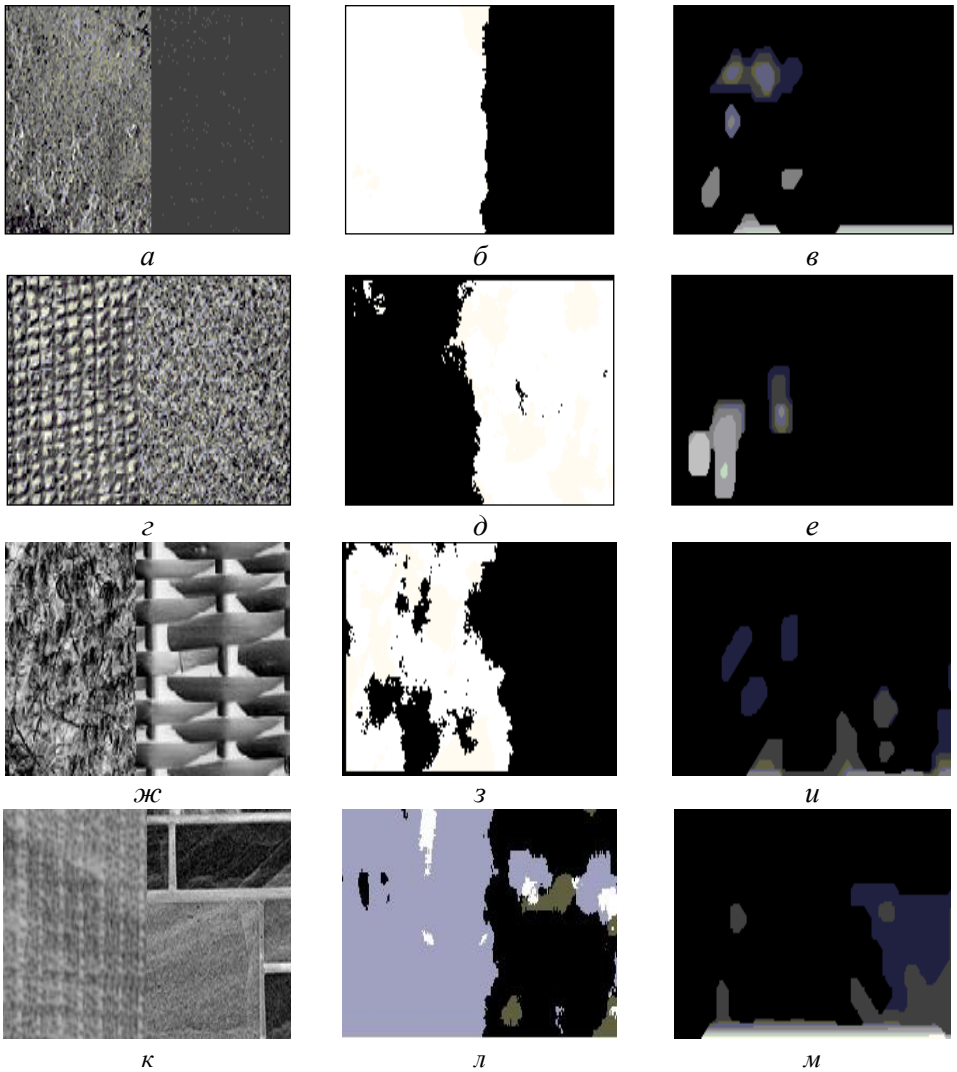


Рис. 3. Тестовые изображения и результаты текстурной сегментации: *а* – тестовое изображение Test 1; *б* – сегментация Test 1 предложенным методом; *в* – сегментация Test 1 с помощью энергетической карты; *г* – тестовое изображение Test 2; *д* – сегментация Test 2 предложенным методом; *е* – сегментация Test 2 с помощью энергетической карты; *ж* – тестовое изображение Test 3; *з* – сегментация Test 3 предложенным методом; *и* – сегментация Test 3 с помощью энергетической карты; *к* – тестовое изображение Test 4; *л* – сегментация Test 4 предложенным методом; *м* – сегментация Test 4 с помощью энергетической карты

Как следует из табл. 2, предложенный метод сегментации обеспечивает меньшую ошибку сегментации. Эксперименты над изображениями баз Brodatz, UIUCTex показали, что предложенный метод обеспечивает меньшую ошибку сегментации по сравнению с методом на основе энергетических карт для всех изображений. При этом средняя ошибка сегментации в 2,4 раза меньше по сравнению с методом на основе энергетических карт.

Таблица 2. **Значения ошибок текстурной сегментации тестовых изображений**

Метод	Изображение			
	Test 1	Test 2	Test 3	Test 4
Предложенный метод	0,1682	0,4634	0,4108	0,1746
Энергетические карты	0,6712	0,5116	0,5982	0,5549

Заключение

Предложен метод сегментации текстурных изображений на основе классификации контурных элементов и логического сложения классов. Показано, что предложенный метод по сравнению с методом на основе энергетических карт обеспечивает меньшую ошибку сегментации стандартных тестовых изображений, опубликованных в тестовых базах Brodatz, UIUCTex; уменьшение средней ошибки сегментации для этих изображений в 2,4 раза.

SEGMENTATION OF TEXTURE-BASED IMAGE CLASSIFICATION CONTOUR ELEMENTS AND LOGICAL ADDITION CLASSES

H.M. ALZAKKI, V.Yu. TSVIATKOU, V.K. KANAPELKA

Abstract

A method for texture segmentation of images based on classification of contour elements and logical addition of classes is offered. The essence of the method is concluded in the contouring of the original image, determining the position of the image of contour elements of different types (points, lines, and shapes), converting closely spaced similar contour elements into binary areal objects, binary coding mutual position obtained polygon objects within the boundaries of the original image segmentation resulting code matrix.

Список литературы

1. *Mastriani M.* // Engineering and Technology International Science Index 48. 2010. №4(12). P. 485-497.
2. *Ватолин Д., Ратушняк А., Смирнов М. и др.* Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. М., 2003.
3. *Materka A., Strzelecki M.* // Technical university of lodz, institute of electronics .1998. №11. P. 9-11.
4. *Tuceryan M., Jain A.K.* // The Handbook of Pattern Recognition and Computer Vision (2nd Edition). 1998. P. 207-248.
5. *Bhosle V.V., Pawar V.P.* // International Journal of Soft Computing and Engineering (IJSCE). 2013. Vol. 3. P. 69-74.
6. *Grigorescu S.E., Petkov N., Kruizinga P.* // IEEE Transactions On Image Processing. 2002. №10. P. 1160-1167.
7. *Reulke R., Lippok A.* // The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. 2008. Vol. 37. P. 615-620.
8. *Clausi D.A.* // Pattern Recognition. 2002. №35. P. 1959-1972.
9. *Liu X., Wang D.* // IEEE Transactions On Image Processing. 2006. Vol. 15. №10. P. 3066-3077.
10. *Kokkinos L., Evangelopoulos G., Maragos P.* // Actions On Pattern Analysis And Machine Intelligence. 2009. Vol. 31. №1. P. 142-157.
11. *Lee D-Ch., Shchenk T.* // A Collection of Papers Presented At the XVII Congress of ISPRS. 1992. №48. P. 75-80.
12. *Jithendra M., Serge B., Thomas L. et. al* // International Journal of Computer. 2001. №43(1). P. 7-27.
13. *Sharma Ritu, Sharma Rajesh* // International Journal of Innovative Research in Computer and Communication Engineering. 2014. Vol. 2. P. 5686-5692.
14. *Shrivakshan G.* // IJCSI International Journal of Computer Science Issues. 2012.Vol. 9. №1. P. 269-276.
15. *Rahman khan A., Thakur K.* // International Journal of Emerging Technology and Advanced Engineering. 2012. Vol. 2. P. 245-250.

УДК 621.391.14

АНАЛИЗ МЕТОДОВ ОРГАНИЗАЦИИ БЕЗЫЗБЫТОЧНОЙ СИНХРОНИЗАЦИИ КОДЕКОВ СВЕРТОЧНЫХ КОДОВ

А.Н. ДРАПЕЗА, Г.А. ВОЛОСТНЫХ, А.И. ПАРФЕНЮК, А.И. КОРОЛЁВ

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

Поступила в редакцию 21 октября 2015

Анализируются методы организации безызбыточной синхронизации распределителей символов кодовых последовательностей кодеков сверточных кодов (СК) с алгоритмом порогового декодирования (ПД). Оцениваются эффективность методов и устройств организации ветвевой (цикловой) синхронизации кодеков СК на основе использования структуры символов синдромной последовательности и символов М-последовательности, накладываемых как на проверочные символы, так и на проверочные и информационные символы. Установлено, что устройства ветвевой синхронизации (УВС), организуемые на основе анализа структуры символов синдромной последовательности, обеспечивают минимальное время вхождения в синхронизм распределителей символов кодовых последовательностей кодека по сравнению с УВС, организуемых на основе М-последовательностей. Однако УВС на основе М-последовательностей обеспечивают на порядок и более высокую помехоустойчивость функционирования работы УВС. Определены методы (направления) повышения эффективности функционирования УВС-кодеков СК в каналах с группирующимися ошибками.

Ключевые слова: сверточный код, синдром, синдромная последовательность, скорость кода, М-последовательность, формирователь интервала анализа, ключ, порог, порождающая и проверочная матрицы.

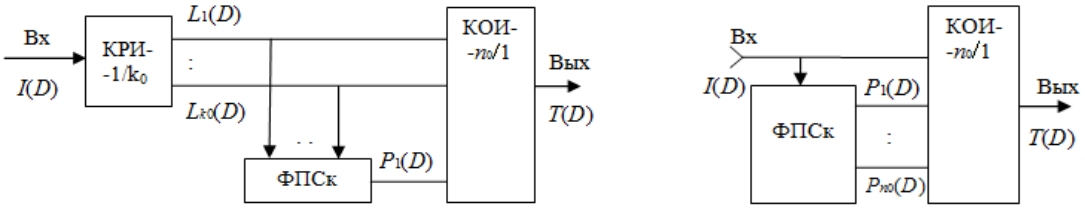
Сверточное кодирование данных

Сверточные коды (СК) с алгоритмом порогового декодирования (ПД) образуют широкий класс помехоустойчивых кодов, обеспечивающих высокую скорость декодирования и достоверность передачи данных [1-3]. На практике широкое применение получили высокоскоростные (малоизбыточные) самоортогональные сверточные коды (ССК), диффузные ССК (ДФССК) и низкоскоростные (высокоизбыточные) ССК или равномерные ССК (РССК). На рис. 1,*а*,*б* приведены структурные схемы канальных кодирующих устройств (кодеров) соответственно ССК (ДФССК) и РССК со скоростью передачи кодов: $R_{\text{ССК}} \geq k_0 / n_0$, $k_0 \geq 2$, $n_0 = k_0 + 1$ – ССК и ДФССК; $R_{\text{РССК}} \leq k_0 / n_0 = 1/n_0$, $n_0 \geq 3$ – РССК; сверточные коды со скоростью передачи кодов $R = k_0 / n_0 = 1/2$ в данной статье не рассматриваются.

При использовании ССК (ДФССК) с $R_{\text{ССК}} \geq k_0 / n_0$ и $k_0 \geq 2$ формируется один дополнительный подпоток проверочных символов $P_1(D)$, а при использовании РССК с $R_{\text{РССК}} \geq 1/n_0$ и $n_0 \geq 3$ дополнительно формируются $n_0 - 1$ подпотоков проверочных символов $P_1, ..., P_{n_0}(D)$. Сформированные символы проверочных подпотоков и информационные символы объединяются в поток кодовых символов $T(D)$. Формирование кодовых символов $T(D)$ осуществляется по правилу [1-3]:

$$T^{(i)}(D) = \sum_{j=1}^{K_0} I^{(j)}(D) \cdot G^{(j)}(D),\ j = 1, 2, ..., k_0,\ i = j + 1, \qquad (1)$$

где $G(D)$ – порождающий полином.



а *б*

Рис. 1. Структурные схемы канальных кодеров ССК (*а*) и РССК (*б*): $I(D)$ – передаваемый поток данных, ФПСк – формирователь проверочных символов кодера, КРИ-1 / k_0 – коммутатор распределения информации на $k_0 (k_0 \geq 2)$ параллельных подпотоков, КОИ-1 / k_0 – коммутатор объединения информации $n_0 (n_0 \geq 3)$ параллельных подпотоков

Из выражения (1) следует, что процесс формирования кодовых символов носит непрерывный характер. В канальном декодере принятые кодовые символы распределяются на n_0 параллельных подпотоков. Следовательно, достоверность передачи данных или вероятность ошибочного декодирования сверточных кодов будет существенно зависеть от синфазной работы коммутаторов объединения информации канального кодера (КОИ-n_0 / 1) и коммутатора распределения информации канального декодера (КРИ-n_0 / 1). Под ветвевой синхронизацией (ВС) канальных кодеков СК (СК, ДФССК, и РССК) с алгоритмом ПД понимают принудительное установление правильных фазовых соотношений между коммутаторами кодера (КОИ-n_0 / 1) и декодера (КРИ-n_0 / 1). Устройства, которые реализуют данную функциюю канального кодека дескретного канала связи, получили название устройств ветвевой синхронизации (УВС).

Классификация методов построения устройств ветвевой синхронизации канальных кодеков сверточных кодов с алгоритмом порогового декодирования

В соответствии с [4-7] наибольшее применение получили следующие методы построения УВС канальных кодеков сверточных кодов с алгоритмом порогового декодирования:

- на основе анализа структуры двоичных символов (битов) одной синдромной последовательности при использовании ССК и ДФССК или структур двоичных символов $n_0 - 1$ $(n_0 \geq 3)$ синдромных последовательностей – при использовании РССК;
- использование символов М-последовательности, накладываемых на проверочные символы одного подпотока при использовании ССК и ДФССК;
- использование символов М-последовательности, накладываемых на проверочные символы $n_0 - 1$ $(n_0 \geq 3)$ подпотоков при использовании РССК.

На основе данных методов разработано большое количество способов технической реализации УВС канальных кодеков сверточных кодов с алгоритмом порогового декодирования. Способы технической реализации УВС канальных кодеков СК с алгоритмом ПД существенно зависят от требований, предъявляемых к данным устройствам при обеспечении допустимой вероятности ошибочного декодирования используемых СК.

Далее выполним анализ качественных характеристик УВС канального кодека СК при использовании символов синдромной последовательности и определим направления повышения эффективности способов технической реализации УВС.

Анализ качественных характеристик УВС канальных кодеков при использовании символов синдромной последовательности

В соответствии с [1-3, 6, 7] следует, что при синхронной работе коммутаторов канального кодека и отсутствии ошибок в канале связи синдромная последовательность (СП) $S(D) = S_0, S_1, S_2, \ldots, S_N, \ldots$ будет последовательностью нулевых двоичных символов (битов).

Если количество ошибок на длине кодового ограничения $n_A = (m+1)\ n_0$ (m – максимальная степень порождающих полиномов СК) бит не превышает корректирующей способности СК ($t_{\text{исп}} \le J/2$, J – число ортогональных проверок), то на интервале $N = m+1$ синдромных символов, анализируемых для принятия решения о достоверности k_0 принятых информационных символов, число ненулевых синдромных символов ($t_{\text{нс}}$) может составлять $t'_{\text{нс}} = t_{\text{исп}}(J-1)$ или $t''_{\text{нс}} = t_{\text{исп}}(J-1)+1 = t_{\text{исп}} \cdot J$ – соответственно при четном и нечетном значении корректируемых ошибок, т.е. $t_{\text{исп}}$. При этом $t'_{\text{нс}}$ и $t''_{\text{нс}} << 0{,}5N$. Кроме того, при вероятности канальных ошибок $P_{\text{к}} = 5 \cdot 10^{-3} - 10^{-4}$ ошибочные кодовые ограничения n'_A будут разделены несколькими безошибочными кодовыми ограничениями n_A, которым будет соответствоввать нулевая СП.

При возникновении в канале связи пакетов ошибок число ненулевых символов в СП зависит от кратности (длины) или структуры ошибок в пакетах. Максимальное число ненулевых символов в СП наблюдается при кратности пакета ошибок равной длине миниблока n_0 СК и состовляет $t_{\text{нс.м.}} = n_0(J-1)+1 < 0{,}5 \cdot N$ символов. При дальнейшим увеличении кратности пакета число ненулевых символов в СП уменьшается. Это обеспечивается алгоритмом формирования ортогональных и самоортогональных проверок СК [6, 7].

Такая закономерность сохраняется при возникновении ошибок кратностью $t'_{\text{п}} = (m+1)/2$. При $t'_{\text{ош}} > t'_{\text{п}}$ число ненулевых символов вновь начинает увеличиваться и при $t'_{\text{ош}} \ge (m+1)$ ошибочных символов число ненулевых символов в СП составляет примерно $0{,}5 \cdot N$. Однако при $P_{\text{к}} = 5 \cdot 10^{-3} - 10^{-4}$ и $n_A = 100 - 500$ символов вероятность возникновения ошибок такой кратности является событием маловероятным, что подтверждается следующим выражением [6, 7]:

$$P(t' \ge (m+1)/2) = C_{n_A}^{(m+1)/2} \cdot (J \cdot P_{\kappa})^{(m+1)/2} \cdot (1 - J \cdot P_{\kappa})^{n_A - (m+1)/2}, \qquad (2)$$

где $P_{\text{к}}$ – вероятность ошибок по символам на выходе ДКС, J – число ортогональных проверок СК, обеспечивающих размножение (увеличение) ненулевых символов в СП при искажении одного информационного символа в канале связи.

Следовательно, если принять, что кодовая последовательность $T^{(i)}(D)$ СК является последовательностью независимых двоичных символов (битов) с равной вероятностью появления «1» и «0» ($P_1 = P_0 = 0{,}5$), то при отсутствии ВС коммутаторов сверточного кодека, сформированная СП $S(0) = S_0, S_1, \ldots, S_{N=m+1} \ldots$ также будет последовательностью независимых двоичных символов с равной вероятностью появления «1» и «0»: $P_1 = P_0 = 0{,}5$. Таким образом, число ненулевых символов в СП на интервале анализа (наблюдения) l_a будет примерно равным, т.е. $0{,}5 \cdot N$ символов. Точность равенства повышается с увеличением интервала анализа.

Сложная зависимость формирования символов СП наблюдается при переключении ветвевых подпотоков коммутаторов кодека, когда осуществляется поиск ВС коммутаторов кодека: при включении кодека в работу и при восстановлении ВС коммутаторов. Это определяется переходным процессом в ФПСд, так как каждый входной кодовый символ оказывает влияние на формирование синдромных символов в течение (m+1) тактов: m – количество ячеек памяти регистра сдвига (RG) ФПСд. С целью исключения из анализа символов СП, сформированных при переходном процессе в ФПСд, целесообразно ввести блокировку счетчика ненулевых символов СП устройства ветвевой синхронизации кодека. Однако это автоматически увеличивает время поиска ВС коммутаторов кодека. С целью снижения влияния данной блоки-

ровки на качественные характеристики УВС кодека длину интервала анализа целесообразно выбирать $l_a \square m$.

Проведенный анализ статистических свойств СП позволяет оптимизировать выбор величины порога «η» счетчика ненулевых символов СП, интервала анализа «l_a» УВС сверточного кодека и ввести следующую методику определения вероятностных и временных характеристик кодеков СК с алгоритмом порового декодирования (ПД). Задача определения наличия ВС коммутаторов кодека СК с алгоритмом ПД сводится к определению количества ненулевых символов $t_{\text{нс}}$ СП на длине интервала анализа l_a и сравнение с выбранным значением порога η: при превышении порога $(t_{\text{нс}} > \eta)$ принимается решение об отсутствии ВС коммутаторов кодека СК и, наоборот, при $(t_{\text{нс}} < \eta)$ – о наличии ВС коммутаторов кодека. Со статистической точки зрения эта задача сводится к задаче проверки двух гипотез H_0 и H_1 соответственно с вероятностью ошибок α и β: α – принять гипотезу H_0, когда верной является гипотеза H_1, β – принять гипотезу H_1, когда верной является гипотеза H_0. Далее условно принимаем, что гипотезе H_0 соответствует отсутствие ВС, а гипотезе H_1 – наличие ВС коммутаторов кодека СК с алгоритмом ПД.

Гипотезы H_0 и H_1 характеризуются условными распределениями вероятностей $P(\varepsilon / H_1)$ и $P(\varepsilon / H_0)$ нормированного числа ненулевых символов ε СП на интервале анализа l_a, т.е. $\varepsilon = t_{\text{нс}} / l_a$. Введение блокировки входа счетчика $t_{\text{нс}}$ на время переходного процесса в ФПСд позволяет считать, что появление ненулевых символов в СП описывается биномиальным законом распределения, что позволяет существенно упростить расчет вероятностей ошибок α и β.

В соответствии с [6, 7] вероятности ошибок α и β принятия решений относительно гипотез H_0 и H_1 определяются следующими выражениями:

$$\beta = \sum_{l_{\text{нс}}=0}^{\eta} \frac{l_a!}{t_{\text{нс}}!(l_a - t_{\text{нс}})!} \cdot P_0^{t_{\text{нс}}} \cdot (1 - P_0)^{l_{\text{нс}} - t_{\text{нс}}}, \qquad (3)$$

$$\alpha = \sum_{t_{\text{нс}}=\eta}^{l_a} \frac{l_a!}{t_{\text{нс}}!(l_a - t_{\text{нс}})!} \cdot P_0^{t_{\text{нс}}} \cdot (1 - P_0)^{l_{\text{нс}} - t_{\text{нс}}}, \qquad (4)$$

где l_a – интервал анализа, η – порог принятия решения, $P_0 = \varphi(y / H_j)$ – вероятность появления «1» для гипотез H_j, $j = 0;1$, $y_\varepsilon = \{y_1, y_2, \ldots, y_\varepsilon\}$ – количество наблюдений (опытов).

Выражения (3) и (4) определяют вероятностные характеристики УВС кодека СК с алгоритмом ПД, а именно, вероятность ошибки $\beta = P_{\text{лс}}$ – вероятности ложной ветвевой синхронизации коммутаторов кодека СК, а $\alpha = P_{\text{проп.с}}$ – вероятности пропуска ветвевой синхронизации или вероятности принятия решения об отсутствии ВС коммутаторов при наличии ВС.

Ввод блокировки входа счетчика ненулевых символов СП при установлении ВС коммутаторов кодека СК с алгоритмом ПД позволяет с достаточной для практики точностью проектирования УВС принять, что в анализируемой СП наблюдается случайный поток равновероятных статистически независимых логических «1» и «0», т.е. $P_0 = 0{,}5$.

Вышеприведенный материал позволяет выполнить проектирование УВС с заданными техническими характеристиками.

Способ построения УВС, обеспечивающий минимальное время установления ВС коммутаторов кодеков СК с пороговым алгоритмом декодирования

Один из способов построения и технической реализации УВС кодека СК, обеспечивающий минимальное время установления ВС коммутаторов, показан в виде следующей схемы.

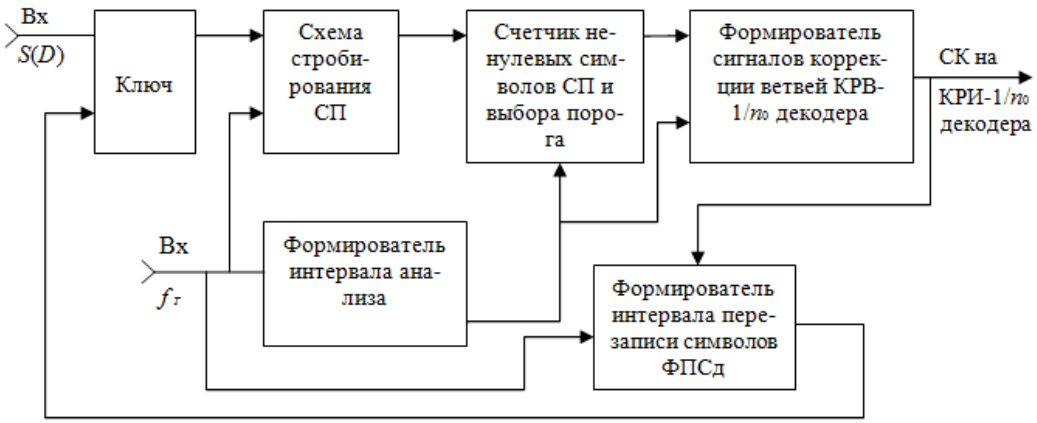


Рис. 2. Схема структурная УВС кодека СК, обеспечивающая минимальное время установления ВС коммутаторов кодека

УВС кодека СК с одним порогом срабатывания работает следующим образом. Сформированные синдромные символы через открытый ключ поступают на схему срабатывания синдромных символов, которые далее поступают на вход счетчика ненулевых символов СП и выбора порога срабатывания. По окончании интервала анализа сигнал с выхода счетчика ненулевых символов СП поступает на вход формирователя сигналов коррекции ветвей КРВ-1/n_0 декодера. Формирование сигнала коррекции ветвей КРВ-1/n_0 декодера выполняется по следующему правилу: если количество ненулевых символов СП не превышает порога, то формируется сигнал коррекции с нулевым уровнем (логический «0») и состояние ветвей КРВ-1/n_0 декодера остается без изменения, и одновременно этим же сигналом блокируется работа формирователя интервала перезаписи символов ФПСд и ключ остается открытым; если же количество ненулевых символов СП превышает порог, то формируется сигнал коррекции с уровнем логической единицы («1»), который осуществляет сдвиг ветвей КРВ-1/n_0 декодера на одну ветвь, включает в работу формирователь интервала перезаписи символов ФПСд, который соответствующим сигналом закрывает вход ключа для поступления символов СП на схему УВС в течение $(m+1)$ тактов, где m – количество ячеек памяти регистра сдвига ФПСд. По окончании интервала перезаписи данным блоком формируется сигнал, открывающий вход ключа для прохождения символов СП и начинается новый цикл анализа структуры символов СП. Таким образом работа УВС продолжается до установления ВС коммутаторов кодека СК. При установлении ВС коммутаторов кодека СК и при наличии ошибок не превышающих корректируемую способность СК порог не будет превышаться и формируемый сигнал коррекции ветвей КРВ-1/n_0 декодера будет иметь всегда уровень логического нуля («0») и переключение ветвей не будет происходить. При нарушении ВС коммутаторов кодека СК УВС перейдет в режим восстановления ВС по правилам, описанным выше. В соответствии с [6, 7] среднее время восстановления ВС коммутаторов кодека СК определяется выражением:

$$\overline{t}_{\text{восст}} = \overline{t}_{\text{оос}} + \overline{t}_{\text{поиск}} + \overline{t}_{\text{вс}}, \qquad (5)$$

где $\overline{t}_{\text{оос}}$ – среднее время обнаружения отсутствия ветвевого синхронизма кодека СК, $\overline{t}_{\text{поиск}}$ – среднее время поиска ветвевого синхронизма кодека СК, $\overline{t}_{\text{увс}}$ – среднее время установления в ветвевом синхронизме кодека СК.

Среднее время установления ветвевого синхронизма кодека СК определяется выражением:

$$\overline{t}_{\text{увс}} = \overline{\delta} \cdot (l_a + m), \qquad (6)$$

где $\overline{\delta} = \dfrac{1}{1 - P_{\text{лс}}}$ – среднее число проверок до первого обнаружения ВС коммутаторов кодека СК, m – длительность интервала перезаписи информации в регистре сдвига ФПСд.

Важнейшим параметром УВС кодека СК является коэффициент надежности работы УВС, определяемый следующим выражением:

$$k_{\text{над}} = \overline{t}_{\text{восст}} / \overline{t}_{\text{удер}}, \qquad (7)$$

где $\overline{t}_{\text{удер}} = \overline{\upsilon} \cdot l_a \cdot \varepsilon_0$ – среднее время удержания ВС коммутаторов кодека СК; $\overline{\upsilon}$ – среднее время наблюдений до первого срыва ВС кодека, которое определяется следующим выражением:

$$\overline{\upsilon} = 1 / P_{\text{проп.ВС}}, \qquad (8)$$

где $P_{\text{проп.ВС}} = 1 - P_{\text{п.ВС}} / 1 - P_{\text{л.ВС}}$ – вероятность пропуска наличия ветвевого синхронизма коммутаторов кодека СК, $P_{\text{п.ВС}}$ и $P_{\text{л.ВС}}$ вероятности соответственно правильного и ложного установления ветвевого синхронизма коммутаторов кодека СК.

Достоинствами данного способа построения и технической реализации УВС кодека СК с алгоритмом ПД являются: минимальное время установления и простота реализации ВС кодека СК.

Недостатками данного способа построения и технической реализации УВС являются: высокая вероятность установления ложной ветвевой синхронизации коммутаторов кодека СК и низкая помехоустойчивость или высокая вероятность срыва ветвевого синхронизма кодека СК. Первый недостаток определяется наличием в передаваемых сообщениях длинных последовательностей, состоящих из нулевых символов, а второй недостаток УВС определяется наличием одного порога или одной решающей схемы. Следовательно, при случайном сбое работы УВС или при поступлении ошибок в 3-4 раза, превышающих корректирующую способность СК, будет превышен установленный порог принятия решения и УВС перейдет в режим восстановления ВС кодека СК. Разработаны способы устранения данных недостатков работы УВС кодеков СК с алгоритмом ПД.

Способ повышения помехоустойчивости и помехозащищенности УВС кодеков СК с алгоритмом порогового декодирования

В [6, 7] рассматривается способ повышения помехоустойчивости работы УВС кодеков СК с алгоритмом ПД при наличии в канале связи ошибок существенно (в 3-4 раза) превышающих корректирующую способность СК, основанный на использовании УВС символов СП с выхода анализатора синдромной последовательности (АСП). Установлено, что использование символов СП с выхода АСП кодека СК вероятность срыва ВС коммутаторов кодека уменьшается на порядок.

Более эффективным способом построения и технической реализации УВС кодеков СК с алгоритмом ПД являются использование в УВС двух порогов или двух решающих схем (РС). Обобщенная структурная схема УВС кодека СК с двумя решающими схемами приведена на рис. 3.

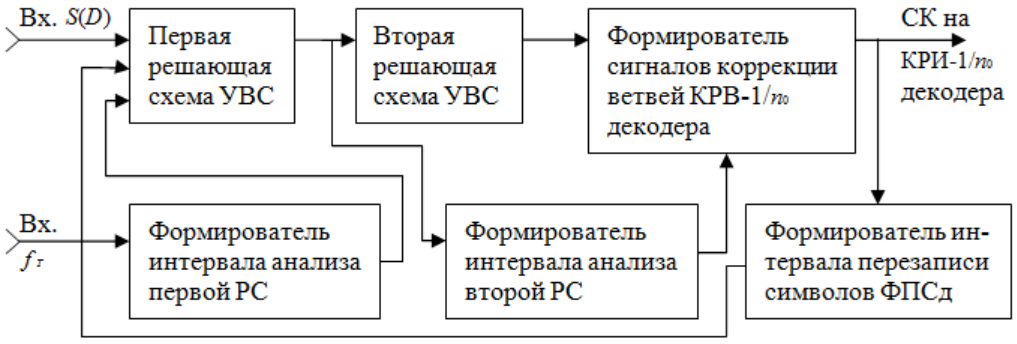


Рис. 3. Схема структурная УВС кодека СК с двумя решающими схемами

Порог принятия решения и принцип работы первой решающей схемы аналогичен УВС, приведенной на рис. 2. Величина порога принятия решения второй РС о наличии или отсутствии ВС коммутаторов кодека СК выбирается, исходя из требований к допустимой вероятно-

сти срыва ВС, и на практике не превышает значения $\mu = 3$, а величина интервала анализа второй РС равна $n = 2\mu + 1$. Это означает, что срыв ВС коммутаторов кодека СК произойдет только при трех и более кратном превышении порога первой РС. Вероятность срыва ВС коммутаторов кодека СК при использовании УВС с двумя РС уменьшается в μ раз по сравнению с использованием УВС с одной ОС, т.е.

$$P_{\text{ср.ВС}_2} \approx P^{\mu}_{\text{ср.ВС}_1}, \qquad (9)$$

где $P_{\text{ср.ВС}_1}$ – вероятность срыва ВС первой РС.

Недостатком данного способа построения УВС является существенное увеличение (в n раз) времени установления и времени восстановления ВС коммутаторов кодека СК. Кроме того, данные способы повышения помехоустойчивости работы УВС не исключают большую вероятность ложной ВС коммутаторов кодека СК при передаче сообщений, содержащих длинные (200 и более) последовательности, состоящие из одних нулевых символов.

Эффективным способом устранения данного недостатка построения УВС кодеков СК с алгоритмом ПД является использование символов псевдослучайной последовательности (М-последовательностей), которые на передающей стороне (в кодере) накладываются (суммируются по модулю два) на проверочные символы кодовой последовательности [5-7]. Это позволяет устранить на приемной стороне (в декодере) кодовые последовательности, состоящие из одних нулевых символов, как при наличии, так и при отсутствии ВС коммутаторов кодека СК с алгоритмом ПД. Принципы построения УВС кодеков СК на основе использования символов М-последовательностей подробно рассматриваются в [7]. К недостатку построения УВС на основе использования символов М-последовательностей следует отнести ухудшение временных характеристик УВС, а именно увеличение $\overline{t}_{\text{уст.ВС}}$ и $\overline{t}_{\text{восст.ВС}}$. Это определяется тем, что в данных УВС необходимо первоначально установить цикловую синхронизацию (ЦС) генераторов М-последовательностей кодера и декодера, а затем ВС коммутаторов кодека СК.

Заключение

Рассмотренные способы построения и технической реализации УВС кодеков СК на основе использования символов синдромной последовательности и символов М-последовательности не исключают возможность разработки новых способов построения и технической реализации УВС, обеспечивающих качественное улучшение временных и вероятностных характеристик данных устройств. Установлено, что практически не рассмотрены способы построения УВС на основе алгоритма последовательного анализа, а также дополнительные функциональные возможности, которые могут выполнять УВС кодеков СК, как, например, устранение фазовой неопределенности когерентных демодуляторов ФМ и др.

ANALYSIS METHODS OF NON-REDUNDANT SYNCHRONIZATION CODECS OF CONVOLUTIONAL CODES

A.N. DRAPEZA, G.A. VOLOSTNYH, A.I. PARFENIUK, A.I. KOROLEV

Abstract

The methods of building and technical implementation UBS of codecs SC on the basis of the use of symbols syndromic sequence and symbols M-sequences do not exclude the possibility of developing new ways of construction and technical implementation of UBS, providing a qualitative improvement temporal and probabilistic characteristics of these devices.

Список литературы

1. *Касами Т., Токура Н., Ивадари Ё. и др.* Теория кодирования. М., 1978.
2. *Блетхут Р*. Теория и практика кодов, контролирующих ошибки. М., 1986.
3. *Кларк Дж. мл., Кейн Дж.* Кодирование с исправлением ошибок в системах цифровой связи. М., 1987.
4. *Воронов В.Е.* // Известия ВУЗов, серия «Радиоэлектроника». 1972. Том 15. №4.
5. *Гизатулин Р.З., Мартин Ю.Н.* // Вопросы радиоэлектроники. Серия ТПС. Вып. 5. 1972.
6. *Королев А.И.* // Деп. в ВИНИТИ №897-СВ. 1986.
7. *Королев А.И.* Ветвевая синхронизация кодеков сверточных кодов. Мн., 1988.

УДК 004.056.53

ОЦЕНКА ПОКАЗАТЕЛЕЙ МОНИТОРИНГА БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ

М.Н. БОБОВ

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

Поступила в редакцию 2 ноября 2015

Приведен подход к оценке мониторинга безопасности информационных систем. Показано, что показателями мониторинга безопасности могут быть вероятности пропуска события безопасности, потери события безопасности при доставке в центр мониторинга и пропадания сигнала тревоги администратору безопасности. Предложены методы оценки указанных показателей мониторинга безопасности.

Ключевые слова: мониторинг безопасности, угрозы безопасности, оценка показателей.

Введение

Большое разнообразие методов и механизмов обеспечения информационной безопасности информационных систем предопределяет необходимость формальных моделей и методик анализа эффективности систем защиты. Данные модели и методики включают агрегированную оценку эффективности методов и механизмов как программно-технического, так и нормативно-организационного характера для защиты активов информационной системы. Решение подобных задач основывается на теоретико-графовом подходе в виде так называемой «модели систем с полным перекрытием [угроз безопасности]». Система защиты в рамках данной формализации представляется дольным графом, вариант которого представлен на рис. 1 [1].

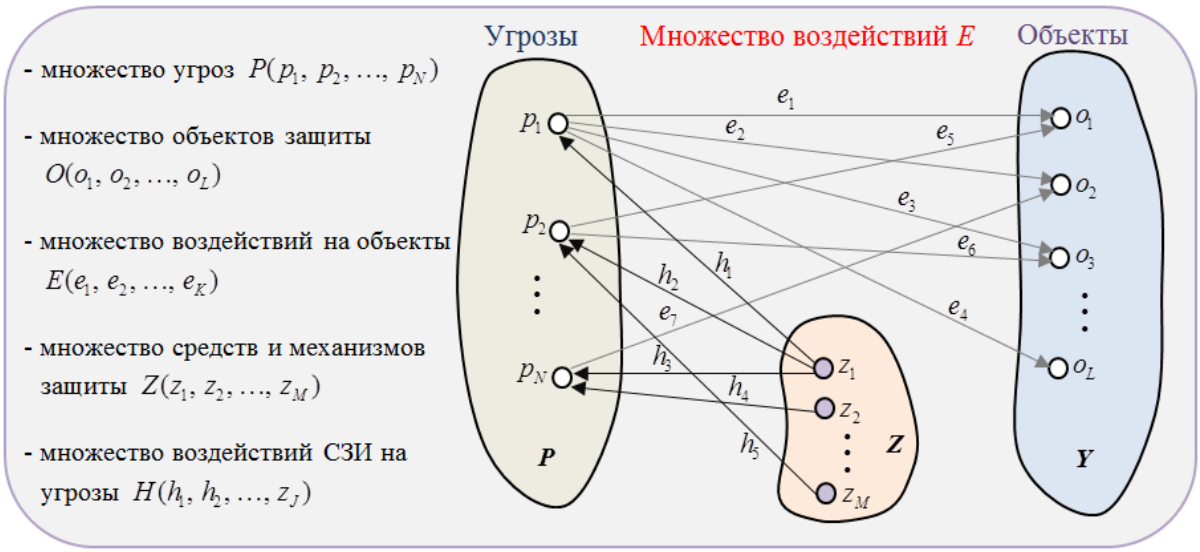


Рис. 1. Модель системы защиты в виде 3-дольного графа

Каждое ребро графа $G(P,O,Z,E,H)$ специфицирует воздействие конкретной угрозы на конкретный актив (объект) или устранение (нейтрализацию) определенным защитным ме-

ханизмом угрозы безопасности. При этом от каждой угрозы может быть несколько воздействий на различные объекты и каждый объект может быть подвергнут нескольким угрозам (связь «многие – ко многим»).

Для получения и оперирования количественными параметрами граф $G(P,O,Z,E,H)$ взвешивается и эквивалентно представляется следующей совокупностью векторов и матриц:

- вектор $P(p_1, p_2, \ldots, p_N)$, где p_i – вероятность осуществления соответствующей угрозы;

- вектор $O(o_1, o_2, \ldots, o_L)$, где o_i – стоимость соответствующего объекта защиты;

- $N \times L$ матрица $E\{eij\}$, где $e, j = 1$ при воздействии i-й угрозы на j-й объект, и $e, j = 0$ в противном случае;

- вектор $Z(z_1, z_2, \ldots, z_M)$, где z_i – стоимость соответствующего способа или средства защиты;

- $N \times M$ матрица $H\{hij\}$, где hij – вероятность устранения (или степень снижения ущерба) i-й угрозы от применения j-го средства защиты.

В рамках данной формализации возможно решение таких важных практических задач, как оценка технико-экономической или тактико- технической эффективности систем защиты.

Теоретический анализ

Оценка технико-экономической эффективности системы защиты основывается на определении количественного критерия, отражающего влияние реализованной системы защиты на снижение опасности от воздействия угроз, или иначе на величину ущерба от угроз безопасности.

В качестве такого критерия технико-экономической эффективности, в частности, можно использовать следующий показатель:

$$\frac{U - U'}{\sum_{i=1}^{L} O_i + \sum_{k}^{M} Z_k},$$

где U – оценка величины ущерба от угроз безопасности при отсутствии защитных мер и механизмов; U' – оценка величины остаточного ущерба при реализации защитных мер и механизмов.

Расчет величин U и U' осуществляется по следующим соотношениям:

$$U = \sum_{i=1}^{L} O_i \left(1 - \prod_{j}^{N} e_{ij} \left(1 - P_j\right)\right),$$

$$U' = \sum_{i=1}^{L} O_i \left(1 - \prod_{j=1}^{N} e_{ij} \left(1 - P_j \left(\prod_{k=1}^{M} (1 - h_{ik})\right)\right)\right).$$

Оценка величин U и U' изложена в методике, приведенной в стандарте банка России СТО БР ИББС-1.2-2014 [2]. В указанной методике используется 10 показателей оценки информационной безопасности (ИБ), из которых для автоматизированных информационных систем можно использовать следующие:

- обеспечение ИБ при назначении и распределении ролей и обеспечении доверия к персоналу;
- обеспечение ИБ на стадиях жизненного цикла;
- обеспечение ИБ при управлении доступом и регистрацией;
- обеспечение ИБ средствами антивирусной защиты;
- обеспечение ИБ при использовании ресурсов сети Интернет;
- обеспечение ИБ при использовании средств криптографической защиты информации;
- обеспечение ИБ информационных технологических процессов.

Вместе с тем, указанные показатели определяются комиссионным образом на стадии ввода информационной системы в эксплуатацию или по истечении определенного периода эксплуатации. Поэтому использовать изложенный в методике подход по оценке информацион-

ной безопасности для текущего контроля состояния информационной безопасности с помощью технических средств не представляется возможным.

Задачи тактико-технического анализа эффективности систем защиты заключаются в определении остаточной вероятности реализации всех возможных угроз без учета стоимости объектов и мер защиты, что может быть вычислено по следующему соотношению:

$$P'_{\text{преод.}} = 1 - \prod_{i=1}^{N}\left(1 - P_i\left(1 - \prod_{k=1}^{M}\left(1 - h_{ik}\right)\right)\right). \quad (1)$$

Считая, что во введенной в действие информационной системе реализована система защиты, средства и механизмы которой в полной мере защищают от угроз безопасности, т.е. $h_{ik} = 1$, выражение (1) можно записать в виде:

$$P'_{\text{преод.}} = 1 - \prod_{i=1}^{N}\left(1 - P_i\right), \quad (2)$$

где вероятности P_i являются остаточными вероятностями реализации угроз и определяются конструктивными свойствами системы мониторинга.

Методика оценки показателей мониторинга безопасности

Применительно к системе мониторинга безопасности информационной системы в условиях, когда все установленные угрозы перекрыты системой защиты, оценка состояния ИБ может быть определена из (2) по формуле

$$P'_{\text{преод.}} = 1 - \prod_{i=1}^{3}\left(1 - P_i\right), \quad (3)$$

где P_1 – вероятность пропуска события безопасности; P_2 – вероятность потери события безопасности при доставке в центр мониторинга; P_3 – вероятность пропадания сигнала тревоги администратору безопасности.

Рассмотрим подходы по определению вероятностей в выражении (3).

Природа угрозы P_1 не позволяет вычислить ее вероятность на основе известных соответствующих физических закономерностей (априорный подход). В некоторых случаях возможен апостериорный подход, основанный на накопленной статистике проявления соответствующей угрозы в данной или подобной компьютерной системе (в подобных условиях) [3]. Оценки вероятности реализации угрозы при этом вычисляются на основе методов статистических оценок. Альтернативой аналитическому и статистическому подходу для определения вероятности P_1 является метод экспертных оценок, широко используемый для оценок сложных, неформализуемых объектов.

Суть метода экспертных оценок заключается в том, что в качестве инструментария оценок (в качестве измерительного прибора) выступают специалисты-эксперты, которые на основе профессионального опыта, глубокого представления многокомпонентной природы оцениваемых объектов, дают эвристические оценки по одному или группе параметров [4]. В кратком изложении методика экспертных оценок включает следующие этапы.

Этап 1. Отбор экспертов (формальные и неформальные требования к специалистам-экспертам, метод «снежного кома», когда известного специалиста просят назвать других ему известных специалистов, в свою очередь, опрашивают их и т.д., до тех пор пока множество экспертов не прекращает расширяться. На практике количество экспертов – 10-12 человек).

Этап 2. Выбор параметров, по которым оцениваются объекты (при этом определяются сущностные параметры оценивания, которые должны выражать природу оцениваемых объектов и быть независимыми друг от друга, определяются веса параметров).

Этап 3. Выбор шкал оценивания и методов экспертного шкалирования. Применяются порядковые, ранговые, интервальные, абсолютные и другие шкалы. В качестве методов шкалирования выступают ранжирование объектов по предпочтительности выраженности оцениваемого параметра (порядковая шкала оценки), попарные оценки сравнительной предпочтительности во всех возможных парах оцениваемых объектов, и непосредственная оценка выражен-

ности оцениваемого параметра (например, эксперты непосредственно дают оценку вероятности реализации угроз, в других случаях на основе специальных балльных шкал оценки).

Этап 4. Выбор и осуществление процедуры опроса экспертов (с непосредственным взаимодействием экспертов или без взаимодействия, т.н. итерационный метод опроса «Дельфи», когда эксперты непосредственно не взаимодействуют, но после каждого тура опроса им сообщают усредненные оценки прежнего тура и просят на этой основе скорректировать свои прежние оценки, исключая тем самым влияние на результаты опроса мнений конкретных «авторитетов», и т.д.).

Этап 5. Агрегирование оценок, анализ их устойчивости и согласованности, осуществляемые на основе подходов, подобных методам обработки статистических данных.

Следует отметить, что экспертные оценки, несмотря на их «субъективность» на основе хорошо подобранных экспертных комиссий, правильно установленных методов шкалирования и опроса, при соответствующей обработке дают результаты, действенность которых многократно апробирована в крупных проектах и процедурах, не допускающих другие, в особенности, аналитические и статистические подходы.

В отличие от вероятности P_1 значения вероятностей P_2 и P_3 могут быть вычислены аналитически. Так, вероятность P_2 может быть вычислена как вероятность потери сообщения, величина которой полностью определяется характеристиками сети передачи. Одним из подходов [5] может быть определение вероятности потери сообщения на основе статистики работы сети по формуле

$$P_{\text{пот}} = 1 - C_{\text{пол}} / C_{\text{отпр}},$$

где $C_{\text{пол}}$ – количество полученных сообщений; $C_{\text{отпр}}$ – количество отправленных сообщений.

Для расчета вероятности потери сообщения можно воспользоваться также методами теории массового обслуживания, приведенными в [6]. В данном случае вероятность потери сообщения можно определить по формуле

$$P_{\text{пот}} = \frac{(1-\rho)}{1-\rho^{s+1}} \cdot \rho^{s}, \qquad (4)$$

где ρ – коэффициент загрузки сети ($\rho < 1$); s – емкость запоминающего устройства (в сообщениях).

Для современных телекоммуникационных сетей расчеты по формуле (4) могут дать неоправданно оптимистические оценки, так как они основаны на марковских моделях потоков заявок и их обслуживания. Так как трафик телекоммуникационной сети обладает свойствами фрактальности, для оценки вероятности потери сообщения используют параметр Херста. Выражение вероятности потерь принимает вид

$$P_{\text{пот}} = \frac{(1-\rho)}{1-\rho^{(S+1)^{2(1-H)}}} \cdot \rho^{S^{2(1-H)}}, \qquad (5)$$

где H – параметр Херста.

Если параметр Херста $H \le 0{,}5,$ то поток не обладает свойством фрактальности и выражение (5) преобразуется к виду (4). В [7] приведены следующие данные о фрактальности различных типов трафика (см. табл.).

Фрактальность различных типов трафика

Тип трафика	Фрактальность трафика (*H*)
Ethernet	0,9
HTTP	0,75-0,92
Видео	0,6-0,9
Аудио	0,6-0,9
P2P	0,6-0,9

Как видно из таблицы, основные типы сетевых приложений имеют коэффициент Херста больше 0,6. При таком трафике вероятность потери сообщения желательно определять по формуле (5).

Для расчета вероятности пропадания сигнала тревоги администратору безопасности P_3 фиксируется совокупность технических средств, составляющих тракт доведения сигналов тревоги администратору. Вероятность P_3 вычисляется по формуле

$$P_3 = 1 - e^{\sum_{i-1}^{m} \lambda_i x t_i},$$

где λ_i – интенсивность отказа или сбоя *i*-го аппаратного средства *i*-го программного средства в тракте сигнализации; t_i – интервал времени, в течение которого функционирует аппаратное или программное средство в тракте сигнализации; m – число технических средств, составляющих тракт сигнализации.

Параметр λ может быть определtn следующим образом [8]. Для периода эксплуатации информационной системы, когда $\lambda = \text{const}$, т.е. интенсивность отказов λ не зависит от времени, вероятность того, что в заданном интервале времени T не произойдет отказа, определяется по формуле

$$P(T) = e^{-\lambda T}.$$

При известном или принятом значении $P(T)$ интенсивность отказа можно вычислить по формуле

$$\lambda = -\frac{\ln P(T)}{T},$$

где T – время безотказной работы (час).

Для программных средств защиты вероятность сбоя может быть определена из соотношения $\lambda_{\text{сб.}} = 0{,}1\lambda_{\text{отк.}}$

Заключение

Разработана методика оценки показателей мониторинга безопасности информационных систем, к которым относятся:

- вероятность пропуска события безопасности;
- вероятность потери события безопасности при доставке в центр мониторинга;
- вероятность пропадания сигнала тревоги администратору безопасности.

Для каждого показателя определены методы расчета численных значений, которые можно легко использовать в практической деятельности служб обеспечения безопасности автоматизированных информационных систем.

ASSESSMENT OF INFORMATION SYSTEMS SECURITY MONITORING

M.N. BOBOV

Abstract

The article shows the approach to assessment of information systems security monitoring. It is revealed that possibilities of omitting or loss the security events in the process of delivering them to the security administrator as well as loss of the alarm signal are important security monitoring showings. The methods of assessment of the revealed security monitoring indexes are offered further in the article.

Список литературы

1. *Гайдамакин Н.А.* Теоретические основы компьютерной безопасности. Екатеринбург, 2008.
2. Стандарт Банка России СТО БР ИББС-1.2-2014.
3. *Антюхов В.И.* Системный анализ и принятие решений. СПб., 2009.
4. *Воробьёв С.Н., Егоров Е.С., Торбин В.У.* Методы выработки решений. Надежность и эффективность в технике. Том 3. М., 1988.
5. *Боев В.Д., Ушкань А.О.* // Сб. докладов IV Всероссийской научно-практической конференции «Имитационное моделирование. Теория и практика». 2009. С. 299-303.
6. *Ланин М.И.* // Автоматика и телемеханика. 1962. Том 23. Вып. 3.
7. *Лосев Ю.И., Руккас К.М.* // Вісник Харківського національного університету. Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління». 2008. №833. С. 163-168.
8. *Рембеза А.И., Патрушев В.И.* Проектный анализ надежности. Том 5. М., 1988.

УДК 621.391.7:512.772

ФУРЬЕ-АНАЛИЗ АМПЛИТУДНО-ОГРАНИЧЕННОЙ СУММЫ СИГНАЛА И НОРМАЛЬНОГО ШУМА

М.Ю. ХОМЕНОК

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

Поступила в редакцию 14 октября 2015

Приведены основные результаты статистического анализа структурных свойств сигнальной компоненты клиппированной суммы сигнала и шума в соответствии с материалами депонированной рукописи [1].

Ключевые слова: ряд Фурье, корреляционный анализ, нормальный узкополосный стационарный процесс, фазоманипулированный сигнал, двумерная функция распределения, отношение сигнал-шум, амплитудный ограничитель.

Ряд Фурье по системе фазоманипулированных функций

Процесс на выходе амплитудного ограничителя, соответствующий знаковой функции от входного процесса, определяемого аддитивной смесью узкополосных сигнала $S(t)$ и шума $n(t)$, равен

$$Y(t)=\operatorname{sign}\left[x(t)\right]=\operatorname{sign}\left[\cos\left(\Phi(t)\right)\right], \tag{1}$$

где $\Phi(t)=2\pi f_o t+\varphi_x(t)$,

$$\varphi_x(t)=\operatorname{arctg}\frac{S(t)\sin\left[\varphi_s(t)\right]+N(t)\sin\left[\varphi_n(t)\right]}{S(t)\cos\left[\varphi_s(t)\right]+N(t)\cos\left[\varphi_n(t)\right]},$$

$S(t)=S_o\cos\left[2\pi f_o t+\varphi_s(t)\right]$ и $n(t)=N(t)\cos\left[2\pi f_o t+\varphi_n(t)\right]$, а $S_o(t)$, $N(t)\geq 0$.

Поскольку $\Phi(t)$ в области определения является однозначной функцией, то система тригонометрических функций вида

$$\left\{\cos\left[n\Phi(t)\right]\right\},\ n=\overline{1,\ldots,N,}\ \text{при}\ N\to\infty \tag{2}$$

является ортогональной в области значений φ. Тогда, учитывая, что в области значений Φ процесс (1) представляет собой периодическую последовательность прямоугольных импульсов $A_{10}(t)$, разложение (1) в ряд Фурье по системе функций (2) имеет вид

$$\operatorname{sign}\left[\cos\Phi(t)\right]=\sum_{k=0}^{\infty}\frac{4}{\pi(2k+1)}(-1)^k\cos\left[(2k+1)\Phi(t)\right]. \tag{3}$$

Таким образом, импульсная последовательность может быть представлена рядом Фурье по системе фазоманипулированных функций (2), ортогональных в области значений фазы $\Phi(t)$, но не ортогональных во временной области, если определен закон изменения фазы, согласно которому формируется структура последовательности.

Проиллюстрируем использование ряда (3) на примере определения функции автокорреляции (ФАК) функционального преобразованного в амплитудном ограничителе нормального узкополосного стационарного шума (НУСШ) $n(t): N\left[0, R(\tau)\right]$, где $R(\tau) = R_o(\tau)\cos(2 f_o \tau)$. Для этого определим ФАК косинуса фазы этого процесса, умноженной на целое n число раз.

Поскольку $\cos\left[n\varphi(t)\right]$ представляет безынерционное функциональное преобразование $\varphi(t)$, то на основе теоремы о среднем ФАК $\cos\left[n\varphi(t)\right]$ определится согласно выражению

$$B_n(\tau) = \int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \cos(n\varphi_1)\cos(n\varphi_2) W_2(\varphi_1, \varphi_2; \tau)\, d\varphi_1 d\varphi_2, \qquad (4)$$

где $\varphi_1 = \varphi(t_1)$, $\varphi_2 = \varphi(\varphi t_2)$, а двумерная плотность вероятности фазы [2]

$$W_2(\varphi_1, \varphi_2; \tau) = \sum_{r=-\infty}^{\infty} A_r e^{jr(\varphi_1 - \varphi_2)} \qquad (5)$$

при $|\varphi_1| \le \pi, |\varphi_2| \le \pi$, $A_r = \dfrac{1 - R_0^2}{4\pi} \sum\limits_{m=0}^{\infty} \dfrac{\Gamma^2\left(m + 1 + \dfrac{r}{2}\right)}{m!(m+r)!} R_0^{r+2}$,

где $\Gamma\left(m + 1 + \dfrac{r}{2}\right)$ – гамма-функция.

Представим (5) в (4) и, меняя порядок суммирования и интегрирования, находим:

$$B_n(\tau) = \sum_{r=-\infty}^{\infty} A_r \int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \cos(n\varphi_1)\cos(n\varphi_2) e^{jr\varphi_1} e^{jr\varphi_2} d\varphi_1 d\varphi_2.$$

В силу ортогональности тригонометрических функций

$$\int_{-\pi}^{\pi} \cos(n\varphi_1) e^{jr\varphi_1} d\varphi_1 = \begin{cases} \pi, & r = \pm n \\ 0 & r \ne \pm n \end{cases}, \quad \int_{-\pi}^{\pi} \cos(n\varphi_2) e^{jr\varphi_2} d\varphi_2 = \begin{cases} \pi, & r = \pm n \\ 0 & r \ne \pm n \end{cases}.$$

С учетом этих значений ФАК $\cos\left[n\varphi(t)\right]$ принимает вид: $B_n(\tau) = 2\pi^2 A_n$, что при $n = 1$ совпадает с результатом, представленным в виде формулы 8.103 [2].

Соответственно, учитывая, что процесс на выходе амплитудного ограничителя определяется выражением (3), а также принимая во внимание наличие детерминированной составляющей фазы $\omega_0 t$, ФАК процесса на выходе идеального симметричного амплитудного ограничителя запишется в виде:

$$B(\tau) = \sum_{k=0}^{\infty} \frac{32}{(2k+1)^2} A_{2k+1} \cos\left[(2k+1)\omega_0 \tau\right] =$$

$$= \frac{8}{\pi^2}\left(1 - R_0^2\right) \sum_{k=0}^{\infty}\sum_{m=0}^{\infty} \frac{1}{(2k+1)^2} \frac{\Gamma^2\left(m + k + \dfrac{3}{2}\right)}{m!(m + 2k + 1)} R_0^{2k+2m+1} \cos\left[(2k+1)\omega_0 \tau\right].$$

При $f_0 = 1/2 f_T$, $\varphi_x(t) \in 0, \pi$ и $t \in \left[n\tau_{\text{И}}, (n+1)\tau\right]$, $n = 0, \pm 1, \pm 2, \ldots$ случайная импульсная последовательность соответствует импульсной последовательности прямоугольных импульсов с тактовой частотой $f_T = 1/\tau_{\text{И}}$.

Эта последовательность может быть представлена рядом Фурье по нечетным фазоманипулированным гармоникам, точки фазовой манипуляции которых совпадают со значащими моментами исходной последовательности:

$$A(t) = \sum_{k=0}^{\infty} (-1)^k \frac{4}{\pi(2k+1)} \cos\left\{(2k+1)\left[\pi f_T t + 1/2\varphi_T + \varphi(t)\right]\right\},$$

где f_T, φ_T – соответственно тактовая частота и начальная фаза гармоники на тактовой частоте, $\varphi(t)$ – закон фазовой манипуляции.

Из него следует, что первая фазоманипулированная гармоника представляет последовательность косинусоидальных импульсов, которая может быть получена путем когерентного преобразования несущей частоты f_0 ФМнС с манипуляцией фазы на 180º до значения, равного полутактовой частоте.

В этом случае структура сформированной импульсной последовательности $A'(t)$ является преобразованной по отношению к исходной $A(t)$ и связана с ней суммированием по модулю два с меандром полутактовой частоты $A_{10}(t)$[1]:

$$A_{10}(t) = A(t) \oplus A'(t).$$

Ряд Фурье сигнальной компоненты случайной импульсной последовательности

Анализируемая модель сигнала соответствует идеальному симметричному амплитудному ограничителю случайного процесса, представляющего собой сумму ФМнС и НУСШ. Двумерная функция распределения этого процесса имеет вид [2]:

$$W_2(x_1, x_2; \mathrm{t}, \tau) = \frac{1}{2\pi\sigma^2\sqrt{1-R^2}} e^{-\frac{(x_1-a_1)^2 - 2R(x_1-a_1)(x_2-a_2)+(x_2-a_2)^2}{2\sigma^2(1-R^2)}},$$

где $a_1 = S(t)$, $a_2 = S(t+\tau)$, $R = R(\tau)$ – коэффициент корреляции и σ^2 – дисперсия стационарного процесса $x(t)$.

Поскольку характеристика $f(x)$ идеального симметричного амплитудного ограничителя допускает представление контурным интегралом, то

$$g(ju) = \int_{-\infty}^{\infty} f(x) e^{-jxu} dx = \begin{cases} (ju)^{-1}, & x > 0 \\ -(ju)^{-1}, & x \le 0 \end{cases}. \qquad (6)$$

ФАК ФМнС и НУСШ может быть определена согласно выражению [2]:

$$B_n(\tau, t) = \frac{1}{4\pi^2} \int_{C1} \int_{C2} g(ju_1) g(ju_2) e^{j(a_1u_1 + a_2u_2)} e^{-\sigma^2/2\left(u_1^2 + 2Ru_1u_2 + u_2^2\right)} du_1 du_2. \qquad (7)$$

В интеграле (7) только множитель $e^{j(a_1u_1+a_2u_2)}$ содержит величины $a_1 = S(t)$ и $a_2 = S(t+\tau)$, зависящие от времени. Поэтому при усреднении корреляционной функции $B(\tau, t)$ по времени, усредняется только этот множитель. Обозначая

$$\Theta_s(u_1, u_2; \tau) = \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} e^{j\left[S(t)u_1 + S(t+\tau)u_2\right]} dt, \qquad (8)$$

находим усредненное выражение корреляционной функции случайного процесса на выходе амплитудного ограничителя в виде

$$B(\tau) = \frac{1}{4\pi^2} \int_{C1} \int_{C2} g(ju_1) g(ju_2) \Theta_S(u_1, u_2; \tau) e^{-\sigma^2/2\left(u_1^2 + 2Ru_1u_2 + u_2^2\right)} du_1 du_2, \qquad (9)$$

Детерминированное слагаемое входного процесса $S(t)$ на интервале $[0, T]$ представим выражением

$$S(t) = S_0(t)\cos\left[2\pi f_o t + \varphi_s(t)\right] = S\cos\Phi.$$

Тогда, воспользовавшись разложением по формуле 8.5114 [3],

$$e^{jzS\cos\Phi}=\sum_{m=0}^{\infty}j^{m}\varepsilon_m J_m(zS)\cos(m\Phi),$$

где $\varepsilon_0=1$, $\varepsilon_m=2$ при $m=1,2,\ldots$, а $J_m(zS)$ – бесселева функция первого рода, получим, что

$$e^{j\left[u_1S(t)+u_2S(t+\tau)\right]}=\sum_{n=0}^{\infty}j^{2n}\varepsilon_n^2 J_n(u_1S_1)J_n(u_2S_2)\cos(n\Phi_1)\cos(n\Phi_2)+$$

$$+\sum_{\substack{n=0\\ n\neq l}}^{\infty}\sum_{l=0}^{\infty}j^{(n+l)}\varepsilon_n\varepsilon_l J_n(u_1S_1)J_l(u_2S_2)\cos(l\Phi_1)\cos(n\Phi_2). \qquad (10)$$

С учетом (10) соотношение (8) запишется в виде

$$\Theta_s(u_1,u_2;\tau)=\sum_{n=0}^{\infty}j^{2n}\varepsilon_n^2\lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{T/2}J_n(u_1S_1)J_n(u_2S_2)\cos(n\Phi_1)\cos(n\Phi_2)dt.$$

Тогда, используя разложение сомножителя

$$e^{-\sigma^2Ru_1u_2}=\sum_{k=0}^{\infty}(-1)^k\frac{\sigma^{2k}}{k!}R^k u_1^k u_2^k$$

и разделяя в (9) переменные интегрирования, выражение для усредненной ФАК процесса на выходе амплитудного ограничителя с характеристикой, определенной в (6), примет вид

$$B(\tau)=\sum_{\substack{k=0\\ k\neq n}}^{\infty}\sum_{n=0}^{\infty}\varepsilon_n^2\frac{\sigma^{2k}}{k!}R^k B_{nk} \qquad (11)$$

где

$$B_{nk}=B_{nk}(\tau)=\lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{T/2}h_{nk}(t)\cos\left[n\Phi(t)\right]h_{nk}(t+\tau)\cos\left[n\Phi(t+\tau)\right]dt, \qquad (12)$$

$$h_{nk}(t)=\left(\frac{1}{\pi}\right)j^{n+k-1}\int_{C1}u^{k-1}J_n(uS)e^{-\frac{\sigma^2u^2}{2}}du=\frac{1}{n!}\left[\frac{\sqrt{2}}{\sigma}\right]^k\left[\frac{S}{\sqrt{2}\sigma}\right]^n\frac{{}_1F_1\left(\frac{n+k}{2}\right),\ n+1,-\frac{S^2}{2\sigma^2}}{\Gamma\left[1-(n+k)/2\right]}, \qquad (13)$$

где ${}_1F_1\left(\frac{n+k}{2},\ n+1,-\frac{S^2}{2\sigma^2}\right)$ – вырожденная гипергеометрическая функция, форм. 7.94 [2].

Выражение (12) представляет собой ФАК сигнальных составляющих на выходе амплитудного ограничителя, которые могут быть определены при анализе энергетического спектра процесса с учетом структурных свойств информационной манипулирующей последовательности. По своей структуре ряд (11) совпадает с выражением ФАК для гармонического сигнала и НУСШ, приведенного в [2]. Поэтому его анализ выполнен применительно только к сигнальной слагаемой процесса, ФАК которой определяется из ряда (11) при $k=0$.

Так как гамма-функция при целом отрицательном не ограничена, то, как видно из (13), $h_{n0}=0$ при $n=2r$, $r=1,2,3,\ldots$. Таким образом, ФАК информационной слагаемой процесса на выходе амплитудного ограничителя может быть представлена в виде суммы нечетных спектральных полос в соответствии с выражением (12), при $n=2r+1$. Поэтому, как следует из (12), информационная составляющая процесса на выходе амплитудного ограничителя определяется в виде

$$y_s(t)=\sum_{r=0}^{\infty}h_{2r+1,0}\cos\left\{(2r+1)\left[2\pi f_o t+\varphi_s(t)\right]\right\}. \qquad (14)$$

С другой стороны, так как $n(t)$ есть НУСШ с нулевым средним, а $s(T)$ и $n(t)$ независимы, то слагаемое процесса на выходе амплитудного ограничителя, соответствующее информационной составляющей, может быть определено как его среднее значение, т.е.

$$y_s(\mathrm{t}) = y_s = \overline{\operatorname{sign}\left[S(t)+n(t)\right]} = erf\left[\frac{S(t)}{\sqrt{2}\sigma}\right], \qquad (15)$$

где $\operatorname{erf}(x) \equiv \frac{1}{\sqrt{2}}\int_0^x \exp\left(-y^2\right)dy.$

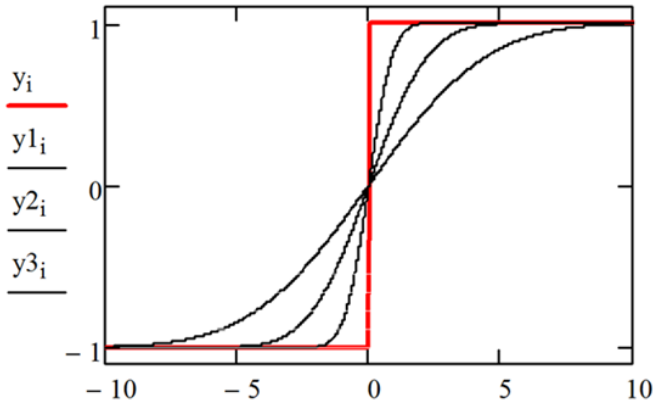
Из сравнения выражений (14) и (15) следует справедливость равенства

$$\operatorname{erf}\left\{\frac{S(t)}{\sqrt{2}\sigma_x}\cos\left[2\pi f_o t+\varphi_s(t)\right]\right\} = \sum_{r=0}^{\infty} h_{2r+1,0}\cos\left\{(2r+1)\left[2\pi f_o t+\varphi_s(t)\right]\right\}. \qquad (16)$$

Заключение

Анализ выражения (16) позволяет выявить следующие особенности преобразования смеси ФМнС и НУСШ в идеальном амплитудном ограничителе:

- отсутствие искаженной фазовой структуры сигнала;
- для сигнальной составляющей на выходе амплитудного ограничителя справедливо представление в ряд Фурье по системе фазоманипулированных функций, определенных в соответствии с фазовой структурой сигнала на его входе, с коэффициентами Фурье, равными $h_{2r+1,0}$. Причем, если отношение сигнал-шум на входе ограничителя стремится к бесконечности, то коэффициенты ряда Фурье в (16) соответствуют коэффициентам Фурье для последовательности прямоугольных импульсов, т.е.: $h_{2r+1,0} \to 4/\left[\pi(2r+1)\right]$;
- при большом отношении сигнал-шум на входе амплитудный ограничитель обладает формирующим по амплитуде свойством в соответствии с функциональной зависимостью $\operatorname{erf}(ax)$. При a, стремящемся к бесконечности, функция $\operatorname{erf}(ax)$ соответствует знаковой функции $\operatorname{sign}(x)$, как это показано на рисунке, иллюстрирующем функциональное отображение среднестатистической компоненты. В этом случае равенство (16) будет соответствовать равенству (3).



Кривая функционального отображения среднестатистической компоненты клиппированной смеси сигнала и нормального узкополосного шума: $\lim_{a\to\infty}\operatorname{erf}(ax) = \operatorname{sign}(x)$

FOURIER ANALYSIS OF AMPLITUDE-LIMITED SIGNAL AND NORMAL NOISE

M.J. HOMENOK

Abstract

The article reflects the main results of statistical analysis of clipped signal plus normal noise in view of Fourier series of the average signal component which is presented in compliance with the deposit manuscript [1].

Список литературы

1. *Хоменок М.Ю.* // Деп. в БелНИИНТИ №805Бе-Д83. 1983.
2. *Левин Б.Р.* Теоретические основы статистической радиотехники. М., 1974.
3. *Градштейн И.С., Рыжик И.М.* Таблицы интегралов, сумм, рядов и произведений. М., 1971.

УДК 621.391

СЖАТИЕ ПОЛУТОНОВЫХ ИЗОБРАЖЕНИЙ НА ОСНОВЕ КЛАСТЕРИЗАЦИИ И ПРОГРЕССИВНОГО ВЛОЖЕННОГО КОДИРОВАНИЯ ВЕЙВЛЕТ-КОЭФФИЦИЕНТОВ

В.В. НОВИЦКИЙ, В.Ю. ЦВЕТКОВ

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

Поступила в редакцию 10 ноября 2015

Разработан алгоритм вейвлет-сжатия изображений на основе кластеризации и прогрессивного вложенного кодирования вейвлет-коэффициентов, обладающий низкой вычислительной сложностью и высокой степенью параллелизма. Он предназначен для аппаратной реализации. Показано, что предложенный алгоритм превосходит известные алгоритмы по коэффициенту сжатия и возможности распараллеливания.

Ключевые слова: вейвлет-сжатие, вложенное кодирование.

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). В то же время разрешающая способность и полоса захвата съемочной аппаратуры неуклонно растут, что приводит к росту количества передаваемой ЦИ. Целевая информация представляет собой последовательность снимков высокого пространственного разрешения (или видеоинформацию, ВИ) и служебную информацию. В данном случае единственным решением проблемы является вейвлет-сжатие ВИ с потерями, которое обеспечивает наилучшее качество восстановленных изображений (среди существующих алгоритмов сжатия изображений с потерями). Однако здесь возникает другая проблема – необходимость работы в режиме реального времени или в поточном режиме. Поскольку даже самый мощный процессор в настоящее время не способен обрабатывать гигабитные потоки информации «на лету», то на борту спутника ДЗЗ приходится устанавливать специализированные устройства для цифровой обработки сигнала с массовым распараллеливанием и большим количеством встроенных гигабитных приемо-передатчиков, а именно: программируемые логические интегральные схемы (ПЛИС). К сожалению, ни один из существующих алгоритмов вейвлет-сжатия не пригоден к реализации на ПЛИС в чистом виде. 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.

Известны четыре базовых метода, позволяющих получать прогрессивный и вложенный поток на основе сжатия вейвлет-коэффициентов. Хронологически первым из них был метод EZW [1], опубликованный Джеромом Шапиро в 1993 г. В 1996 г. Амир Саид и Вильям Перлман представили алгоритм SPIHT [2], который являлся модификацией метода EZW и выигрывал у него в среднем 1 дБ по метрике PSNR при сжатии с потерями. В 1998 году Давид Таубман разработал алгоритм EBCOT [3], который использовал отличную от методов EZW и SPIHT

концепцию. Именно EBCOT стал прототипом для создания стандарта JPEG2000. Еще один значимый метод в этом направлении сжатия под названием SPECK был впервые представлен Асадом Исламом и Вильямом Перлманом в 1998 г. и стал основным конкурентом EBCOT за базовый метод стандарта JPEG2000. Впоследствии этот метод претерпел множество модификаций [4] и одна из них под названием SBHT [5] вошла в верификационную модель стандарта JPEG2000 как менее сложная альтернатива EBCOT при несущественном уменьшении качества сжатия.

Главным недостатком приведенных базовых методов вейвлет-сжатия с точки зрения распараллеливания и вычислительной сложности является наличие адаптивного арифметического кодера, который в настоящее время обеспечивает самое близкое к оптимальному энтропийное кодирование, но является сложным и достаточно медленным устройством. Методы EBCOT и JPEG2000 помимо этого используют контекстное моделирование по битовым плоскостям кодового блока, что дополнительно усложняет кодер с учетом роста разрядности пикселов исходных снимков. В то же время самым простым и хорошо распараллеливаемым алгоритмом является SPECK, который за время своего существования стал прототипом для множества других алгоритмов. С точки зрения низкой вычислительной сложности наиболее подходящим является алгоритм MECT [6], разработанный сотрудниками кафедры СиУТ БГУИР. Однако данный алгоритм имеет ряд существенных недостатков, в случае устранения которых можно добиться решения поставленной задачи.

Цель работы: разработать алгоритм вейвлет-сжатия с низкой вычислительной сложностью и высокой степенью параллелизма, базирующийся на принципах алгоритма SPECK.

Разработка нового алгоритма вейвлет-сжатия

Предлагается алгоритм HBCT (Hardware Block Cluster Tree) на основе двумерного дискретного вейвлет-преобразовании полутонового изображения, построении кластерных деревьев в пределах квадратных блоков битовых плоскостей матрицы вейвлет-коэффициентов и прогрессивного вложенного кодирования.

В качестве прототипов данного алгоритма выступают SPECK (Set-Partitioning Embedded bloCK) и MECT (Multiscale Embedded Cluster Tree). Поэтому новый алгоритм назван HBCT. Он использует тот же принцип сжатия (Cluster Tree) и блочную структуру, подобно SPECK, но более простую аппаратную реализацию. Следовательно, он должен обладать вычислительной сложностью не выше MECT.

Все алгоритмы вейвлет-сжатия основаны на вейвлет-преобразовании исходного изображения, которое перераспределяет энергию пикселов по различным пространственно-частотным поддиапазонам так, что в самом узком поддиапазоне содержится наибольшая энергия. Для изображений, как правило, используют пирамидальную схему разложения в несколько уровней, в результате которой получается матрица вейвлет-коэффициентов такой же размерности, что и исходное изображение, но другой разрядности и со знаком. Для сжатия без потерь, как правило, используют семейство биортогональных вейвлетов CDF 5/3, с потерями – CDF 9/7. На рис. 1 приведен пример трехуровневого разложения (декомпозиции) популярного тестового изображения «Lena» по пирамидальной схеме с применением вейвлетов CDF 9/7.

Метод SPECK, в отличие от EZW и SPIHT, устраняет не междиапазонную избыточность, а внутридиапазонную, что дает ему ряд преимуществ, прежде всего в низкой вычислительной сложности [4]. При этом он, подобно SPIHT, может работать как в режиме сжатия с потерями, так и без потерь, показывая при этом практически идентичные результаты в качестве сжатия по метрике PSNR. Метод SPECK итеративно разделяет каждую битовую плоскость матрицы вейвлет-коэффициентов (кроме знаковой) на два множества S и I, которым ставятся в соответствие два контрольных списка (список значимых бит и список незначимых множеств).

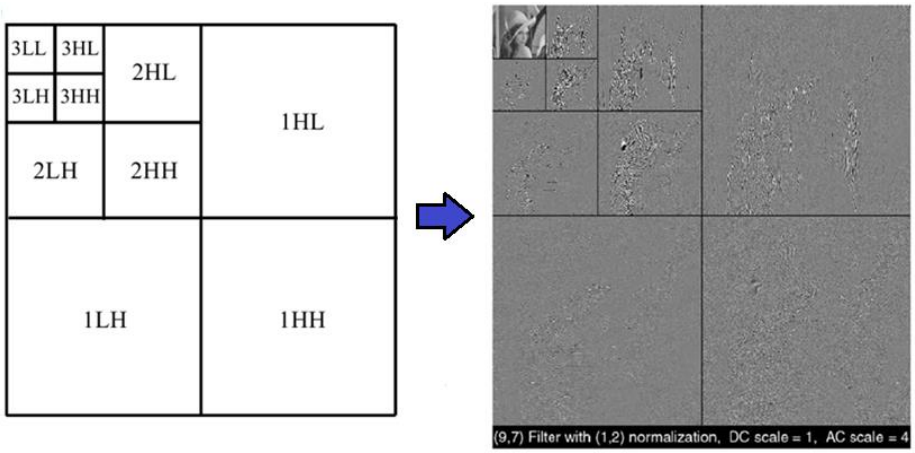


Рис. 1. Пример трехуровневой вейвлет-декомпозиции изображения «Lena»

Если упростить описание алгоритма кодирования в SPECK, то можно показать, что в процессе кодирования кодер проходит по всем пространственно-частотным диапазонам битовой плоскости, начиная с *LLn* и заканчивая *HH1*. Каждый диапазон компактно описывается с помощью «нуль-дерева», знаков значимых бит и уточняющих бит. «Нуль-дерево» строится в два этапа: на первом этапе кодовый блок (поддиапазон) проходит кластеризацию, строя квадродерево (каждый последующий уровень квадродерева получается разбиением предыдущего на кластеры размером 2×2 и присвоения соответствующей позиции последующего уровня «0» в случае, если в кластере все биты равны «0», и «1» в любом другом случае); на втором этапе кодер двигается по единичным битам квадродерева, начиная с вершины и игнорируя нулевые кластеры (рис. 2). «Карта значимости» здесь, как и в SPIHT, состоит из «нуль-деревьев», которые затем дополнительно сжимаются арифметическим кодером.

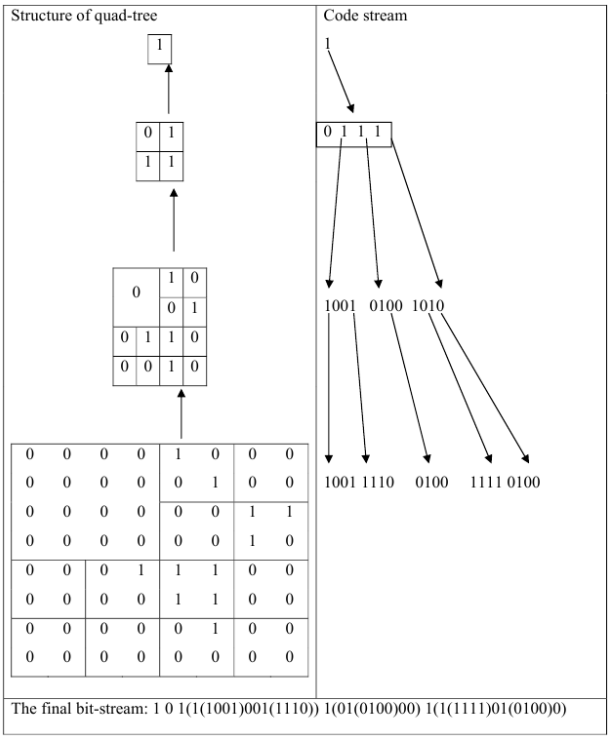


Рис. 2. Построение «нуль-дерева» с помощью кластеризации блока в методе SPECK

Основным недостатком SPECK, помимо использования арифметического кодера, являются блоки переменной размерности, по которым строятся карты значимости («нуль-деревья»). Поэтому в алгоритме HBCT используются блоки фиксированной размерности (оптимальная размерность 32×32). Кроме этого размер кластера 2×2 является наиболее эффективным для сбора статистики по блоку, поэтому в HBCT также используются квадродеревья, од-

нако введена адаптация по количеству уровней подобных деревьев, что также используется в MECT и позволяет приблизиться по уровню сжатия к арифметическому кодеру.

Алгоритм HBCT состоит из следующих шагов.

Шаг 1. Выполнение ДВП над входным полутоновым аэрокосмическим изображением в зависимости от режима сжатия (целочисленные вейвлеты CDF 5/3 в случае сжатия без потерь или рациональные вейвлеты CDF 9/7 в случае сжатия с потерями).

Шаг 2. Сканирование матрицы вейвлет-коэффициентов *Z*-разверткой в одномерный массив.

Шаг 3. Вычисление максимального числа бит $B_{\max}$ по заданному коэффициенту сжатия и инициализация маски *MS* (в MECT – M_{SP}) нулями.

Шаг 4. Определение номера старшей битовой плоскости $n_{\max}$ и запись его в начало выходного прогрессивного потока *PS* (в MECT − M_{CD}). Текущее количество бит потока B_{PS} равно восьми. Текущий номер битовой плоскости *n* равен $n_{\max}$.

Шаг 5. Чтение из одномерного массива вейвлет-коэффициентов одного или нескольких блоков бит BP_j размером 1024 по текущему номеру битовой плоскости. Чтение соответствующих участков (размером 1024) маски MS_j и плоскости знаков SP_j (находится перед старшей битовой плоскостью в одномерном массиве).

Шаг 6. Разбиение битовых блоков на четыре равные части (размером 256) и их бинарная кластеризация [6] с параметрами $N_C = 4$ и $L_I = 4$.

Шаг 7. Вычисление длины вложенного кода по сформированным на шаге 6 бинарным кластерным деревьям для уровней 0, 1 и 4 (L_{EC0}, L_{EC1} и L_{EC4}). Вложенный код нулевого уровня эквивалентен исходному битовому блоку BP_j, поэтому его длина L_{EC0} равна 1024.

Шаг 8. Оценка компактности вложенного кодирования путем выбора наименьшего значения из L_{EC0}, L_{EC1} и L_{EC4}. Формирование флага уровней вложенного кода *F* (соответственно «11», «10» и «01»/«00») в зависимости от выбранного значения длины. Запись флага в поток *PS* и увеличение значения B_{PS} на два.

Шаг 9. Формирование вложенного кода на основе бинарных кластерных деревьев в соответствии с полученным на шаге 8 флагом *F* для частей битового блока BP_j (соответственно EC_0, EC_1, EC_2 и EC_3).

Шаг 10. Формирование кода знаков на основе частей битового блока BP_j, маски MS_j и плоскости знаков SP_j ((соответственно SC_0, SC_1, SC_2 и SC_3).

Шаг 11. Последовательная запись пар сегментов кода EC_i и SC_i в поток *PS*, увеличение значения B_{PS} на соответствующее количество бит и проверка выполнения условия $B_{PS} < B_{\max}$. В случае невыполнения условия − переход к шагу 13.

Шаг 12. Проверка на конец битовой плоскости, в случае истины – уменьшение *n* на единицу. Если *n* меньше нуля, то переход на шаг 13, иначе – переход на шаг 5.

Шаг 13. Завершение работы алгоритма HBCT. Освобождение памяти.

Выходной поток кодера HBCT (рис. 3) является хорошо структурированным, поэтому существует возможность его разбиения на *M* пакетов на этапе формирования прогрессивного потока. Разбивать можно по битовым плоскостям либо по иному критерию. При этом важность пакетов для восстановления исходного изображения (отсутствие более важных пакетов приводит к большей среднеквадратичной ошибке) убывает прямо пропорционально порядковому номеру пакета. Этот факт дает возможность использовать неравномерные коды для помехоустойчивого кодирования. Например, более важные пакеты кодируются с большей избыточностью, чем менее важные. Это обеспечивает экономию общего битового бюджета и повышение защищенности от помех.

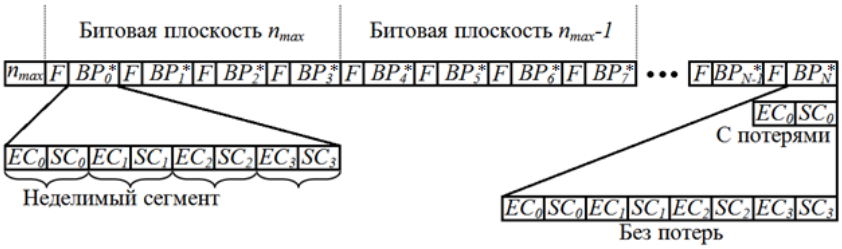
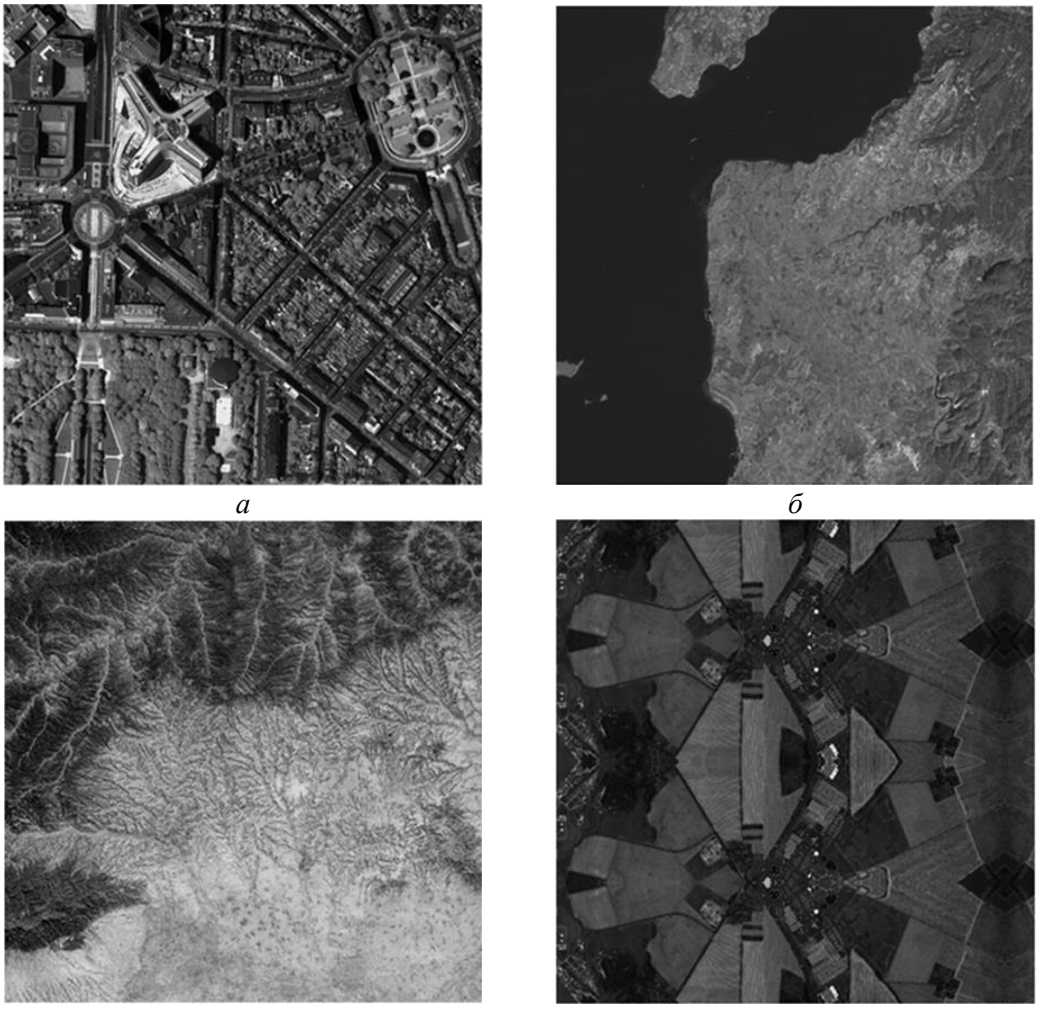


Рис. 3. Структура выходного потока в алгоритме HBCT

Оценка эффективности алгоритмов сжатия

На рис. 4 представлены четыре тестовых аэрокосмических изображения, а в табл. 1-4 – численные характеристики алгоритмов JPEG2000, МЕСТ и НВСТ для этих изображений.



а *б*

в *г*

Рис. 4. Тестовые аэрокосмические изображения:
а – «City»; *б* – «Sealine»; *в* – «Mountains»; *г* – «Plains»

Таблица 1. **Численные результаты алгоритмов для изображения «City»**

Алгоритм	Режим сжатия	Без потерь		Потери, 2 bpp		Потери, 1 bpp		Потери, 0.5 bpp	
JPEG2000	Семейство вейлетов	CDF 5/3		CDF 9/7		CDF 9/7		CDF 9/7	
	Схема, уровни ДВП	Пирамида, 5		Пирамида, 5		Пирамида, 5		Пирамида, 5	
	CR	1,232		4,027		8,057		16, 149	
	PSNR, дБ	∞		**31,680**		**25,999**		**22,656**	
МЕСТ	Семейство вейлетов	CDF 5/3		CDF 9/7		CDF 9/7		CDF 9/7	
	Схема, уровни ДВП	Пирамида, 7		Пирамида, 7		Пирамида, 7		Пирамида, 7	
	CR	1,186		3,998		7,995		15,981	
	PSNR, дБ	∞		**28,204**		**23,690**		**21,822**	
НВСТ	Семейство вейлетов	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	1,203	1,189	3,999	3,998	7,995	7,999	15,977	15,998
	PSNR, дБ	∞	∞	**29,105**	**28,674**	**24,220**	**23,917**	**21,989**	**21,856**
	Семейство вейлетов	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	1,201		3,998	3,999	7,998	7,999	15,999	15,999
	PSNR, дБ	∞		**26,554**	**28,430**	**21,584**	**23,787**	**20,001**	**21,257**

При вычислении 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 установлено, что алгоритм HBCT выигрывает 0,6 дБ у алгоритма MECT и проигрывает 1 дБ алгоритму JPEG2000 (кодек Jasper) независимо от типа изображения.

Таблица 2. **Численные результаты алгоритмов для изображения «Sealine»**

Алгоритм	Режим сжатия	Без потерь		Потери, 2 bpp		Потери, 1 bpp		Потери, 0.5 bpp	
JPEG2000	Семейство вейлетов	CDF 5/3		CDF 9/7		CDF 9/7		CDF 9/7	
	Схема, уровни ДВП	Пирамида, 5		Пирамида, 5		Пирамида, 5		Пирамида, 5	
	CR	1,786		4,018		8,083		16, 076	
	PSNR, дБ	∞		**42,410**		**35,050**		**31,298**	
MECT	Семейство вейлетов	CDF 5/3		CDF 9/7		CDF 9/7		CDF 9/7	
	Схема, уровни ДВП	Пирамида, 7		Пирамида, 7		Пирамида, 7		Пирамида, 7	
	CR	1,636		3,998		7,995		15,982	
	PSNR, дБ	∞		**39,437**		**33,620**		**30,339**	
HBCT	Семейство вейлетов	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	1,721	1,655	3,998	3,998	7,997	7,994	15,999	15,981
	PSNR, дБ	∞	∞	**41,027**	**40,132**	**34,259**	**33,767**	**30,659**	**30,416**
	Семейство вейлетов	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	1,714		3,999	3,998	7,998	7,997	15,984	15,979
	PSNR, дБ	∞		**39,625**	**40,554**	**32,916**	**33,882**	**28,518**	**30,458**

Таблица 3. **Численные результаты алгоритмов для изображения «Mountains»**

Алгоритм	Режим сжатия	Без потерь		Потери, 2 bpp		Потери, 1 bpp		Потери, 0.5 bpp	
JPEG2000	Семейство вейлетов	CDF 5/3		CDF 9/7		CDF 9/7		CDF 9/7	
	Схема, уровни ДВП	Пирамида, 5		Пирамида, 5		Пирамида, 5		Пирамида, 5	
	CR	1,186		4,027		8,104		16, 172	
	PSNR, дБ	∞		**30,066**		**25,183**		**22,748**	
MECT	Семейство вейлетов	CDF 5/3		CDF 9/7		CDF 9/7		CDF 9/7	
	Схема, уровни ДВП	Пирамида, 7		Пирамида, 7		Пирамида, 7		Пирамида, 7	
	CR	1,172		3,998		7,995		15,980	
	PSNR, дБ	∞		**27,729**		**24,075**		**22,036**	
HBCT	Семейство вейлетов	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	1,182	1,172	3,998	3,999	7,996	7,991	15,979	15,981
	PSNR, дБ	∞	∞	**28,438**	**28,141**	**24,348**	**23,856**	**22,329**	**22,278**
	Семейство вейлетов	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	1,178		3,998	3,999	7,998	7,999	15,993	15,992
	PSNR, дБ	∞		**26,195**	**28,203**	**20,983**	**24,387**	**20,034**	**22,237**

Таблица 4. **Численные результаты алгоритмов для изображения «Plains»**

Алгоритм	Режим сжатия	Без потерь		Потери, 2 bpp		Потери, 1 bpp		Потери, 0.5 bpp	
JPEG2000	Семейство вейлетов	CDF 5/3		CDF 9/7		CDF 9/7		CDF 9/7	
	Схема, уровни ДВП	Пирамида, 5		Пирамида, 5		Пирамида, 5		Пирамида, 5	
	CR	1,502		4,037		8,074		16, 207	
	PSNR, дБ	∞		**38,177**		**33,269**		**30,394**	
MECT	Семейство вейлетов	CDF 5/3		CDF 9/7		CDF 9/7		CDF 9/7	
	Схема, уровни ДВП	Пирамида, 7		Пирамида, 7		Пирамида, 7		Пирамида, 7	
	CR	1,450		3,999		7,996		15,982	
	PSNR, дБ	∞		**36,454**		**32,322**		**29,267**	
HBCT	Семейство вейлетов	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	1,464	1,438	3,998	3,999	7,994	7,992	15,997	15,992
	PSNR, дБ	∞	∞	**37,218**	**36,656**	**32,672**	**32,299**	**29,703**	**29,541**
	Семейство вейлетов	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	1,459		3,998	3,999	7,992	7,995	15,996	15,981
	PSNR, дБ	∞		**36,169**	**36,696**	**31,314**	**32,174**	**27,244**	**29,370**

Заключение

Разработан простой и эффективный алгоритм вейвлет-сжатия HBCT, который обладает низкой вычислительной сложностью и высокой степенью параллелизма, что делает его хорошо пригодным для реализации в кристалле ПЛИС. Сжатие с потерями тестовых аэрокосмических изображений показало, что по метрике PSNR алгоритм HBCT выигрывает у алгоритма MECT порядка 0,6 дБ и проигрывает 1 дБ у алгоритма JPEG2000.

HALF-TONE IMAGE COMPRESSION BASED ON PROGRESSIVE EMBEDDED ENCODING OF WAVELET-COEFFICIENTS

V.V. NAVITSKI, V.YU. TSVIATKOU

Abstract

The new algorithm of image wavelet-compression has been developed, 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 quality of compression with losses.

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.* // IEEE Transactions on Image Processing. 2000. Vol. 9. №7. P. 1158-1170.
4. *Pearlman W.A., Islam A., Nagaraj N. et al.* // IEEE transactions on circuits and systems for video technology. 2004. Vol. 14. №11. P. 1219-1235.
5. *Chrysafis C., Said A., Drukarev A. et al.* // IEEE Int. Conf. Acoust., Speech Signal Processing (ICASSP). 2000. Vol. 4. P. 2035-2038.
6. *Борискевич А.А., Цветков В.Ю.* // Докл. НАН Беларуси. 2009. Том 53. №3. С. 38-48.

УДК 654.1.02:004.357

АНАЛИЗ ПРОИЗВОДИТЕЛЬНОСТИ КОДЕКОВ ВИДЕОКОНФЕРЕНЦСВЯЗИ

Н.М. ФИЛОНЧИК, Ж. ИБРАГИМ, И.И. АСТРОВСКИЙ

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

Поступила в редакцию 29 ноября 2015

Приведены результаты анализа производительности кодеков, используемых при передаче видеоинформации и проведении видеоконференцсвязи: H.264 Advanced Video Coding (AVC), H.264 High Profile, H.264 Scalable Video Coding (SVC), Real-Time Video (RTV), H.265 High-Efficiency Video Coding (HEVC) и VP8.

Ключевые слова: видеоконференцсвязь, видеокодек, проприетарные решения передачи видеоинформации.

Введение

На современном этапе развития компьютерных и телекоммуникационных технологий каждое предприятие уже способно иметь свою компьютерную сеть, как для внутреннего сообщения и взаимодействия, так и для выхода в интернет и коммуникаций с представителями других предприятий, клиентами и заказчиками. В последнее время многие предприятия и организации оснащаются видеоконференцсвязью в связи с достоинствами этого вида коммуникаций и приемлемыми затратами на ее организацию. Уже сейчас для ее организации требуется только наличие вебкамеры, кроме непосредственно самого канала связи, и компьютера. Большинству современных мобильных телефонов, кроме наличия 3G-соединения для организации видеосвязи, уже ничего не нужно. Сложность может возникнуть исключительно при организации видеоконференций [1]. Следует понимать, что термины видеосвязь и видеоконференцсвязь в эпоху IP-телефонии означают одно и то же, поскольку возможности современного сетевого оборудования позволяют организовать видеоконференцсвязь высокой четкости с участием десятков оконечных устройств, генерирующих и получающих мультимедийные данные. Даже связь всего только двух абонентов уже является видеоконференцией.

Организация видеоконференцсвязи

Видеоконференцсвязь (сокращенное название ВКС) – это телекоммуникационная технология интерактивного взаимодействия двух и более удаленных абонентов, при которой между ними возможен обмен аудио- и видеоинформацией в реальном масштабе времени с учетом передачи управляющих данных. Другими словами, ВКС дает возможность слышать, видеть, обмениваться данными со своими сотрудниками в режиме онлайн. Таким образом, можно проводить виртуальные встречи нескольких сотрудников одной компании, филиалы которой могут находиться на большом расстоянии друг от друга. Это позволяет существенно сократить расходы на командировки. Помимо этого видеоконференции позволяют более качественно проводить дистанционное обучение, тренинги и семинары [2].

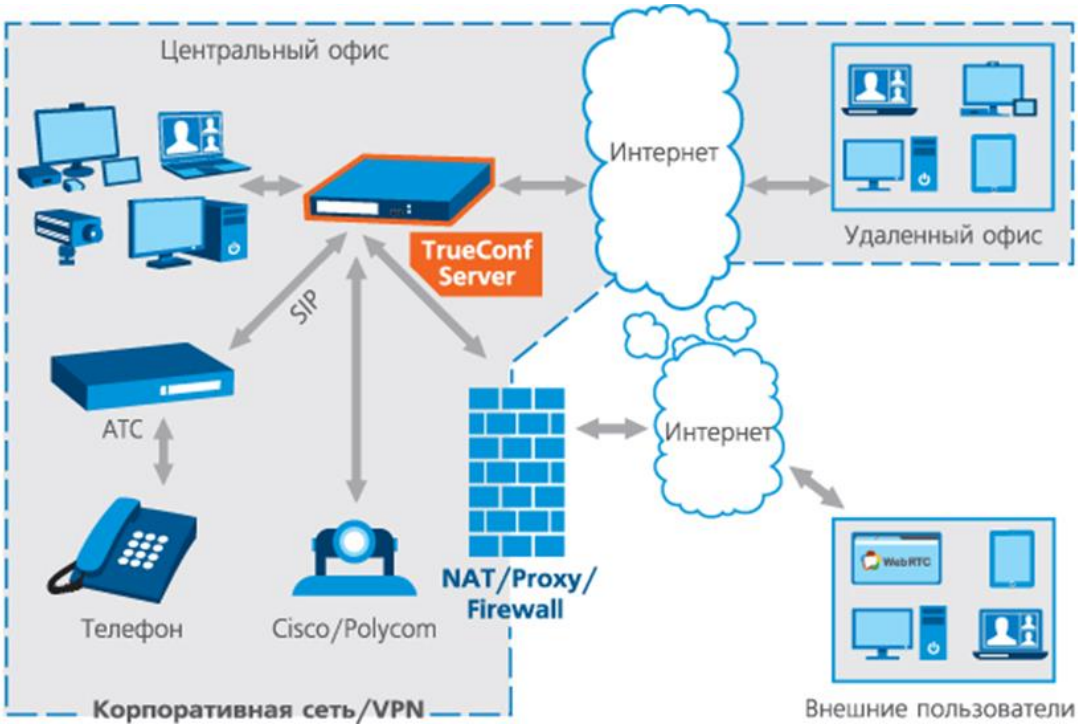


Рис. 1. Схема организации видеоконференцсвязи

Весь спектр решений для видеоконференцсвязи, присутствующих сегодня на рынке, разделяется на две большие категории:

- системы видеоконференцсвязи, основанные на проприетарных (принадлежащих одной компании и, как правило, запатентованных) кодеках;
- системы видеоконференцсвязи с кодеками, основанными на стандартах.

Хотя проприетарные решения на основе инновационных разработок, произведенных в одной компании, могут показаться революционными, этот путь часто приводит к закрытости решения, что, по большому счету, снижает преимущества над системами, разработанными на основе промышленных стандартов.

Организации, которые инвестируют в проприетарные разработки, могут нести значительные расходы. К тому же они привязаны к точке зрения производителя и лишаются, таким образом, всей широты возможностей, доступных при работе с системами, основанными на стандартах, которые были разработаны широким сообществом экспертов.

Технологии, которые могут взаимодействовать между собой благодаря совместимости стандартов, используют инвестиции как внутри своего предприятия, так и вне его рамок [3].

Таким образом, для обеспечения расширяемости инвестиций в технологии видеоконференцсвязи решающее значение имеет то, что решения видеосотрудничества способны взаимодействовать с другими решениями между предприятиями без организационных и корпоративных границ. Чтобы обеспечить совместимость с продукцией других производителей, решения об организации видеоконференцсвязи целесообразно строить на основе промышленных стандартов.

Анализ производительности видеокодеков

Существует множество факторов, которые следует учитывать при выборе решения для видеоконференцсвязи и выборе оборудования, состав которого, в основном, определяется типом кодека.

На рис. 2 приведена попытка суммировать основные критерии по выбору кодека видеоконференцсвязи. Баллы, присвоенные каждому из кодеков, начислены из реальных соображений по использованию кодеков на сегодняшнем рынке видеоконференцсвязи. Единственным исключением из этого правила является кодек H.265, по причине того, что в настоящее время не производятся системы видеоконференцсвязи, использующие этот кодек.

Рис. 2. Таблица анализа кодеков

На данный момент производители решений для видеоконференцсвязи используют следующие кодеки: H.264 Advanced Video Coding (AVC), H.264 High Profile, H.264 Scalable Video Coding (SVC), Real-Time Video (RTV), H.265 High-Efficiency Video Coding (HEVC) и VP8 [4].

Видеокодек – это не что иное, как устройство или программное обеспечение для компрессии и декомпрессии цифрового видеосигнала. Однако, как и во всех других случаях жизни, именно детали выбора одного кодека вместо другого определяют правильность принятого решения. Здесь необходимо учитывать целый комплекс факторов, включая: баланс между качеством видео и скоростью передачи данных, сложность алгоритмов кодирования и декодирования, защиту от потери данных и возможность коррекции ошибок, временную задержку сигнала от точки до точки и еще множество факторов, по которым одни кодеки существенно отличаются от других.

В соответствии со сравнительной таблицей из кодеков, производимых на сегодняшний день, наилучшую оценку получает H.264 AVC. Это связано с тем, что наибольшее количество систем видеоконференцсвязи поддерживают кодек H.264 AVC.

Кроме того, H.264 AVC до сих пор эффективно реализуется в системах ВКС. В самом деле, во многих случаях, когда кодек H.264 AVC был внедрен в высокопроизводительные системы, например, в системы Cisco TelePresence, было показано его превосходство над кодеком H.264 High Profile.

Заключение

Из таблицы (рис. 2) видно, что кодек H.265 имеет значительные перспективы на будущее. Однако для полной реализации потенциала этого кодека и преобразования индустрии видеоконференцсвязи потребуется время. Тем не менее, работа, проделанная для стандартизации сигналов и улучшения существующих стратегий по реализации более старого кодека – H.264 SVC, сможет обеспечить его применение еще достаточно долго, поскольку группа 264-х кодеков распространена повсеместно и соответствует большинству требований.

VIDEOCONFERENCE CODEC PERFORMANS ANALYSIS

N.M. FILONCHIK, J. IBRAHIM, I.I. ASTROVSKY

Abstract

The analysis results of the codecs performance used in the transmission of video information and conduct video conferencing: H.264 Advanced Video Coding (AVC), H.264 High Profile, H.264 Scalable Video Coding (SVC), Real-Time Video (RTV), H.265 High-Efficiency Video Coding (HEVC) and VP8 are given.

Список литературы

1. О государственной программе информатизации Республики Беларусь на 2010 годы и на перспективу до 2015 года. Постановление РБ. 10.05.2009. Мн., 2009.
2. *Артамонова Е.В.* Видеоконференции. Нужны ли они вашей компании и с чего их начать. М., 2005.
3. Cisco Unified Conferencing: Codec Technology Overview (Simulcast SVC). Вашингтон, 2012.
4. Conformance specification for ITU-T H.264 [Электронный ресурс]. Режим доступа: http://www.itu.int/rec/T-REC-H.264.1/en.

УДК 512.624.95:378.147.091.3

ТРЕХРАЗРЯДНЫЙ ВАРИАНТ АЛГОРИТМА «BABY-STEP GIANT-STEP» В ПРОБЛЕМЕ ДИСКРЕТНОГО ЛОГАРИФМИРОВАНИЯ

В.А. ЛИПНИЦКИЙ, Т.Г. КРУПЕНКОВА

Военная академия Республики Беларусь
Минск-57, 220057, Беларусь

Поступила в редакцию 2 октября 2015

Наряду с методом дискретного логарифмирования Д. Шенкса «шаг ребенка – шаг гиганта», авторы предлагают трехразрядный вариант «шага гиганта» для расшифровки сообщений в криптосистеме Эль Гамаля.

Ключевые слова: криптографическая система Эль Гамаля, проблема дискретного логарифмирования, Шенкс Д., алгоритм «шаг ребенка – шаг гиганта».

Введение и цель работы

Современная криптография имеет точную дату своего рождения – выход в свет в 1976 г. статьи [1] Уитни Диффи и Мартина Хелмана. Здесь авторы предложили три революционные идеи: идею применения открытых ключей, методику выработки общего секретного ключа в открытых каналах связи, идею односторонних функций. Это функции, вычисление значений которых не представляет особых трудностей, но вычисление значений обратных к ним функций практически невозможно без дополнительных, секретных данных. Диффи У. и Хелман М. предложили две кандидатуры на роль таких функций:

1) произведение двух больших натуральных чисел (обращение которого упирается в проблему факторизации больших натуральных чисел);

2) вычисление большой степени элементов в кольце классов вычетов по большому простому модулю (обращение которой сталкивается с проблемой дискретного логарифмирования).

Первая из названных функций сразу же оказалась в основе криптосистемы RSA [2], наверное, самой известной из современных криптографических систем, хотя и весьма вязкой даже для легальных пользователей [3, 4]. Вторая обеспечивает криптографическую стойкость криптосистемы Эль Гамаля [5], появившейся в 1985 г. как реакция на излишнюю сложность криптосистемы RSA. Легкая и элегантная в применении, она стала прототипом стандартов шифрования во многих странах мира, в том числе в России, а также в Республике Беларусь [6].

Строго математически существование односторонних функций не доказано [7]. Тем не менее, следует отметить, что ведутся исследования по разработке и применению в криптографии и других кандидатов на односторонние функции. Из наиболее экстравагантных направлений, пожалуй, следует назвать криптосистемы на помехоустойчивых кодах [8, 9].

В 1971 г. Даниэл Шэнкс на симпозиуме по чистой математике американского математического общества прочитал доклад [10], содержавший названный выше метод «baby-step giant-step». И этот метод и доклад знали и цитировали ученые во всем мире.

В 1978 г. Стивен Полиг и Мартин Хелман в работе [11], опираясь на факторизацию мультипликативной группы кольца классов вычетов в произведение циклических подгрупп, построили дальнейшее развитие метода «baby-step giant-step». Вскоре выяснилось, что этот же метод независимо и ранее изобрел и развил (но не опубликовал вовремя) американский математик Роланд Силвер.

В 1994 г. появилась работа [12] московского математика В.И. Нечаева. В работе утверждались следующие факты:

а) метод малых и больших шагов, именуемый на Западе как «baby-step giant-step», в Советском Союзе известен с 1962 г. и был открыт советским математиком А.О. Гельфондом;

б) метод Силвера-Полига-Хелмана был открыт еще в 1965 г. самим В.И. Нечаевым;

в) публикуемый в статье результат получен им еще в 1972 г.;

г) результат этот утверждает, что методы «baby-step giant-step» и Силвера-Полига-Хелмана в определенном смысле являются наилучшими среди детерминированных алгоритмов решения проблемы дискретного логарифмирования. В своей последней книге [13], с. 67, В.И. Нечаев полностью и категорично подтверждает свои слова.

Уважая труд ученых-первооткрывателей, авторы считают, что обсуждаемый метод должен носить название: «метод Нечаева-Силвера-Полига-Хелмана», или сокращенно: «метод НСПХ».

Алгоритм «baby-step giant-step» базируется на искусном использовании двузначного представления искомого логарифма. В данной работе рассматриваются многозначное, а точнее, трехзначное представление искомой степени, достоинства и недостатки такого подхода.

Проблема дискретного логарифмирования

Пусть $p-$ простое число, $p>2$. Мультипликативная группа Z/pZ^* кольца Z/pZ классов вычетов по модулю p является циклической порядка $p-1$ [13, 14]. Каждый элемент a этой группы имеет свой мультипликативный порядок $l(a)-$натуральное число, удовлетворяющее следующим четырем условиям:

1) $1 \le l(a) \le p-1$;

2) $l(a)-$делитель числа $p-1$;

3) $a^{l(a)}=1$;

4) среди степеней $a, a^2, ..., a^{l(a)}=1$ не имеется одинаковых (они и образуют циклическую подгруппу $<a>$ в группе Z/pZ^*, порожденную элементом a).

Если два класса вычетов $a, b \in Z/pZ^*$ связаны соотношением $a^x=b$ для некоторого целого x, и если $r-$остаток от деления x на $l(a)$, то $a^x=a^r=b$; при этом число r называют (дискретным) $\log_a b$ по модулю p.

Самый быстрый способ вычисления конкретной m-й степени элемента $a \in Z/pZ^*$, $a \ne 1$, $1<m<l(a)$ известен со времен Лейбница и заключается в последовательном вычислении квадратов $a, a^2, (a^2)^2=a^4, ...$ с последующим составлением из них степени a^m в соответствии с представлением числа m в двоичной системе счисления. Сложность такого возведения в m-ю степень оценивается в $O(\log_2 m)$ умножений [14, 15].

Обратная задача нахождения степени x из соотношения $a^x=b$ по известным $a, b \in Z/pZ^*$ несравнимо сложнее. На сегодняшний день не известно детерминированного алгоритма ее решения, исключающего перебор. Прямой перебор степеней a или метод малых шагов требует $O(p)$ умножений. Алгоритм малых и больших шагов уменьшает эту величину до $O(\sqrt{p})$.

Суть алгоритма «baby-step giant-step»

В задаче дискретного логарифмирования порядок $l(a)$ элемента a не всегда известен. Поэтому предполагаем, что $l(a)$ имеет максимальное значение $p-1$. Тем более, что в группе Z/pZ^* имеется достаточно много таких элементов (точное их число определяет функция Эйлера: $\varphi(p-1)$).

Пусть d – наименьшее натуральное число, такое, что $\sqrt{l(a)} \le d$. По теореме о делении с остатком $x = d \cdot Q + r$ для некоторых целых Q и r, таких, что $0 \le r < d$, $0 \le Q < d$. Тогда соотношение $b \equiv a^x \pmod p = a^{Qd} a^r \pmod p$ эквивалентно сравнению

$$b \cdot (\mathrm{a}^{-d})^Q \equiv a^r \pmod p . \qquad (1)$$

«Baby step giant step algorithm» заключается в поиске пары целых чисел Q, r, удовлетворяющих условиям $0 \le r < d$, $0 \le Q < d$ и соотношению (1). Поиск реализуется в два этапа. Первый этап – «baby-step» – состоит в составлении таблицы степеней a^i, $2 \le i \le d$. Если в процессе ее составления не встретится значение a^i, равное b, то следует перейти ко второму этапу – «giant-step». Для этого необходимо с помощью расширенного алгоритма Евклида вычислить $a^{-d} \pmod p$. Второй этап состоит в последовательном вычислении величин $b \cdot (a^{-d})^j \pmod p$, $1 \le j < d$, и в сравнении их с данными таблицы степеней a^i, $2 \le i \le d$. Если на каком-то шаге найдутся Q_0, r_0, удовлетворяющие сравнению $b \cdot (\mathrm{a}^{-d})^{Q_0} \equiv a^{r_0} \pmod p$, тогда однозначно опрелеляем искомое $x = d \cdot Q_0 + r_0$. Ясно, что количество вычислений в таком алгоритме оценивается величиной $O(d) = O(\sqrt{p})$.

Трехразрядный вариант метода малых и больших шагов

Пусть δ – наименьшее натуральное число, такое, что $\sqrt[3]{\gamma} \le \delta$. Искомую величину x в задаче дискретного логарифмирования можно представить в виде: $x = \alpha\delta^2 + \beta\delta + \gamma$ для некоторых целых α, β, γ, удовлетворяющих неравенствам $0 \le \alpha < \delta$, $0 \le \beta < \delta$, $0 \le \gamma < \delta$. Тогда соотношение $b \equiv a^x \pmod p = a^{\alpha\delta^2} a^{\beta\delta+\gamma} \pmod p$ эквивалентно сравнению

$$b \cdot a^{-\alpha\delta^2} \equiv a^x \pmod p = a^{\beta\delta+\gamma} \pmod p . \qquad (2)$$

Первый этап предлагаемого алгоритма – «baby-step» – состоит в составлении таблицы степеней $b \cdot (a^{-\delta^2})^i \pmod p$, $1 \le i < \delta$. Второй этап – «giant-step» – состоит в последовательном вычислении величин $a^{\delta j + k} \pmod p$, $0 \le j < \delta$, $0 \le k < \delta$, и в сравнении их с данными таблицы степеней $b \cdot (a^{-\delta^2})^i \pmod p$, $1 \le i < \delta$. Если на каком-то шаге найдутся $\alpha_0, \beta_0, \gamma_0$, удовлетворяющие сравнению $b \cdot (a^{-\delta^2})^{\alpha_0} \equiv a^{\delta\beta_0+\gamma_0} \pmod p$, то тогда однозначно опрелеляем искомое $x = \alpha_0\delta^2 + \beta_0\delta + \gamma_0$.

Пример 1. Найдем с помощью рассмотренных алгоритмов наименьшее натуральное число x, удовлетворяющее условию $3^x = 691$ в группе $Z/1327Z^*$.

Решение. Непосредственно или же пользуясь данными из книги [14] можно убедиться, что в группе $Z/1327Z^*$ элемент $a = 3$ имеет показатель $l(a) = 1326$. В таком случае $\delta = 37$.

Прямой путь малых шагов приводит к результату: $x = 731$. При решении задачи методом малых и больших шагов придется составить таблицу степеней 3^i из 37 данных. Зная ответ, можно выяснить, что $x = 731 = 37 \cdot 19 + 28$. Это означает, что после вычисления $3^{-37} (\mathrm{mod} 1327)$ расширенным алгоритмом Евклида мы на 19-м шаге составления таблицы значений $691 \cdot (3^{-37})^j (\mathrm{mod}\, 1327)$ получим требуемое решение задачи дискретного логарифмирования. В целом мы затратим 56 умножений по модулю числа $p = 1327$, не считая обращения числа по модулю p.

Найдем решение этой же задачи с помощью трехразрядного «baby step giant step algorithm». В данном случае величина $\delta = 11$. Несложно устанавливается, что в поле $Z/1327Z$

$a^{-1} = \left(\overline{3}\right)^{-1} = \overline{885}$ и $a^{-\delta^2} = \overline{885}^{121} = \overline{927}$. Теперь составляем таблицу значений $b \cdot (a^{-\delta^2})^i \pmod p = 691 \cdot 927^i \pmod{1327}$, $0 \le i < d = 11$.

Таблица 1. **Значения $691 \cdot 927^i \pmod{1327}$, $0 \le i < 11$**

i	0	1	2	3	4	5	6	7	8	9	10
	691	943	995	100	1137	361	243	998	227	763	10

Для выполнения второго этапа алгоритма составляем таблицу значений $a^{\beta j+k} \pmod p$, $0 \le j < \delta$, и сравниваем их с данными табл. 1. Дальнейшие вычисления сведем в табл. 2.

Таблица 2. **Значения $3^{11j+k} \pmod{1327}$, $0 \le j < 11$, $0 \le k < 11$**

j/k	0	1	2	3	4	5
0	1	3	9	27	81	243

Вычисленное $3^{11 \cdot 0+5} \pmod{1327} = 243$ находится в табл. 1 при $i = 6$. Таким образом, найдены: $\alpha_0 = 6, \beta_0 = 0, \gamma_0 = 5$. Следовательно, $x = \alpha_0 \delta^2 + \beta_0 \delta + \gamma_0 = 6 \cdot 121 + 731$.

Для вычисления дискретного логарифма новый алгоритм потребовал вычисления в поле $Z/1327Z$ 17 умножений вместо 730 и 56 соответственно без учета нахождения величины $a^{-\delta^2} = \overline{927}$. Конечно, в общем случае трехразрядный метод малых и больших шагов для своей реализации может потребовать $O(\delta^2)$ вычислений. Однако же приведенный пример наглядно демонстрирует несомненную эффективность трехразрядного метода «baby step giant step algorithm».

Заключение

Трехразрядный метод больших и малых шагов является прямым развитием двухшагового метода «baby step giant step algorithm». Данный метод демонстрирует свою реальную эффективность на многих и достаточно разнообразных примерах. Он должен занять достойное место в арсенале тех, кто занят практическим решение проблемы дискретного логарифмирования.

THREE-STEP VARIANT OF «BABY-STEP GIANT STEP» ALGORITHM IN THE DISCRETE LOGARITHM PROBLEM

V.A. LIPNITSKI, T.G. KRUPENKOVA

Abstract

In addition to Shanks' «baby-step giant step» algorithm of finding discrete logarithms the authors suggest a three-step variant of «giant step» to decrypt the message in the Elgamal cryptosystem.

Список литературы

1. *Diffie W. and Helman M.E.* // IEEE Trans. Inf. Theory. 1976. Vol. 22. P. 644-654.
2. *Rivest R., Shamir A., Adleman L.* // Commun. ACM. 1978. Vol. 21. №2. P. 120-126.
3. *Мао В.* Современная криптография: теория и практика. М., 2005.
4. *Фергюсон Н., Шнайдер Б.* Практическая криптография. М., 2005.
5. *Elgamal T.* // IEEE Trans. Inf. Theory. 1985. Vol. 31. P. 469-472.
6. *Харин Ю.С., Берник В.И., Матвеев Г.В. и др.* Математические и компьютерные основы криптологии. Мн., 2003.
7. *Левин Л.А.* // Проблемы передачи информации. 2003. Том 39. Вып. 1. С. 103-117.
8. *Сидельников В.М.* Теория кодирования. М., 2008.
9. *Костелецкий А.В., Липницкий В.А.* // Проблемы защиты информации. 2011. Вып. 10. С. 57-64.
10. *Shanks D.* // Proc. Symp. Pure Math. 1971. Vol. 20. 415-440.
11. *Pohlig S.C. and Helman M.E.* // IEEE Trans. Inf. Theory. 1978. Vol. 1. №24. P. 106-110.
12. *Нечаев В.И.* // Математические заметки. 1994. Том 55. Вып. 2. С. 91-101.
13. *Нечаев В.И.* Элементы криптографии. Основы теории защиты информации. М., 1999.
14. *Виноградов И.М.* Основы теории чисел. М., 1982.
15. *Липницкий В.А.* Современная прикладная алгебра. Математические основы защиты информации от помех и несанкционированного доступа. Мн., 2006.

УДК 621.391

ПРОГРЕССИВНАЯ СЕГМЕНТАЦИЯ ПОЛУТОНОВЫХ ИЗОБРАЖЕНИЙ НА ОСНОВЕ ВОЛНОВОГО КВАЗИПАРАЛЛЕЛЬНОГО ВЫРАЩИВАНИЯ ОБЛАСТЕЙ

О.М. АЛЬМИЯХИ, В.К. КОНОПЕЛЬКО

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

Поступила в редакцию 20 сентября 2015

Предложен метод прогрессивной сегментации полутоновых изображений на основе волнового квазипараллельного выращивания областей. В отличие от известных методов сегментации, предложенный метод позволяет разделять области с плавными перепадами яркости и адаптироваться к ограниченному времени сегментации.

Ключевые слова: сегментация изображений, выращивание областей.

Введение

Сегментация изображений находит широкое применение при решении различных задач обработки видеоинформации. В некоторых случаях время сегментации может быть ограничено. Возможна также задача сегментации изображений с плавными перепадами яркости. Известные методы сегментации, основанные на формировании областей с использованием водораздела [1], квантования по гистограмме [2], разделении и слиянии областей с использованием квадра-дерева [3], выращивании областей [4], не эффективны в данных условиях. Сегментация с использованием водораздела не обеспечивает выделение плавных перепадов яркости изображений. Сегментация на основе квантования по гистограмме не обеспечивает точное разделение областей из-за присвоения одинаковых номеров сегментам с одинаковой яркостью. Методы на основе разделения и слияния областей с использованием квадра-дерева и на основе выращивания областей позволяют точно сегментировать изображения, но также не позволяют находить границы областей на плавных перепадах яркости, что приводит к ошибкам сегментации. Кроме того, все рассмотренные методы не обеспечивают адаптацию к ограничению на время сегментации. В этой связи актуальна задача разработки метода сегментации изображений, учитывающего перечисленные недостатки.

Целью работы является разработка метода сегментации изображений, позволяющего разделять области с плавными перепадами яркости и адаптироваться к ограниченному времени сегментации.

Метод сегментации изображений на основе волнового выращивания областей

Для прогрессивной сегментации полутоновых изображений предлагается метод на основе волнового квазипараллельного выращивания областей. Сущность метода заключается в квазипараллельном выращивании областей вокруг выбранных начальных точек роста, что обеспечивает автоматическое разделение областей с плавным перепадом яркости, которые известные методы сегментируют с ошибками.

Алгоритм сегментации изображений на основе волнового выращивания областей состоит из следующих шагов.

Шаг 1. Инициализация. На данном шаге осуществляется буферизация изображения и определение порога, определяющего условие присоединения пикселя к выращиваемой области.

Для выбора порога может использоваться гистограмма первых производных значений пикселей по строкам и столбцам. Формируется нулевая матрица сегментации, размер которой равен размеру изображения. Счетчику сегментированных значений присваивается значение 0. Инициализируется нулевой стек коллизий. Указатель стека коллизий устанавливается на 0.

Шаг 2. Начало цикла сегментации. Выбор начальных точек роста областей. Для этого могут использоваться, например, локальные максимумы гистограммы значений пикселей изображения, которые соответствуют нулевым значениям в матрице сегментации.

Шаг 3. Инициализация стеков роста областей. Число стеков совпадает с числом начальных точек роста. В качестве начальных значений в стеки заносятся координаты выбранных начальных точек роста – каждая пара координат в отдельный стек. Указатель каждого стека роста области устанавливается в значение 1. Счетчику сегментированных значений присваивается число начальных точек роста. Элементам матрицы сегментации, координаты которых соответствуют координатам начальных точек роста, присваиваются номера сегментов (каждому элементу присваивается неиспользуемый ранее номер).

Шаг 4. Инициализация счетчика циклов перебора выращиваемых областей (устанавливается в ноль).

Шаг 5. Начало цикла перебора выращиваемых областей. Из стека роста области, номер которого соответствует значению счетчика циклов перебора выращиваемых областей, извлекаются координаты текущего выращиваемого пикселя, на который указывает значение соответствующего указателя стека. Значение указателя стека уменьшается на единицу.

Шаг 6. Инициализация счетчика окрестных пикселей.

Шаг 7. Начало цикла анализа окрестных пикселей. На основе координат текущего выращиваемого пикселя вычисляются координаты текущего окрестного пикселя, номер которого определяется значением счетчика окрестных пикселей.

Шаг 8. Проверяется на ноль значение элемента матрицы сегментации, координаты которого соответствуют найденным координатам окрестного пикселя. Если это значение не равно нулю, то переход на шаг 13, иначе – переход на следующий шаг.

Шаг 9. Абсолютное значение разности значений текущего выращиваемого пикселя и текущего окрестного пикселя сравниваются с заданным порогом. Если абсолютное значение разности меньше порога (окрестный пиксель должен быть присоединен к области), то переход на следующий шаг, иначе – переход на шаг 16.

Шаг 10. Элементу матрицы сегментации, координаты которого соответствуют координатам текущего окрестного пикселя, присваивается значение элемента матрицы сегментации, координаты которого соответствуют координатам текущего выращиваемого пикселя.

Шаг 11. Указатель текущего стека роста области увеличивается на единицу. В текущий стек роста области заносятся координаты текущего окрестного пикселя.

Шаг 12. Значение счетчика сегментированных значений увеличивается на 1. Переход на шаг 16.

Шаг 13. Проверка коллизии. Сравниваются значения элементов матрицы сегментации, координаты которых соответствуют координатам текущего окрестного пикселя и текущего выращиваемого пикселя. Если эти значения равны, то переход на шаг 16, иначе – переход на следующий шаг.

Шаг 14. Абсолютное значение разности значений пикселей изображения, координаты которых соответствуют координатам текущего окрестного пикселя и текущего выращиваемого пикселя, сравниваются с заданным порогом. Абсолютное значение разности значений пикселей изображения, координаты которых соответствуют координатам начальных точек роста рассматриваемых сегментов, также сравниваются с заданным порогом. Если абсолютные значения разностей меньше порога (окрестный пиксель должен быть присоединен к данной области, но уже присоединен к другой области – имеет место коллизия), то переход на следующий шаг, иначе – переход на шаг 16.

Шаг 15. Указатель стека коллизий увеличивается на единицу. Текущему элементу стека коллизий присваиваются координаты текущего окрестного пикселя и текущего выращиваемого пикселя.

Шаг 16. Окончание цикла анализа окрестных пикселей. Счетчик окрестных пикселей увеличивается на единицу. Проверяется неравенство значения счетчика окрестных пикселей и

числа окрестных пикселей (8 пикселей). Если счетчик окрестных пикселей меньше 8, то осуществляется переход на шаг 7, иначе – выход из цикла.

Шаг 17. Окончание цикла перебора выращиваемых областей. Счетчик циклов перебора выращиваемых областей увеличивается на единицу. Проверяется неравенство значения счетчика циклов перебора выращиваемых областей и числа начальных точек роста. Если счетчик циклов перебора выращиваемых областей меньше числа начальных точек роста, то осуществляется переход на шаг 5, иначе – выход из цикла.

Шаг 18. Окончание цикла сегментации. Если счетчик сегментированных значений равен числу пикселей сегментируемого изображения, то выход из цикла (сформирована промежуточная матрица сегментации), иначе – переход на шаг 2.

Шаг 19. Инициализируется результирующая матрица сегментации. Значения матрицы сегментации переносятся в результирующую матрицу сегментации.

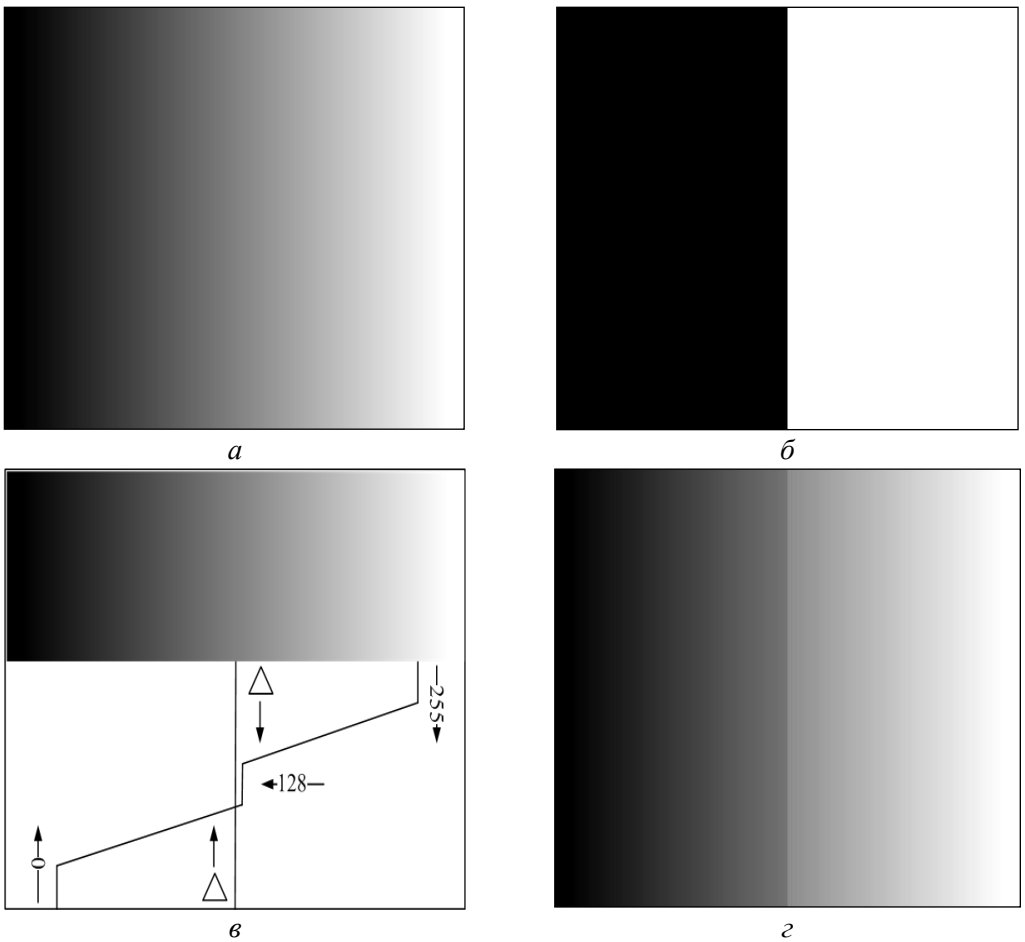
Шаг 20. Проверка указателя стека коллизий на ноль. Если указатель равен нулю, то и осуществляется выход из алгоритма, иначе – переход на следующий шаг.

Шаг 21. Разрешение коллизий. В стеке коллизий отыскиваются связанные номера сегментов, которым присваиваются новые номера. Эти номера заносятся в результирующую матрицу сегментации. Указатель стека коллизий уменьшается на соответствующее число. Когда указатель стека коллизий равен нулю, осуществляется выход из алгоритма.

В результате выполнения данного алгоритма формируется матрица сегментации, значение каждого элемента которой указывает на номер сегмента, которому принадлежит пиксель сегментируемого изображения с соответствующими координатами. С каждым циклом перебора выращиваемых областей размеры сегментов постепенно увеличиваются, в чем проявляется прогрессивный характер сегментации, осуществляемой предложенным методом. В случае прерывания (например при поиске мелких объектов) алгоритма часть изображения останется несегментированной, однако будут найдены все доминирующие центры областей и эти области будут равномерно сегментированы. Это позволяет при необходимости а) интерполировать несегментированные области; б) предсказать положение границ в несегментированных областях; в) определить число, местоположение и оценить размеры объектов интереса.

Оценка эффективности алгоритмов сегментации

На рис. 1 приведено тестовое изображение, содержащее 4 области: 2 прямоугольные контрастные области с постоянной яркостью по краям изображения и 2 прямоугольные смежные области в центре изображения с плавно изменяемой яркостью (рис. 1,*а*). В начале эксперимента яркости смежных областей отличаются от яркости граничных областей на значение порога, в результате чего в центре изображения образуется резкий перепад яркости и два малых перепада яркости по краям изображения (рис. 1,*б*). Сегментация данного изображения всеми методами (предложенным методом и методами на основе выращивания областей, разделения и слияния областей, водораздела) дает одинаковый результат – две сегментированные области. В процессе эксперимента яркости центральных смежных областей плавно меняются по закону, приведенному на рис. 1,*в*, в результате чего перепад яркости в центре изображения на границе этих областей становится менее резким (рис. 1,*г*). Предложенный метод обеспечивает стабильное положение границы и выделение 2 областей при любом значении перепада яркости (от 255 до 0) в центре изображения. Остальные методы при некоторых значениях перепада яркости в центре тестового изображения показывают ошибку сегментации, выделяя на изображении только одну область. Для метода сегментации на основе выращивания областей это значение равно 126. Таким образом, предложенный метод обеспечивает повышение чувствительности сегментации к перепадам яркости на 1,6 % по сравнению с методом сегментации на основе выращивания областей.



а *б* *в* *г*

Рис. 1. Тестовые изображения для оценки чувствительности методов сегментации к перепадам яркости: *а* – четыре области распределения яркости; *б* – изображение с резким перепадом яркости; *в* – закон распределения яркости; *г* – изображение с относительно плавным перепадом яркости

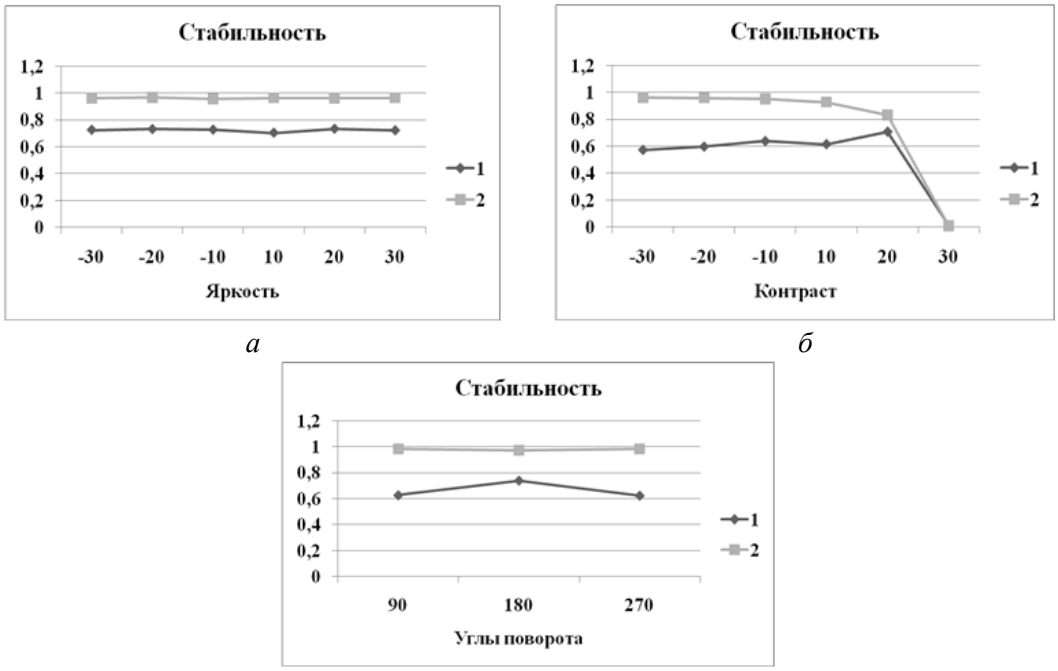
Произведена оценка эффективности предложенного метода прогрессивной сегментации на основе волнового квазипараллельного выращивания областей и известных методов сегментации на основе выращивания областей, разделения и слияния областей, водораздела. В качестве показателей эффективности использованы время сегментации, стабильность границ и числа сегментов.

В табл. приведены результаты оценки времени сегментации для рассматриваемых методов. Из табл. следует, что предложенный метод проигрывает в скорости сегментации до 5 раз при размере изображения 512×512 пикселей; и до 2,5 раз при размере 1024×1024 пикселей методам сегментации на основе выращивания областей.

Время сегментации тестовых изображений, с

Методы сегментации	**Тестовые изображения**		
	Lena 512×512	Barbara 512×512	France 8 1024×1024
Предложенный метод	14,1416	5,8029	88,6354
Выращивания областей	3,0438	0,8005	35,9598

На рис. 2 приведены зависимости площадей сегментов от изменения яркости, контраста и поворота изображения, характеризующие устойчивость границ сегментов. Устойчивость оценивается по отношениям площадей сегментов для базового изображения к площадям сегментов для модифицированного изображения, подвергнутого изменению яркости, контраста и угла поворота. Из рис. 2 следует, что предложенный метод проигрывает в стабильности площадей сегментов до 6,75; 1,76; 9 раз при изменении яркости, контраста и угла поворота по сравнению с методом сегментации на основе выращивания областей.

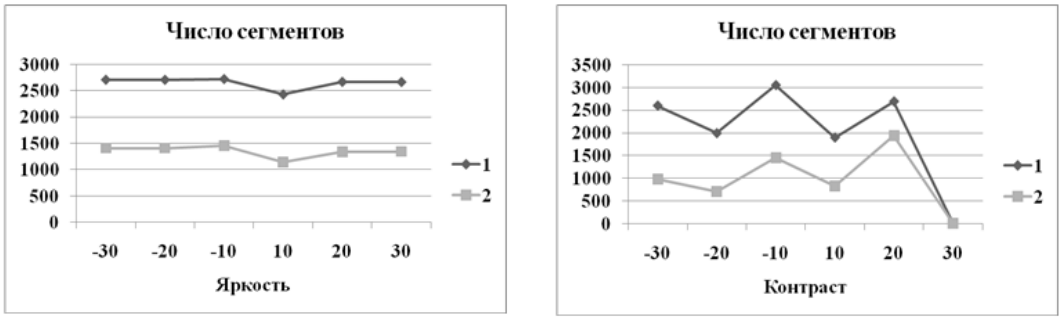


а *б*

в

Рис. 2. Зависимости площадей сегментов от изменения условий формирования изображения (1 – предложенный метод; 2 – выращивания областей): *а* – при изменении яркости; *б* – при изменении контраста; *в* – при изменении угла поворота

На рис. 3 приведены зависимости числа сегментов от изменения яркости, контраста и поворота изображения, характеризующие устойчивость результатов сегментов. Устойчивость оценивается по отношениям числа сегментов для базового изображения к числу сегментов для модифицированного изображения, подвергнутого изменению яркости, контраста и угла поворота. Из рис. 3 следует, что предложенный метод выигрывает в стабильности числа сегментов до 2,4; 2; 2 раз при изменении яркости, контраста и угла поворота по сравнению с методом сегментации на основе выращивания областей.

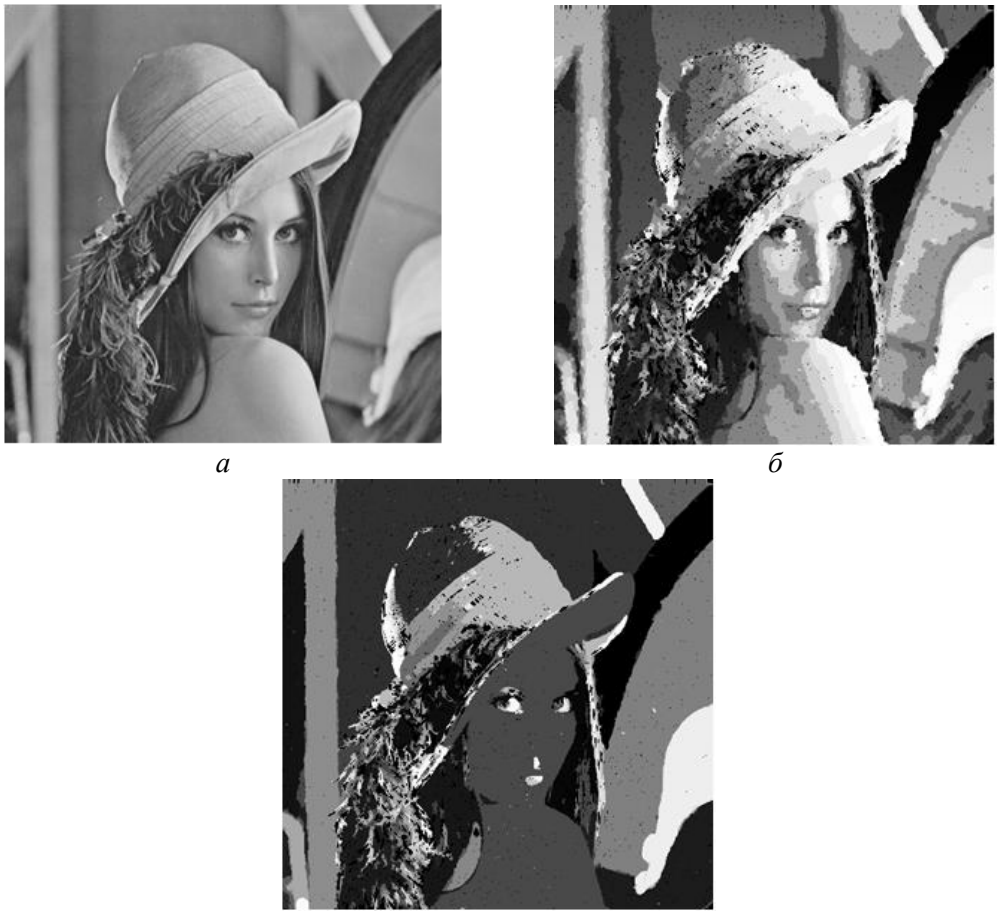


а *б*

Рис. 3. Зависимости числа сегментов от изменения условий формирования изображения (1 – предложенный метод; 2 – выращивания областей): *а* – при изменении яркости; *б* – при изменении контраста

Стабильность числа сегментов к изменению условий формирования изображений является основным достоинством предложенного метода сегментации.

На рис. 4 показаны результаты сегментации стандартного тестового изображения Lena с помощью предложенного метода и метода сегментации на основе выращивания областей.



а *б*

в

Рис. 4. Результаты сегментации стандартного тестового изображения Lena:
а – исходное изображение; *б* – результат для предложенного метода;
в – результат для метода выращивания областей

Заключение

Для прогрессивной сегментации полутоновых изображений предложен метод на основе волнового квазипараллельного выращивания областей. Сущность метода заключается в квазипараллельном выращивании областей вокруг выбранных начальных точек роста, что обеспечивает автоматическое разделение областей с плавным перепадом яркости, которые известные методы сегментируют с ошибками. Показано, что предложенный метод обеспечивает повышение чувствительности сегментации к перепадам яркости в 1,6 % по сравнению с методом сегментации на основе выращивания областей. Установлено, что предложенный метод проигрывает в стабильности площадей сегментов до 6,75; 1,76; 9 раз по сравнению с методом сегментации на основе выращивания областей, но выигрывает в стабильности числа сегментов по сравнению с этими методом до 2,4; 2; 2 раз соответственно. Установлено, что предложенный метод проигрывает в скорости сегментации до 5; 2,5 раз методу сегментации на основе выращивания областей, но обеспечивает прогрессивный характер сегментации, позволяя в условиях ограниченного времени обработки интерполировать несегментированные области; предсказывать положение границ в несегментированных областях; определять число, местоположение и оценивать размеры объектов интереса.

PROGRESSIVE HALFTONE IMAGE SEGMENTATION BASED ON THE WAVE QUASI PARALLEL REGION GROWING

O.M. ALMIAHI, V.K. KANAPELKA

Abstract

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

Список литературы

1. *Lalitha M., Kiruthiga M., Loganathan C.* // International Journal of Science and Research (IJSR). 2013. Vol. 2. №2. P. 348-358.
2. *Chang J.H., Fan K.Ch., Chang Y.L.* // Image and Vision Computing. 2002. Vol. 20. P. 203-216.
3. *Muhsin, Z.F., Rehman A., Altameem A. et. al* // The Imaging Science Journal. 2014. Vol. 62. №1. P. 56-62.
4. *Singh K.K., Singh A.* // International Journal of Computer Science Issues. 2010. Vol. 7. №5. P. 414-417.

УДК 621.391

ЗАЩИТА УДАЛЕННОГО ДОСТУПА К СЕРВЕРУ ОТ АТАКИ МЕТОДОМ «ГРУБОЙ СИЛЫ»

Р.М. ГОРБУЛЬ, Д.Д. ТРОФИМЕНКО, Ф. АЛИХАНОВ

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

Поступила в редакцию 23 октября 2015

Рассмотрена атака методом «грубой силы». Произведена оценка факторов риска атаки. Рассмотрены методики защиты удаленного доступа. Показана эффективность методик защиты от атаки методом «грубой силы».

Ключевые слова: удаленный доступ, метод «грубой силы», защита от атаки методом «грубой силы».

Введение

Задача обеспечения безопасного удаленного доступа к серверу крайне важна. Одной из распространенных атак на службу ssh является атака методом «грубой силы» (метод перебора). Цель данной атаки – удаленно получить доступ к аккаунтам пользователей путем многократных попыток угадать пароль пользователя или группы пользователей. Если веб-сервер не имеет никаких защитных мер против этого типа атак, то злоумышленнику довольно просто взломать систему, основанную на парольной аутентификации, осуществив сотню попыток ввода пароля с помощью автоматизированных программ.

Целью данной работы является исследование методик защиты удаленного доступа по ssh от атаки методом «грубой силы» и их применения.

Атака методом «грубой силы»

Хотя метод «грубой силы» в большинстве прикладных задач (особенно не связанных со взломом) на практике не применяется, но есть ряд исключений. В частности, когда данный метод все же оказывается оптимальным, либо представляет собой начальный этап в разработке алгоритма, его использование оправдано. Примером оптимальности «грубой силой» является алгоритм оценки времени вычисления цепочечных произведений матриц, который не удается ускорить по сравнению с алгоритмом, основанным на методе «грубой силы». Этот алгоритм используется для решения классической задачи динамического программирования – определения приоритетов вычислений матричных произведений следующего вида: $A_1A_2A_3\ldots A_n$.

Исходная задача заключается в вычислении данной цепочки (матричного произведения) за наименьшее время. Можно реализовать тривиальный последовательный алгоритм, вычисляющий искомое произведение. Поскольку матричное произведение является ассоциативной операцией, можно вычислить цепочечное произведение, произвольно выбирая пару элементов цепочки $\left(A_iA_{i+1}\right), i=1\ldots n-1$ и заменяя ее результирующей матрицей $A_i^1 : A_i^1 = \left(A_iA_{i+1}\right)$. Если повторять описанную процедуру $n-1$ раз, то оставшаяся результирующая матрица A_k^{n-1} и будет ответом: $A_k^{n-1} = \left(A_k^{n-2} \cdot A_{k+1}^{n-2}\right) = \ldots = A_1A_2A_3\ldots A_n,\ k=1\ldots n-1.$ Эта формула может быть проиллюстрирована следующим образом. Рассмотрим матричную цепочку: $\langle A_1A_2A_3A_4 \rangle$.

Существуют следующие 5 способов вычисления соответствующего этой цепочке произведения $A_1A_2A_3A_4$:

$\left(A_1\left(A_2\left(A_3A_4\right)\right)\right)$

$\left(A_1\left(\left(A_2A_3\right)A_4\right)\right)$

$\left(\left(A_1A_2\right)\left(A_3A_4\right)\right)$

$\left(\left(A_1\left(A_2A_3\right)\right)A_4\right)$

$\left(\left(\left(A_1A_2\right)A_3\right)A_4\right)$

Выбрав правильный порядок вычислений, можно добиться значительного ускорения вычислений. Чтобы убедиться в этом, рассмотрим простой пример цепочки из 3-х матриц. Положим, что их размеры равны соответственно 10×100, 100×5, 5×50. Стандартный алгоритм перемножения двух матриц размерами $p\times q$, $q\times r$ требует время вычисления, пропорциональное числу pqr (число вычисляемых скалярных произведений). Следовательно, вычисляя цепочку в порядке $\left(\left(A_1A_2\right)A_3\right)$, получаем $10\cdot100\cdot5=5000$ скалярных произведений для вычисления $\left(A_1A_2\right)$, плюс дополнительно $10\cdot5\cdot50=2500$ скалярных произведений, чтобы вычислить второе матричное произведение. Общее число скалярных произведений: 7500. При ином выборе порядка вычислений получаем $100\cdot5\cdot50=25000$ плюс $10\cdot100\cdot50=50000$ скалярных произведений, то есть 75000 скалярных произведений.

Таким образом, решение данной задачи может существенно сократить временные затраты на вычисление матричной цепочки. Это решение может быть получено полным перебором: необходимо рассмотреть все возможные последовательности вычислений и выбрать из них ту, которая при вычислении цепочки занимает наименьшее число скалярных произведений. Однако надо учитывать, что этот алгоритм сам по себе требует экспоненциального времени вычисления, так что для длинных матричных цепочек выигрыш от вычисления цепочки самым эффективным образом (оптимальная стратегия) может быть полностью потерян временем нахождения этой стратегии.

В криптографии на полном переборе основывается криптографическая атака методом «грубой силы». Ее особенностью является возможность применения против любого практически используемого шифра. Однако такая возможность существует лишь теоретически, зачастую требуя нереалистичные временные и ресурсные затраты. Наиболее оправдано использование атаки методом «грубой силы» в тех случаях, когда не удается найти слабых мест в системе шифрования, подвергаемой атаке (либо в рассматриваемой системе шифрования слабых мест не существует). При обнаружении таких недостатков разрабатываются методики криптоанализа, основанные на их особенностях, что способствует упрощению взлома.

Устойчивость к атаке методом «грубой силы» определяет используемый в криптосистеме ключ шифрования. Так, с увеличением длины ключа сложность взлома этим методом возрастает экспоненциально. В простейшем случае шифр длиной в N битов взламывается, в наихудшем случае, за время, пропорциональное $2N$. Среднее время взлома в этом случае в два раза меньше и составляет $2N-1$. Существуют способы повышения устойчивости шифра к атаке методом «грубой силы», например запутывание (обфускация) шифруемых данных, что делает нетривиальным отличие зашифрованных данных от незашифрованных.

Методика защиты от атаки методом «грубой силы»

Для обеспечения безопасного удаленного подключения к серверу необходимо произвести корректные настройки демона ssh, а также встроенного брандмауэра iptables. Также в качестве дополнительной меры безопасности можно использовать программное обеспечение psad и методику port knocking.

Несмотря на то, что протокол ssh использует шифрование, для безопасного использования требуется корректная настройка ssh-сервера. Одним из самых уязвимых мест доступа с

использованием ssh является доступ по пользовательскому паролю. Такой доступ подвержен атакам методикой «грубой силы». Любой веб-сервер или сервер с прямым доступом в интернет ежедневно сканируется на наличие открытых портов с дальнейшим подбором учетных данных с целью получения доступа. Подбор осуществляется, как правило, на основе заранее созданных словарей с наиболее используемыми учетными данными. Для предотвращения несанкционированного доступа настоятельно рекомендуется отказаться от использования парольного доступа и использовать только аутентификацию по публичным ключам. Также не менее важным правилом является запрет удаленного доступа к учетной записи суперпользователя – «root». В качестве дополнительной меры безопасности можно указать список разрешенных пользователей. Также существует ложное правило, которое потенциально является не мерой безопасности, а лишь потенциальной уязвимостью. Правило смены порта ssh – 22, на случайный непривилегированный порт. Данное правило является потенциальной уязвимостью, так как в ОС на основе ядра Linux привилегированные порты могут быть открыты только от имени суперпользователя, а непривилегированные могут использоваться обычными пользователями. При несанкционированном доступе к серверу, злоумышленник сможет установить скрипт, эмулирующий работу ssh-сервера и осуществить кражу паролей пользователей.

Приведенные выше правила можно осуществить следующими директивами в конфигурационном файле /etc/ssh/sshd_conf:

PermitRootLogin no – запрещает аутентификацию суперпользователя «root»,
PasswordAuthentication no – запрещает аутентификацию по паролю,
AllowUsers <users> – разрешает только перечисленных пользователей.

Результат функционирования правил можно увидеть на рис. 1.

Рис. 1. Результат функционирования правил

Эффективной мерой защиты может послужить использование утилиты fail2ban. Данная утилита позволяет установить лимит на количество попыток авторизации посредством ssh, при превышении которого создается политика блокировки ip-адреса на заданный промежуток времени. Для конфигурирования требуется указать следующие настройки в конфигурационном файле /var/fail2ban/jail.local:

bantime = 86400 – блокировка ip-адреса на 24 ч (в сек),
findtime = 120 – время, за которое считаются попытки,
maxretry = 3 – количество попыток,
[ssh] – включает мониторинг сервиса ssh,

enabled = true,
port = ssh,
filter = sshd,
logpath = /var/log/auth.log.

Результат работы утилиты можно увидеть на рис. 2.

```
r[illegible] me:~$ sudo iptables -S
-P INPUT ACCEPT
-P FORWARD ACCEPT
-P OUTPUT ACCEPT
-N fail2ban-nginx-http-auth
-N fail2ban-ssh
-A INPUT -p tcp -m multiport --dports 22 -j fail2ban-ssh
-A fail2ban-ssh -s 80.153.99.58/32 -j REJECT --reject-with icmp-port-unreachable
-A fail2ban-ssh -s 188.116.52.14/32 -j REJECT --reject-with icmp-port-unreachable
-A fail2ban-ssh -s 40.121.48.153/32 -j REJECT --reject-with icmp-port-unreachable
-A fail2ban-ssh -j RETURN
r[illegible] me:~$
```

Рис. 2. Результат работы утилиты

Для защиты от сканирования может быть применена утилита psad. Утилита psad посредством мониторинга логов брандмауэра определяет попытки сканирования или атак на сервер. Принцип работы предельно прост: psad определяет, что с конкретного ip-адреса производится попытка сканирования сервера на открытые порты и по заранее заданному правилу осуществляет блокировку данного ip-адреса. Для конфигурирования требуется указать следующие настройки в файле конфигурации /etc/psad/psad.conf:

EMAIL_ADDRESSES address@domain.com,
HOSTNAME domain.com,
DANGER_LEVEL1 5,
DANGER_LEVEL2 15,
DANGER_LEVEL3 150,
DANGER_LEVEL4 1500,
DANGER_LEVEL5 10000,
PORT_RANGE_SCAN_THRESHOLD 1,
IPT_SYSLOG_FILE /var/log/syslog,
IGNORE_PORTS ports_or_range_to_ignore,
MIN_DANGER_LEVEL 1; # Controls psad logging and email alerts,
EMAIL_ALERT_DANGER_LEVEL 1; # Applies only for email alerts,
EMAIL_LIMIT 0,
ENABLE_AUTO_IDS Y,
AUTO_IDS_DANGER_LEVEL 5,
AUTO_BLOCK_TIMEOUT 3600.

При помощи утилиты nmap можно осуществить попытку сканирования сервера на наличие открытых портов:

sudo nmap -PN -sS server_domain_or_ip
Nmap scan report for server_domain_or_ip
Host is up (0.013s latency).
Not shown: 999 filtered ports
PORT STATE SERVICE
22/tcp open ssh
Nmap done: 1 IP address (1 host up) scanned in 6.84 seconds

Результатом служит наличие новых политик в брандмауэре iptables:

-A FORWARD -j PSAD_BLOCK_FORWARD
-A FORWARD -j LOG
-A OUTPUT -j PSAD_BLOCK_OUTPUT

Методика port knocking относится к так называемому принципу «безопасность через неясность». Принцип данной методики заключается в том, что нужный порт заблокирован политикой брандмауэра и для доступа к нему пользователь должен осуществлять доступ через

строго заданную цепочку портов. Для применения данной методики можно использовать утилиту knockd. Для конфигурирования требуется указать следующие настройки в файле конфигурации /etc/kockd.conf, а также сконфигурировать брандмауэр iptables. Для начала потребуется добавить следующие правила в iptables:

```
sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
sudo iptables -I INPUT 1 -i lo -j ACCEPT
sudo iptables -P INPUT DROP
```

Конфигурация knockd в файле /etc/knockd.conf:

```
[options]
UseSyslog
[openSSH]
sequence = 7000,8000,9000
seq_timeout = 5 command = /sbin/iptables -A INPUT -s %IP% -p tcp --dport 22 -j ACCEPT
tcpflags = syn
[closeSSH] sequence = 9000,8000,7000
seq_timeout = 5
command = /sbin/iptables -D INPUT -s %IP% -p tcp --dport 22 -j ACCEPT
tcpflags = syn
```

Результатом работающего сервиса knockd является недоступность подключения по порту 22 - ssh:

```
ssh <server-ip-address>
sh: connect to host server_ip_address port 22: Operation timed out
```

Чтобы получить доступ к серверу, требуется осуществлять подключение через цепочку портов:

```
knock 7000 8000 9000 && ssh <server-ip-address>
```

Для закрытия порта требуется всего лишь пройти цепочку в обратном порядке:

```
knock 9000 8000 7000
```

Заключение

Проведено исследование методик защиты удаленного подключения к серверу. Для тестирования использовался настроенный веб-сервер с доступом через сеть Интернет. В результате были получены работающие методики защиты удаленного подключения и навыки по их применению. Получена статистика, согласно которой в среднем за день осуществляется 35-40 попыток получения доступа перебором учетных данных. Данная статистика актуальна после применения приведенных выше методик (без использования методики port knocking).

PROTECTION OF REMOTE ACCESS TO SERVER AGAINST BRUTEFORCE ATTACK

R.M. HARBUL, D.D. TRAFIMENKA, F. ALIHANOV

Abstract

Remote access policy against bruteforce attacks was examined during research work of bruteforce attack characteristics and its risk factors. As research results remote access policy is developed and its efficiency against bruteforce attacks is shown.

Список литературы

1. *Олифер В.Г., Олифер Н.А.* Компьютерные сети. Принципы, технологии, протоколы. СПб., 2010.
2. *Таненбаум Э., Уэзеролл Д.* Компьютерные сети. СПб., 2012.
3. Iptables. [Электронный ресурс]. Режим доступа: http://ipset.netfilter.org/iptables.man.html.
4. Port Scan Attack Detector. [Электронный ресурс]. Режим доступа: https://cipherdyne.org/psad/docs/manpages/psad.html.
5. Fail2Ban. [Электронный ресурс]. Режим доступа: http://www.fail2ban.org/wiki/index.php/MANUAL_0_8.
6. OpenSSH. [Электронный ресурс]. Режим доступа: http://www.openssh.com/manual.html.
7. Port knocking manual pages. [Электронный ресурс]. Режим доступа: http://www.portknocking.org.

УДК 528.854.2

ПАРАМЕТРИЗАЦИЯ КРИВЫХ ЛИНИЙ НА ОСНОВЕ АПРОКСИМАЦИИ ОКРУЖНОСТЯМИ

Д.И. КИРИЛЮК, Ю.И. КУЛАЖЕНКО

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

Поступила в редакцию 3 декабря 2015

Приведены примеры различных подходов к параметризации кривых линий. Получена оценка стабильности параметров кривых линий при поворотной трансформации изображения.

Ключевые слова: параметризация кривых линий, оценка стабильности параметров.

Введение

Важными задачами в обработке изображений являются идентификация и параметризация линий. Прямые линии встречаются на изображении достаточно редко, поэтому стоит более общая задача – параметризовать произвольную линию на изображении.

Подходы к параметризации кривых линий

При параметризации кривых линий известны следующие подходы.

1. Приближение кривых линий кривыми, свойства которых хорошо изучены, например кривыми второго порядка, в частности, окружностями или эллипсами [1].

2. Аппроксимация функции кривой сплайнами различных степеней [2].

3. Разложения в ряды Фурье или по другим ортогональным системам [3].

4. Методы Хафа [4].

Для того, чтобы применить второй и третий подходы, необходимо рассмотреть кривую как функцию, что является трудоемким процессом при работе с большим количеством кривых на изображении. Сложность заключается в выборе системы координат, и разбитии кривой на подкривые для того, чтобы выполнились условия существования функции. После этого в случае второго подхода необходимо зафиксировать некоторое количество точек, и в зависимости от этого количества выбрать степень апроксимирующего сплайна. Далее необходимо решить систему линейных алгебраических уравнений относительно коэффициентов сплайна. Методы Хафа не обладают достаточной точностью. Первый подход является менее ресурсозатратным и достаточно эффективным. Поэтому в дальнейшем будет дана оценка стабильности параметризации именно на его основе.

Оценка стабильности характеристических параметров кривых линий

Рассмотрены две тестовые кривые на изображении размером 68×68 (см. рис. 1).

а *б*

Рис. 1. Тестовые кривые: *а* – кривая 1, *б* – кривая 2

Построены окружности малого радиуса, аппроксимирующие данные кривые. Для кривой 1 – 18 окружностей, для кривой 2 – 22. В качестве параметров использованы длины отрезков, соединяющие центры окружностей, и углы, образованные этими отрезками. Были вычислены параметры этих кривых при повороте изображений на 15, 30, 45, 60, 90, 120, 135, 150 и 180 градусов. Для оценки стабильности была вычислена среднеквадратическая ошибка (MSE), полученных параметров. Результаты приведены на графике (по длинам отрезков – рис. 2, 3, по углам – рис. 3, 4).

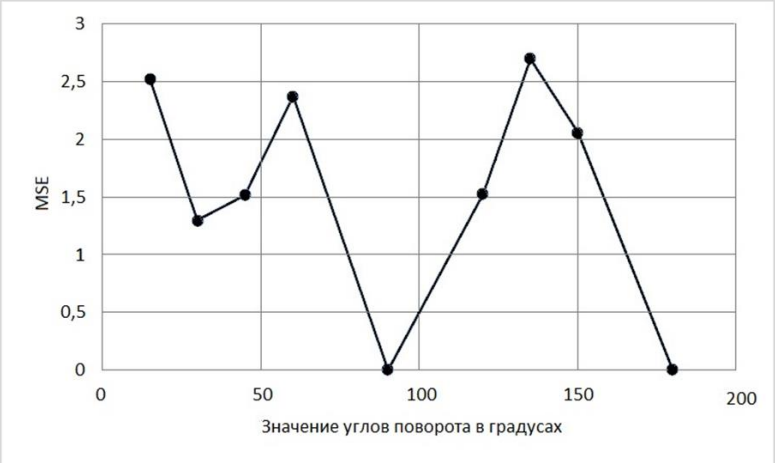


Рис. 2. Значение MSE по длине отрезков кривой 1

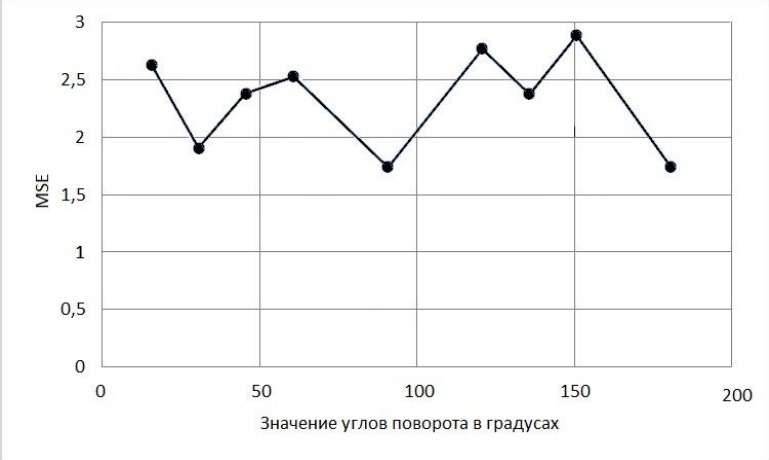


Рис. 3. Значение MSE по длине отрезков кривой 2

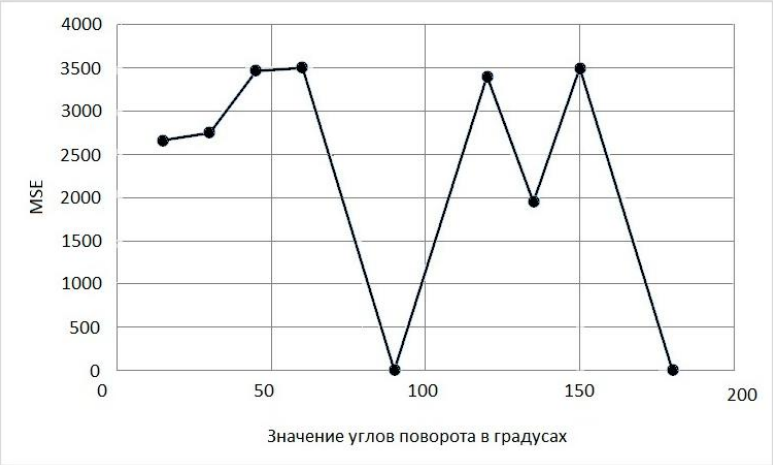


Рис. 4. Значение MSE по величине углов кривой 1

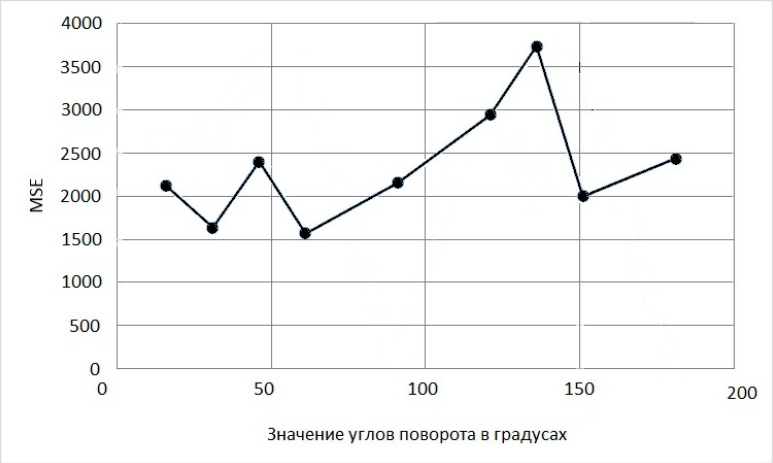


Рис. 5. Значение MSE по величине углов кривой 2

Заключение

На основании полученных данных можно сделать вывод, что параметризация кривой по значению длин отрезков более устойчива к повороту изображения, чем параметризация по значению углов.

PARAMETRIZATION OF CURVES ON THE BASIS OF APPROXIMATION BY CIRCLES

D.I. KIRYLUK, YU.I. KULAZHENKO

Abstract

Examples of various approaches to parametrization of curves are given. The assessment of stability of parameters of curves at rotary transformation of the image is received.

Список литературы

1. *Канатников А.Н., Крищенко А.П.* Аналитическая геометрия. М., 2002.
2. *Роджерс Д., Адамс Дж.* Математические основы машинной графики. М., 2001.
3. *Жук В.В., Натансон Г.И.* Тригонометрические ряды Фурье и элементы теории аппроксимации. Л., 1983.
4. *Guil N.* IEEE Transactions on Image Processing. 1995. Vol. 4. №11. P. 1541-1548.

УДК 621.396

ОСНОВНЫЕ ПОДХОДЫ ПРИ ОРГАНИЗАЦИИ SMS-СПАМА И СПОСОБЫ ПРОТИВОДЕЙСТВИЯ

А.С. ШЕЛКОВ, Н.В. НАСОНОВА

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

Поступила в редакцию 25 октября 2015

Рассмотрены вопросы, связанные с актуальностью борьбы с SMS-спамом, основные способы организации SMS-спама и основные методы для его блокировки в сетях сотовой связи.

Ключевые слова: SMS-спам, сотовая связь, блокировка спама.

Введение

Статистика последних лет показывает рост SMS-трафика в сетях операторов сотовой связи. В основном данная тенденция наблюдается для трафика, генерируемого различными приложениями для спам-рассылок пользователям сетей сотовой связи. Для SMS-сообщений, которые рассылаются вредоносными программами, действующими на телефонных аппаратах абонентов, напротив, характерен спад из-за широкого распространения программ-мессенджеров: Skype, Viber и т. д.

Актуальность

Из-за особенностей организации сетей сотовой связи, поддерживающих технологию SMS, абоненты практически не защищены от спам-рассылок через SMS-сообщения. Динамика роста SMS-трафика оператора сотовой связи Республики Беларусь (ИП «Велком») изображена на рис. 1-3.

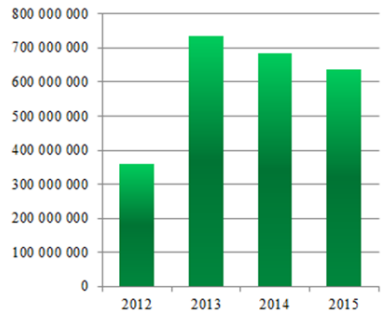


Рис. 1. Динамика роста абонентского SMS-трафика внутри сети

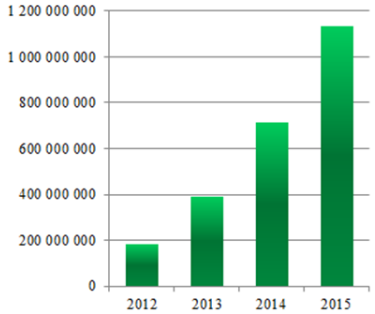


Рис. 2. Динамика роста SMS-трафика от приложений внутри сети

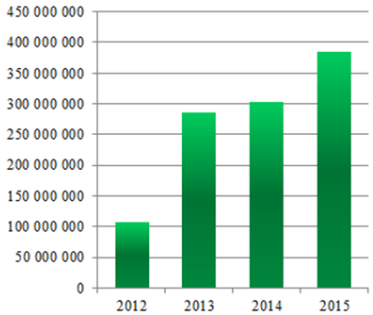


Рис. 3. Динамика роста SMS-трафика из других сетей

Анализ жалоб от абонентов РБ на различных информационных сайтах и порталах подтверждает рост доли SMS-спама среди общего объема трафика.

Угрозы SMS-спама для конечных пользователей:

1) расход памяти на телефонном аппарате (ТА) абонента;
2) заражение ТА вредоносным программным обеспечением (ПО);
3) мошенничество (SMS-викторины, SMS-казино);
4) отслеживание местоположения абонента;

5) причинение неудобств пользователям сетей сотовой связи.

Анализ способов организации спама

SMS-спам – массовая смс-рассылка коммерческого либо другого характера, организованная с использованием технологии SMS. По назначению он подразделяется на спам:

1) коммерческого содержания – используются рекламными организациями (рассылка рекламных сообщений, информаций о ссылках), банками (рассылка SMS онлайн-банкинга), торговыми предприятиями, а также любыми другими организациями, поскольку целевой диапазон использования таких рассылок достаточно широк;

2) вредоносного содержания – используются злоумышленниками для распространения вирусов, передачи бинарного кода на телефонные аппараты (ТА) абонентов, добывание персональных данных абонентов, причинение неудобств.

SMS-спам по способам организации подразделяется на:

1) MO-MT (Mobile Originated – Mobile Terminated) – SMS-трафик, формируемый вредоносными программами на ТА абонентов;

2) AO-MT (Application Originated – Mobile Terminated) – SMS-трафик, формируемый приложениями, которые имеют подключение к SMS-центру (SMSC) оператора сотовой связи.

На рис. 4, 5 изображены схемы организации MO-MT и AO-MT SMS-спама соответственно.

Рис. 4. Схема организации MO-MT SMS-спама

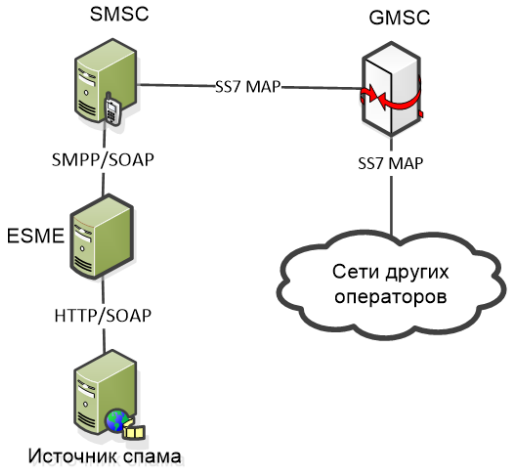


Рис. 5. Схема организации AO-MT SMS-спама

Обозначения на рисунках:

GSM (Global System for Mobile Communications) – технология сотовой связи второго поколения (2G);

UMTS (Universal Mobile Telecommunications System) – технология сотовой связи третьего поколения (3G);

LTE (Long-Term Evolution) – технология сотовой связи четвертого поколения (5G);

BSC (Base Station Controller) – контроллер базовых станций в 2G-сетях;

RNC (Radio Network Controller) – контроллер радиосети в 3G-сетях;

MSC (Mobile Switching Center) – коммутатор в сетях сотовой связи;

VLR (Visitors Location Register) – временная база данных абонентов;

SMSC (SMS Center) – сервер обработки и отправки SMS-сообщений;

GMSC (Gateway MSC) – коммутатор, который обрабатывает вызовы из внешних сетей и наоборот [1, 2];

SS7 MAP (Signalling System №7 Mobile Appication Part) – протокол системы общеканальной сигнализации №7 для обеспечения работы мобильных приложений в сетях сотовой связи, в частности SMS [3];

SMPP (Short Message Peer-to-peer Protocol) – протокол передачи коротких сообщений между приложением (ESME) и SMSC;

ESME (External Short Messaging Entities) – приложения рассылок коротких сообщений в спецификации протокола SMPP [4];

SOAP (Simple Object Access Protocol) – протокол обмена структурированными сообщениями в распределенной вычислительной среде;

HTTP (HyperText Transfer Protocol) – протокол прикладного уровня передачи данных.

Рассылка первым способом возможна при определенном количестве ТА с вредоносным ПО, иначе рассылка не будет массовой.

Наиболее используемый способ рассылки – второй. Источник рассылки (коммерческая организация либо злоумышленники) организовывает подключение с контент-агрегатором (ESME). Контент-агрегатор имеет постоянное подключение к SMSC оператора(ов) сотовой связи по протоколу SMPP (реже SOAP). Контент-агрегаторы – организации, предоставляющие свои услуги различным организациям в организации рассылок. Операторы сотовой связи, как правило, не взаимодействуют напрямую с другими организациями, так как последние не могут обеспечить необходимую интенсивность SMS-рассылок. Далее от SMSC рассылки идут через сети сигнализации SS7 и попадают в сети других операторов сотовой связи. Для транспортировки используется протокол MAP. Уже через сети других операторов SMS-рассылки достигают конечных абонентов [5].

Анализ возможных способов защиты

Перед анализом возможных методов защиты на уровне операторов сотовой связи необходимо отметить следующее: SMS-рассылки незаконного содержания, организованные через SMSC, легко могут быть приостановлены силами владельца этого же SMSC. Именно поэтому в дальнейшем будут рассматриваться SMS-спам из одной сети абонентам другой сети по SS7 MAP.

Путь попадания SMS в сеть сотовой связи извне по SS7 MAP в соответствии со спецификациями ETSI показан на рис. 6.

Рис. 6. Путь попадания SMS из других сетей

В соответствии со спецификациями ETSI для SMS в сети оператора 2 блокировать SMS-спам можно на GMSC и MSC. Реализация блокировки на уровне радиосети более затратная, так как компонентов радиосети во много раз больше и дальнейшая настройка механизмов блокировки будет занимать значительное время. Использование блокировки на уровне GMSC также не является предпочтительным, так как в современных коммутационных сетях каждый MSC является GMSC. Для стабильной и бесперебойной работы коммутационной сети в РБ операторам мобильной связи необходимо иметь от 7 до 10 MSC, поэтому настройка правил блокировки на таком количестве узлов также связана с большими затратами.

На данный момент для решения данной задачи применяется технология «Home Routing». Суть данной технологии для SMS заключается в том, что все SMS маршрутизируются на один MSC, который может одновременно включать в себя функционал SMSC. Такое решение значительно упрощает процесс настройки правил блокировки SMS-спама, поскольку все настройки осуществляются на одном узле сети и этот узел имеет функционал для обработки SMS [6].

SMS в SS7 сетях передаются с использованием протокола MAP. На входе в сеть поступающее SMS сообщение содержит следующую служебную информацию:

1) Originating Address/Destination Address (OA/DA) – номер отправителя/номер получателя;

2) OA/DA Type of Number (TON) – тип номера отправителя/получателя;

3) OA/DA Numbering Plan Indicator (NPI) – индикатор номерного плана отправителя и получателя;

4) Длина SMS в байтах;

5) DCS (Data Coding Scheme) – тип кодировки SMS [7];

6) Global Tittle (GT) SMSC оператора отправителя.

Это основные параметры, которые используются для блокировки SMS-спама. Просмотр текста SMS запрещен в рамках закона «Об информации, информатизации и защите информации» [8].

Методы блокировки SMS-спама.

1. По номеру отправителя.

Достоинства: простая реализация.

Недостатки: организатор спам-рассылки может сменить номер.

2. По TON и NPI номера отправителя. SMS, как правило, имеют следующие значения TON и NPI:

а) TON = 1, NPI = 1 – для абонентских номеров, например «375296452789»;

б) TON = 2, NPI = 1 – для коротких номеров, например «611»;

в) TON = 0, NPI = 5 – для альфанумерических номеров, например «Izbirkom».

SMS-спам обычно рассылается с коротких и альфанумерических номеров.

Достоинства: а) все SMS-рассылки с коротких и альфанумерических номеров из других или определенных сетей можно легко заблокировать; б) нагрузка на MSC/SMSC ниже, чем при блокировке отдельных номеров; в) смена номера при организации рассылок не поможет преодолеть блокировку.

Недостатки: а) SMS-спам можно рассылать с абонентских номеров; б) некоторые интернет-сервисы могут проводить рассылку через внешние SMSC и рассмотренный метод блокировки может привести к отказам для легальных пользователей сети.

3. По GT SMSC отправителя.

Достоинства: а) все подозрительные (серые) маршруты доставки SMS можно блокировать, оставив только разрешенные SMSC (яркий пример – австрийский оператор «A1»); б) нагрузка на MSC/SMSC становится ниже, чем при двух предыдущих методах; в) смена номера при организации рассылок не поможет преодолеть блокировку.

Недостатки: блокируется весь трафик, включая абонентский, а это может привести к большому числу жалоб от абонентов.

Указанный метод максимально эффективен, когда договоры о взаимодействии между операторами сотовой связи включают в себя пункты об ответственности за SMS-спам через свои SMSC.

На сегодняшний день серьезный вклад в борьбу с SMS-спамом вносят SMS-хабы – провайдеры, имеющие договорные отношения с различными операторами мобильной связи на предмет SMS-рассылок и ответственности за рассылку спама [9].

Заключение

Выбор метода блокировки SMS-спама должен основываться на используемых организаторами спама параметрах SMS-сообщений (адрес отправителя, GT SMSC отправителя) и на существующих договорных взаимоотношениях, регулирующих межсетевой обмен SMS-трафиком между различными операторами сотовой связи, SMS-хабами и другими сервис-провайдерами, которые осуществляют передачу SMS-трафика.

BASIC APPROACHES OF SMS-SPAM ORGANISATION AND PROTECTION

A.S. SHELKOV, N.V. NASONOVA

Abstract

SMS-spam protection issues, main approaches of SMS-spamming and basic methods of its blocking in cellular networks are discussed.

Список литературы

1. Specification GSM 03.02 – Network architecture. [Электронный ресурс]. Режим доступа: http://www.etsi.org/deliver/etsi_gts/03/0302/ 05.01.00_60/gsmts_0302v050100p.pdf.
2. Specification 3GPP TS 23.01U – General UMTS architecture. [Электронный ресурс]. Режим доступа: http://www.3gpp.org/DynaReport/2301U.htm.
3. Specification ETS 300 599 – GSM 09.02 version 4.17.1 – Mobile Application Part (MAP) specification. [Электронный ресурс]. Режим доступа: http://www.etsi.org/deliver/etsi_i_ets/300500_300599/300599/07_60/ets_300599e07p.pdf.
4. Short Message Peer to Peer. Protocol Specification v3.4. [Электронный ресурс]. Режим доступа: http://opensmpp.org/specs/smppv34_gsmumts_ig_v10.pdf.
5. Specification ETSI TS 100 901 – Technical realization of the Short Message Service (SMS) – Point-to-Point (PP). [Электронный ресурс]. Режим доступа: http://www.etsi.org/deliver/etsi_ts/100900_100999/100901/07.05.00_60/ts_100901v070500p.pdf.
6. Specification 3GPP TR 23.840 – Study into routeing of MT-SMs via the HPLMN. [Электронный ресурс]. Режим доступа: http://www.3gpp.org/DynaReport/23840.htm.
7. Specification GSM 03.38 – Alphabets and language-specific information. [Электронный ресурс]. Режим доступа: http://www.etsi.org/deliver/etsi_ts/100900_100999/100900/07.02.00_60/ts_100900v070200p.pdf.
8. Закон Республики Беларусь от 10 ноября 2008 г. №455-З «Об информации, информатизации и защите информации»
9. Официальный сайт компании «GMS». [Электронный ресурс]. Режим доступа: http://www.gms-worldwide.com/.

УДК 004.716

ОБРАБОТКА ТРАФИКА КОЛЛ-ЦЕНТРА НА ОСНОВЕ ОФИСНОЙ IP ТЕЛЕФОННОЙ СТАНЦИИ

А.А. АНТОННИКОВ, С.Н. ПЕТРОВ, С.В. ВЛАСЮК*, Т.А. ПУЛКО

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

**Институт информационных технологий*
Козлова, 28, Минск, 220037, Беларусь

Поступила в редакцию 29 октября 2015

Рассмотрены вопросы анализа и обработки сигнального трафика колл-центра на основе офисной IP телефонной станции (IP-PBX). Описана локальная вычислительная сеть с интеграцией системы телефонии 3CX PHONE SYSTEM. Проведен анализ обработки трафика колл-центра. Рассмотрены процедуры обработки и анализа трафика колл-центра на основе IP-PBX с помощью программного анализатора протоколов Wireshark.

Ключевые слова: интернет-телефония, офисные телефонные станции, анализ трафика, джиттер.

Введение

Офисные (учрежденческие) телефонные станции PBX (от англ. Private Branch Exhange) обеспечивают весь процесс коммутации между аппаратами: установление, поддержание и разрыв соединений. Использование PBX позволяет разделять ограниченные ресурсы (городские линии и номера) между неограниченным числом внутренних пользователей, при помощи таких телефонных функций, как внутренний номерной план, перевод звонков, постановка на удержание, и других. По этой причине PBX-система необходима любой организации, она позволяет эффективно организовать телефонную связь на предприятии. На сегодняшний день IP-телефония все больше вытесняет традиционные телефонные сети за счет легкости развертывания, низкой стоимости звонка, простоты конфигурирования, высокого качества связи и сравнительной безопасности соединения. Офисная IP-телефонная станция, это станция на основе межсетевого протокола IP.

Из всего множества протоколов интернет-телефонии в настоящее время стандартом дефакто считается протокол SIP (от англ. Session Initiation Protocol) – стандарт на способ установления и завершения пользовательского интернет-сеанса, включающего обмен мультимедийным содержимым (видео- и аудиоконференции, мгновенные сообщения, онлайн-игры, и пр.) [1]. Протокол SIP больше приспособлен к работе в сетях TCP/IP и более универсален (предназначен для передачи медиа контента разного типа). Также он поддерживает услуги интеллектуальной сети, такие как преобразование имен, переадресация и маршрутизация, что существенно для использования SIP в качестве протокола сигнализации в сети общего пользования, где приоритетной задачей оператора является предоставление широкого спектра телефонных услуг. Другой важной особенностью протокола SIP является поддержка мобильности пользователя, т.е. его способности получать доступ к заказанным услугам в любом месте и с любого терминала, а также способности сети идентифицировать и аутентифицировать пользователя при его перемещении из одного места в другое. Данный режим работы требует дистанционной регистрации пользователей на сервере идентификации и аутентификации [2].

На сегодняшний день можно встретить IP-PBX двух видов – аппаратные в виде специализированного оборудования с предустановленным ПО и программные в виде дистрибутивов или исполняемых файлов. Примером программной IP АТС (VoIP АТС) является 3CX Phone System, которая осуществляет вызовы через сеть передачи данных вместо традиционной телефонных линий. Используя VoIP-шлюзы, осуществляется подключение существующих традиционных линий к IP АТС и пользование телефонной связью так же, как это происходит при использовании обычной АТС. 3CX Phone System дает возможность использовать и программные телефоны (софтфоны), объединять удаленные офисы и отдельных абонентов в общую сеть через интернет. Программная АТС 3CX использует в качестве протокола сигнализации открытый стандарт SIP и, базируясь на обычном Windows-сервере, дает возможность интеграции с различными бизнес-приложениями (например CRM-системами) [3].

Целью работы является исследование сети связи предприятия с использованием протокола прикладного уровня SIP для повышения качества телефонной связи и снижения затрат на телефонные разговоры. Мониторинг параметров сети проводится с помощью программного анализатора протоколов Wireshark.

Схема исследуемой сети и оцениваемые параметры

Система связи предприятия должна обеспечить реализацию следующих функций:

- передачу голоса по существующей сети передачи данных;
- коммутацию абонентов внутри сети;
- возможность абонентов принимать звонки из телефонной сети общего пользования;
- возможность абонентов совершать звонки в телефонную сеть общего пользования;
- маршрутизацию входящих звонков.

Для выполнения вышеуказанных функций в сеть будет подключен сервер, с установленным на него программным обеспечением IP-АТС, поддерживающим протокол SIP. В качестве конечных аппаратов будут использоваться IP-телефоны и программные IP-телефоны, также поддерживающие протокол SIP. Необходимо настроить программную IP-АТС, настроить активное и пассивное сетевое оборудование, организовать выделенный центр обслуживания вызовов. Также выявить и устранить неисправностей в VoIP-системе.

Передача речевой информации в IP-сетях или IP-телефония предъявляет к сетевой инфраструктуре жесткие требования. Степень доступности сети, потеря пакетов и временные задержки оказывают существенное влияние на качество передачи речи в IP-сетях. Такие явления, как потеря пакетов, джиттер (разброс времени доставки пакетов) или ошибки последовательности пакетов являются неотъемлемыми для IP-сетей и успешно обнаруживаются и корректируются протоколом передачи данных. Потеря пакетов, джиттер и беспорядочно следующие пакеты тесно связаны между собой, поэтому исправление какой-либо одной ошибки зачастую снижает действие всех трех и ведет к существенному улучшению качества голосовых звонков.

Структурная схема исследуемой локальной вычислительно сети (ЛВС) приведена на рис 1.

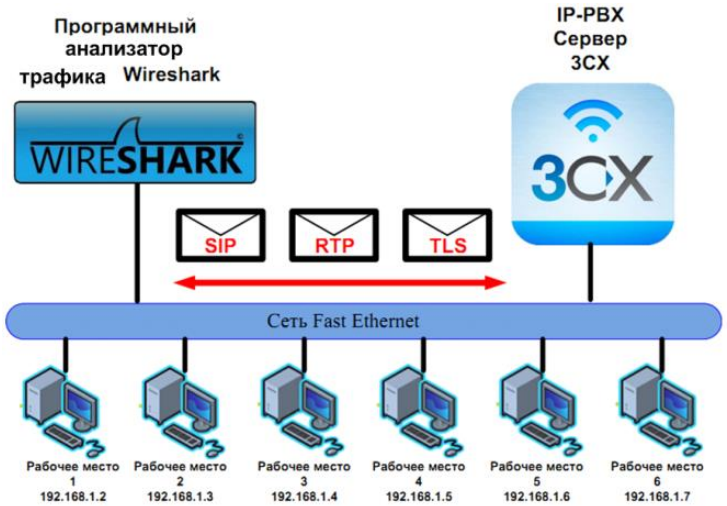


Рис. 1. Схема ЛВС с интеграцией 3CX и программным анализатором Wireshark

В качестве средства выявления и устранения неисправностей предлагается использовать анализатор протоколов Wireshark, позволяющий установить некоторые логические условия для захвата отдельных пакетов и выполнить полное декодирование захваченных пакетов, показать в удобной форме вложенность пакетов протоколов разных уровней друг в друга с расшифровкой содержания отдельных полей каждого пакета. Wireshark переводит сетевой адаптер в режим «беспорядочного» приема кадров, записывает в свой буфер отфильтрованные кадры сетевого трафика, по запросам пользователя выводит на экран те или иные кадры из буфера и посредством декодера протоколов предоставляет пользователю информацию о значениях полей заголовка протокола и содержимое его блока данных.

Wireshark содержит основные компоненты, такие как фильтр захвата, буфер кадров, декодер протоколов, фильтр отображения захваченных кадров и модуль статистики с элементами экспертной системы, присущие всем программным анализаторам этого класса. Основными исследуемыми параметрами являются джиттер, MOS и R-Factor.

Результаты и их обсуждение

Wireshark позволяет анализировать протокол SIP и его RTP-трафик. Каждому совершенному звонку соответствует SIP-сессия (рис. 2). Диаграмма наглядно показывает временные метки отправки SIP-сообщений, графическое представление и комментарии к каждой посылки. RTP-сессия устанавливается для каждого звонка либо мультимедийного потока. Сессия состоит из IP-адреса и пары портов для RTP и RTCP.

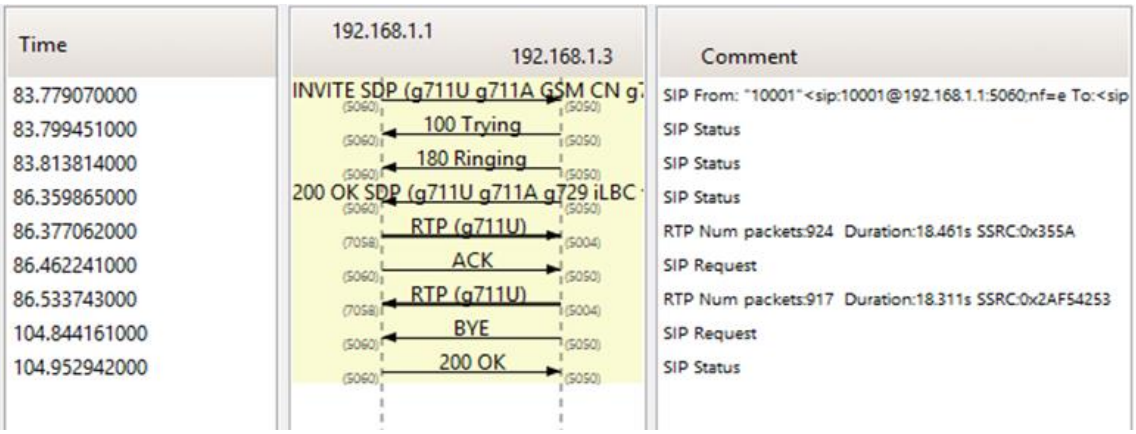


Рис. 2. SIP-сессия

Для оценки потерянных пакетов нужно перейти по вкладкам Telephony → Rtp → Show all streams. Далее в открывшемся окне в столбце Lost будут определены потери RTP-пакетов. Потери до 5 % считаются допустимыми.

Для количественной оценки качества VoIP-переговоров используется шкала MOS – усредненная оценка разборчивости речи. MOS включает в себя показатель воспринимаемого качества звука по балльной шкале от 1 до 5. Изначально MOS представлял собой среднее арифметическое всех оценок качества, данных людьми, которые прослушивали тестовый звонок и давали ему свою оценку. Показатель MOS крайне субъективен.

Альтернативным способом оценки качества звука является R-Factor. Балльная шкала от 0 до 120 в отличие от сокращенной шкалы MOS позволяет делать более точную оценку показателя качества. R-Factor рассчитывается с учетом ощущений пользователя и объективных факторов, которые влияют на общее качество VoIP системы; соответственно, отдельно рассчитываются сетевой R-Factor и пользовательский R-Factor. MOS и R-Factor не дают полной картины качества VoIP-системы. Эти показатели являются лишь оценками ожидаемого качества звука при внедрении VoIP-системы. Для получения точной взаимосвязи между численными измерениями и реальным качеством звука следует прослушать несколько разговоров и вынести свое решение по поводу приемлемого качества. Возможность воспроизведения звонка очень важна для успешного внедрения VoIP-решений. Для оценки качества звонка путем прослушивания в сетевом анализаторе имеются средства записи и воспроизведения звонков. Для этого необхо-

димо выбрать в меню Statistics (или Telephony, в зависимости от версии программы) → VoIP Calls. Откроется окно VoIP Calls со списком VoIP-звонков. Затем выбрать VoIP-вызов и нажать кнопку Player. Появится окно VoIP-Player с графической диаграммой речи. Необходимо выбрать поток и нажать Decode. Результат этих действий показан на рис. 3.

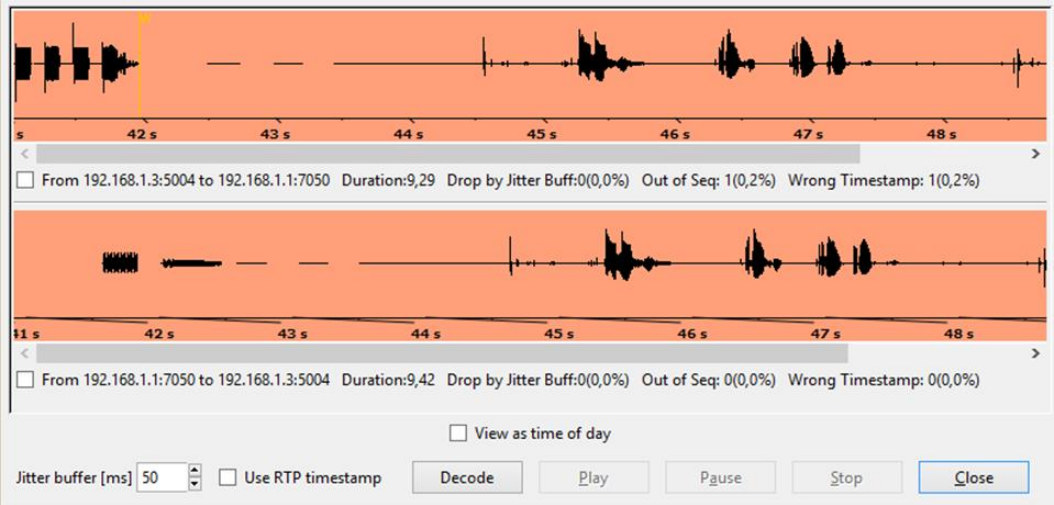


Рис. 3. Окно прослушивания голосового трафика RTP-потоков

Заключение

Проведен анализ разработанной системы связи предприятия, работающей по протоколу SIP, с использованием анализатора протоколов Wireshark. Поскольку размер буфера существенно влияет на качество воспринимаемой речи, оптимизация джиттер-буфера в VoIP устройстве также существенно влияет на результат. Размер буфера, превышающий 150 мс, существенно влияет на воспринимаемое качество разговора. Подбирая размер джиттер-буфера в зависимости от используемого кодека и среды передачи, можно добиться оптимального качества принимаемой речи.

PROCESSING CALL CENTER TRAFFIC BASED ON IP PRIVATE BRANCH EXHANGE

A.A. ANTONNIKOV, S.N. PETROV, S.V. VLASYUK, T.A. PULKO

Abstract

The problems of the analysis and processing of IP-PBX call center signaling traffic are discussed. Call center traffic processing is analyzed. A local area network with the integration of IP-PBX phone system 3CX PHONE SYSTEM is described. The procedures of processing and analyzing of IP-PBX call center traffic using a software analyzer Wireshark are performed.

Список литературы

1. СТБ 2156-2014. Средства электросвязи мультсервисных сетей.
2. *Гольдштейн Б.С., Пинчук А.В., Суховицкий А.Л.* IP-телефония. М., 2001.
3. *Дэвидсон Д.* Основы передачи голосовых данных по сетям IP (IP Voice over IP Fundamentals). М., 2007.

УДК 539.24

ЦИФРОВАЯ ОБРАБОТКА И АНАЛИЗ 2D ИЗОБРАЖЕНИЙ ПЛЕНОК НАНОТРУБЧАТОГО ОКСИДА ТИТАНА С ИСПОЛЬЗОВАНИЕМ IMAGEJ

К.В. ЧЕРНЯКОВА, И.А. ВРУБЛЕВСКИЙ, М.Ф.С.Х. АЛЬ-КАМАЛИ

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

Поступила в редакцию 30 октября 2015

Представлен цифровой способ обработки изображения для анализа микроструктуры пленок нанотрубчатого оксида титана по данным 2D изображений сканирующей электронной микроскопии с использованием программы ImageJ. Полученные результаты показывают, что разработанный метод может быть использован в качестве инструмента для оценки микроструктуры нанопористых пленок оксида титана.

Ключевые слова: 2D СЭМ изображения, ImageJ, цифровая обработка изображения, нанопористые пленки, оксид титана.

Введение

Изучение морфологических и структурных характеристик нанопористых материалов является одной из трудных и важных задач современного материаловедения. Это связано с тем, что создание и разработка новых материалов с улучшенными характеристиками, имеющих массивы высокоупорядоченных наноразмерных пор, невозможны без определения размеров пор и плотности их распределения. В данной работе для анализа морфологии поверхности выбраны пленки нанотрубчатого оксида титана. Такие пленки вызывает большой интерес благодаря своей уникальной самоорганизованной нанопористой структуре, а также потенциальной возможности управления ее размерными параметрами, что обеспечивает возможность широкого применения в конструкциях газовых сенсоров, солнечных элементов, фотокаталитических и биосовместимых покрытиях [1, 2]. При анализе изображений фотографии объекта исследования с нанопористой микроструктурой должны решаться следующие задачи: сегментация, фильтрование недостатков и выделение объектов из фона, определение пределов объектов и распознавание образов [3]. Для успешного проведения анализа основным является вопрос надежности сегментации изображения. На сегодняшний день существует много различных прикладных программ для анализа изображений. С учетом функциональных возможностей среди многообразия программного обеспечения для анализа изображений наиболее успешными являются следующие прикладные программы – «Photom», «Optimas», «Видиотест», «Image Expert Pro», «ImageJ», «Avizo», «Smart-eye». В своих исследованиях авторы использовали программу «ImageJ». В арсенале данной программы есть все необходимые алгоритмы для обработки технических изображений: высокочастотного и низкочастотного фильтрования, выделение пределов изображений, арифметических и логических операций, коррекции яркость/контраст [4-6].

Целью работы было использование алгоритмов анализа 2D изображений для определения геометрических параметров пленок нанотрубчатого TiO_2 с наноразмерными порами. Решение таких задач, включающих анализ и обработку изображений, проводилось при помощи программного продукта ImageJ, алгоритмы работы которого позволяли анализировать характеристики сотовой структуры пористых пленок по данным сканирующей электронной микроскопии (СЭМ).

Экспериментальная часть

Для экспериментов использовали титановую фольгу 99,7 % чистоты и толщиной 0,127 мкм (Aldrich). Из фольги вырезали образцы размерами 12×12 мм и обезжиривали поверхность титана в ультразвуковой ванне в растворах ацетона, этанола и воды в течение 6 мин в каждом растворе. Затем подготовленные образцы погружали в раствор и анодировали в потенциостатическом режиме при 50 В в течение 0,5-2 ч. В качестве электролита для получения пленок нанотрубчатого оксида титана использовался следующий состав: 93 масс.% этиленгликоля, 0,3 масс.% фторида-аммония, 2 масс.% воды.

С помощью процесса электрохимического окисления металлического титана формировались высокоупорядоченные пористые анодные пленки TiO_2. Структура пористых пленок представляла собой плотно упакованные массивы нанотрубок оксида титана, ориентированных перпендикулярно металлической подложке. Сформированные пленки характеризовались узким распределением пор по размерам. Морфологию полученных образцов изучали с помощью СЭМ на микроскопе Zeiss DSM 982. Напряжение на ускоряющем электроде варьировали от 10 до 20 кВ. На рис. 1 представлена микрофотография пленок нанотрубчатого TiO_2, полученных анодированием в электролите на основе этиленгликоля с добавками фторид-ионов.

Для обработки и получения характеристик нанопористой структуры пленок использовалась следующая последовательность алгоритмов в программе «ImageJ»:

1) фильтрование изображения с целью исключения случайного шума;

2) предыдущая сегментация, которая направлена на выделение однородных областей;

3) коррекция объекта с целью определения порога яркостей;

4) окончательная сегментация с использованием определенного фонового значения, что позволяет полностью определить объекты;

5) анализ выделенных объектов с целью определения их параметров.

На рис. 1 показана блок-схема процесса цифровой обработки изображения в программе ImageJ.

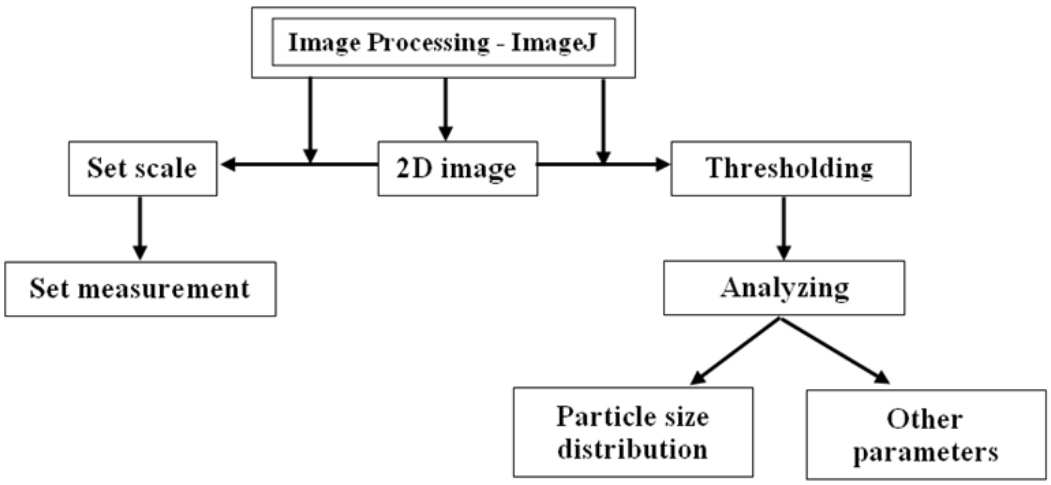


Рис. 1. Схема процесса цифровой обработки 2D изображения в программе ImageJ

В ImageJ можно вычислять площади, статистические показатели пиксельных значений различных выделенных областей интереса на изображениях, которые выделены вручную или при помощи пороговых функций. Программа ImageJ поддерживает стандартные функции обработки изображений, такие как логические и арифметические операции между изображениями, манипуляции с контрастностью, свертки, Фурье-анализ, повышение резкости, сглаживание, обнаружение границ и медианный фильтр. Программа позволяет производить различные геометрические преобразования, масштабирование, поворот или отражение. Конечными задачами анализа изображений являлись статистическая обработка результатов, полученных при измерении характеристик объекта с нанопористой структурой, определение средних значений диаметров пор, а также построение графических зависимостей для визуализации процесса анализа.

Результаты и их обсуждение

Полученные образцы с пленками нанотрубчатого TiO_2 характеризовались самоорганизованной пористой структурой (рис. 2). Для оценки геометрических параметров пленок нанотрубчатого TiO_2 до и после обработки для осаждения наночастиц оксидов металлов из раствора была выбрана методика, основанная на анализе морфологии поверхности по данным сканирующей электронной микроскопии с помощью цифровых методов обработки изображений (программа ImageJ).

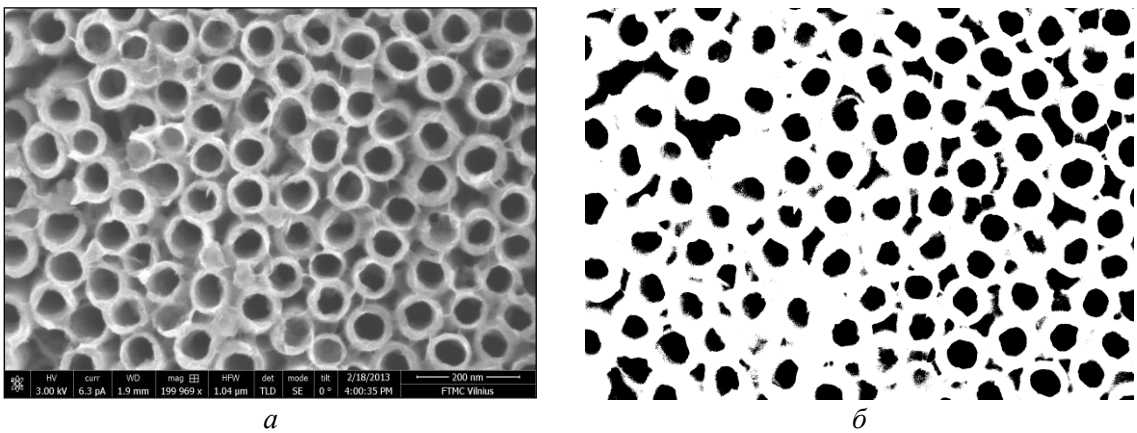
На рис. 2, 3 представлены примеры начальных (рис. 2,*а*,*б*) и конечных (рис. 3) результатов работы программы по обработке СЭМ изображений для пленок нанотрубчатого TiO_2, отожженных на воздухе при 500 ºС в течение 2 ч.

Работа с использованием программы ImageJ включала в себя несколько этапов. В начале для получения информации о размере пор необходимо провести калибровку области изображения. В меню настройки (Settings) выбирается функция Calibrate Spatial Measurements и проводится линия заданной длины. В качестве такой линии выбирается маркер, обычно находящийся в правом нижнем углу микрофотографии. В появляющемся диалоговом окне выбирается нужная размерность и задается длина линии. После того как проведена калибровка изображения, переходят непосредственно к операциям для измерения размеров пор.

На первом этапе оттенки серого цвета, присутствующие в СЭМ изображении, импортировались в ImageJ (рис. 2,*а* и 4,*а*). Затем задавались размеры анализируемой области, проводили обрезание до выбранного размера и остальное изображение преобразовывали в настоящее черно-белое изображение (рис. 2,*б* и 4,*б*). На втором этапе, перед преобразованием, в программе ImageJ выбирали значение порогового серого, выше которого связанные пиксели преобразовывались в черный цвет, в свою очередь ниже которого пиксели преобразовывались в чисто белый цвет.

Для проведения анализа лучше всего подходило СЭМ изображение, где имелся значительный контраст в оттенках серого между круглыми порами. На третьем этапе, с помощью установок программы автоматически удалялись мелкие темные объекты и, таким образом, проводилась очистка изображение. В конце цикла программа выдавала данные для построения гистограммы распределения пор определенного диаметра на анализируемой поверхности (рис. 3 и 5).

Как видно из результатов, представленных на рис. 3, пленки нанотрубчатого TiO_2 имели размер пор 50,3 нм. Полученный результат хорошо согласуется с данными в литературе для размеров пор пленок нанотрубчатого TiO_2, формируемых в электролите на основе этиленгликоля с добавками фторид-ионов. Результаты обработки экспериментальных данных показали, что размер пор пленок нанотрубчатого TiO_2 увеличивался с увеличением приложенного напряжения анодирования.



а *б*

Рис. 2. СЭМ изображение поверхности пленок нанотрубчатого TiO_2 (*а*) и ее окончательный вид (*б*) для идентификации пор после преобразования в черно-белую графику с помощью компьютерной программы ImageJ. Режимы получения пленок: напряжение анодирования 50 В, продолжительность анодирования – 40 мин; отжиг на воздухе при 500 ºС в течение 2 ч

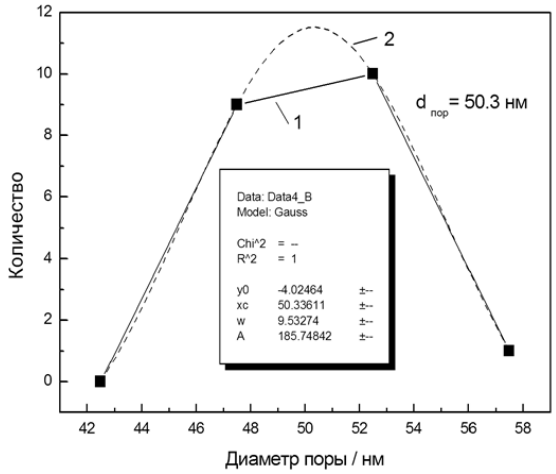
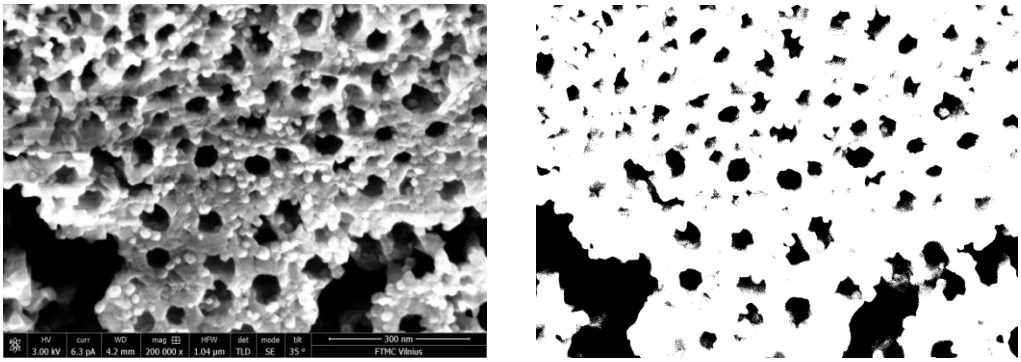


Рис. 3. Распределение пор по диаметру (1) и подгоночная кривая Гаусса (2), полученные по результатам обработки СЭМ изображения пленок нанотрубчатого TiO_2 (рис. 2,*а*) с помощью компьютерной программы ImageJ

На рис. 4-5 представлены примеры начальных (рис. 4,*а*,*б*) и конечных (рис. 5) результатов работы программы по обработке СЭМ изображений для образца с пленками нанотрубчатого TiO_2, обработанного переменным током для осаждения наночастиц оксидов металлов из раствора в поры.



а *б*

Рис. 4. СЭМ изображение поверхности пленок нанотрубчатого TiO_2 после обработки переменным током для осаждения наночастиц оксидов металла из раствора в поры (*а*) и ее окончательный вид (*б*) для идентификации пор после преобразования в черно-белую графику с помощью компьютерной программы ImageJ. Режимы обработки пленок: раствор, содержащий 0,1 $Cu(CH_3COO)_2$, 0,1 mol L^{-1} $Mg(CH_3COO)_2$ и CH_3COOH с pH 5,3; переменный ток плотностью 0,5 А $дм^{-2}$ в течение 10 мин

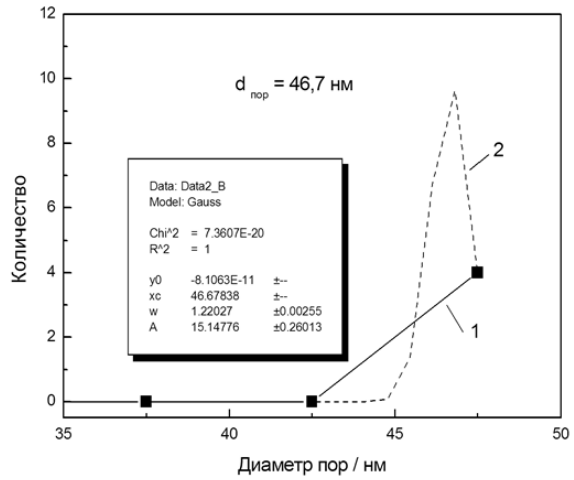


Рис. 5. Распределение пор по диаметру (1) и подгоночная кривая Гаусса (2), полученные по результатам обработки СЭМ изображения пленок нанотрубчатого TiO_2 (рис. 4,*а*) с помощью компьютерной программы ImageJ

Как видно из результатов, представленных на рис. 5, образцы с нанотрубчатыми пленками TiO_2 после электрохимической обработки на переменном токе имели размер пор 46,7 нм. Уменьшение размера пор на 3,6 нм (с 50,3 до 46,7 нм) по сравнению с исходными пленками связано с осаждением наночастиц оксидов металлов на стенки пор пленок TiO_2 в процессе электрохимической обработки. Таким образом, алгоритмы работы программы ImageJ по анализу изображений позволили зафиксировать столь незначительное уменьшение размеров диаметра пор (несколько нм) для нанотрубчатых пленок TiO_2, вызванное их электрохимической обработкой.

Заключение

Показано, что микроскопические методы анализа морфологии поверхности являются достаточно простыми и очень информативными методами изучения структуры нанотрубчатых пленок TiO_2. Проведенное изучение позволило сделать вывод, что программа ImageJ для анализа микроизображений является подходящим инструментом для количественного определения параметров микроструктуры пленок TiO_2 с наноразмерными порами. Она поддерживает стандартные функции обработки изображений, такие, как логические и арифметические операции между изображениями, манипуляции с контрастностью, свертки, Фурье-анализ, повышение резкости, сглаживание, обнаружение границ и медианный фильтр. Результаты измерений в программе ImageJ позволяют рассчитать средний диаметр пор нанотрубчатых пленок TiO_2, а также стандартное отклонение. Построение графических зависимостей для визуализации процесса анализа реализуется в виде гистограмм или кривых численного распределения диаметра пор по размерам с использованием математического пакета Origin компании OriginLab Corporation.

DIGITAL PROCESSING AND ANALYSIS OF 2D IMAGES OF NANOTUBE TITANIUM OXIDE BY IMAGEJ

K.V. CHERNYAKOVA, I.A. VRUBLEVSKY, M.F.S.H. AL-KAMALI

Abstract

A digital image processing method for the analysis of structure of nanotube titanium oxide films from 2D SEM images using ImageJ program was presented. The results obtained show that the developed method can be used as a practical tool to estimate the structure of nanoporous titanium oxide films.

Список литературы

1. *Морозов А.Н., Михайличенко А.И.* // Химическая промышленность сегодня. 2013. №5. С. 74-78.
2. *Lazarouk S., Xie Z., Chigrinov V. et al.* // Japanese Journal of Applied Physics. 2007. Vol. 46. №7A. P. 4390-4394.
3. *Serra J.* Image Analysis and Mathematical Morphology. London, 1992.
4. *Bodla K.K., Murthy J.Y., Garimella S.V.* // Numerical Heat Transfer. 2010. Part A. №58(7). P. 527.
5. *Feldkamp L.A., Davis L.C., Kress J.W.* // J. Microsc. 1997. Vol. 185. P. 67-75.
6. *Whitehouse D.* Handbook of Surface Metrology. Bristol and Philadelphia, 1994.

СВЕДЕНИЯ ОБ АВТОРАХ

1.	Алиханов Фархад	- магистрант кафедры сетей и устройств телекоммуникаций БГУИР
2.	Альзаки Хайдер Макки Хамид	- аспирант кафедры сетей и устройств телекоммуникаций БГУИР
3.	Аль-Камали Марван Фархан Саиф Хассан	- магистрант кафедры микро- и наноэлектроники БГУИР
4.	Альмияхи Осама Мажид Хилал	- аспирант кафедры сетей и устройств телекоммуникаций БГУИР
5.	Антонников Артур Алексеевич	- магистрант кафедры защиты информации БГУИР
6.	Астровский Иван Иванович	- к.т.н., доцент кафедры сетей и устройств телекоммуникаций БГУИР
7.	Бобов Михаил Никитич	- д.т.н., профессор кафедры сетей и устройств телекоммуникаций БГУИР
8.	Власюк Светлана Владимировна	- зам. декана факультета компьютерных технологий ИИТ БГУИР
9.	Волостных Григорий Александрович	- магистрант кафедры сетей и устройств телекоммуникаций БГУИР
10.	Врублевский Игорь Альфонсович	- к.т.н., доцент кафедры защиты информации БГУИР
11.	Горбуль Роберт Михайлович	- магистрант кафедры сетей и устройств телекоммуникаций БГУИР
12.	Драпеза Алексей Николаевич	- магистрант кафедры сетей и устройств телекоммуникаций БГУИР
13.	Ибрагим Жад	- магистрант кафедры сетей и устройств телекоммуникаций БГУИР
14.	Кирилюк Денис Игоревич	- аспирант кафедры сетей и устройств телекоммуникаций БГУИР
15.	Конопелько Валерий Константинович	- д.т.н., профессор, зав. кафедрой сетей и устройств телекоммуникаций БГУИР
16.	Королев Алексей Иванович	- к.т.н., доцент кафедры сетей и устройств телекоммуникаций БГУИР
17.	Костусев Алексей Владимирович	- магистрант кафедры сетей и устройств телекоммуникаций БГУИР

СВЕДЕНИЯ ОБ АВТОРАХ

18.	Крупенкова Татьяна Георгиевна	- ассистент кафедры инженерной математики БНТУ
19.	Кулаженко Юрий Иванович	- д.ф.-м.н., доцент, начальник управления подготовки научных кадров высшей квалификации БГУИР
20.	Липницкий Валерий Антонович	- д.т.н., профессор, зав. кафедрой высшей математики и физики ВА
21.	Макейчик Екатерина Геннадьевна	- старший преподаватель кафедры сетей и устройств телекоммуникаций БГУИР
22.	Насонова Наталья Викторовна	- к.т.н., доцент кафедры защиты информации БГУИР
23.	Новицкий Виталий Владимирович	- магистрант кафедры сетей и устройств телекоммуникаций БГУИР
24.	Парфенюк Александр Иванович	- магистрант кафедры сетей и устройств телекоммуникаций БГУИР
25.	Петров Сергей Николаевич	- к.т.н., доцент кафедры защиты информации БГУИР
26.	Пулко Татьяна Александровна	- к.т.н., доцент кафедры защиты информации БГУИР
27.	Спичекова Наталья Викторовна	- к.ф.-м.н., доцент кафедры высшей математики БГУИР
28.	Трофименко Дарья Дмитриевна	- магистрант кафедры сетей и устройств телекоммуникаций БГУИР
29.	Филончик Николай Михайлович	- магистрант кафедры сетей и устройств телекоммуникаций БГУИР
30.	Хоменок Михаил Юлианович	- к.т.н., доцент кафедры сетей и устройств телекоммуникаций БГУИР
31.	Цветков Виктор Юрьевич	- к.т.н., доцент кафедры сетей и устройств телекоммуникаций БГУИР
32.	Чернякова Екатерина Викторовна	- к.ф.-м.н., доцент кафедры защиты информации БГУИР
33.	Шевчук Оксана Геннадьевна	- аспирант кафедры сетей и устройств телекоммуникаций БГУИР
34.	Шелков Андрей Сергеевич	- магистрант кафедры защиты информации БГУИР

ISBN 978-985-488-834-7