

## ВЛОЖЕННОЕ И ЭНТРОПИЙНОЕ КОДИРОВАНИЕ ГИПЕРСПЕКТРАЛЬНЫХ СПУТНИКОВЫХ ИЗОБРАЖЕНИЙ

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Ловецкий М.Ю., Мирончик Д.Ю.

Рост объёма данных, передаваемых по различного рода сетям, в настоящее время опережает рост пропускной способности каналов, на которых эти сети построены. Примером сети с ограниченной пропускной способностью является система дистанционного зондирования земли (ДЗЗ), требующая алгоритм сжатия изображений, который обладал бы высоким коэффициентом сжатия и потенциалом для распараллеливания вычислений.

Как правило, соседние пиксели изображения имеют близкие значения яркости, т.е. коррелированы. Следовательно, сжатие изображений можно осуществлять с помощью декорреляции – такого перераспределения общей энергии изображения, что большая её часть лежит в относительно узком диапазоне. На сегодняшний день наиболее эффективны (по качеству и быстродействию) спектральные алгоритмы сжатия изображений, одним из которых является вейвлет-преобразование – стандарт в области декорреляции пикселей.

Полученную в результате такого преобразования матрицу вейвлет-коэффициентов необходимо развернуть в одномерный массив так, чтобы сохранить вложенность её пространственно-частотных диапазонов (это особенно важно для сжатия с потерями, т.к. поток должен быть упорядочен по убыванию значимости его составляющих) – эту задачу выполняет Z-развёртка. Её алгоритм может быть описан так: матрица разделяется на четыре равные части и считывается в следующем порядке – верхний левый сегмент, верхний правый, нижний левый, нижний правый. Процедура повторяется вплоть до отдельных элементов.

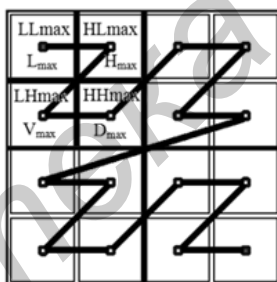


Рис. 1 – Z-развёртка на примере матрицы 4 на 4

В целях снижения объёмов используемой памяти и создания возможности распараллеливания вычислений было принято решение разбивать матрицу вейвлет-коэффициентов на блоки размером 32 на 32. Каждый из этих блоков проходит Z-развёртку и вложенное кодирование; результат становится частью выходного потока.

После Z-развёртки одного блока образовывается одномерный массив из 1024 элементов, который содержит вейвлет-коэффициенты в виде целых чисел со знаком. Точно так же, как все эти числа представимы в виде значений двоичных разрядов, блок представим в виде набора битовых плоскостей, каждая из которых (под номером  $n$ ) содержит  $n$ -ые разряды этих чисел. Одна из таких плоскостей будет являться плоскостью знаков.

Битовые плоскости кодируются по убыванию, при этом номер старшей битовой плоскости определяется номером старшего разряда самого большого абсолютного значения среди кодируемого блока. Используется бинарное кластерное дерево с размером кластера, равным четырём, показавшем в ходе экспериментов наилучшие показатели сжатия. Число уровней кластерного дерева выбирается из нулевого (отсутствие вложенного кодирования), первого и четвёртого – на каждом шаге алгоритма из трёх вариантов для записи в выходной поток выбирается кратчайший. Принцип прост – один элемент текущего уровня соответствует четырём элементам предыдущего уровня. Если среди этих четырёх элементов есть хоть одна единица, то соответствующий элемент следующего уровня равен единице, иначе – нулю.

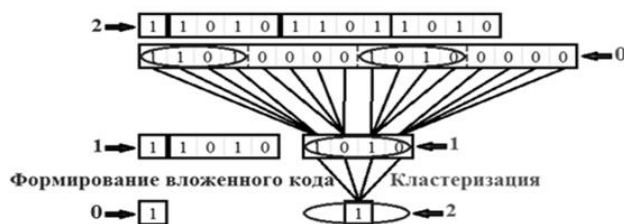


Рис. 2 – Формирование вложенного кода

Помимо формирования вложенного кода, отдельно обрабатываются знаки значащих (ненулевых) бит блока. Поскольку некоторое количество вейвлет-коэффициентов равно нулю, то выделение значащих бит на каждой битовой плоскости позволит помимо обеспечения прогрессивности потока сжать плоскость знаков (нули не имеют знака). Принцип формирования кода знаков тоже несложен: если число значащее – в выходной поток передаётся его знак. Для этого достаточно проверки знаковой плоскости на равенство нулю. Если элемент плоскости не равен нулю, происходит сверка с маской – не была ли произведена запись знака данного элемента. Если маска в этом месте пуста – в код знаков добавляется знак данного элемента, а маска обновляется.

Итоговый выходной поток состоит из набора последовательностей «четыре бита, в которых закодирован номер старшей битовой плоскости – флаг, указывающий на уровень вложенного кода – вложенный код – код знаков», соответствующих каждому закодированному блоку.

Дальнейшее увеличение коэффициента сжатия возможно при использовании энтропийного кодирования.

Энтропийное кодирование основывается на следующем факте: если вероятности появления различных символов в последовательности одинаковы, то энтропия – количество информации, приходящейся на один символ источника – максимальна. Таким образом, необходимо усреднить вероятности появления символов в описанном выше потоке, для чего используется код Хаффмана.

Кодирование по Хаффману состоит в сопоставлении наиболее вероятным символам исходной последовательности кодов наименьшей длины, а менее вероятным – кодов большей длины. Соответственно, предполагается, что вероятности появления тех или иных символов известны, так что необходимо провести статистический анализ выходного потока.

Из проведённого для разнородных изображений анализа следует, что распределение вероятностей появления различных символов в сегментах вложенного кода (ЕС) не является равномерным, что означает возможность их сжатия. Поскольку эти сегменты состоят из четырёхбитных кластеров, сплошное энтропийное кодирование с четырёхбитным окном требует мало аппаратных ресурсов и показывает неплохие результаты, однако для дальнейшего повышения эффективности нужно учесть внутреннюю структуру этих сегментов.

Вложенный код нулевого уровня (ЕС0) отличается от других вариантов вложенного кода своей шумоподобностью, так что энтропийное кодирование таких сегментов существенного выигрыша не принесёт (однако в целом оно возможно для 8-битных символов). Сегменты ЕС4 и ЕС1 похожи по структуре, но каждый из них следует разбить далее – на первый уровень сегмента и оставшуюся часть сегмента, которая получается из первого уровня путём логического суммирования битов кластера.

Сегменты кода знаков (SC) ещё более шумоподобны, чем ЕС0, и энтропийному кодированию не подлежат.

Таким образом, задача энтропийного кодирования заключается в построении деревьев кода Хаффмана для первого и последующих уровней ЕС4 (окно 4-битное), для первого и второго уровня ЕС1 (окно также 4-битное) и, возможно, для ЕС0 (окно 8-битное) – итого пяти деревьев, которые достаточно построить один раз и занести в память, и замене исходных 4-битных и 8-битных символов на код Хаффмана. Служебная информация – первые четыре бита закодированного блока, флаги – остаётся неизменной, так как необходима для декодирования, сегменты кода знаков так же передаются без изменений.

Код Хаффмана строится так: символы располагают в порядке убывания их вероятностей, складывают вероятности двух последних символов, переписывают ряд снова с учетом новой вероятности (суммы). Операцию повторяют, пока не получится 1. Нижнюю букву всегда кодируют нулем, а верхнюю – единицей.

Код Хаффмана является префиксным, что означает выполнение условия Фано: никакое кодовое слово не может быть началом другого кодового слова. Это позволяет записывать получившийся код без каких-либо разделительных символов и при необходимости однозначно его декодировать.

Программное моделирование описанного алгоритма показало, что его коэффициент сжатия близок к таковому у эталонного JPEG2000, а скорость существенно больше за счёт высокой степени распараллеливания. Кроме того, данный алгоритм лучше подходит для реализации в ПЛИС.

Список использованных источников:

1. Новицкий В.В., Цветков В.Ю. Сжатие полутоновых изображений на основе кластеризации и прогрессивного вложенного кодирования вейвлет коэффициентов / В.В. Новицкий, В.Ю. Цветков // Телекоммуникации: сети и технологии, алгебраическое кодирование и безопасность данных: материалы междунар. научно-технич. семинара. Минск, апрель–декабрь 2015 г. – Мн.: БГУИР, 2015. – С. 45-51.
2. Борискевич, А.А. Метод масштабируемого вложенного кодирования изображений на основе иерархической кластеризации вейвлет структур / А.А. Борискевич, В.Ю. Цветков // Доклады НАН Беларуси. – 2009. Т. 53, № 3. – С. 38 – 48.
3. Гонсалес Р. Цифровая обработка изображений. / Р. Гонсалес, Р. Вудс. – Пер. с англ. – Москва. – Техносфера, – 2006. – 1072 с.