

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. Бровка, 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 решающая 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 пикселей 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 МЕСТ [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 НВСТ (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 МЕСТ (Multiscale Embedded Cluster Tree). Поэтому новый алгоритм назван НВСТ. 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 МЕСТ.

Все алгоритмы вейвлет-сжатия основаны на вейвлет-преобразовании исходного изображения, которое перераспределяет энергию пикселей 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).

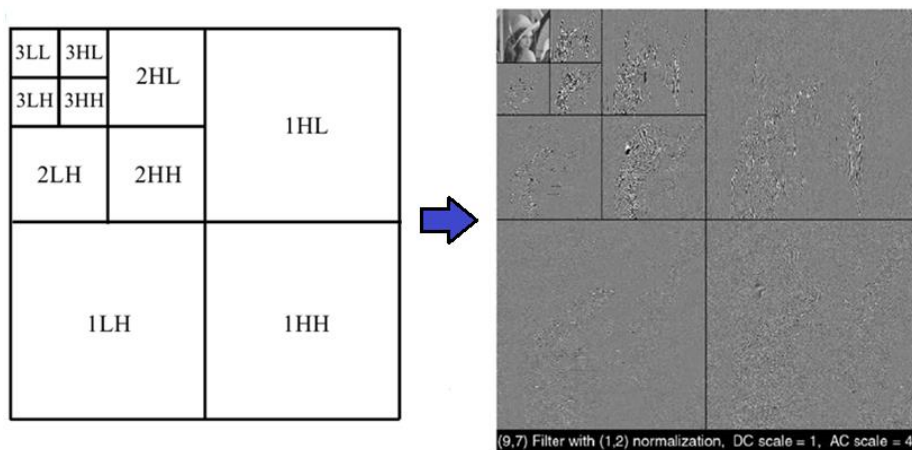


Рис. 1. Пример трехуровневой вейвлет-декомпозиции изображения «Lena»

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.

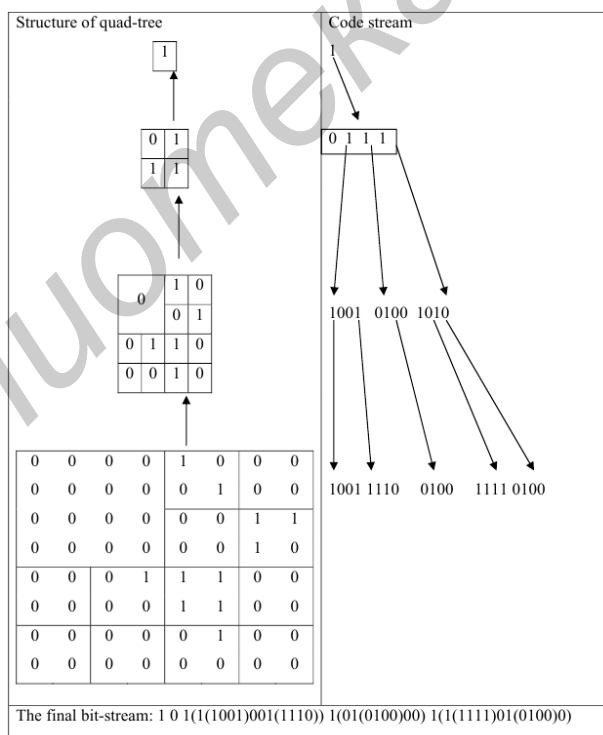


Рис. 2. Построение «нуль-дерева» с помощью кластеризации блока в методе SPECK

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

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

Алгоритм НВСТ 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 МЕСТ – M_{SP}) нулями.

Шаг 4. Определение номера старшей битовой плоскости n_{max} и запись его в начало выходного прогрессивного потока PS (в МЕСТ – 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 $L_l = 4$.

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 НВСТ. Освобождение памяти.

Выходной поток кодера НВСТ (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 НВСТ

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.

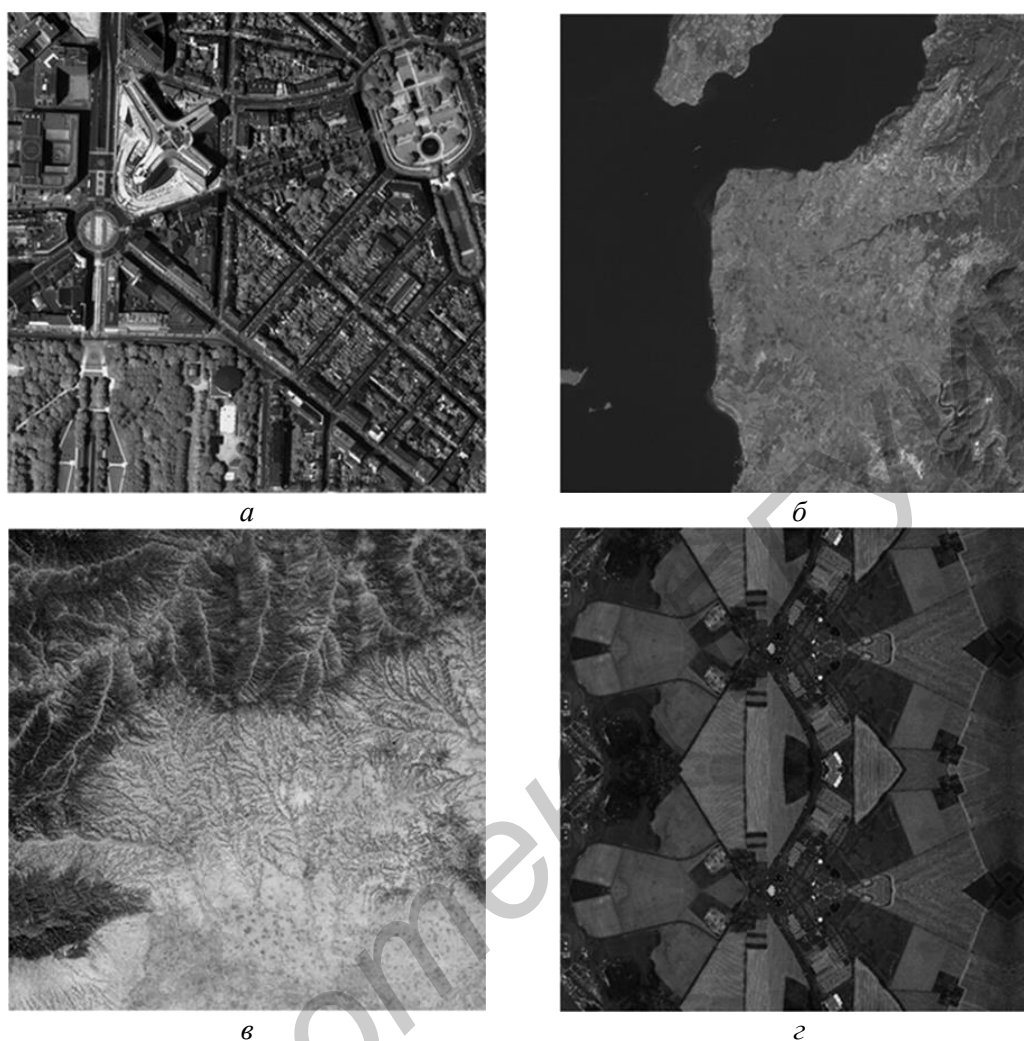


Рис. 4. Тестовые аэрокосмические изображения:
а – «City»; б – «Sealine»; в – «Mountains»; z – «Plains»

Таблица 1. Численные результаты алгоритмов для изображения «City»

Алгоритм	Режим сжатия	Без потерь		Потери, 2 бпр		Потери, 1 бпр		Потери, 0,5 бпр	
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 вейлетов. 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 НВСТ выигрывает 0,6 дБ у алгоритма МЕСТ 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	
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	1,636		3,998		7,995		15,982	
	PSNR, дБ	∞		39,437		33,620		30,339	
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	1,721	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	Наар	CDF 5/3	Наар	CDF 5/3	Наар
	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	
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	1,172		3,998		7,995		15,980	
	PSNR, дБ	∞		27,729		24,075		22,036	
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	1,182	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	Наар	CDF 5/3	Наар	CDF 5/3	Наар
	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	
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	1,450		3,999		7,996		15,982	
	PSNR, дБ	∞		36,454		32,322		29,267	
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	1,464	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	Наар	CDF 5/3	Наар	CDF 5/3	Наар
	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

Заклучение

Разработан простой и эффективный алгоритм вейвлет-сжатия НВСТ, 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 НВСТ выигрывает у алгоритма МЕСТ 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 МЕСТ, were chosen as prototypes. Numerical 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.