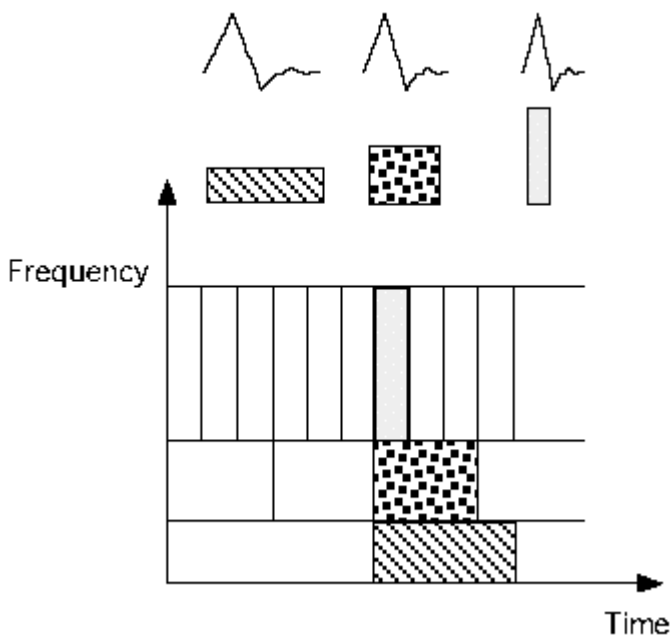


Военный университет связи

В.И.ВОРОБЬЕВ, В.Г.ГРИБУНИН

Теория и практика вейвлет - преобразования



С.-Петербург
1999

УДК:621.391
519.21

Теория и практика вейвлет-преобразования. ВОРОБЬЕВ В.И., ГРИБУНИН В.Г. ВУС, 1999. С.1-204.

Излагаются основные вопросы теории вейвлет-преобразования; рассмотрены принципы построения вейвлет-фильтров, практические аспекты осуществления преобразования, современные направления исследований в этой области; обсуждаются алгоритмы сжатия изображений с использованием вейвлет-преобразования; приведены технические данные о микросхемах ADV6xx, осуществляющих сжатие видео на основе этой технологии; в приложениях представлены характеристики и коэффициенты некоторых вейвлет-фильтров, а также пример программы на языке C++, выполняющей прямое и обратное вейвлет-преобразование.

Книга предназначена для инженерно-технических работников, а также аспирантов и студентов старших курсов вузов.

Владимир Иванович Воробьев
Вадим Геннадьевич Грибунин

Теория и практика вейвлет-преобразования

Редактор И. В. Тимофеева
Технический редактор Г. Н. Кузей

Подписано к печати 31 мая 1999 года
Объем: 12,75 п.л.; 12,5 уч.-изд.л. Тираж 500 экз. Зак.105

Типография ВУС, 194064, СПб., Тихорецкий пр., 3.

На базе Военного университета связи образован постоянный семинар **«Цифровые цепи, сигналы и системы»**. Он проводится под эгидой Комитета по высшему образованию Администрации Санкт-Петербурга и фирмы АВТЭКС. На семинаре рассматриваются актуальные вопросы теории и практики построения систем связи и управления, современные схмотехнические решения и электронные компоненты.

Семинар проводится в виде лекций, читаемых профессорско-преподавательским составом ВУЗов Санкт-Петербурга и Москвы, а также представителями фирм-производителей. За время работы семинара в нем приняли участие представители таких фирм, как **Analog Devices, Motorola, Siemens, Texas Instruments, Zilog** и многих других.

Периодичность семинара – один раз в месяц. Посещение **бесплатное**. Руководитель семинара - кандидат технических наук, доцент Воробьев Владимир Иванович. Справки по телефону **556-98-06**.

Желающие глубже изучить теорию вейвлет-преобразования, познакомиться с современными направлениями работ в этой области приглашаются принять участие в работе постоянно действующего семинара **«Вейвлеты (всплески) и их приложения»**.

Семинар организован благодаря энтузиазму профессорско-преподавательского состава СПбГУ, СПбГТУ и других вузов Санкт-Петербурга. Идея семинара заключается в обеспечении взаимопонимания «инженеров» и «математиков», взаимопроникновения теории и практики, которое может дать импульс к появлению качественно новых результатов.

Подробнее узнать о работе семинара, его истории, ознакомиться с аннотациями состоявшихся и темой очередного докладов можно на интернет-странице семинара, расположенной на сервере СПбГУ. Справки по телефону **136-90-79** (вечером).

ПРЕДИСЛОВИЕ

В последнее десятилетие в мире возникло и оформилось новое научное направление, связанное с так называемым вейвлет-преобразованием. Слово «wavelet», являющееся переводом французского «ondelette», означает небольшие волны, следующие друг за другом. Можно без преувеличения сказать, что вейвлеты произвели революцию в области теории и практики обработки нестационарных сигналов. В настоящее время вейвлеты широко применяются для распознавания образов; при обработке и синтезе различных сигналов, например речевых, медицинских; для изучения свойств турбулентных полей и во многих других случаях.

Особо большое развитие получила практика применения вейвлетов для решения задач сжатия и обработки изображений, являющихся нестационарными по своей природе. В этой области применение вейвлет-преобразования позволило достичь одновременного снижения сложности и повышения эффективности кодеров. В настоящее время уже находятся в разработке международные стандарты по сжатию неподвижных изображений и видео – JPEG2000 и MPEG-4. Ядром этих стандартов будет вейвлет-преобразование.

Огромный интерес к изучению теории и практики вейвлет-преобразования вызвал лавинообразный поток издающейся литературы. В США и других развитых странах ежегодно издаются десятки книг, учебных пособий, тематических выпусков журналов, посвященных данной тематике. На этом фоне почти полное отсутствие публикаций в отечественных журналах выглядит достаточно странно. Целью данной книги является заполнение существующего пробела в этой области.

Теория и практика вейвлет-преобразования находится на стыке различных наук: математики, физики и т.д. В настоящей книге изложение ведется с позиций цифровой обработки сигналов, с учетом специализации авторов.

К каждой главе прилагается список используемых источников. Ссылок на книги в тексте практически нет. Это связано с тем, что ссылки имели бы для нашего читателя чисто теоретический интерес: зарубежные книги по данной тематике на русский язык не переводятся и практически отсутствуют даже в Публичной библиотеке. С другой стороны, количество работ различного рода в "основном" источнике информации – Интернете – весьма велико. Поэтому приводится несколько наиболее интересных, с точки зрения авторов, адресов.

Книга является обобщением научно-технической литературы и состоит из десяти глав. В первой главе кратко рассматриваются различные линейные преобразования изображений. Для понимания сути вейвлет-преобразования особое внимание следует обратить на схемы пирамидального кодирования и квадратурно-зеркальные фильтры. Во второй главе дано введение в теорию вейвлет-преобразования. Хотя авторы старались минимизировать число формул, эта глава вышла наиболее «математизированной». В третьей главе разбираются практические вопросы, связанные с осуществлением фильтрации изображений. Здесь же более подробно рассматриваются квадратурно-зеркальные фильтры. В четвертой главе показано отличие «классических» фильтров от вейвлет-фильтров и приведены возможные методики расчета некоторых классов вейвлет-фильтров. В пятой главе рассматривается построение адаптивных кодеров изображений на основе вейвлет-преобразования. Приводится их сравнение по различным критериям. Шестая глава посвящена относительно новому методу выполнения вейвлет-преобразования – лифтинговой схеме. В седьмой главе рассматриваются возможные способы осуществления целочисленного вейвлет-преобразования. Мультивейвлеты, или вейвлеты с несколькими масштабирующими функциями, описаны в восьмой главе. В девятой главе представлены некоторые потенциальные характеристики сжатия изображений с применением вейвлет-преобразования, а в десятой - основные современные алгоритмы сжатия. Одинадцатая глава посвящена описанию возможностей и принципов функционирования вейвлет-кодексов изображения семейства ADV, выпускаемых фирмой Analog Devices. Это – первые и пока единственные микросхемы, использующие вейвлет-преобразование изображений.

Авторы выражают искреннюю признательность фирме АВТЭКС и ее региональному представителю Цуканову Юрию Васильевичу, а также Йоханнесу Хорвату – представителю фирмы Analog Devices, любезно ознакомившими нас со своими материалами и предоставившими возможность издания данной книги.

ВВЕДЕНИЕ

Первое упоминание о вейвлетах появилось в литературе по цифровой обработке и анализу сейсмических сигналов (работы А.Гроссмана и Ж.Морлета). Так как интерес авторов заключался в анализе сигналов, набор базисных функций был избыточным. Далее, математик И.Мейер показал существование вейвлетов, образующих ортонормальный базис в $L^2(R)$. Дискретизация вейвлет-преобразования была описана в статье И.Добеши, которая перекинула мост между математиками и специалистами в области обработки сигналов. Добеши разработала семейство вейвлет-фильтров, имеющих максимальную гладкость для данной длины фильтра. Популярность вейвлетов увеличилась после введения С.Маллатом концепции кратномасштабного анализа. Он же, по-видимому, первым применил вейвлеты для кодирования изображений.

И И.Добеши, и С.Маллат показали, что практическое выполнение вейвлет-преобразования осуществляется посредством двухполосного банка фильтров анализа-синтеза, известного ранее в теории субполосного кодирования (см. главу 1). Эта теория может быть описана в терминах вейвлетов. Главное различие между этими двумя направлениями заключается в критериях построения фильтров, как это будет показано далее.

Некоторые идеи теории вейвлетов частично были разработаны уже очень давно. Например, А.Хаар опубликовал в 1910 году полную ортонормальную систему базисных функций с локальной областью определения. Эти функции называются теперь вейвлетами Хаара.

В настоящее время исследования в области вейвлетов ведутся по многим направлениям. В главе 6 будет представлена лифтинговая схема выполнения вейвлет-преобразования, имеющая ряд преимуществ по сравнению с традиционной. Материалы этой главы в основном основаны на работах В.Сведенса, ссылки на которые имеются в списке литературы. Активно исследуется целочисленное вейвлет-преобразование, которому посвящена глава 7. Многие перспективные направления в области теории и практики вейвлет-преобразования, такие как вейвлеты, оптимальные по некоторому критерию, вейвлеты, определенные на интервале, вейвлеты, адаптивные к сигналу и т.д., не нашли своего отражения на страницах книги.

Несмотря на то, что теория вейвлет-преобразования уже в основном разработана, точного определения, что же такое "вейвлет", какие функции можно назвать вейвлетами, насколько известно, не существует. Обычно под

вейвлетами понимаются функции, сдвиги и растяжения которых образуют базис многих важных пространств, в том числе и $L^2(\mathbb{R})$. Эти функции являются компактными как во временной, так и в частотной области. Вейвлеты непосредственно связаны с кратномасштабным анализом сигналов.

Вейвлеты могут быть ортогональными, полуортогональными, биортогональными. Эти функции могут быть симметричными, асимметричными и несимметричными. Различают вейвлеты с компактной областью определения и не имеющие таковой. Некоторые функции имеют аналитическое выражение, другие – быстрый алгоритм вычисления связанного с ними вейвлет-преобразования. Вейвлеты различаются также степенью гладкости. Для практики желательно было бы иметь ортогональные симметричные (асимметричные) вейвлеты. К сожалению, доказана теорема о том, что такими вейвлетами являются лишь вейвлеты Хаара. Функции Хаара не обладают достаточной гладкостью и не подходят для большинства приложений. Поэтому для кодирования изображений обычно используют биортогональные вейвлеты.

В настоящее время многие исследователи понимают под вейвлетами более широкий класс функций. Это и вейвлет-пакеты, рассматриваемые в главе 5, и локальные тригонометрические базисы (вейвлеты Малвара), и мультивейвлеты, и так называемые вейвлеты второго поколения, не являющиеся сдвигами и растяжениями одной функции. Базисы преобразования Фурье не являются вейвлетами, так как у них отсутствует локализация в пространстве (времени).

Российские математики вейвлеты иногда называют всплесками. На наш взгляд, этот термин является неудачным, а попытка русификации терминологии может ввести в заблуждение и породить ошибки.

Некоторым может показаться, что вейвлеты не являются чем-то фундаментально новым. В самом деле, сходные идеи появлялись на протяжении последних десятилетий : субполосное кодирование, успешно применяемое при кодировании речи, пирамидальные схемы кодирования изображений, преобразование и функции Габора (вейвлеты Габора). С развитием теории вейвлетов произошло как бы объединение, взаимопроникновение, взаимообогащение этих идей, что привело к качественно новому результату.

Так как с точки зрения практики наиболее интересными представляются быстрые алгоритмы вычисления вейвлет-преобразования, то в первой главе кратко рассматриваются вопросы, связанные с так называемым субполосным кодированием.

СУБПОЛОСНОЕ КОДИРОВАНИЕ

Одним из основных средств обработки сигналов является линейное преобразование. Субполосное кодирование (применяются также термины «субполосная фильтрация», «субполосное преобразование», «подполосное кодирование») является частным случаем линейного преобразования и имеет многочисленные полезные свойства. В данной главе мы обсуждаем различные аспекты субполосного кодирования и иллюстрируем их применение для кодирования изображений.

Традиционно кодеры, основанные на линейном преобразовании, делятся на две группы: кодеры с преобразованием и субполосные кодеры. Как будет показано, два типа преобразований различаются в основном методами их вычислений.

Кодирование с преобразованием обычно основывается на ортогональном линейном преобразовании. Классическим примером такого преобразования является дискретное преобразование Фурье (ДПФ), которое декомпозирует сигнал на синусоидальные компоненты. Двумя другими примерами являются дискретное косинусное преобразование (ДКП) и преобразование Карунена-Лозва (ПКЛ), известное также и под другими названиями. Эти преобразования находятся путем вычисления свертки сигнала конечной длины с семейством базисных функций. В результате получается ряд коэффициентов, который и подвергается дальнейшей обработке. На практике многие из этих преобразований имеют эффективные алгоритмы вычислений. Отличительной особенностью преобразований является то, что они обычно применяются к неперекрывающимся блокам сигнала.

Субполосное кодирование реализуется путем свертки сигнала с несколькими полосовыми фильтрами и децимацией результата. Совокупность набора фильтров с дециматорами называется банком или блоком фильтров. Каждый получившийся в результате преобразования сигнал несет в себе информацию о спектральной составляющей исходного сигнала при некотором пространственном (временном) масштабе. Так происходит анализ сигнала. Для обратного синтеза сигнала (его реконструкции) выполняется операция интерполяции субполосных сигналов, фильтрация и их сложение. Большинство методов синтеза фильтров направлено на устранение наложения спектров («элайзинга»), возникающего при децимации. В пространственной области элайзинг проявляется в виде дискретной структуры синтезированного изображения. Идеальный банк фильтров должен включать фильтры с прямоугольной характеристикой, предотвращающие вместе с тем элайзинг. Такие

фильтры, однако, приводят к так называемому эффекту Гиббса, вследствие которого искажения на реконструированном изображении слишком заметны.

Хотя кодеры с линейным преобразованием обычно делят на две категории, можно показать, что различия между ними весьма незначительны. Например, кодер, использующий ДКП, принято относить к кодерам с преобразованием. Однако вычисление ДКП неперекрывающихся блоков эквивалентно свертке всего исходного изображения с соответствующими базисными функциями и децимации результата в N раз, где N – число блоков. Таким образом, ДКП можно также рассматривать как разновидность субполосного кодирования. Фурье-преобразование базисных функций имеет определенную (хотя и недостаточно хорошую) локализацию и несет, таким образом, информацию о некоторой субполосе.

Результаты, представленные в данной главе, основаны на анализе как во временной, так и в частотной области. Поэтому используются два подхода к их описанию: матричное описание, применяемое в линейной алгебре, и описание в частотной области, наиболее часто применяющееся в цифровой обработке сигналов. Далее будут представлены два типа обозначений и показана их взаимосвязь. Для простоты ограничимся анализом одномерных сигналов, хотя результаты могут быть легко обобщены и для двумерного случая.

1.1. Требования, предъявляемые к преобразованиям

Итак, у кодирования с преобразованием есть много общего с субполосным кодированием. Какое преобразование выбирать и по какому критерию, зависит от конкретной задачи. Рассмотрим свойства, которые являются важными при кодировании изображений.

1. Масштаб и ориентация.

Для эффективного представления изображения важную роль играет масштаб. В изображениях имеются объекты самых различных размеров. Поэтому, преобразование должно позволять анализировать изображение одновременно (и независимо) на различных масштабах. В главе 2, изучая теорию вейвлетов, мы будем говорить о кратномасштабном анализе. В частотной области это эквивалентно логарифмической шкале.

Для двумерного сигнала некоторая спектральная область соответствует определенному масштабу и ориентации. Ориентация базисных функций определяет способность преобразования корректно анализировать ориентированные структуры, типичные для изображений. Примером могут служить контуры и линии. Таким образом, для решения задачи анализа желательно иметь преобразование, которое бы делило входной сигнал на локальные частотные области.

2. Пространственная локализация.

Кроме частотной локализации, базисные функции должны быть локальными и в пространстве. Необходимость в пространственной локализации

преобразования возникает тогда, когда информация о местоположении деталей изображения является важнейшей. Эта локальность, однако, не должна быть «абсолютной», блочной, как при ДКП, так как это ведет к потере свойства локальности в частотной области.

Наиболее часто применяемый подход при анализе заключается в следующем: сигнал дискретизируется, затем выполняется ДПФ. Что же получается в результате? Сначала сигнал раскладывается по базису единичного импульса, который не имеет частотной локальности, а затем по базису синусоид с четными и нечетными фазами, не имеющих пространственной локальности. Конечно, представление сигнала в частотной области исключительно важно для его анализа. Однако это не означает, что выбор функций импульса и синусоиды для решения этой задачи является наилучшим. Еще в 1946 году Д.Габор предложил класс линейных преобразований, которые обеспечивают локальность и в частотной, и во временной области. Базис единичного импульса и базис синусоиды могут рассматриваться как два экстремальных случая этих преобразований. Функции Габора будут рассмотрены в разделе 1.3. Вейвлеты являются еще одним примером функций, хорошо локализованных в пространственной и частотной областях.

3. Ортогональность.

Вообще говоря, преобразование не обязательно должно быть ортогональным. Так, ортогональность обычно не рассматривается в контексте субполосного кодирования, хотя вейвлет-преобразование, как правило, является ортогональным. Ортогональность функций упрощает многие вычисления. Кроме того, как будет показано, «сильно» неортогональное преобразование может быть неприемлемо для кодирования.

4. Быстрые алгоритмы вычисления.

Это, наверное, наиболее важное свойство. Так как невозможность практической реализации преобразования в реальном масштабе времени сводит на нет все его положительные свойства.

1.2. Линейные преобразования конечных сигналов

1.2.1. Система фильтров анализа-синтеза

Будем рассматривать линейные преобразования сигналов конечного размера, которые могут быть выражены в терминах свертки с КИХ-фильтрами. На рис. 1.1 показана система, осуществляющая такое преобразование при помощи банков фильтров анализа-синтеза.

Обозначения на рисунке являются стандартными для цифровой обработки сигналов. $|H_i(\omega)|$ означает операцию круговой свертки входного сигнала

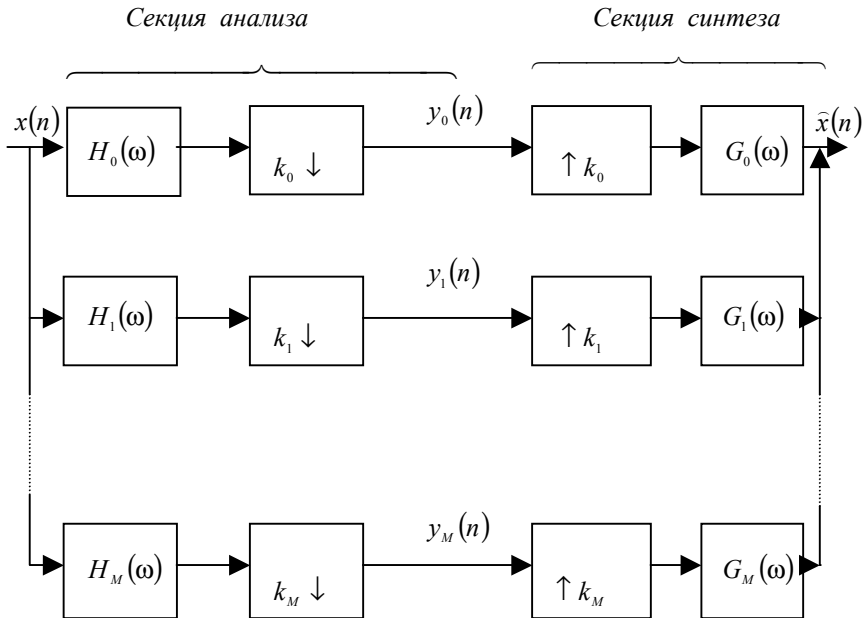


Рис. 1.1. Банк фильтров анализа-синтеза

длиной N с импульсной характеристикой КИХ-фильтра $h_i(n)$ и Фурье-преобразование результата:

$$H_i(\omega) = \sum_n h_i(n) e^{-j\omega n}. \quad (1.1)$$

Естественным образом предполагается, что длина фильтра меньше размера изображения. Блоки $k_i \downarrow$ означают децимацию в k_i раз, блоки $\uparrow k_i$ - интерполяцию в k_i раз. Напомним, что децимация означает оставление лишь каждого k_i отсчета, интерполяция означает вставку $k_i - 1$ нулей между этими отсчетами. Предполагается, что k_i - целые числа и делят N .

Будем называть такую систему системой А-С. Секция анализа системы А-С выполняет линейное преобразование над входным сигналом $x(n)$ длиной n . В результате получается M последовательностей $y_i(n)$ длиной N/k_i .

Операции, выполняемые секцией синтеза, являются обратными операциям секции анализа. В результате получается сигнал $\hat{x}(n)$. Точно так же строится система А-С и для многомерного сигнала.

Таким образом, коэффициенты преобразования вычисляются через свертку. Интуитивно понятно, что это хорошо, так как различные участки сигнала будут обрабатываться одинаковым образом. Далее, формулирование проблемы в частотной области позволяет легко разделить ошибку реконструкции $\varepsilon(n) = \hat{x}(n) - x(n)$ на две части: элайзинговую составляющую и составляющую, инвариантную к сдвигу. Для этого запишем выходной сигнал схемы анализа в частотной области:

$$Y_i(\omega) = \frac{1}{k} \sum_{j=0}^{k-1} H_i \left(\frac{\omega + 2\pi j}{k} \right) X \left(\frac{\omega + 2\pi j}{k} \right). \quad (1.2)$$

Тогда выходной сигнал схемы А-С

$$\hat{X}(\omega) = \sum_{i=0}^{M-1} Y_i(k\omega) G_i(\omega) \quad (1.3)$$

с учетом эффекта интерполяции и децимации в частотной области. Объединяя выражения (1.2) и (1.3), получаем

$$\begin{aligned} \hat{X}(\omega) &= \frac{1}{k} \sum_{i=0}^{M-1} \left[\sum_{j=0}^{k-1} H_i \left(\omega + \frac{2\pi j}{k} \right) X \left(\omega + \frac{2\pi j}{k} \right) \right] G_i(\omega) = \\ &= \frac{1}{k} \sum_{i=0}^{M-1} H_i(\omega) G_i(\omega) X(\omega) + \frac{1}{k} \sum_{j=1}^{k-1} X \left(\omega + \frac{2\pi j}{k} \right) \sum_{i=0}^{M-1} H_i \left(\omega + \frac{2\pi j}{k} \right) G_i(\omega). \end{aligned} \quad (1.4)$$

Здесь первое слагаемое соответствует отклику линейной времянезависимой системе, а второе соответствует элайзингу системы.

1.2.2. Каскадное соединение систем А-С

Преимуществом введения понятия систем А-С является то, что оно позволяет наглядно представить и проанализировать иерархически построенные преобразования. Предположим, что система А-С удовлетворяет требованию полного восстановления (то есть $\hat{x}(n) = x(n)$). Промежуточный сигнал этой системы $y_i(n)$ может подаваться на вход некоторой другой системы А-С, в результате чего получается иерархическая каскадно соединенная система. Пример такой системы показан на рис. 1.2, где система А-С применяется повторно к своему же промежуточному сигналу $y_0(n)$.

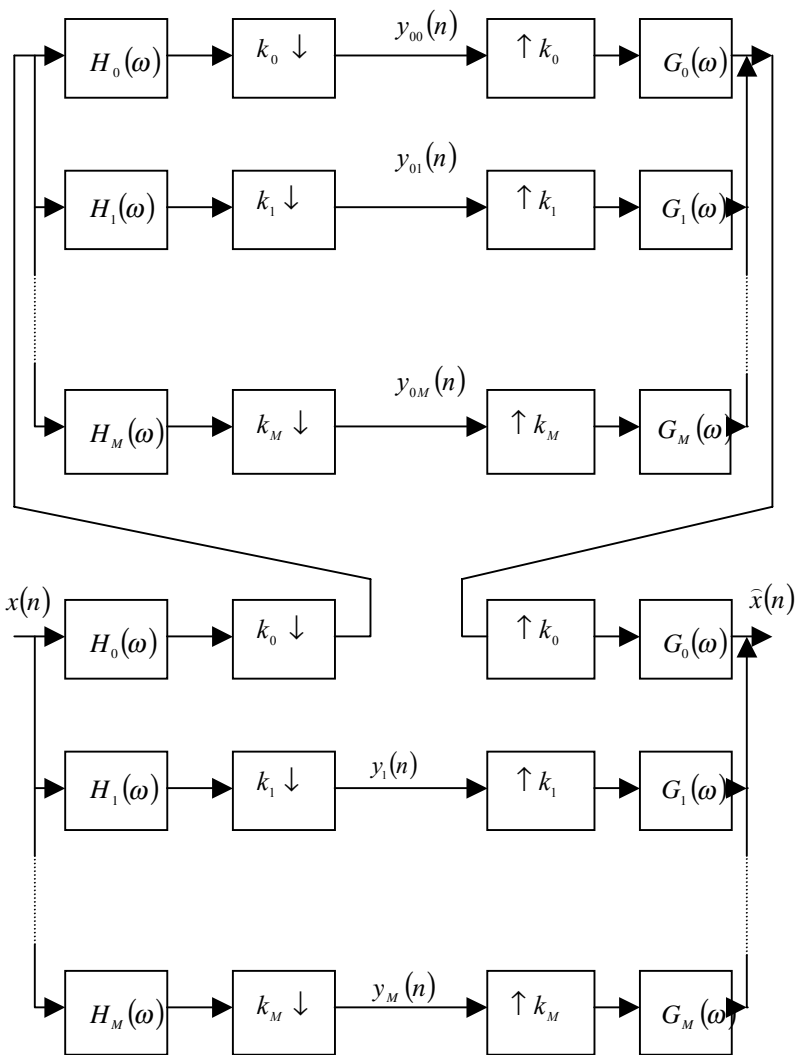


Рис. 1.2. Неравномерный каскадный банк фильтров анализа - синтеза

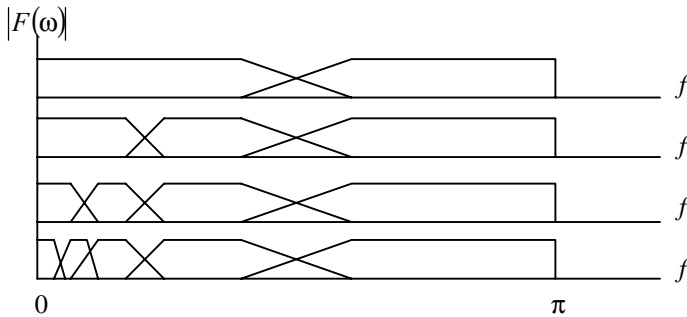


Рис. 1.3. Октавополосное разбиение частотного плана четырехуровневой пирамиды, построенной на основе двухканальной системы А-С

Если первоначальная система А-С обладала свойством полного восстановления, то и получившаяся двухкаскадная система также будет обладать этим свойством. Если дальнейшему разложению подвергается каждый промежуточный сигнал, то такая система называется равномерной системой.

В противном случае мы имеем дело с неравномерной, или пирамидальной системой, как показано на рис. 1.2. В разделе 1.3 будут обсуждаться пирамидальные системы, построенные на основе двухканальных систем А-С. Такое каскадирование приводит к октавополосному разбиению частотного плана, как показано на идеализированной частотной диаграмме (см. рис.1.3). Здесь на верхней диаграмме показано разбиение частотного плана двухканальной системой А-С. Следующие диаграммы демонстрируют последовательное повторение применения той же системы к низкочастотной части сигнала.

Таким образом, нижняя диаграмма соответствует четырехуровневому разбиению частотной области. Как будет показано в дальнейшем, такие системы лежат в основе быстрого алгоритма вычисления вейвлет-преобразования.

1.2.3. Представление субполосного кодирования при помощи аппарата матриц

Дискретный сигнал или изображение можно представить в виде некоторого вектора-столбца x длиной N . Тогда линейному преобразованию изображения будет соответствовать умножение вектора-столбца x на матрицу размером $M \times N$.

Система А-С, показанная на рис. 1.1, соответствует линейному преобразованию. Поэтому ее можно представить в следующем виде:

$$y_i(m) = \sum_{l=0}^{N-1} x(l)h_i(k_i m - l) \quad (1.5)$$

и

$$\hat{x}(n) = \sum_{i=0}^{M-1} \sum_{m=0}^{\frac{N}{k_i}-1} y_i(m)g_i(n - k_i m), \quad (1.6)$$

где позиции отсчетов фильтра и сигнала $(k_i m - l)$ и $(n - k_i m)$ вычисляются по модулю N . Эти выражения могут быть записаны в матрично-векторной форме:

$$y = \mathbf{H}^T x \quad (1.7)$$

и

$$\hat{x} = \mathbf{G}y, \quad (1.8)$$

или, объединив эти равенства,

$$\hat{x} = \mathbf{G}\mathbf{H}^T x, \quad (1.9)$$

где y и \hat{x} - векторы длиной N , \mathbf{H}^T означает выполнение операции транспонирования и

$$\mathbf{H} = \begin{bmatrix} h_0(0) & h_0(k_0) & h_1(0) & h_1(k_1) & & \\ h_0(-1) & h_0(k_0 - 1) & h_1(-1) & h_1(k_1 - 1) & & \\ h_0(-2) & h_0(k_0 - 2) & h_1(-2) & h_1(k_1 - 2) & & \\ \vdots & h_0(k_0 - 3) \dots & \vdots & h_1(k_1 - 3) & \dots & \\ & h_0(k_0 - 4) & & h_1(k_1 - 4) & & \\ h_0(2) & \vdots & h_1(2) & \vdots & & \\ h_0(1) & & h_1(1) & & & \end{bmatrix}, \quad (1.10)$$

$$\mathbf{G} = \begin{bmatrix} g_0(0) & g_0(k_0) & g_1(0) & g_1(k_1) \\ g_0(1) & g_0(k_0+1) & g_1(1) & g_1(k_1+1) \\ g_0(2) & g_0(k_0+2) & g_1(2) & g_1(k_1+2) \\ \vdots & g_0(k_0+3) \dots & \vdots & g_1(k_1+3) \dots \\ & g_0(k_0+4) & & g_1(k_1+4) \\ g_0(-2) & \vdots & g_1(-2) & \vdots \\ g_0(-1) & & g_1(-1) & \end{bmatrix}. \quad (1.11)$$

Столбцы матрицы \mathbf{G} , состоящие из сдвинутых импульсных характеристик фильтров, называются базисными функциями синтеза, а столбцы матрицы \mathbf{H} – функциями анализа.

Итак, мы можем представить любую линейную систему А-С в матричной форме. Верно и обратное: существует система А-С, соответствующая некоторому линейному преобразованию и обратному ему преобразованию, задаваемому обратимой матрицей \mathbf{M} . Для данной матрицы \mathbf{M} , имеющей l строк, блок фильтров анализа будет содержать l различных фильтров, каждый из которых определяется строкой матрицы \mathbf{M} .

1.2.4. Обратное преобразование

Преимуществом матричного представления преобразования является то, что мы всегда можем определить условия существования обратного преобразования. Как видно из равенства (1.9), для того, чтобы система А-С обладала свойством полного восстановления, необходимо

$$\mathbf{G}\mathbf{H}^T = \mathbf{I}, \quad (1.12)$$

где \mathbf{I} – единичная матрица. Если \mathbf{H} имеет ранг N и является квадратной, матрица синтеза будет следующего вида:

$$\mathbf{G} = (\mathbf{H}^{-1})^T, \quad (1.13)$$

и тоже будет являться квадратной ранга N . Далее мы увидим, что обратное преобразование применяется для анализа систем А-С. Кроме того, можно показать, что матрицы \mathbf{H} и \mathbf{G} можно поменять местами. Тогда функции анализа будут использоваться как функции синтеза и наоборот.

Если матрица \mathbf{H} ранга N не квадратная (то есть представление избыточное), можно построить систему с полным восстановлением путем выбора в качестве \mathbf{G} псевдоинверсной матрицы:

$$\mathbf{G} = (\mathbf{H}\mathbf{H}^T)^{-1} \mathbf{H}. \quad (1.14)$$

Если \mathbf{H} – квадратная, равенство (1.14) вырождается в (1.13). Аналогично, если мы имеем (возможно, неквадратную) матрицу \mathbf{G} ранга N , $\mathbf{H} = (\mathbf{G}\mathbf{G}^T)^{-1} \mathbf{G}$.

1.2.5. Ортогональное преобразование

Как было отмечено ранее, ортогональность обычно не рассматривается в контексте субполосного кодирования. Тем не менее, это свойство весьма важно для кодирования изображений, как будет показано в разделе 1.3. Матрица ортогонального преобразования является квадратной и обладает следующим свойством:

$$\mathbf{M}\mathbf{M}^T = \mathbf{M}^T\mathbf{M} = \mathbf{I}. \quad (1.15)$$

Это означает, что скалярное произведение любых двух ее столбцов (или базисных функций преобразования) должно быть равно нулю. Кроме того, скалярное произведение столбца с самим собой должно давать единичную матрицу.

Условие ортогональности накладывает ряд ограничений на систему А-С. Так как матрица преобразования является квадратной, число коэффициентов преобразования должно равняться числу отсчетов в исходном сигнале. Для системы А-С это означает, что

$$\sum_{i=0}^{M-1} \frac{1}{k_i} = 1, \quad (1.16)$$

где N является делимым всех k_i . Такая система называется критически дискретизированным банком фильтров.

Второе, и более важное условие, налагаемое ортогональностью, заключается в следующем. Условие ортогональности (1.15) с учетом условия полного восстановления (1.12) приводит к равенству

$$\mathbf{G} = \mathbf{H}. \quad (1.17)$$

Из выражений для матриц преобразования \mathbf{H} и \mathbf{G} через импульсные характеристики фильтров h и g (1.10) и (1.11) получим взаимосвязь между фильтрами анализа и синтеза:

$$g_i(n) = h_i(-n), \quad \forall i. \quad (1.18)$$

Другими словами, фильтры синтеза ортогонального преобразования являются инвертированными во времени копиями фильтров анализа.

1.3. Некоторые примеры преобразований

В данном разделе мы рассмотрим три примера одномерных преобразований. Представлены достоинства и недостатки преобразований в свете кодирования изображений.

1.3.1. Преобразование Габора

Как было сказано, базисные функции преобразования должны быть локализованы как в пространственной (временной), так и в частотной областях. Одно из решений этой проблемы было предложено Д.Габором. Габор представил преобразование, в котором базисными функциями являются синусоиды, модулированные гауссовским окном. Преобразование Габора можно рассматривать как выполнение локализованной частотной декомпозиции в ряд перекрывающихся окон. Базисные функции Габора локализованы по частоте и по времени. Габор показал, что эти функции являются оптимальными с точки зрения локализации относительно выбранной им меры. (Позднее было показано, что выбор другой меры ведет к другим оптимальным функциям). Первые пять базисных функций преобразования Габора вместе с их спектрами показаны на рис. 1.4. Как сами базисные функции, так и их спектр являются гладкими и компактными. Функции Габора можно перенести и на двумерный случай. Они могут быть применены для сжатия изображения.

Главный недостаток преобразования Габора заключается в неортогональности базисных функций (то есть функции анализа коренным образом отличаются от функций синтеза). Функции анализа преобразования Габора являются плохо обусловленными как в пространственной, так и в частотной областях. Это приводит к распространению ошибок квантования коэффициентов по всей частотной и пространственной областям, несмотря на то, что значения коэффициентов вычислялись для локальной области.

Интересно отметить, что локализация базисных функций Габора может быть значительно улучшена, если использовать избыточное представление. Оно выполняется путем более частого, чем требуется, наложения окна Гаусса либо путем деления на части каждого частотного окна. Однако это приводит к увеличению числа коэффициентов, что неприменимо для кодирования. Таким образом, возможность применения избыточного преобразования Габора для целей кодирования требует дополнительного исследования.

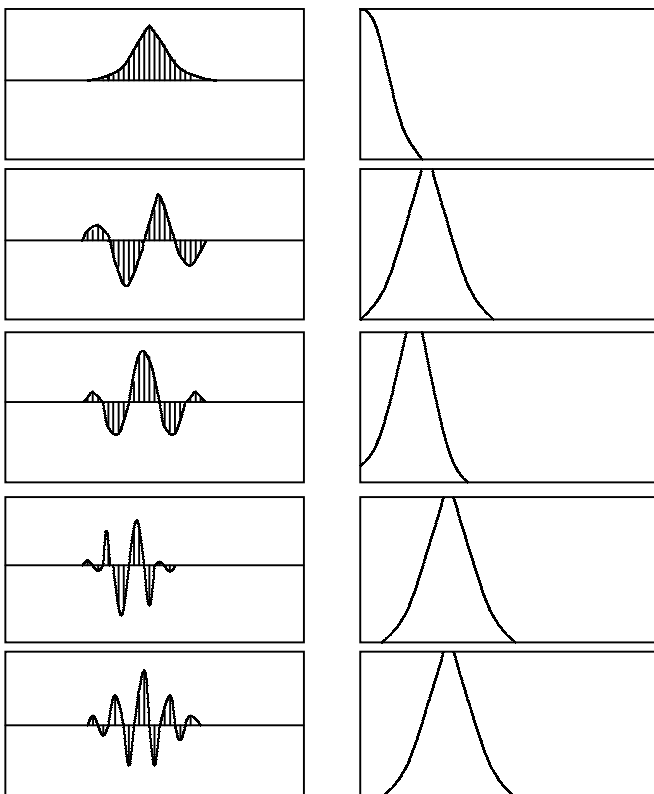


Рис. 1.4. Пять из шестнадцати базисных функций Габора с соответствующими спектрами Фурье. Преобразования изображены на линейной шкале в диапазоне от 0 до π

Некоторыми авторами обсуждалось применение похожих избыточных преобразований для целей кодирования. Однако высоких результатов достичь не удалось.

1.3.2. Дискретное косинусное и перекрывающееся ортогональное преобразования

Использование дискретного косинусного преобразования для кодирования изображений стандартизировано в ряде международных стандартов: JPEG, MPEG и других. Его применение основано на представлении изобра-

жения как источника с гауссовой статистикой. Для такого источника оптимальным является преобразование Карунена-Лозва, у которого отсутствует быстрый алгоритм выполнения. Кроме того, оно требует знания статистики кодируемого сигнала. ДКП достаточно точно аппроксимирует преобразование Карунена-Лозва. Обычно преобразование применяется не ко всему изображению, а только к его неперекрывающимся блокам размером 8×8 или 16×16 . Блочное ДКП можно рассматривать как субполосное кодирование, при котором базисные функции плохо локализованы в частотной области. Рассматривая ДКП в контексте системы А-С, можно показать, что в коэффициентах преобразования будет иметь место элайзинг. Так как преобразование обратимое, этот элайзинг будет устранен на этапе синтеза. Однако если коэффициенты квантуются или отбрасываются (например, в схеме сжатия), элайзинг не устраняется и проявляется в виде артефакта блочности в реконструированном изображении.

Существует возможность уменьшения элайзинга в блочном ДКП. Для этого над коэффициентами из соседних блоков выполняется еще одно ортогональное преобразование. В результате получается перекрывающееся ортогональное преобразование (ПОП). Базисные функции соседних блоков этого преобразования перекрываются, а импульсные характеристики сужаются возле границ. Х.Малваром разработан быстрый алгоритм вычисления ПОП, имеющий аналогии с «бабочкой» при БПФ. Существенным недостатком ПОП является то, что оно делит спектр на равные субполосы, тогда как во многих случаях желательно иметь логарифмическое разбиение спектра.

1.3.3. Пирамида Лапласа

Один из первых методов для получения октавополосной декомпозиции был разработан и применен для кодирования изображения П.Буртом и Э.Адельсоном. Они использовали каскадно включенные гауссовские фильтры для получения избыточного представления сигнала, которое они назвали пирамидой Лапласа. Схема получения одного уровня пирамиды Лапласа (для одномерного сигнала) показана на рис. 1.5.

Сигнал пропускается через НЧ-фильтр $B(\omega)$ и затем прореживается. В результате получается низкочастотная субполоса W_0 . Высокочастотная субполоса W_1 формируется за счет последовательного выполнения следующих операций: интерполяции W_0 , свертки с интерполирующим фильтром $A(\omega)$ и вычитания результата из исходного сигнала. Реконструкция сигнала происходит путем интерполяции W_0 , свертки с интерполирующим фильтром

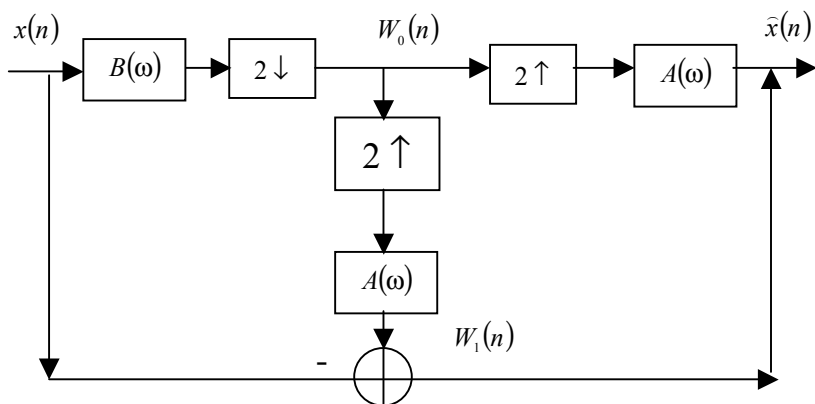


Рис. 1.5. Схематическое изображение одного уровня пирамиды Лапласа

$A(\omega)$ и сложения с W_1 . Восстановленный сигнал точно соответствует исходному, вне зависимости от выбора фильтров $A(\omega)$ и $B(\omega)$. Полная пирамида строится рекурсивно, с применением схемы рис.1.5 к низкочастотной субполосе. Фильтры $A(\omega)$ и $B(\omega)$ обычно выбираются одинаковыми НЧ фильтрами, хотя лучшие результаты в кодировании достигаются при независимом выборе фильтров.

Пирамида Лапласа обладает дополнительным привлекательным свойством – многомасштабностью представления. Изображение получается представленным одновременно на нескольких уровнях разрешения. Такой подход позволяет осуществлять прогрессивную передачу изображения по каналу с ограниченной пропускной способностью. При этом вначале передается самое грубое приближение (низкочастотная часть), а затем передаются детали, от уровня к уровню.

Для сравнения пирамиды Лапласа с другими субполосными преобразованиями представим ее как трехканальную систему А-С (см. рис.1.1), полученную путем деления W_1 на два сигнала: Y_1 , содержащего четные коэффициенты, и Y_2 , содержащего нечетные коэффициенты. Так как децимация во всех трех ветвях осуществляется в два раза, пирамидальное представление является избыточным в $3/2$ раза. Фильтры системы А-С выражаются через фильтры пирамиды следующим образом:

$$\begin{aligned}
H_0(\omega) &= B(\omega), & G_0(\omega) &= A(\omega), \\
H_1(\omega) &= \frac{1}{2} [1 - B(\omega)A(\omega) - B(\omega)A(\omega + \pi)], & G_1(\omega) &= 1, \\
H_2(\omega) &= \frac{e^{j\omega}}{2} [1 - B(\omega)A(\omega) - B(\omega)A(\omega + \pi)], & G_2(\omega) &= e^{-j\omega}.
\end{aligned} \tag{1.19}$$

Так как в данной системе не выполняется условие (1.18), пирамида Лапласа является неортогональным преобразованием. Пирамида Лапласа может строиться и для двумерного сигнала с использованием двумерного разделимого сглаживающего фильтра. Такая пирамида представляется как 5-полосная система А-С, в которой каждая субполоса прореживается в два раза как по горизонтали, так и по вертикали.

Пирамида Лапласа имеет несколько недостатков для кодирования изображений. Главным из них является то, что ошибки квантования высокочастотных коэффициентов распространяются на другие субполосы и проявляются в реконструированном изображении в виде шума. Как и в случае с преобразованием Габора, причина заключается в неортогональности преобразования. Другим недостатком является увеличение числа коэффициентов по сравнению с числом отсчетов исходного сигнала в 3/2 раза. И, наконец, двумерные базисные функции пирамиды Лапласа являются изотропными. Следовательно, они не подходят для устранения избыточности ориентированных структур, имеющих в естественных изображениях.

Необходимо отметить, что данное преобразование с успехом может применяться для компенсации вектора движения при кодировании видео, где избыточность пирамиды делает ее робастной к ошибкам компенсации движения.

1.4. Квадратурно – зеркальные фильтры

В предыдущем разделе были описаны три примера преобразований, которые могут применяться для целей кодирования. Теперь рассмотрим преобразование, которое обладает всеми достоинствами предыдущих, но лишено их недостатков.

Итак, октавополосное преобразование может быть получено путем каскадирования по низкочастотной области двухканальной системы А-С. При этом возникает проблема элайзинга. Решить эту проблему помогает применение квадратурно-зеркальных фильтров (КЗФ), впервые предложенных для кодирования речи. КЗФ – это фильтры с конечной импульсной характеристикой. Свое название они получили из-за того, что их частотная характеристика симметрична относительно частоты, равной половине частоты дискретизации. Выход двухканальной системы А-С, построенной на КЗФ, свободен от элайзинга. Такая система осуществляет ортогональное субполосное коди-

рование сигнала. Во второй главе будет показана тесная связь КЗФ с теорией вейвлет-преобразования. Рассмотрим вкратце основные свойства и принципы построения КЗФ. При этом ограничимся одномерным случаем.

Сигнал на выходе двухканальной системы А-С может быть записан в виде

$$\begin{aligned} \widehat{X}(\omega) = & \frac{1}{2} [H_0(\omega)G_0(\omega) + H_1(\omega)G_1(\omega)]X(\omega) + \\ & + \frac{1}{2} [H_0(\omega + \pi)G_0(\omega) + H_1(\omega + \pi)G_1(\omega)]X(\omega + \pi). \end{aligned} \quad (1.20)$$

Первый член в (1.20) есть отклик линейной времянезависимой системы, второй – элайзинговая паразитная составляющая.

Для предотвращения элайзинга КЗФ можно выбрать следующим образом:

$$\begin{aligned} H_0(\omega) &= G_0(-\omega) = F(\omega), \\ H_1(\omega) &= G_1(-\omega) = e^{j\omega} F(-\omega + \pi), \end{aligned} \quad (1.21)$$

где $F(\omega)$ - произвольная функция. Отсюда видно, что фильтры анализа и синтеза удовлетворяют условию (1.18), и, следовательно, преобразование является ортогональным.

С учетом (1.21) равенство (1.20) запишется в виде

$$\begin{aligned} \widehat{X}(\omega) = & \frac{1}{2} [H(\omega)H(-\omega) + H(-\omega + \pi)H(\omega + \pi)]X(\omega) + \\ & + \frac{1}{2} [H(\omega + \pi)H(-\omega) + e^{j\pi} H(-\omega)H(\omega + \pi)]X(\omega + \pi). \end{aligned} \quad (1.22)$$

Второе, элайзинговое слагаемое равно нулю, и

$$\widehat{X}(\omega) = \frac{1}{2} [H(\omega)H(-\omega) + H(-\omega + \pi)H(\omega + \pi)]X(\omega). \quad (1.23)$$

Отметим полную ликвидацию элайзинга вне зависимости от выбора функции $F(\omega)$. Необходимо подчеркнуть, однако, что элайзинг ликвидирован лишь на выходе всей системы А-С, тогда как в отдельных субполосах он остался.

1.4.1. Построение КЗФ

Проблема конструирования КЗФ сводится к поиску НЧ фильтра, преобразование Фурье которого удовлетворяет ограничению:

$$\frac{1}{2}[H(\omega)H(-\omega) + H(-\omega + \pi)H(\omega + \pi)] = 1, \quad (1.24)$$

или

$$|H(\omega)|^2 + |H(\omega + \pi)|^2 = 2. \quad (1.25)$$

Существуют различные методы получения таких фильтров как во временной, так и в частотной областях. Большинство из них основано на выборе некоторой функции ошибок. Варьируя свободные параметры, добиваются минимизации этой функции. Подробнее вопрос построения КЗФ с полным восстановлением будет рассмотрен в разделе 3.2.

Для обработки изображения обычно применяются разделимые КЗФ. Для получения многомасштабной пирамиды преобразование рекурсивно применяется к низкочастотной части изображения. Это делит частотную область на октавные ориентированные субполосы.

Таким образом, пирамида КЗФ удовлетворяет требованиям, описанным в начале главы: она является многомасштабной, ориентированной, пространственно локализованной и ортогональной. Так что ошибки квантования коэффициентов ограничиваются одной субполосой. Недостатком преобразования является то, что базисные функции ориентированы только по вертикали и горизонтали.

1.4.2. Асимметричная система

До сих пор мы не рассматривали вычислительную сложность преобразований. Для некоторых приложений она не так важна в силу экспоненциального роста производительности средств обработки сигналов, для других, напротив, является первостепенным ограничением. Существуют приложения, в которых ограничение накладывается только на сложность кодера или декодера. Например, для проигрывания видеодисков нужен простой декодер, а алгоритм студийного кодирования может быть сложным. Разведывательный спутник должен иметь простой кодер, а на сложность аппаратуры декодирования ограничений не налагается.

Для таких приложений желательно было бы разработать асимметричную систему, в которой простота одного из устройств компенсировалась бы сложностью другого. Для преобразований с использованием КЗФ вычисли-

тельная сложность прямо пропорциональна длине используемых фильтров. Например, нам требуется простой декодер. Эффективная система для этого случая была разработана Ф.Леголом. Он предложил следующий набор простых фильтров для использования в системе А-С:

$$\begin{aligned} H_0(\omega) &= A(\omega), & G_0(\omega) &= B(\omega), \\ H_1(\omega) &= e^{j\omega} B(\omega + \pi), & G_1(\omega) &= e^{-j\omega} A(\omega + \pi), \end{aligned} \quad (1.26)$$

где импульсные характеристики фильтров, соответствующих $A(\omega)$ и $B(\omega)$:

$$\begin{aligned} a(n) &= [1, 2, 1], \\ b(n) &= [-1, 2, 6, 2, -1]. \end{aligned} \quad (1.27)$$

Видно, что многие умножения при выполнении свертки с этими фильтрами будут замещены сдвигами. Фильтры $a(n)$ и $b(n)$ можно поменять местами. Таким образом, они позволяют создавать вычислительно простые кодеры и декодеры при обеспечении полного восстановления (полностью подавляется элайзинг).

1.5. О преимуществе преобразования при помощи блоков фильтров перед преобразованием Фурье

Итак, преобразование Фурье и ему аналогичные применяются для декорреляции отсчетов сигнала. Блоки фильтров предоставляют альтернативный путь достижения этой цели и обладают некоторыми преимуществами.

Для понимания того, каким образом блоки фильтров декоррелируют сигнал, рассмотрим следующий пример. Пусть имеется гауссовский источник с памятью и спектральной плотностью мощности $\Phi_x(\omega)$. Необходимо произвести квантование данного источника. Функция скорость-искажение для него имеет вид

$$D(\theta) = \frac{1}{2\pi^2} \int_{\omega} \min(\theta, \Phi_x(\omega)) d\omega, \quad (1.28)$$

$$R(\theta) = \frac{1}{4\pi^2} \int_{\omega} \max\left(0, \log \frac{\Phi_x(\omega)}{\theta}\right) d\omega. \quad (1.29)$$

Каждое значение θ соответствует точке на кривой скорость-искажение. Цель любой схемы квантования состоит в приближении к этой кривой, кото-

рая является оптимальной. Поэтому из равенств (1.28) и (1.29) можно вывести простой метод: на частотах, на которых мощность сигнала меньше θ , нет смысла тратить биты на квантование. Таким образом, мощность сигнала на этих частотах будет равна мощности шума. Для кодирования сигнала на тех частотах, на которых его мощность больше θ , отводится ровно столько бит, чтобы мощность шума была точно равна θ . Тогда мощность сигнала всегда будет больше мощности шума. Данная процедура известна в мире под названием “inverse waterfilling”, что переводится примерно как “заполнение водой наоборот”. Суть метода показана на рис. 1.6.

Разумеется, непрактично рассматривать каждую частоту в отдельности: их бесконечно много. Необходимо принимать решение не по отдельным частотам, а по полосам частот. Это возможно в том случае, когда внутри них спектральная плотность мощности постоянна. Для разделения сигнала на полосы и применяются блоки фильтров.

Например, двухканальный блок фильтров делит спектр сигнала на две субполосы – высокочастотную и низкочастотную. Гауссовский процесс может быть декоррелирован путем разбиения его спектра на примерно плоские сегменты и умножения сигнала в каждом сегменте на некоторый коэффициент. Далее сигналы складываются. Результирующий сигнал будет иметь плоскую плотность распределения вероятности (то есть является белым шумом).

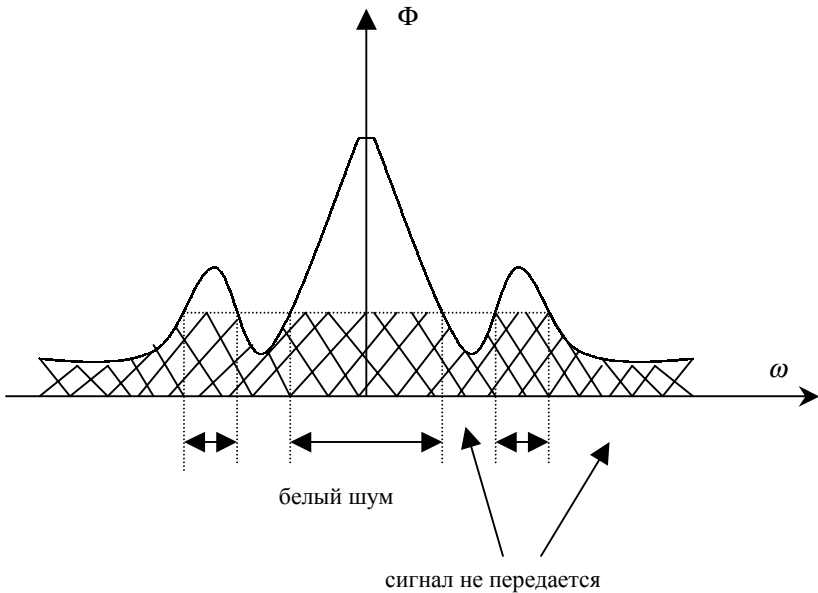


Рис. 1.6. Процедура «заполнения водой наоборот» для гауссовского источника без памяти

Таким образом, и преобразование Фурье, и блоки фильтров «работают» в частотной области. Почему же мы говорим о преимуществе блоков фильтров? Ответ заключается в пространственно-частотных свойствах этих методов. Базисы Фурье локализованы по частоте, но не в пространстве. Для кодирования сигнала, описываемого гауссовским процессом, это не является недостатком. Однако на изображениях присутствуют контуры, которые не могут быть описаны этой моделью и требуют локализованных в пространстве базисов. Поэтому блоки фильтров, являясь локальными и в пространстве, обеспечивают в среднем лучшую декорреляцию.

Возникает следующий вопрос: каким образом произвести разбиение спектра сигнала для заданного числа блоков фильтров? Известно, что корреляция между пикселями изображения убывает экспоненциально с увеличением расстояния. То есть

$$R_x(\delta) = e^{-\alpha_0|\delta|}, \quad (1.30)$$

где δ - переменная расстояния. Соответствующая спектральная плотность мощности

$$\Phi_x(\omega) = \frac{2\omega_0}{\omega_0^2 + (2\pi\omega)^2}. \quad (1.31)$$

Эта функция показана на рис.1.7. Из рисунка видно, что для получения плоских сегментов спектра необходимо точно делить спектр на низких частотах и грубо – на высоких. Субполосы, получаемые в результате выполнения такой процедуры, будут описываться белым шумом с дисперсией, пропорциональной спектру мощности в данном диапазоне. Как мы увидим в следующих главах, такое разделение спектра осуществляется посредством вейвлет-преобразования.

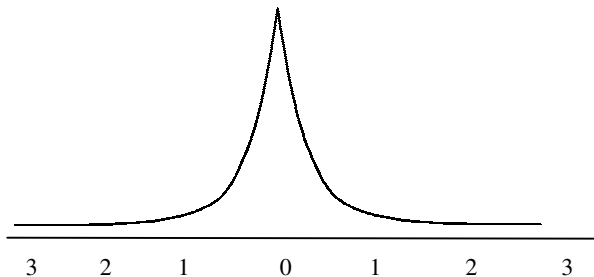


Рис. 1.7. Спектральная плотность мощности, соответствующая экспоненциальной корреляции

Итак, мы обсудили свойства различных линейных преобразований, которые могут использоваться для сжатия изображений. В частности отмечено, что базисные функции анализа и синтеза преобразования должны быть локализованы как в пространственной, так и в частотной областях. Кроме того, желательно, чтобы преобразование было ортогональным.

Несколько приведенных примеров иллюстрируют эти свойства. Базисные функции синтеза Габора хорошо локализованы, но неортогональность преобразования ведет к плохой локализации функций анализа. Блочное ДКП является преобразованием с равными размерами субполос с плохой частотной локализацией. Перекрывающееся ортогональное преобразование улучшает частотную локализацию ДКП. Пирамида Лапласа служит примером октавополосного преобразования. Она является неортогональным, неориентированным и избыточным разложением сигнала и плохо пригодна для кодирования неподвижных изображений. Однако для кодирования видео пирамида Лапласа может найти применение.

Субполосное преобразование, основанное на банках КЗФ, хорошо локализовано, ортогонально и может применяться рекурсивно для получения октавополосного разбиения. Это преобразование является, по сути, быстрым алгоритмом вычисления вейвлет-преобразования и тесно связано с теорией вейвлет-функций и концепцией кратномасштабного анализа, которые будут рассмотрены в следующей главе.

ОСНОВЫ ТЕОРИИ ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЯ

Данная глава является введением в теорию вейвлет-преобразования. В ней рассмотрены непрерывное и дискретное вейвлет-преобразования, ряды вейвлетов, быстрые алгоритмы вычислений. Также обсуждается свойство гладкости базисных вейвлет-функций. Математическое описание предполагает знакомство читателя с теорией преобразования Фурье.

2.1. Непрерывное вейвлет-преобразование

Важнейшим средством анализа стационарных непрерывных сигналов является преобразование Фурье непрерывного времени (СТФТ). При этом сигнал раскладывается в базис синусов и косинусов различных частот. Количество этих функций – бесконечно большое. Коэффициенты преобразования находятся путем вычисления скалярного произведения сигнала с комплексными экспонентами:

$$F(\omega) = \int_{-\infty}^{+\infty} f(x) e^{-i\omega x} dx, \quad (2.1)$$

где $f(x)$ означает сигнал, а $F(\omega)$ - его преобразование Фурье. С практической точки зрения СТФТ имеет ряд недостатков. Во-первых, для получения преобразования на одной частоте требуется вся временная информация. Это означает, что должно быть известно будущее поведение сигнала. Далее, на практике не все сигналы стационарны. Пик в сигнале во временной области распространится по всей частотной области его преобразования Фурье. Для преодоления этих недостатков СТФТ вводится кратковременное, или оконное преобразование Фурье (STFT):

$$\text{STFT}_f(\omega, b) = \int_{-\infty}^{+\infty} f(x) e^{-i\omega x} w(x-b) dx, \quad (2.2)$$

в котором применяется операция умножения сигнала на окно перед применением преобразования Фурье. Окном $w(x-b)$ является локальная функция, которая сдвигается вдоль временной оси для вычисления преобразования в нескольких позициях b . Преобразование становится зависимым от времени,

и в результате получается частотно-временное описание сигнала. В качестве окна часто выбирается функция Гаусса, и в этом случае обратное преобразование тоже будет выполняться с использованием оконной функции Гаусса. Используются также многочисленные другие окна, в зависимости от конкретного приложения.

Недостаток STFT состоит в том, что при его вычислении используется фиксированное окно, которое не может быть адаптировано к локальным свойствам сигнала.

Вейвлет-преобразование, рассматриваемое далее, решает эту и некоторые другие проблемы. Непрерывное вейвлет-преобразование (CTWT) есть скалярное произведение $f(x)$ и базисных функций

$$\psi_{a,b}(x) = a^{-1/2} \psi\left(\frac{x-b}{a}\right), \quad a \in R^+, b \in R, \quad (2.3)$$

так что

$$\text{CTWT}_f(a, b) = a^{-1/2} \int_{-\infty}^{+\infty} \psi\left(\frac{x-b}{a}\right) f(x) dx. \quad (2.4)$$

Базисные функции $\psi_{a,b} \in L^2(R)$ являются вещественными и колеблются вокруг оси абсцисс. Они определены на некотором интервале. Данные функции называются вейвлетами (в переводе – короткие волны) и могут рассматриваться как масштабированные и сдвинутые версии функции-прототипа $\psi(x)$. Параметр b показывает расположение во времени, а a – параметр масштаба. Большие значения a соответствуют низким частотам, малые – высоким. Операция умножения на окно как бы содержится в самой базисной функции, которая позволяет сужать и расширять это окно. Отсюда появляется возможность адаптивного к сигналу выбора параметров окна.

На рис. 2.1 показано разбиение частотно-временного плана для STFT и для CTWT. В соответствии с принципом неопределенности сужение окна анализа во временной области вызывает расширение его в частотной. Таким образом, площадь окна остается постоянной.

Для того чтобы было возможно обратное получение $f(x)$ из результата CTWT, функция $\psi(x)$ должна удовлетворять следующему условию:

$$C_\psi = \int_0^\infty \frac{|\Psi(\omega)|^2}{\omega} d\omega < \infty, \quad (2.5)$$

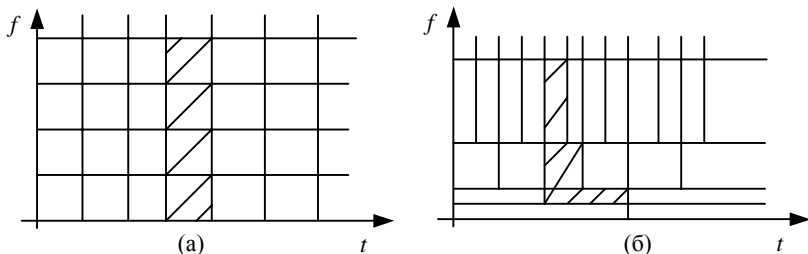


Рис. 2.1. Разбиение частотно-временного плана при STFT (а) и при STFT (б)

где через $\Psi(\omega)$ обозначено преобразование Фурье $\psi(x)$. Если $\psi(x)$ - локальная функция, то из (2.5) следует, что ее среднее значение равно нулю:

$$\int_{-\infty}^{\infty} \psi(x) dx = 0. \quad (2.6)$$

Тогда формула реконструкции имеет вид

$$f(x) = \frac{1}{C_{\psi}} \iint_{-\infty}^{\infty} \text{CTWT}_f(a, b) a^{-1/2} \psi\left(\frac{x-b}{a}\right) \frac{dadb}{a^2}. \quad (2.7)$$

Как видно из (2.7), $f(x)$ может быть выражена через сумму базисных функций $\psi_{a,b}(x)$ с весами $\text{CTWT}_f(a, b)$.

Параметры a и b меняются непрерывно, и поэтому множество базисных функций избыточно. Необходима дискретизация значений a и b при сохранении возможности восстановления сигнала из его преобразования. Можно показать, что дискретизация должна осуществляться следующим образом:

$$a = a_0^m; \quad b = nb_0 a_0^m, \quad m, n \in Z, a_0 > 1, b_0 \neq 0. \quad (2.8)$$

Возможен произвольный выбор параметра b_0 . Без потери общности выберем $b_0 = 1$. Из (2.8) видно, что параметр местоположения зависит от параметра масштаба. С увеличением масштаба увеличивается размер шага сдвига. Это интуитивно понятно, так как при анализе с большим масштабом детали не так важны.

Для дискретных значений a и b вейвлет-функции представляются в виде

$$\psi_{m,n}(x) = a_0^{-m/2} \psi(a_0^{-m} x - n). \quad (2.9)$$

Иногда дискретизированное преобразование называется вейвлет-преобразованием. Однако нам кажется более правильным ввести по аналогии с терминологией преобразований Фурье название рядов вейвлетов непрерывного времени (CTWS), так как мы имеем дело с дискретным представлением непрерывного сигнала. CTWS определяется путем дискретизации CTWT:

$$(\text{CTWS}_f)_{m,n} = d_{m,n} = \int_{-\infty}^{\infty} a_0^{-m/2} \psi(a_0^{-m} x - n) f(x) dx. \quad (2.10)$$

Восстановление $f(x)$ из последовательности возможно в том случае, если существуют числа $A > 0$ и $B < \infty$, такие что

$$A \|f(x)\|^2 \leq \sum_{m \in \mathbb{Z}} \sum_{n \in \mathbb{Z}} |d_{m,n}|^2 \leq B \|f(x)\|^2 \quad (2.11)$$

для всех $f(x)$ в $L^2(\mathbb{R})$. Это означает, что хотя реконструкция $f(x)$ из ее вейвлет-коэффициентов может не совпадать точно с $f(x)$, она будет близка к ней в среднеквадратическом смысле. Если $A = B = 1$ и $a_0 = 2$, то возможно полное восстановление, и семейство базисных функций $\psi_{a,b}(x)$ образует ортогональный базис. Тогда

$$f(x) = C_\psi \sum_{m \in \mathbb{Z}} \sum_{n \in \mathbb{Z}} d_{m,n} a_0^{-m/2} \psi(a_0^{-m} x - n). \quad (2.12)$$

Если базисные функции нормализованы, то $C_\psi = 1$.

Итак, мы дали определения вейвлет-преобразования и ряда вейвлетов для функций непрерывного времени по аналогии с соответствующими формулами для преобразования и ряда Фурье. Прежде чем перейти к рассмотрению дискретного вейвлет-преобразования, введем концепцию кратномасштабного анализа, которая является краеугольным камнем в теории вейвлет-преобразования.

2.2. Кратномасштабное представление функций

При анализе сигналов часто полезно представить сигнал в виде совокупности его последовательных приближений. Например, при передаче изображения можно сначала передать грубую его версию, а затем последовательно ее уточнять. Такая стратегия передачи имеет выгоды, например при осуществлении выбора изображений из некоторой базы данных, когда необходимо

быстро просмотреть большое количество картинок. Другим примером может являться телевизионный приемник, на экране которого одновременно отображены несколько программ. Разрешение и размеры выбранной программы должны затем кратномасштабно увеличиться.

Теория кратномасштабного анализа базируется на теории функциональных пространств.

Под кратномасштабным анализом понимается описание пространства $L^2(R)$ через иерархические вложенные подпространства V_m , которые не пересекаются и объединение которых дает нам в пределе $L^2(R)$, то есть

$$\begin{aligned} \dots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \subset \dots, \\ \bigcap_{m \in Z} V_m = \{0\}, \quad \overline{\bigcup_{m \in Z} V_m} = L^2(R). \end{aligned} \quad (2.13)$$

Далее, эти пространства имеют следующее свойство: для любой функции $f(x) \in V_m$ ее сжатая версия будет принадлежать пространству V_{m-1} ,

$$f(x) \in V_m \Leftrightarrow f(2x) \in V_{m-1}. \quad (2.14)$$

И, наконец, последнее свойство кратномасштабного анализа: существует такая функция $\phi(x) \in V_0$, что ее сдвиги $\phi_{0,n}(x) = \phi(x-n)$, $n \in Z$ образуют ортонормированный базис пространства V_0 . На рис. 2.2 схематично показаны данные вложенные пространства.

Так как функции $\phi_{0,n}(x)$ образуют ортонормированный базис пространства V_0 , то функции

$$\phi_{m,n}(x) = 2^{-m/2} \phi(2^{-m}x - n) \quad (2.15)$$

образуют ортонормированный базис пространства V_m . Эти базисные функции называются масштабирующими, так как они создают масштабированные версии функций в $L^2(R)$. Из кратномасштабного анализа, определенного выше, следует, что функция $f(x)$ в $L^2(R)$ может быть представлена множеством последовательных ее приближений $f_m(x)$ в V_m . Другими словами, функция $f(x)$ есть предел аппроксимаций $f_m(x) \in V_m$ при m стремящемся к минус бесконечности:

$$f(x) = \lim_{m \rightarrow -\infty} f_m(x), \quad (2.16)$$

Отсюда появляется возможность анализа функции или сигнала на различных уровнях разрешения, или масштаба. Переменная m называется масштабным коэффициентом, или уровнем анализа. Если значение m велико, то функция в V_m есть грубая аппроксимация $f(x)$, и детали отсутствуют. При малых значениях m имеет место точная аппроксимация. Из определения кратномасштабного анализа следует, что все функции в V_m могут быть представлены как линейная комбинация масштабирующих функций. В действительности, $f_m(x)$ есть ортогональная проекция $f(x)$ на V_m ,

$$f_m(x) = \sum_n \langle \phi_{m,n}(x), f(x) \rangle \phi_{m,n}(x) = \sum_n c_{m,n} \phi_{m,n}(x). \quad (2.17)$$

Так как $\phi(x) = \phi_{0,0}(x) \in V_0 \subset V_{-1}$, можно записать

$$\phi_{0,0}(x) = 2^{1/2} \sum_n h_n \phi_{-1,n}(x) = 2 \sum_n h_n \phi(2x - n), \quad (2.18)$$

где h_n - некоторая последовательность. Равенство (2.18) является одним из основных в теории вейвлет-анализа и имеет различные названия в литературе. Мы будем называть его далее масштабирующим уравнением.

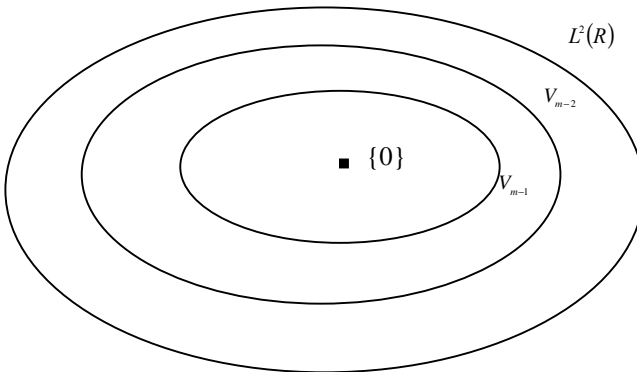


Рис. 2.2. Кратномасштабное представление $L^2(\mathbb{R})$

Функция $\phi(x)$ и последовательность h_n тесно связаны между собой. Выведем соответствующие отношения. Из (2.18) можно получить

$$\phi_{m+1,k}(x) = 2^{1/2} \sum_p h_p \phi_{m,p-2k}(x) = 2^{(-m+1)/2} \sum_p h_p \phi(2^{-m}x - (p-2k)). \quad (2.19)$$

Выполним операцию скалярного произведения $\phi_{m,n-2k}(x)$ с обеих сторон равенства (2.19):

$$\begin{aligned} \langle \phi_{m+1,k}(x), \phi_{m,n-2k}(x) \rangle &= \left\langle 2^{1/2} \sum_p h_p \phi_{m,p-2k}(x), \phi_{m,n-2k}(x) \right\rangle = \\ &= 2^{1/2} \sum_p h_p \langle \phi_{m,p-2k}(x), \phi_{m,n-2k}(x) \rangle = 2^{1/2} h_n. \end{aligned} \quad (2.20)$$

Отметим, что это равенство выполняется для любого m . Далее, если переписать (2.18) в частотной области, можно получить

$$\Phi(\omega) = H\left(\frac{\omega}{2}\right) \Phi\left(\frac{\omega}{2}\right). \quad (2.21)$$

При рекурсивном повторении формулы (2.21) получается выражение

$$\Phi(\omega) = \prod_{m=1}^{\infty} H\left(\frac{\omega}{2^m}\right). \quad (2.22)$$

Итак, последовательность h_n тесно связана с масштабирующей функцией. Кроме того, из концепции кратномасштабного анализа вытекают следующие свойства.

Во-первых, интегрируя (2.18) по всей числовой оси x , можно получить

$$\sum_n h_n = 1, \quad (2.23)$$

так как для построения кратномасштабного анализа среднее значение функции $\phi(x)$ не должно быть равно нулю. Во-вторых, в силу ортонормальности базисных функций

$$\delta_{0,k} = \langle \phi_{0,0}(x), \phi_{0,k}(x) \rangle = 2 \sum_n h_n h_{n+2k}. \quad (2.24)$$

Третье свойство последовательности h_n сформулируем в спектральной области. Из записи условия ортонормальности функций $\phi_{m,n}(x)$ в области спектра

$$\sum_k |\Phi(\omega + 2k\pi)|^2 = 1 \quad (2.25)$$

можно получить следующее выражение:

$$|H(\omega)|^2 + |H(\omega + \pi)|^2 = 1. \quad (2.26)$$

Равенство (2.23) эквивалентно тому, что $H(0)=1$. Тогда из (2.26) следует, что $H(\pi)=0$.

Эти свойства последовательности h_n будут использованы позднее. А пока оставим на время теорию и перейдем к простейшему примеру множества масштабирующих функций, образующих $L^2(R)$.

Рассмотрим множество сдвигов и растяжений единичной функции на единичном интервале:

$$\phi(x) = \begin{cases} 1, & 0 \leq x < 1, \\ 0, & \text{в остальных случаях.} \end{cases} \quad (2.27)$$

Так, базисные функции с коэффициентом масштаба -1 имеют вид

$$\phi_{-1,n}(x) = \begin{cases} \sqrt{2}, & n/2 \leq x < (n+1)/2, \\ 0, & \text{в других случаях.} \end{cases} \quad (2.28)$$

Базисная функция и соответствующая последовательность изображены на рис. 2.3.

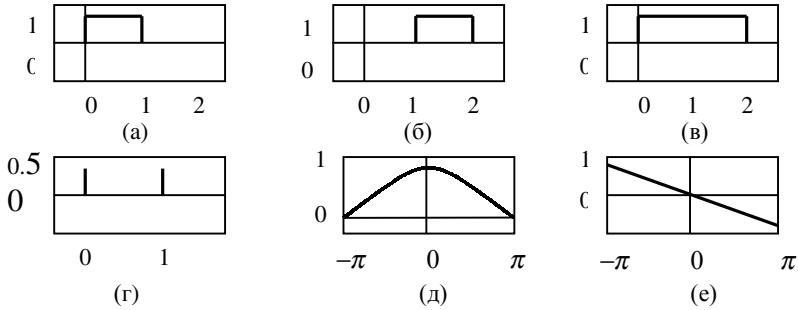


Рис. 2.3. Пример масштабирующей функции: (а) $\phi(x)$; (б) $\phi_{0,1}(x)$; (в) $\phi_{1,0}(x)$; (г) последовательность h_n ; (д) $|H(\omega)|$; (е) $\arg(H(\omega))$

2.2.1. Представление функций при помощи вейвлетов

При рассмотрении рис.2.2 видно, что область $L^2(R)$ построена из множества «колец», которые есть разность между двумя соседними пространствами. Эти разностные пространства обозначаются через W_m и определяются как ортогональные дополнения областей V_m до V_{m-1} :

$$V_{m-1} = V_m \oplus W_m, \quad \bigcap_{m \in Z} W_m = \{0\}, \quad \overline{\bigcup_{m \in Z} W_m} = L^2(R). \quad (2.29)$$

Пусть $\psi(x) = \psi_{0,0}(x)$ есть базисная функция W_0 . Так как $\psi_{0,0}(x) \in W_0 \subset V_{-1}$, можно записать

$$\psi_{0,0}(x) = 2^{1/2} \sum_n g_n \phi_{-1,n}(x) \quad (2.30)$$

для некоторой последовательности g_n . По аналогии с ранее рассмотренным множеством функций $\phi_{m,n}(x)$ определим семейство вейвлет-функций:

$$\psi_{m,n}(x) = 2^{-m/2} \psi(2^{-m}x - n). \quad (2.31)$$

Функции $\psi_{m,n}(x)$ идентичны полученным в разделе 2.1 после дискретизации (выражение (2.8)). Параметр a_0 в (2.9) в данном случае равен 2. Эти функции образуют ортонормированный базис $L^2(R)$.

Существуют строгие зависимости между $\psi(x), \phi(x), g_n, h_n$. Вначале получим формулу, аналогичную (2.22). Перепишем (2.30) для частотной области:

$$\Psi(\omega) = G\left(\frac{\omega}{2}\right)\Phi\left(\frac{\omega}{2}\right), \quad (2.32)$$

заменяем $\Phi(\omega)$ бесконечным произведением (2.22) и получим

$$\Psi(\omega) = G\left(\frac{\omega}{2}\right)\prod_{m=2}^{\infty} H\left(\frac{\omega}{2^m}\right). \quad (2.33)$$

Отметим, что $\Psi(\omega)$ пропорционально бесконечному произведению $H(2^{-m}\omega)$, а не $G(2^{-m}\omega)$, так же, как и в (2.30), вейвлет $\psi(x)$ был выражен в виде линейной комбинации масштабирующих функций.

Теперь получим выражения, связывающие последовательности g_n и h_n . Так как W_m есть ортогональное дополнение V_m , функции $\psi_{0,0}(x)$ и $\phi_{0,0}(x)$ должны быть ортогональны, и из (2.18) и (2.30) следует, что

$$0 = \langle \phi_{0,0}, \psi_{0,0} \rangle = 2 \sum_n \sum_p h_n g_p \langle \phi_{-1,n}, \psi_{-1,p} \rangle = 2 \sum_n h_n g_n. \quad (2.34)$$

Легко увидеть, что выбор

$$g_n = (-1)^n h_{-n+2t+1} \quad (2.35)$$

будет корректен для всех $t \in Z$. Эквивалент (2.35) в частотной области представляется в виде

$$G(\omega) = -H(-\omega + \pi)e^{-i\omega(2t+1)}. \quad (2.36)$$

С учетом этого из (2.32) получим

$$\Psi(\omega) = -e^{i\omega/2} H\left(-\frac{\omega}{2} + \pi\right)\Phi\left(\frac{\omega}{2}\right), \quad (2.37)$$

где без потери общности выбрано $t = 0$.

Наконец отметим, что и функция $\psi(x)$ и последовательность g_n имеют нулевое среднее. Этот факт легко проверить, подставляя $\omega = 0$ в (2.37) и (2.36) и используя свойство $H(\pi) = 0$:

$$\Psi(0) = \int_{-\infty}^{\infty} \psi(x) dx = 0 \quad (2.38)$$

и

$$G(0) = \sum_n g_n = 0. \quad (2.39)$$

Определение функций вейвлетов позволяет нам записать любую функцию $f(x) \in L^2(R)$ в виде суммы проекций на W_j , $j \in R$:

$$f(x) = \sum_{j=-\infty}^{\infty} e_j(x), \quad (2.40)$$

где

$$e_j(x) = \sum_k \langle \psi_{j,k}(x), f(x) \rangle \psi_{j,k}(x). \quad (2.41)$$

Если осуществлять анализ функции вплоть до некоторого масштаба m , то $f(x)$ будет представлена суммой ее грубой аппроксимации $f_m(x) \in V_m$ и множества деталей $e_j(x) \in W_j$:

$$\begin{aligned} f(x) &= f_m(x) + \sum_{j=-\infty}^m e_j(x) = \\ &= \sum_n \langle \phi_{m,n}(x), f(x) \rangle \phi_{m,n}(x) + \sum_{j=-\infty}^m \sum_k \langle \psi_{j,k}(x), f(x) \rangle \psi_{j,k}(x) = \\ &= \sum_n c_{m,n} \phi_{m,n}(x) + \sum_{j=-\infty}^m d_{j,k} \psi_{j,k}(x). \end{aligned} \quad (2.42)$$

В качестве примера семейства вейвлет-функций, образующих ортонормальный базис пространства $L^2(R)$, на рис.2.4 показан вейвлет, соответствующий масштабирующей функции рис.2.3. Это семейство вейвлетов называется вейвлетами Хаара.

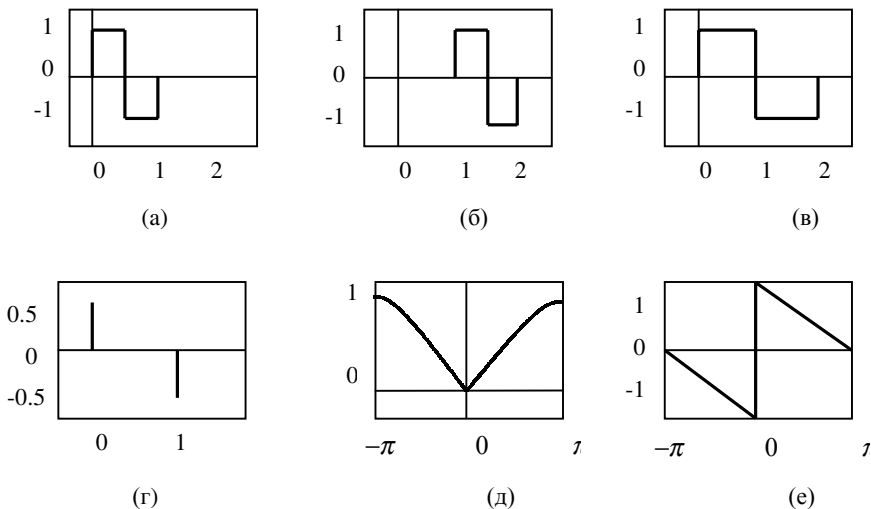


Рис. 2.4. Пример вейвлет-функции: (а) $\psi(x)$; (б) $\psi_{0,1}(x)$; (в) $\psi_{1,0}(x)$; (г) последовательность g_n ; (д) $|G(\omega)|$; (е) $\arg(G(\omega))$

Из теории известно, что в случае ортогональных вейвлетов последовательности h_n и g_n не могут быть симметричными, если длина каждой из них превышает 2. Однако во многих приложениях свойство симметричности является важным. В этом случае отказываются от требования ортогональности и на вейвлет-функции налагают менее строгое требование биортогональности. Выражения для биортогонального кратномасштабного анализа аналогичны выписанным выше и здесь не приводятся.

2.3. Вейвлет-ряды дискретного времени

В большинстве приложений мы имеем дело с дискретными сигналами. Поэтому с точки зрения практики представляют интерес дискретные аналоги СТWT и СТWS, которые преобразуют дискретный сигнал в непрерывный и дискретный сигналы, соответственно. К сожалению, формулы для вейвлет-преобразования и рядов вейвлетов дискретного времени (DTWT и DTWS) нельзя получить простой дискретизацией соответствующих формул для непрерывного времени. Также невозможно определить кратномасштабный анализ для дискретных сигналов, так как не существует базисных функций, масштабированные и смещенные версии которых давали бы нам базис про-

пространства $L^2(\mathbb{R})$, пространства квадратично суммируемых последовательностей бесконечной длины.

Попробуем вывести формулы для DTWS из формул кратномасштабного анализа раздела 2.2. В приложении 1 обобщены все формулы для вейвлет-преобразований и рядов. Там же даны для сравнения аналогичные формулы преобразования и рядов Фурье.

Пусть имеется некоторая непрерывная функция $f_0(x) \in V_0$. Наш дискретный сигнал c_n представим как последовательность коэффициентов при масштабирующих функциях, по которым раскладывается $f_0(x)$:

$$f_0(x) = \sum_n c_{0,n} \phi_{0,n}(x), \quad (2.43)$$

где $c_{0,n} = c_n$. Другими словами, мы интерпретируем наш сигнал как последовательность коэффициентов разложения, полученную в ходе кратномасштабного анализа функции $f_0(x)$. Тогда мы можем вычислить аппроксимации этой функции, принадлежащие пространствам V_1, V_2, \dots . Пространства V_{-1}, V_{-2}, \dots не имеют значения при данной интерпретации.

Согласно концепции кратномасштабного анализа функция $f_0(x)$ декомпозируется на две функции $f_1(x) \in V_1$ и $e_1(x) \in W_1$:

$$f_0(x) = f_1(x) + e_1(x) = \sum_k c_{1,k} \phi_{1,k}(x) + \sum_k d_{1,k} \psi_{1,k}(x). \quad (2.44)$$

Таким образом, получили две новые последовательности $c_{1,n}$ и $d_{1,n}$. Этот процесс может быть продолжен по $f_1(x)$, и функция $f_0(x)$ (а также и последовательность c_n) будет представлена совокупностью коэффициентов $d_{m,n}$, $m \in \mathbb{Z}^+$, $n \in \mathbb{Z}$.

Итак, концепция DTWS определена. Однако вычисления пока зависят от непрерывных функций $\phi(x)$ и $\psi(x)$. Поэтому покажем, как вычисления DTWS могут быть выполнены с использованием операций только над дискретными сигналами.

С учетом того, что масштабирующая функция образует базис соответствующего пространства, из (2.20) можно получить

$$\begin{aligned}
c_{1,k} &= \langle \phi_{1,k}(x), f_1(x) \rangle = \langle \phi_{1,k}, f_0(x) - e_1(x) \rangle = \left\langle \phi_{1,k}(x), \sum_n c_{0,n} \phi_{0,n}(x) \right\rangle = \\
&= \sum_{n \in \mathbb{Z}} c_{0,n} \langle \phi_{1,k}(x), \phi_{0,n}(x) \rangle = 2^{1/2} \sum_{n \in \mathbb{Z}} c_{0,n} h_{n+2k}.
\end{aligned} \tag{2.45}$$

Так что оказывается возможным итеративное вычисление коэффициентов $c_{j,k}$ и $d_{j,k}$ без непосредственного использования функций $\phi(x)$ и $\psi(x)$. По аналогии с (2.44) можно записать для произвольного j

$$c_{j,k} = 2^{1/2} \sum_n c_{j-1,n} h_{n+2k}, \tag{2.46}$$

$$d_{j,k} = 2^{1/2} \sum_n c_{j-1,n} g_{n+2k}, \tag{2.47}$$

получив, таким образом, полностью дискретный процесс декомпозиции. Последовательности h_n и g_n называются фильтрами. Отметим, что $c_{j,k}$ и $d_{j,k}$ имеют «половинную» длину по сравнению с $c_{j-1,k}$ (хотя, конечно, на данном этапе все последовательности бесконечны). Таким образом, не вводится избыточности.

Обратный процесс заключается в получении c_{j-1} из c_j и d_j :

$$\begin{aligned}
c_{j-1,n} &= \langle \phi_{j-1,n}, f_{j-1}(x) \rangle = \langle \phi_{j-1,n}(x), f_j(x) + e_j(x) \rangle = \\
&= \left\langle \phi_{j-1,n}, \sum_k c_{j,k} \phi_{j,k}(x) \right\rangle + \left\langle \phi_{j-1,n}, \sum_k d_{j,k} \psi_{j,k}(x) \right\rangle = \\
&= \sum_k c_{j,k} \langle \phi_{j-1,n}, \phi_{j,k}(x) \rangle + \sum_k d_{j,k} \langle \phi_{j-1,n}, \psi_{j,k}(x) \rangle = \\
&= 2^{1/2} \sum_k c_{j,k} h_{n+2k} + 2^{1/2} \sum_k d_{j,k} g_{n+2k}.
\end{aligned} \tag{2.48}$$

Отметим, что в данном случае суммирование производится по другим переменным по сравнению с формулами (2.45) и (2.46). Длина последовательности c_{j-1} вдвое больше длины последовательности c_j или d_j .

Подставляя (2.45) и (2.46) в (2.47), получаем следующие ограничения на фильтры h_n и g_n :

$$2 \sum_k (h_{n+2k} h_{p+2k} + g_{n+2k} g_{p+2k}) = \delta_{n,p}, \quad (2.49)$$

$$2 \sum_n h_{n+2k} h_{n+2p} = 2 \sum_n g_{n+2k} g_{n+2p} = \delta_{k,p}, \quad (2.50)$$

$$2 \sum_n h_{n+2k} h_{n+2p} = 0. \quad (2.51)$$

Выражение (2.48) для временной области эквивалентно выражениям (2.26) и (2.36) для частотной. Равенства (2.49) и (2.50) уже появлялись ранее, но в менее общей форме ((2.24) и (2.34), соответственно).

2.4. Дискретное вейвлет-преобразование

На практике DTWS должно применяться к сигналам конечной длины. Таким образом, его необходимо модифицировать, чтобы из сигнала какой-то длины получать последовательность коэффициентов той же длины. Получившееся преобразование называется дискретное вейвлет-преобразование (DWT).

Вначале опишем DWT в матричном виде, а затем – на основе банков фильтров, что наиболее часто используется при обработке сигналов.

В обоих случаях мы предполагаем, что базисные функции $\phi(x)$ и $\psi(x)$ компактно определены. Это автоматически гарантирует финитность последовательностей h_n и g_n . Далее предположим, что сигнал, подвергаемый преобразованию, имеет длину $N = 2^d$, $d \in Z^+$.

2.4.1. Матричное описание DWT

Обозначим через вектор v^j последовательность конечной длины $c_{j,n}$ для некоторого j . Этот вектор преобразуется в вектор v^{j+1} , содержащий последовательности $c_{j+1,n}$ и $d_{j+1,n}$, каждая из которых половинной длины. Преобразование может быть записано в виде матричного умножения $v^{j+1} = \mathbf{M}_j v^j$, где матрица \mathbf{M}_j - квадратная и состоит из нулей и элементов h_n , умноженных на $\sqrt{2}$. В силу свойств h_n , полученных в разделе 2.3, матрица \mathbf{M}_j является ортонормированной, и обратная ей матрица равна транспонированной. В качестве иллюстрации рассмотрим следующий пример. Возьмем фильтр длиной $L = 4$, последовательность длиной $N = 8$, а в качестве начального

риодического продолжения $c_{0,n}$. Так что DWT, будучи определенным таким образом, использует периодичность сигнала, как и в случае с DFT.

Матричное описание DWT кратко и ясно. Однако при обработке сигналов DWT чаще всего описывается посредством блок-диаграммы, аналогичной диаграмме системы анализа-синтеза (см. рис.1.1).

2.4.2. Описание DWT посредством блоков фильтров

Рассматривая в главе 1 субполосные преобразования, мы интерпретировали равенства, аналогичные (2.45) и (2.46), как фильтрацию с последующим прореживанием в два раза. Так как в данном случае имеется два фильтра h_n и g_n , то банк фильтров – двухполосный и может быть изображен, как показано на рис.2.5.

Фильтры F и E означают фильтрацию фильтрами h_{-n} и g_{-n} , соответственно. В нижней ветви схемы выполняется низкочастотная фильтрация. В результате получается некоторая аппроксимация сигнала, лишенная деталей низкочастотная (НЧ) субполоса. В верхней части схемы выделяется высокочастотная (ВЧ) субполоса. Отметим, что при обработке сигналов константа $2^{1/2}$ всегда выносится из банка фильтров и сигнал домножается на 2 (см. рис.3.2, глава 3).

Итак, схема рис.2.5 делит сигнал уровня $j = 0$ на два сигнала уровня $j = 1$. Далее, вейвлет-преобразование получается путем рекурсивного применения данной схемы к НЧ части. При осуществлении вейвлет-преобразования изображения каждая итерация алгоритма выполняется вначале к строкам, затем – к столбцам изображения (строится так называемая пирамида Маллата). В видеокодеках ADV6xx применена модифицированная пирамида Маллата, когда на каждой итерации не обязательно выполняется преобразование и по строкам, и по столбцам. Это сделано для более полного учета зрительного восприятия человека.

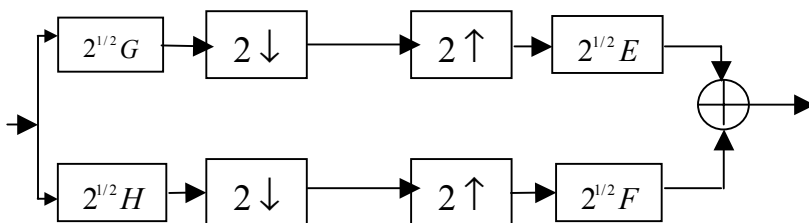


Рис. 2.5. Схема двухполосного банка фильтров

Получившееся преобразование аналогично (2.51). Однако существуют некоторые различия. При фильтрации сигнала конечной длины мы сталкиваемся с проблемой его продолжения на границе. Матричное выполнение DWT эквивалентно периодическому продолжению сигнала на границе. Этот тип продолжения является обязательным для ортогональных фильтров. В случае применения биортогональных фильтров появляются некоторые другие возможности в силу симметричности их характеристик. Подробнее этот вопрос будет рассматриваться в главе 3.

Схему, выполняющую DWT, можно представить еще и как показано на рис.2.6. Здесь рекурсивная фильтрация и прореживание заменены одной операцией фильтрации и одной операцией прореживания на каждую субполосу. Определение итерационных фильтров h_n^j и g_n^j легче всего дать в частотной области:

$$\begin{aligned}
 H^j(\omega) &= H(\omega) \times H(2\omega) \times H(4\omega) \times \dots \times H(2^{j-1}\omega) = \prod_{m=1}^j H(2^{m-1}\omega), \\
 G^j(\omega) &= G(\omega) \times H(2\omega) \times H(4\omega) \times \dots \times H(2^{j-1}\omega) = G(\omega) \times \prod_{m=2}^j H(2^{m-1}\omega).
 \end{aligned}
 \tag{2.54}$$

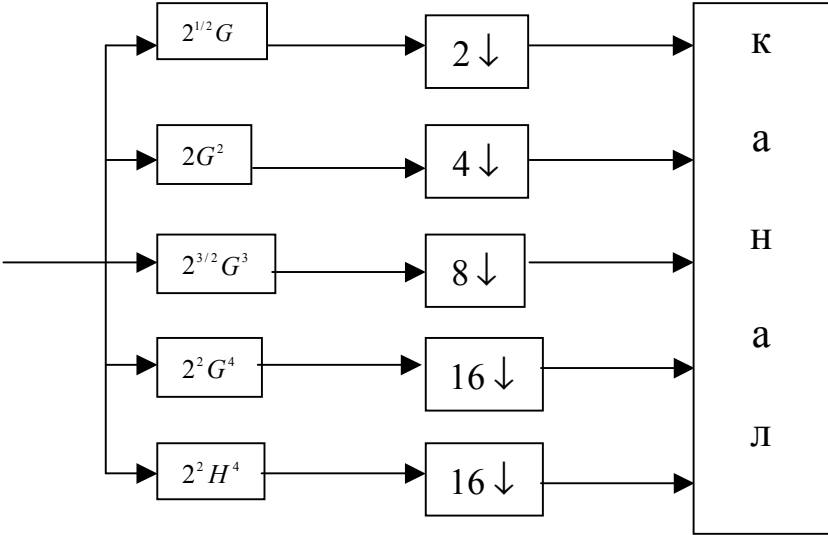


Рис. 2.6. Эквивалентная схема вейвлет-преобразования

В пределе итерационный фильтр $H^j(2^{-j}\omega)$ сходится к $\Phi(\omega)$ в соответствии с (2.22):

$$\lim_{j \rightarrow \infty} H^j(2^{-j}\omega) = \lim_{j \rightarrow \infty} \prod_{m=1}^j H(2^{m-1-j}\omega) = \lim_{j \rightarrow \infty} \prod_{p=1}^j H(2^{-p}\omega) = \Phi(\omega) \quad (2.55)$$

Во временной области это означает, что график последовательности $2^j h_n^j$, построенной против 2^{-jn} , сходится к $\phi(x)$ при j , стремящемся к бесконечности. На рис.2.7 это изображено для фильтра Добеши длиной 4.

Определение DWT может быть дано по аналогии с DFT. Предположим, что сигнал, подвергаемый преобразованию, c_n имеет длину $N = 2^d$. Периодически продолжим его. Получим периодический сигнал \tilde{c}_n с периодом N . Тогда

$$\begin{aligned} (\text{DWT}_c)_{j,k} &= 2^{j/2} \sum_{n=-\infty}^{\infty} g_{n-2^j k}^j \tilde{c}_n, \quad \text{при } \begin{cases} 1 \leq j < d, \\ 0 \leq k < N2^{-j}, \end{cases} \\ (\text{DWT}_c)_{j,k} &= 2^{j/2} \sum_{n=-\infty}^{\infty} h_{n-2^j k}^j \tilde{c}_n, \quad \text{при } \begin{cases} j = d, \\ k = 0. \end{cases} \end{aligned} \quad (2.56)$$

Заметим, что в действительности суммы конечные, так как итерируемые фильтры имеют конечную длину. Ряд DTWS может быть записан аналогично выражению (2.56):

$$(\text{DTWS}_c)_{j,k} = 2^{j/2} \sum_{n=-\infty}^{\infty} g_{n-2^j k}^j c_n, \quad j \geq 0, k \in Z. \quad (2.57)$$

Отметим, что (2.57) не есть дискретизированная версия непрерывного ряда СТWS (2.10), так как вместо функции $\psi(x)$ здесь мы имеем последовательность $2^{j/2} g_n^j$. Однако с учетом (2.33) и (2.54) дискретная формула сходится в пределе к непрерывной.

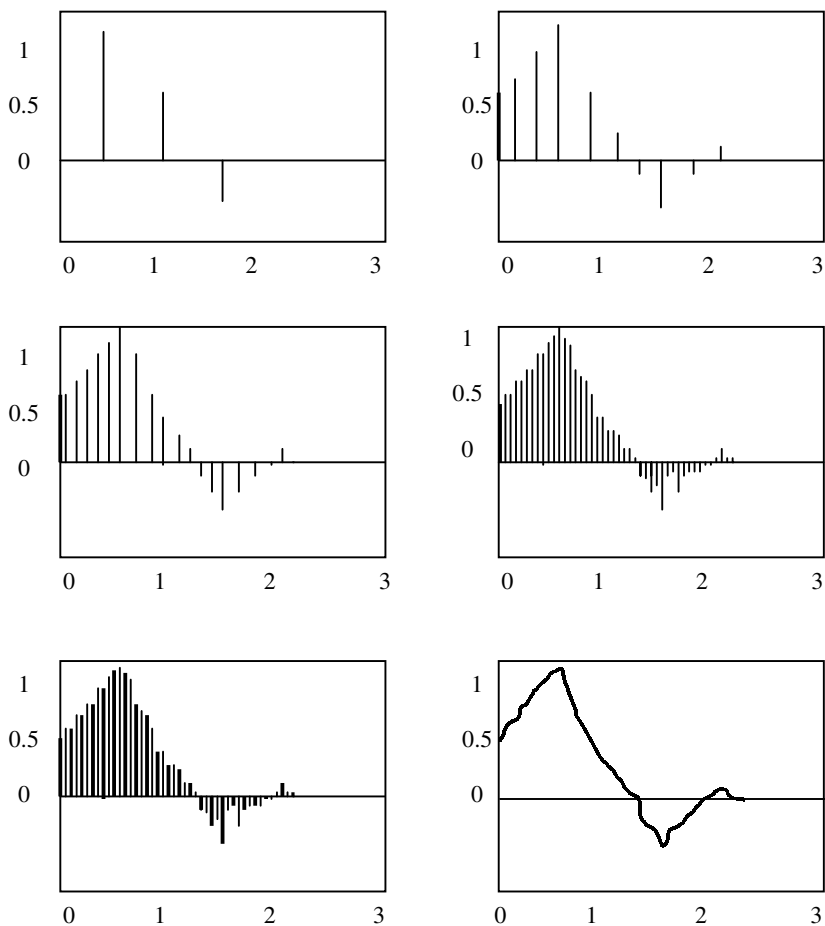


Рис. 2.7. Демонстрация сходимости итерационных масштабирующих фильтров к непрерывной функции для фильтра Добеши длиной 4

2.5. Гладкость базисных функций

Обсудим свойство гладкости базисных функций $\phi(x)$ и $\psi(x)$. Вначале покажем, почему гладкость должна учитываться в приложениях DWT. Далее продемонстрируем, как может быть определена гладкость базисных функций и соответствующих фильтров.

Функция, показанная на рис.2.7, непрерывная, но не гладкая. Применение такой функции (и связанного с ней фильтра) для анализа сигнала приведет к появлению в коэффициентах неоднородностей, отсутствующих в исходном сигнале. Поэтому для лучшего представления сигнала желательно было бы иметь гладкие базисные функции. Существуют и другие аргументы в пользу необходимости гладкости. Так, ошибка квантования вейвлет-коэффициента на некотором уровне декомпозиции дает ошибку в реконструированном сигнале, пропорциональную базисным функциям этого уровня. При кодировании изображения однородная ошибка менее заметна глазу, чем неоднородная, даже если численно они одинаковы.

Так что принцип обеспечения гладкости базисных функций может быть хорошим критерием при проектировании фильтров для кодирования изображений. Следует сказать, однако, что некоторые специалисты не считают гладкость важным критерием, влияющим на эффективность кодирования.

Разработаны различные методы для определения гладкости базисных функций. Рассмотрим один из них. Данный метод заключается в преобразовании всех свойств гладкости $\phi(x)$ в эквивалентные свойства итерируемых фильтров h_n^j .

Гладкость масштабирующей функции $\phi(x)$ равна регулярности вейвлет-функции $\psi(x)$, так как обе они могут быть записаны как бесконечное число сверток с одним и тем же фильтром (2.22), (2.33). Так что далее будет рассматриваться только $\phi(x)$.

Говорят, что функция обладает гладкостью порядка N , если она N раз дифференцируема и N -я производная непрерывна. Таким образом, порядок гладкости – это некоторое целое число.

Чтобы функция $\phi(x)$ обладала гладкостью N -го порядка, необходимо как минимум N нулей на частоте $\omega = \pi$ для соответствующего фильтра $H(\omega)$. Наличие у фильтра большего количества нулей на данной частоте вовсе не означает больший порядок гладкости $\phi(x)$, так как некоторые нули могут ухудшать гладкость.

Пусть $2^{-(N+1)}F(\omega)$ - множитель, остающийся от $H(\omega)$ после удаления N нулей на частоте $\omega = \pi$. Тогда можно записать

$$H(\omega) = 2^{-(N+1)}(1 + e^{-i\omega})F(\omega). \quad (2.58)$$

Как было отмечено, гладкость ухудшается в силу наличия нулей у $F(\omega)$. Если нули отсутствуют, то порядок гладкости – N . В противном случае нижняя граница для гладкости определяется как $R = N - \beta$, где ухудшение гладкости

$$\beta = \lim_{j \rightarrow \infty} \beta_j = \lim_{j \rightarrow \infty} \frac{1}{j} \log_2 \left(\max_{0 \leq n \leq 2^j - 1} \left[\sum_k |f_{n-2^j k}^j| \right] \right). \quad (2.59)$$

Как и в (2.53), f_n^j есть итерируемый фильтр, относящийся к $F^j(\omega)$. В общем, значение β_j близко к β уже после 20 итераций ($j = 20$).

Так как вычислительная сложность и требования к памяти данного метода возрастают экспоненциально, часто пользуются упрощенной формулой, дающей нижнюю оценку для β и, следовательно, верхнюю границу для гладкости:

$$\beta = \lim_{j \rightarrow \infty} \beta_j = \lim_{j \rightarrow \infty} \frac{1}{j} \log_2 \left(\max \left[\sum_k |f_{2^j k}^j|, \sum_k |f_{2^{j-1}-2^j k}^j| \right] \right). \quad (2.60)$$

Можно показать, что гладкость базисных функций наиболее важна для представления полиномиальных сигналов. Если порядок гладкости функции N , она может эффективно использоваться в качестве базисной для представления полинома степени меньше N . В этом случае все вейвлет-коэффициенты будут равны нулю. Так как изображения обычно относят к кусочно-полиномиальным сигналам степени 2, для их кодирования требуются вейвлет-функции со степенью гладкости не меньше 2.

Итак, в данной главе было осуществлено введение в теорию вейвлетов с точки зрения обработки сигналов. Дано определение СТWT как логического продолжения СТFT и STFT. После введения концепции кратномасштабного анализа мы сконцентрировались на обработке дискретных сигналов. Рассматривалось описание DWT в матричной форме и через банки фильтров. Также была обсуждена регулярность фильтров. В следующей главе будет проведено сравнение вейвлет-фильтров и фильтров, применяемых в субполосном кодировании.

Глава 3

ВЕЙВЛЕТ – ДЕКОМПОЗИЦИЯ СИГНАЛОВ ПРОИЗВОЛЬНОЙ ДЛИНЫ

Итак, вейвлет-преобразование выполняется при помощи древовидно соединенных двухканальных блоков фильтров. Пусть глубина дерева d . Тогда обычно подразумевается, что сигнал имеет длину 2^d . Если это не выполняется, можно добавить недостающие отсчеты, например дописать сигнал нулями. Другим подходом является осуществление периодического или симметричного продолжения сигнала. Однако это приводит к увеличению объема данных, что нежелательно при решении задач сжатия сигнала. В данной главе рассматривается эффективный метод «удлинения» сигнала на этапе фильтрации. При помощи данного метода может осуществляться вейвлет-декомпозиция сигналов произвольной длины без увеличения объема данных.

Необходимо отметить, что существует еще одна возможность решения проблемы границ сигнала: конструирование граничных фильтров. Этот подход в книге не рассматривается. Интересующимся рекомендуем обратиться к статьям Е.Ковачевич (см. Интернет-ссылки).

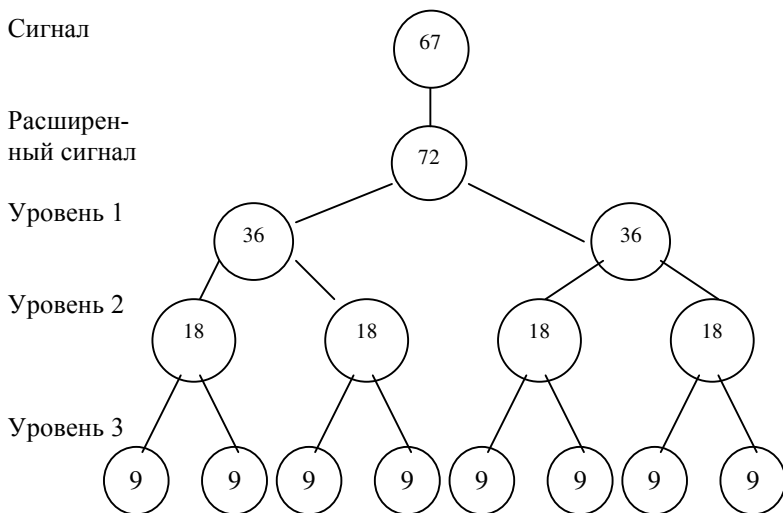
3.1. Условия полного восстановления сигнала

Вейвлет-преобразование (DWT) и субполосное кодирование - два популярных и очень похожих метода. В большинстве случаев используется двухканальная схема, в которой исходный сигнал делится на две субполосы, каждая вдвое меньше размером, чем исходная. В результате рекурсивного повторения этого процесса для обеих субполос получаем древовидное разбиение спектра на определенное количество уровней. Полное восстановление сигнала из субполос возможно лишь в отсутствие квантования коэффициентов. Будем предполагать это на протяжении всех последующих рассуждений. Однако и при квантовании коэффициентов можно построить схему, осуществляющую почти полное восстановление. Полное восстановление зависит от выполнения двух условий:

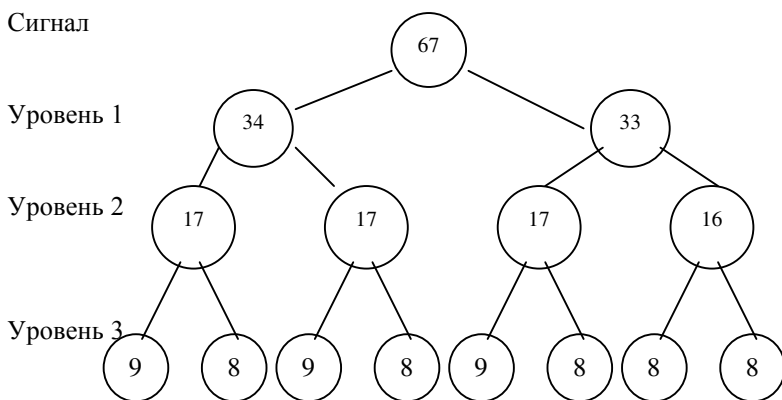
соответствующего расчета фильтров анализа и синтеза;

соответствующего продолжения сигнала конечной длины после границы.

В большинстве публикаций последний аспект не рассматривается, хотя от него так же зависит эффективность кодирования, как будет показано в разделе 3.4 при обсуждении эффективного метода продолжения сигнала.



(а)



(б)

Рис. 3.1. Трехуровневая декомпозиция сигнала длиной 67 отсчетов:
(а) обычный способ; (б) эффективный способ

Данный метод основан на следующем факте. На каждом уровне древо-видного DWT сигнал должен быть четной длины. Если она нечетная, то после прореживания мы либо потеряем какую-то информацию, либо добавим один лишний отсчет. Поэтому для дерева глубиной d необходимо иметь

сигнал длиной 2^d . В противном случае он удлинится некоторым образом до этого размера. На рис. 3.1(а) показано, как сигнал длиной 67 должен быть увеличен до длины 72 для построения дерева глубиной 3. Ясно, что декомпозиция, представленная на рис. 3.1(б) лучше подходит для целей кодирования, так как не увеличивается количество отсчетов. Как будет показано в разделе 3.4, такая декомпозиция возможна с сохранением свойства полного восстановления.

Иными словами, нами будет показано, каким образом может быть осуществлено вейвлет-преобразование сигнала нечетной длины, в результате которого не происходит увеличения количества отсчетов. В результате сигнал произвольной длины N может быть декомпозирован, в принципе, на N субполос длиной 1 без потери свойства полного восстановления.

3.2. Методика расчета фильтров, позволяющих осуществить полное восстановление сигнала

Двухканальная схема анализа-синтеза представлена на рис.3.2. Все фильтры данной схемы являются казуальными. Сдвиги фильтров осуществляются путем умножения на комплексную экспоненту. Входной сигнал S делится на две части путем фильтрации НЧ фильтром H , который производит смещение сигнала на p отсчетов, и ВЧ фильтром G , который сдвигает сигнал на q отсчетов. После прореживания осуществляется кодирование сигналов, передача их по каналу (на схеме эти операции не показаны). Секция синтеза выполняет интерполяцию сигналов и фильтрацию смещенными фильтрами F и E . После умножения на 2 для сохранения величины амплитуды, оба сигнала складываются. В результате чего получается исходный сигнал, если фильтры обладают свойством полного восстановления.

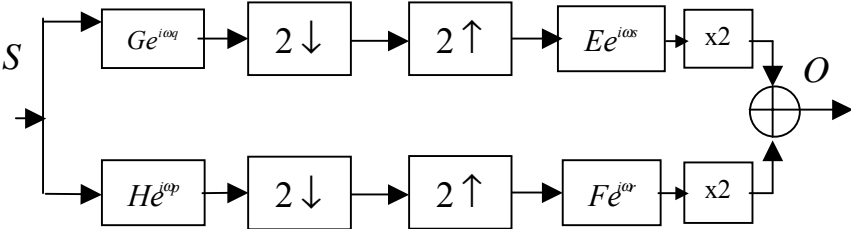


Рис. 3.2 . Схема двухполосной схемы анализа-синтеза: фильтры H и F низкочастотные, G и E - высокочастотные

Система анализа-синтеза описывается в области Фурье следующим образом:

$$O(\omega) = [F(\omega)H(\omega)e^{i\omega(p+r)} + E(\omega)G(\omega)e^{i\omega(q+s)}]S(\omega) + [F(\omega)H(\omega + \pi)e^{i\omega(p+r)}e^{i\pi} + E(\omega)G(\omega + \pi)e^{i\omega(q+s)}e^{i\pi}]S(\omega + \pi), \quad (3.1)$$

где O – выходной сигнал системы. Вторая часть (3.1) представляет собой искажение из-за наложения спектров и должна быть устранена для полного восстановления сигнала. Это может быть достигнуто в случае $p + r = q + s$, и тогда

$$F(\omega) = G(\omega + \pi), \quad E(\omega) = -H(\omega + \pi), \quad |p - q| \bmod 2 = 0 \quad (3.2)$$

или

$$F(\omega) = G(\omega + \pi), \quad E(\omega) = H(\omega + \pi), \quad |p - q| \bmod 2 = 1. \quad (3.3)$$

Далее будем использовать (3.2). В этом случае функция передачи

$$T(\omega) = [H(\omega)G(\omega + \pi) - G(\omega)H(\omega + \pi)]e^{i\omega(p+r)}. \quad (3.4)$$

Для достижения полного восстановления фильтры должны рассчитываться так, чтобы $H(\omega)G(\omega + \pi) - G(\omega)H(\omega + \pi)$ равнялось чистой задержке. Далее, необходимо выбрать p, q, r и s так, чтобы общая задержка системы равнялась нулю. Тогда из (3.4) имеем

$$p = (L_H - 2)/2, \quad q = (L_G - 2)/2 \quad (3.5)$$

для четных фильтров и

$$p = (L_H - 1)/2, \quad q = (L_G - 3)/2 \quad (3.6)$$

для нечетных, где L_H и L_G означают длины фильтров H и G . Для симметричных фильтров задержка системы будет равна нулю, если положить $r = q + 1$ и $s = p + 1$.

Для того чтобы $H(\omega)G(\omega + \pi) - G(\omega)H(\omega + \pi)$ равнялось чистой задержке, мы можем взять, например, фильтр G , коэффициенты которого равны

коэффициентам H , но записаны в обратном порядке и через один умножены на -1 . В частотной области это означает

$$G(\omega) = H(-\omega + \pi) e^{i(-\omega + \pi)(L_n - 1)}. \quad (3.7)$$

Отметим казуальность фильтра G . Отсюда вытекает

$$T(\omega) = [H(\omega)H(-\omega) - H(\omega + \pi)H(-\omega + \pi) e^{i\pi(L_n - 1)}] e^{i\omega(p+r+1-L_n)}. \quad (3.8)$$

Чистая задержка достигается, если

$$H(\omega)H(-\omega) + (-1)^{L_n} H(\omega + \pi)H(-\omega + \pi) = 1, \quad (3.9)$$

в результате чего получается система с полным восстановлением. Отметим, что равенства (3.7) и (3.9) являются аналогами (2.36) и (2.26), соответственно. Другой возможной системой с полным восстановлением является биортогональная система, которая будет описана позднее.

3.3. Продолжения сигналов, сохраняющие свойство полного восстановления

В этом разделе будут рассмотрены традиционные методы продолжения сигналов, используемые при фильтрации сигналов конечной длины. Проблема фильтрации таких сигналов заключается в пересечении фильтром границы сигнала. Существуют четыре основных метода решения данной проблемы:

- добавление нулей;
- повторение граничного значения;
- периодизация сигнала (круговая свертка);
- симметричное отражение сигнала относительно границы.

Можно ввести следующий критерий сохранения полного восстановления при продолжении сигнала.

Предположим, сигнал длиной N продолжен до бесконечности с обеих сторон. Далее он фильтруется, прореживается. Если все коэффициенты обеих бесконечных субполос могут быть определены из $N/2$ отсчетов в каждой субполосе, то продолжение сигнала сохраняет полное восстановление.

Более строго можно сказать, что в каждой из бесконечных субполос может быть только $N/2$ различных отсчетов. Это невозможно при применении первых двух методов продолжения сигнала. Третий и четвертый методы обеспечивают полное восстановление. Периодическое и симметричное продолжения сигналов далее описываются более подробно и иллюстрируются

диаграммами, наглядно показывающими сохранение полного восстановления в этих случаях.

3.3.1. Периодическое продолжение

Метод периодического продолжения является наиболее часто используемым, так как он пригоден для любого типа фильтра. На рис.3.3 представлена диаграмма, поясняющая периодическое продолжение. На ней показана система анализа-синтеза (см. рис.3.2), повернутая на 90^0 . Для демонстрации продолжения сигнала использован ортогональный нелинейно-фазовый фильтр длиной 8, удовлетворяющий (3.2), (3.5) и (3.7).

В обоих блоках анализа в верхней строке показан исходный сигнал длиной $N = 8$ в темной рамке. Строчными буквами в квадратных рамках обозначаются отдельные отсчеты сигнала. Вне пределов темной рамки сигнал продолжается периодически, что отображается на диаграмме путем использования одних и тех же букв. $N/2 = 4$ строки в середине блока показывают четыре положения фильтра. Коэффициенты фильтра представлены прописными буквами. Для каждого положения вычисляется скалярное произведение коэффициентов фильтра и части сигнала, расположенного в блоке непосредственно над ним. Значения $N/2$ скалярных произведений показаны в нижней строке в темной рамке. Например, 'k' есть результат скалярного произведения фильтра в его первой позиции и части сигнала, обозначенного символами 'f', 'g', ..., 'e'. Пустые квадраты означают прореживание. В нижней строке показаны также значения, появляющиеся в выходном сигнале в случае свертки с больше чем $N/2$ позициями фильтра. Как видно, новых значений не возникает. Это означает, что свойство полного восстановления сохраняется.

В блоках синтеза субполосный сигнал продолжается и интерполируется, как показано в верхней строке. Для реконструкции требуются все N позиций фильтра (показаны только три из них). Для нашего выбора фильтров выполняется условие (3.7) и общая задержка системы $p + r + 1 - L_H$. Так что, с учетом (3.5), выбор $r = s = 4$ приводит к общей задержке, равной 0.

Для некоторых приложений, например для кодирования изображений, периодическое продолжение сигнала нежелательно. Левая и правая части сигнала, как правило, различаются. Это приводит к значительному перепаду в сигнале возле границы, следовательно, и к большим коэффициентам разложения, что неприемлемо для кодирования. Метод симметричного продолжения сигнала решает эту проблему, однако требует для своего применения симметричных фильтров.

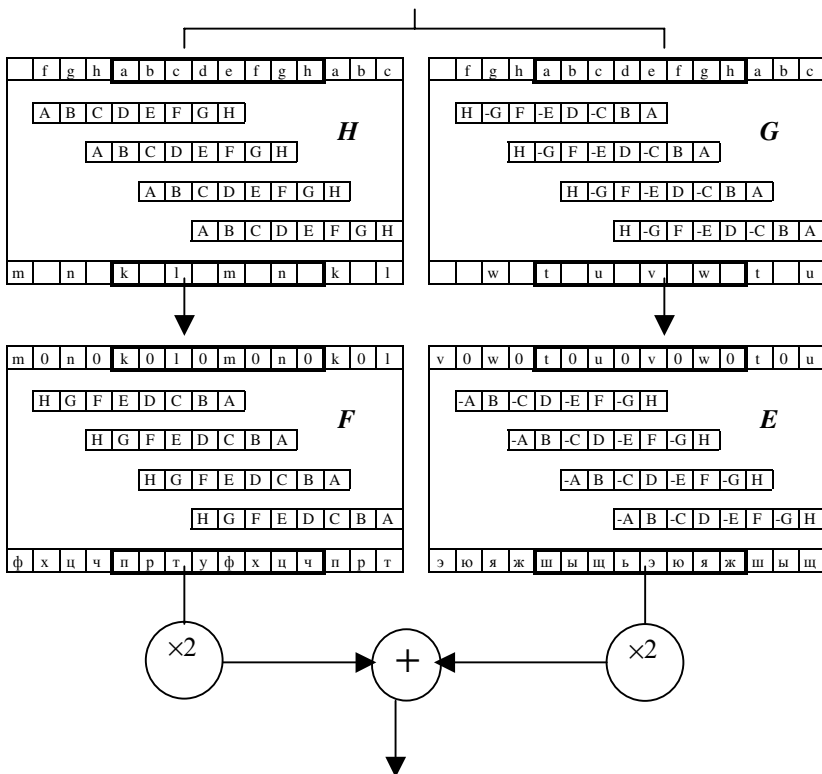


Рис.3.3. Периодическое продолжение сигнала (независимо от типа фильтра)

3.3.2. Симметричное продолжение

Симметричное продолжение сигнала может применяться при использовании симметричных фильтров и зависит от четности длины фильтра. Именно такое продолжение применено в видеокодеках ADV6xx. На диаграмме 3.4(а) показан случай четного фильтра. Примером такого фильтра может являться фильтр Джонстона, обладающий свойством почти полного восстановления. Симметрия фильтров показана на рисунке повторением букв. Входной сигнал отражается относительно границы, что приводит к гладкой склейке.

Для фильтров нечетной длины симметричное продолжение должно выполняться по-другому для получения $N/2$ различных отсчетов. Это показано на рис.3.4(б) для пары фильтров длиной 9 и 7. В блоках анализа ось симметрии проходит через отсчет (нечетная симметрия). В блоке синтеза

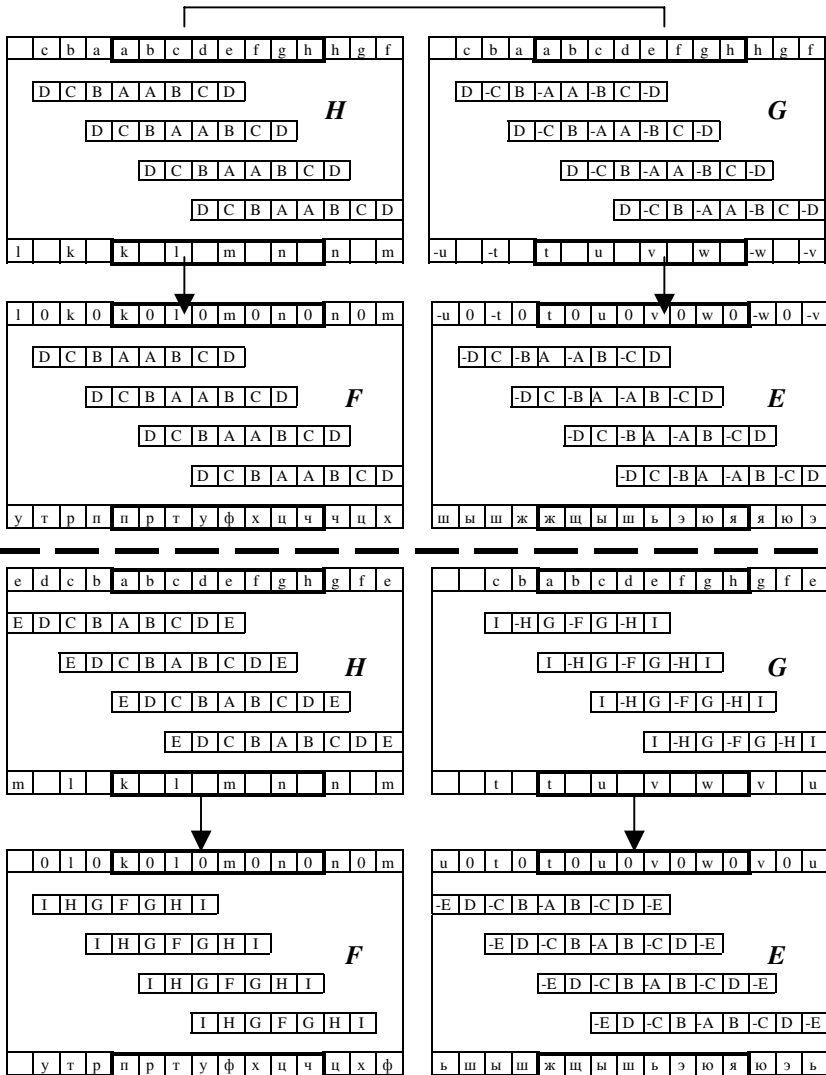


Рис. 3.4. Симметричное продолжение; (а) симметричные фильтры четной длины; (б) симметричные фильтры нечетной длины.

продолжение зависит от сигнала. Для одного канала левая часть сигнала продолжается путем симметричного нечетного отображения, правая – путем четного. Для другого канала – наоборот. Поэтому продолжение сигнала при синтезе отличается от продолжения его при анализе. Понятно, почему

в случае нечетной длины фильтров нельзя применять четное расширение: значения отсчетов субполос вне темной рамки не равны тем, что внутри ее. Следовательно, они несут информацию о сигнале, которая теряется при четной симметрии, так как эти отсчеты не передаются.

3.4. Эффективный метод продолжения для декомпозиции сигнала произвольной длины

В данном разделе мы покажем, каким образом сигнал нечетной длины может быть подвергнут вейвлет-преобразованию без увеличения количества коэффициентов. Это достигается путем применения продолжения сигнала, которое имеет три варианта: периодическое для любого типа фильтра и два варианта симметричного продолжения для симметричного фильтра.

Вернемся к схемам продолжения сигналов (см. рис.3.3 и 3.4). В случае сигнала нечетной длины (например, $N = 7$) отсчет, обозначенный h , отсутствует в нашем сигнале. Таким образом, темная рамка покрывает отсчеты a - g . Новый метод продолжения сигнала заключается в добавлении одного отсчета h и обработке получившегося четного сигнала обычным образом. Значение h выбирается так, чтобы один из отсчетов в субполосе принял какое-то фиксированное значение. Это фиксированное значение не зависит от сигнала, не несет о нем информации, значит, его не надо передавать. Так что, суммарное количество отсчетов в обеих субполосах равно количеству отсчетов исходного сигнала. Это позволяет осуществить разложение сигнала произвольной длины, показанное на рис.3.1(б).

В приведенных примерах в качестве фиксированного значения взят правый отсчет ВЧ полосы, обозначенный w на диаграмме. Значение w получается в результате фильтрации ВЧ фильтрами, находящимися на последних четырех позициях. Получается линейное уравнение относительно h . Приравняв w к нулю, всегда можно определить h .

Для осуществления реконструкции на приемной стороне не требуется никакой дополнительной информации в силу того, что известно, куда надо добавить нуль.

Точное значение, к которому мы приравниваем w , не важно для полного восстановления. Однако его выбор будет влиять на отсчеты в НЧ субполосе. Общее правило такое, что выбор w не должен изменять статистики НЧ субполосы.

В табл.3.1 представлены значения w , обеспечивающие полное восстановление для всех возможных длин фильтров и продолжений сигнала. Сигнал длиной N обозначается s_n , а ВЧ фильтр - g_n . Значение q выбирается в соответствии с (3.5) и (3.6).

**Формулы для добавляемого отсчета
при вейвлет-преобразовании сигнала нечетной длины**

Тип расширения / длина фильтра	Формула
Периодическое/четная	$-\left(\sum_{k=0}^q S_{N-q-1+k} g_k + \sum_{k=0}^{q-1} S_k g_{q+2+k}\right) / g_{q+1}$
Периодическое/нечетная	$-\left(\sum_{k=0}^q S_{N-q-1+k} g_k + \sum_{k=0}^q S_k g_{q+2+k}\right) / g_{q+1}$
Симметричное/четная	$-\left(\sum_{k=0}^q S_{N-q-1+k} g_k + \sum_{k=0}^{q-2} S_k g_{q+3+k}\right) / (g_q + g_{q+1})$
Симметричное/нечетная	$-2\left(\sum_{k=0}^q S_{N-q-1+k} g_k\right) / g_{q+1}$

Метод эффективного продолжения сигнала не увеличивает вычислительной сложности. Вместо последнего скалярного произведения при фильтрации (не требующегося) мы вычисляем значение добавляемого отсчета. Это требует того же количества умножений и сложений.

К сожалению, при применении данного метода для многомерной фильтрации значительно увеличивается его вычислительная сложность.

3.5. Симметрично-периодическое продолжение сигнала

Симметричное продолжение для фильтров нечетной длины приводит к «смещению» периода сигнала. В результате продолжения сигнала, показанного на рис.3.4 (б), получается сигнал периодичностью $2N - 2$, а не $2N$. Для фильтров четной длины периодичность всегда $2N$.

Используя метод, описанный в предыдущем разделе, можно осуществить симметрично-периодическое продолжение сигнала, дающее периодичность $2N$ для фильтров нечетной длины. Оно показано на рис.3.5 для той же пары фильтров 9-7. Добавляемый символ обозначен `z`. При соответствующем выборе `z`, как и в разделе 3.4, отсчет `x` на левой границе становится равен нулю, что уменьшает количество отсчетов, зависящих от сигнала с $N/2 + 1$ до $N/2$.

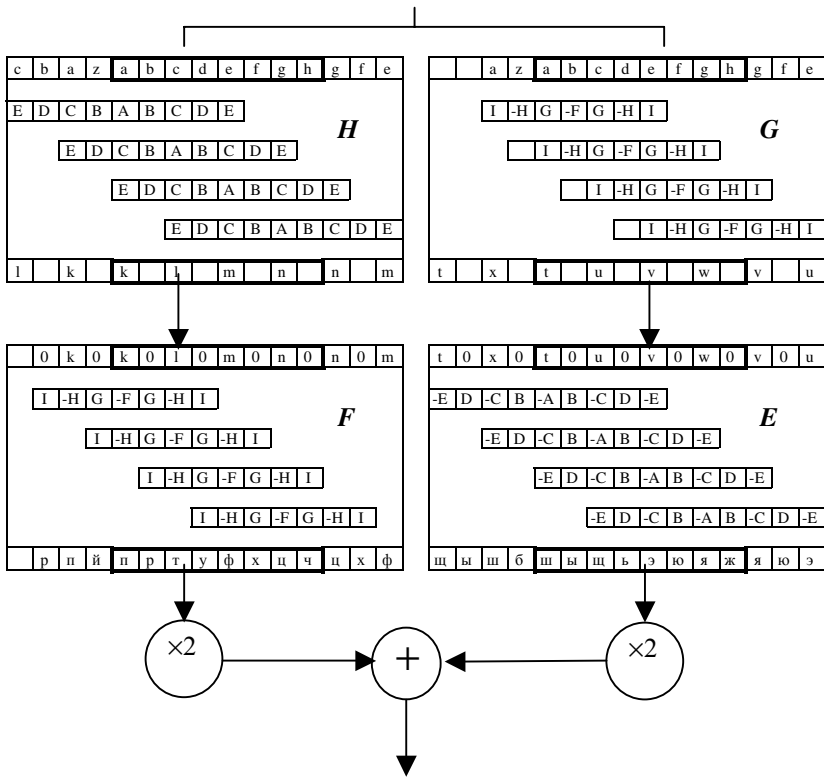


Рис.3.5. Симметрично-периодическое продолжение сигнала

Итак, в данной главе были показаны методы продолжения сигналов для осуществления двухканального субполосного и вейвлет-преобразования с полным восстановлением. Представленные диаграммы наглядно свидетельствуют, что полное восстановление тесно связано с правильным продолжением сигнала. Был представлен новый метод для продолжения сигнала нечетной длины, не увеличивающий числа отсчетов и сохраняющий свойство полного восстановления.

Метод также пригоден для многомерной фильтрации, хотя в случае неразделимых фильтров является непрактичным.

В главе показан также метод симметрично-периодического продолжения сигнала при фильтрации фильтром нечетной длины. Его отличительным свойством является то, что сигнал на выходе имеет период $2N$, а не $2N - 2$. Это является важным в некоторых приложениях.

СРАВНЕНИЕ ВЕЙВЛЕТ-ФИЛЬТРОВ С ФИЛЬТРАМИ, ПРИМЕНЯЕМЫМИ ПРИ СУБПОЛОСНОМ КОДИРОВАНИИ

Проводя в данной главе сравнение двух типов фильтров, мы ограничимся рассмотрением фильтров с конечной импульсной характеристикой, так как они наиболее часто применяются при кодировании изображений. В разделе 4.1 будут рассмотрены свойства идеального фильтра. Процедура построения реального фильтра заключается в минимизации отклонения заданного числа свойств от свойств идеального фильтра. При конструировании обычных и вейвлет-фильтров для этой минимизации используются различные критерии, как будет показано. Под обычными в этой главе понимаются фильтры, используемые в субполосном кодировании (см. главу 1).

В разделе 4.2 приведен пример расчета обычного фильтра. За основу взяты фильтры Джонстона, которые нашли применение во многих приложениях. Некоторые из этих фильтров дают низкое объективное качество кодирования изображений. Пример расчета вейвлет-фильтров приведен в разделе 4.3. Различия между обычными и вейвлет-фильтрами приведены в разделе 4.4.

4.1. Критерии для расчета фильтров

При субполосном кодировании применяемые фильтры должны строго разделить частотную область сигнала на непересекающиеся участки. Амплитудно-частотная характеристика (АЧХ) идеального фильтра низких частот для двухполосного банка фильтров приведена на рис.4.1(а). Такую прямоугольную характеристику может иметь фильтр бесконечной длины, что неприменимо. Существует несколько методов получения импульсной характеристики фильтра конечной длины, например применение окна. После такой операции АЧХ фильтра аппроксимирует идеальный прямоугольник (рис.4.1(б)).

Для оценки качества аппроксимации существует ряд критериев. Почти все они зависят от четырех параметров, показанных на рис.4.1(б):

δ_p - неравномерность в полосе пропускания;

δ_s - неравномерность в полосе задерживания;

ω_p - граничная частота полосы пропускания;

ω_s - граничная частота полосы задерживания.

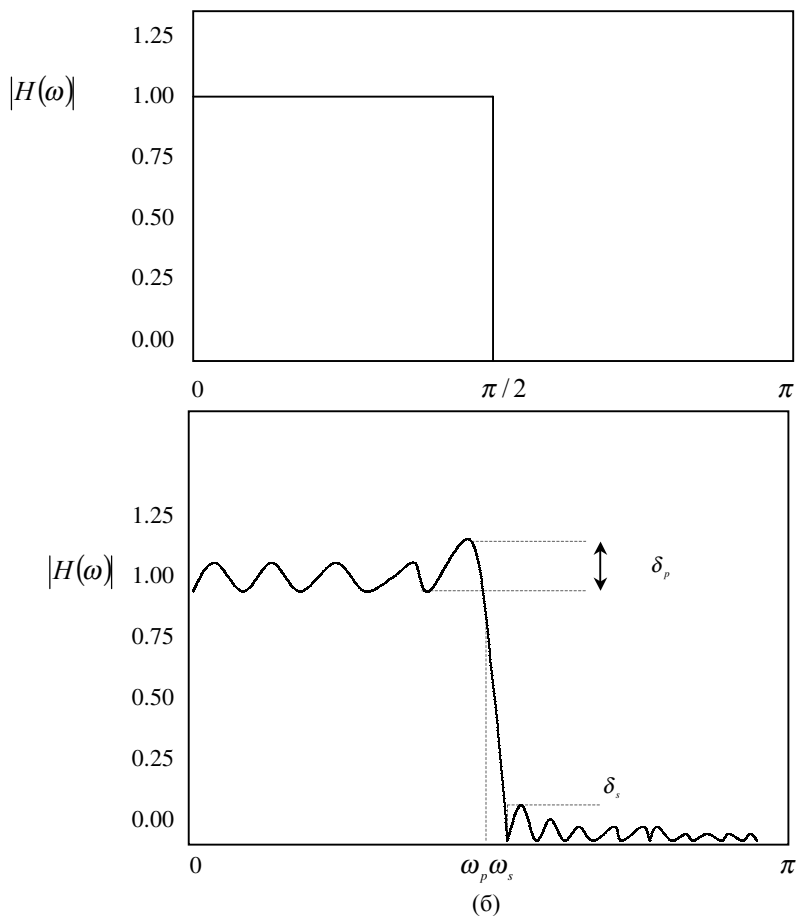


Рис. 4.1. Амплитудно-частотная характеристика: (а) идеального фильтра; (б) аппроксимация идеальной АЧХ реальным фильтром

Малое значение δ_p означает умножение гармоник сигнала на равные коэффициенты. Малое значение δ_s предотвращает появление в фильтрованном сигнале высокочастотных составляющих, которые могут появиться за счет эйлинга (см. главу 1). Эйлинг отрицательно влияет на визуальное качество изображения. Как отмечалось в главе 1, и в системах анализа-синтеза с подавлением эйлинга квантование коэффициентов может привести к его

появлению. Граничные частоты полосы пропускания должны быть как можно более близки к $\pi/2$.

При проектировании фильтров часто задаются тремя другими критериями, зависящими от вышеперечисленных. Это – затухание в полосе пропускания $A_p = -20\lg(1 - \delta_p)$, минимальное ослабление сигнала в полосе задерживания $A_s = -20\lg\delta_s$ и полоса перехода, определяемая как $\Delta\omega = \omega_s - \omega_p$.

Какой критерий важнее в каждом конкретном случае, решается в зависимости от приложения.

При расчете вейвлет-фильтров не пытаются аппроксимировать прямоугольную характеристику. Для построения вейвлет-фильтра важным является степень плоскости характеристики около частоты $\omega = \pi$, которая зависит от числа нулей низкочастотного фильтра на частоте $\omega = \pi$. Как обсуждалось в разделе 2.5, число нулей связано со степенью гладкости соответствующих функций. На рис.4.2 показана типичная характеристика вейвлет-фильтра.

Наконец, имеются еще две характеристики фильтров, которые необходимо учитывать при построении фильтров обоих типов. Во-первых, сумма коэффициентов фильтра, которая должна быть равна 1:

$$\sum_n h_n = 1. \quad (4.1)$$

При расчете вейвлет-фильтров равенство (4.1) выполняется автоматически. Для обычных фильтров оно выполняется в общем случае лишь приближенно. Чем больше отличается сумма коэффициентов от 1, тем больше смещение функции передачи.

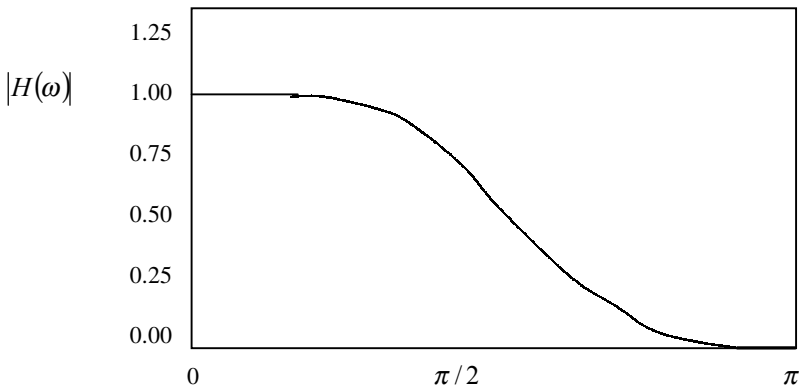


Рис. 4.1 Типичная частотная характеристика вейвлет-фильтра

Вторым критерием является энергия коэффициентов, которая должна быть равна 0,5:

$$\sum_n h_n^2 = 0,5. \quad (4.2)$$

Если не выполняется этот критерий, субполосы содержат разное количество энергии сигнала. Следовательно, ошибки квантования в одной субполосе будут более значимыми, чем в другой.

На этом обсуждение свойств фильтров закончим. В следующих двух разделах приводятся два различных примера построения фильтров.

4.2. Построение обычных фильтров: фильтры Джонстона

Фильтры Джонстона популярны во многих областях обработки сигналов, в том числе и в кодировании изображений. Наиболее важным критерием построения этих фильтров является минимальная ширина полосы перехода.

Будем использовать те же обозначения, что и в разделе 3.2. Выход двухканальной схемы анализа-синтеза связан с входом равенством (3.4):

$$T(\omega) = [H(\omega)G(\omega + \pi) - G(\omega)H(\omega + \pi)]e^{i\omega(p+r)}, \quad (4.3)$$

где $T(\omega)$ - функция передачи; p и r - смещения фильтров; $H(\omega)$ и $G(\omega)$ - НЧ и ВЧ фильтры, соответственно. Однако Джонстоном были выбраны другие соотношения между этими фильтрами, чем в выражении (3.7). При его выборе теряется свойство полного восстановления, если длина фильтра превышает 2. ВЧ фильтр становится отражением НЧ фильтра относительно частоты, четверо меньшей частоты дискретизации, т.е.

$$G(\omega) = H(\omega + \pi). \quad (4.4)$$

Поэтому эти фильтры получили название квадратурно-зеркальных (КЗФ). Тогда функция передачи будет

$$T(\omega) = [H(\omega)H(\omega) - H(\omega + \pi)H(\omega + \pi)]e^{i\omega(p+r)}. \quad (4.5)$$

Если фильтры симметричны и имеют четную длину, то

$$T(\omega) = [|H(\omega)|^2 + |H(\omega + \pi)|^2] e^{i\omega(1-L+p+r)}, \quad (4.6)$$

где L - длина фильтра h_n . Таким образом, все фильтры в блоке КЗФ зависят от одного фильтра. Так как фильтры симметричные, то фазовые искажения отсутствуют, и фаза может быть равной нулю при соответствующем выборе смещений p и r .

Остаются только амплитудные искажения, которые должны быть минимизированы. Нулевые искажения возможны лишь в случае $L = 2$. Джонстон наложил ограничение на полосу перехода. Тогда коэффициенты фильтра оптимизируются путем минимизации функции ошибки

$$C = 2 \int_0^{\pi/2} (T(\omega) - 1) d\omega + \alpha \int_{\omega_1}^{\pi} |H^2(\omega)| d\omega, \quad (4.7)$$

где параметр α взвешивает ослабление в полосе задержки. Равенство (4.7) вычисляется в дискретном виде путем применения 512-точечного БПФ и замены интеграла суммой.

Джонстоном были разработаны фильтры различных длин. Однако для кодирования изображения длинные фильтры неприемлемы, так что мы рассмотрим только фильтры, длина которых не превышает 32.

Джонстон разработал фильтры для 5 полос перехода. Комбинации этих полос с длинами фильтров показаны в табл.4.1, вместе с выбором параметра α .

В зависимости от ширины полосы перехода различают пять классов фильтров, обозначаемых буквами А, В, С, D и Е, как в таблице. Обычно фильтры Джонстона обозначаются следующим образом: 'joh16C' означает фильтр Джонстона класса С длиной 16.

Таблица 4.1

Длина, полоса перехода и параметр α фильтров Джонстона

				$\Delta\omega$		
		0.14 (A)	0.10 (B)	0.0625 (C)	0.43 (D)	0.23 (E)
	8	$\alpha = 1$	-	-	-	-
L	12	$\alpha = 1$	$\alpha = 1$	-	-	-
	16	$\alpha = 1$	$\alpha = 1$	$\alpha = 1$	-	-
	24	-	$\alpha = 1$	$\alpha = 1$	$\alpha = 1$	-
	32	-	-	$\alpha = 2$	$\alpha = 2$	$\alpha = 2$

$$\begin{aligned} h_3 - h_2 + h_1 - h_0 &= 0, \\ 0h_3 - 1h_2 + 2h_1 - 3h_0 &= 0, \end{aligned} \quad (4.15)$$

то из (4.14) и (4.15) получается единственное решение:

$$\begin{aligned} h_0 &= (1 + \sqrt{3})/8, & h_1 &= (3 + \sqrt{3})/8, \\ h_2 &= (3 - \sqrt{3})/8, & h_3 &= (1 - \sqrt{3})/8. \end{aligned} \quad (4.16)$$

Все фильтры Добеши могут быть получены посредством этой процедуры. Для фильтра длиной $L = 2N$ необходимо N равенств относительно нулевых моментов для получения единственного решения. В общем случае решения могут быть получены численно.

4.3.2. Расчет пары биортогональных фильтров

Процесс расчета линейно-фазовой биортогональной пары фильтров нечетной длины основан на фиксировании числа нулей на частоте $\omega = \pi$. Это приводит к максимально плоским фильтрам. Существуют и другие критерии построения этих фильтров.

Как известно, линейная фаза не может быть обеспечена у ортогональных фильтров длиной больше 2. Линейно-фазовый фильтр может быть симметричным или антисимметричным.

Из формул раздела 3.2 получаем модуль функции передачи

$$|T(\omega)| = |H(\omega)\tilde{H}(\omega) - H(\omega + \pi)\tilde{H}(\omega + \pi)| = 1, \quad (4.17)$$

где $H(\omega)$ и $\tilde{H}(\omega)$ - низкочастотные фильтры блоков анализа и синтеза, соответственно. Из этого равенства мы и получим пару фильтров. Кроме того, для описания фильтров в частотной области потребуем, чтобы они обладали некоторыми дополнительными свойствами. Во-первых, свойством симметрии в сочетании с нечетной длиной фильтров. Во-вторых, чтобы они имели определенное число нулевых моментов фильтров на частоте, равной половине частоты дискретизации.

Симметричные фильтры нечетной длины могут быть представлены как полиномы от $\cos \omega$:

$$\begin{aligned}
H(\omega) &= \sum_{n=0}^{L_H-1} h_n e^{-i\omega n} = \\
&= h_{M_H} e^{-i\omega M_H} + \sum_{n=0}^{M_H-1} h_n (e^{-i\omega n} + e^{-i\omega(L_H-1-n)}) = \\
&= e^{-i\omega M_H} \left(h_{M_H} + \sum_{n=0}^{M_H-1} h_n (e^{-i\omega(n-M_H)} + e^{-i\omega(M_H-n)}) \right) = \\
&= e^{-i\omega M_H} \left(h_{M_H} + 2 \sum_{n=0}^{M_H-1} h_n \cos((M_H - n)\omega) \right), \tag{4.18}
\end{aligned}$$

где $M_H = (L_H - 1)/2$. Далее, если $H(\omega)$ имеет l нулевых моментов, то выражение для фильтра можно записать в виде

$$H(\omega) = (1 + e^{-i\omega})^l Q_0(\omega) = \left(\cos\left(\frac{\omega}{2}\right) \right)^l Q_1(\omega), \tag{4.19}$$

где $Q_0(\omega)$ и $Q_1(\omega)$ - некоторые функции. Однако (4.19) будет полиномом по $\cos\omega$ только в случае четного l . Другими словами, симметричные фильтры нечетной длины могут иметь только четное количество нулей на частоте $\omega = \pi$. Поэтому, если $H(\omega)$ и $\tilde{H}(\omega)$ имеют $2k$ и $2\tilde{k}$ нулей, соответственно, из (4.18) и (4.19) получаем

$$|H(\omega)| = (1 + \cos\omega)^k Q(\cos\omega), \quad |\tilde{H}(\omega)| = (1 + \cos\omega)^{\tilde{k}} \tilde{Q}(\cos\omega), \tag{4.20}$$

где $Q(\cos\omega)$ и $\tilde{Q}(\cos\omega)$ - полиномы от $\cos\omega$. Теперь подставим (4.20) в (4.17) и получим

$$(1 + \cos\omega)^m Q(\cos\omega)\tilde{Q}(\cos\omega) + (1 - \cos\omega)^m Q(-\cos\omega)\tilde{Q}(-\cos\omega) = 1, \tag{4.21}$$

где $m = k + \tilde{k}$. Для удобства перепишем это равенство в виде полинома от $x = \sin^2(\omega/2)$:

$$(1-x)^m P(x) + x^m P(1-x) = 1. \tag{4.22}$$

Нашей целью является найти полином $P(x)$, который удовлетворяет

(4.22), и факторизовать функцию $(1-x)^m P(x) = H(\omega)\tilde{H}(\omega)$ на две составляющие. Тогда мы получим пару биортогональных фильтров. Согласно теореме Безу, существует единственный полином $P(x)$ степени меньше m , который является решением (4.22). Записав это уравнение в виде

$$P(x) = (1-x)^{-m} - x^m(1-x)^{-m} P(1-x), \quad (4.23)$$

можно разложить его правую часть в ряд Тейлора. Нам требуется только m его первых членов, так как известно, что $P(x)$ имеет максимальную степень $m-1$. Поэтому решение (4.22) будет иметь вид

$$P(x) = \sum_{n=0}^{m-1} \binom{m-1+n}{n} x^n. \quad (4.24)$$

После подстановки этого решения и записи вместо x $\sin^2(\omega/2)$ получаем

$$H(\omega)\tilde{H}(\omega) = \left(\cos\left(\frac{\omega}{2}\right) \right)^{2(k+\tilde{k})} \sum_{n=0}^{m-1} \binom{m-1+n}{n} \sin^{2n}\left(\frac{\omega}{2}\right). \quad (4.25)$$

Дальнейший расчет пары биортогональных фильтров заключается в выборе степеней гладкости k и \tilde{k} для соответствующих фильтров и разделении правой части (4.25) на две части.

Например, можно взять $k = \tilde{k} = 2$ и попробовать построить фильтры почти равной длины (равная длина невозможна). Каждый из двух фильтров дает свой косинусный член, и сумма квадратов синусов делится на две части. Сумма квадратов синусов становится равной

$$1 + 4x^2 + 10x^4 + 20x^6, \quad (4.26)$$

и единственный нетривиальный путь факторизовать ее на два полинома с вещественными коэффициентами есть

$$(a_0 + a_1x^2)(a_2 + a_3x^2 + a_4x^4), \quad (4.27)$$

с

$$\begin{aligned} a_0 &= 1.53118822, & a_1 a_4 &= 4.47213595, \\ a_2 &= 0.65308757, & a_3 &= 0.70487976. \end{aligned} \quad (4.28)$$

Тривиальная факторизация имеет члены 1 и $1 + 4x^2 + 10x^4 + 20x^6$ и приводит к фильтрам намного различающейся длины (11 и 5). Решение (4.28) приводит к паре фильтров длиной 9 и 7, считающихся в среднем лучшими для сжатия изображения. Именно эта пара фильтров и была использована в видеокодеках ADVbхх.

Так как биортогональная пара фильтров имеет два различных фильтра НЧ, необходимо решить, какой из них включать в секцию анализа, какой – в секцию синтеза. Решение обычно принимают исходя из необходимости того, чтобы фильтр синтеза был как можно более гладким. Это приводит к менее заметным ошибкам квантования в НЧ регионах изображения, к которым человеческий глаз очень чувствителен.

Из (4.25) могут быть получены и другие фильтры, если выбрать большие значения для k и \tilde{k} . Тогда возможно большее количество решений, приводящих к разнообразным парам фильтров. Например, при $k = \tilde{k} = 8$ имеем три решения, приводящих к фильтрам с длинами, различающимися на 2 (17 и 15).

Отметим, что вышеприведенные фильтры могли быть получены и матричным методом. Например, для пары фильтров 9 и 7 получается следующая матрица 12×12 :

$$\mathbf{M} = \begin{bmatrix} h_4 & h_3 & h_2 & h_1 & h_0 & & & & & h_0 & h_1 & h_2 & h_3 \\ h_2 & h_3 & h_4 & h_3 & h_2 & h_1 & h_0 & & & & & & h_1 \\ h_0 & h_1 & h_2 & h_3 & h_4 & h_3 & h_2 & h_1 & h_0 & & & & \\ & & h_0 & h_1 & h_2 & h_3 & h_4 & h_3 & h_2 & h_1 & h_0 & & \\ h_0 & & & & h_0 & h_1 & h_2 & h_3 & h_4 & h_3 & h_2 & h_1 & \\ h_2 & h_1 & h_0 & & & & h_0 & h_1 & h_2 & h_3 & h_4 & h_3 & \\ f_2 & -f_3 & f_2 & -f_1 & f_0 & & & & & & & & \\ f_0 & -f_1 & f_2 & -f_3 & f_2 & -f_1 & f_0 & & & & & & \\ & & f_0 & -f_1 & f_2 & -f_3 & f_2 & -f_1 & f_0 & & & & \\ & & & & f_0 & -f_1 & f_2 & -f_3 & f_2 & -f_1 & f_0 & & \\ f_0 & & & & & & f_0 & -f_1 & f_2 & -f_3 & f_2 & -f_1 & \\ f_2 & -f_1 & f_0 & & & & & & f_0 & -f_1 & f_2 & -f_3 & \end{bmatrix}. \quad (4.29)$$

Таким же образом можно получить и обратную матрицу (4.30). Отметим, что обратная матрица не является просто транспонированной, так как свойство ортонормальности отсутствует. Для обеспечения полного восстановления

ния матричное произведение $\mathbf{M}^{-1}\mathbf{M}$ должно быть равно единичной матрице, что дает нам четыре уравнения. Далее, потребуем наличия 4 нулей на частоте $\omega = \pi$, то есть 4 нулевых моментов ВЧ фильтров, связанных с h_n и f_n . Получаем еще 4 уравнения. Наконец, сумма коэффициентов обоих фильтров должна быть равна 1. Решив получившуюся систему 10 уравнений с 9 неизвестными, получим ту же пару фильтров, что и в предыдущем случае. Главным отличием является то, что в случае матричного метода требуется заранее знать длины фильтров.

$$\mathbf{M}^{-1} = 2 \begin{bmatrix} f_3 & f_1 & & & f_1 & h_3 & h_1 & & & h_1 & h_3 \\ f_2 & f_2 & f_0 & & f_0 & -h_4 & -h_2 & -h_0 & & -h_0 & -h_2 \\ f_1 & f_3 & f_1 & & & h_3 & h_3 & h_1 & & & h_1 \\ f_0 & f_2 & f_2 & f_0 & & -h_2 & -h_4 & -h_2 & -h_0 & & -h_0 \\ & f_1 & f_3 & f_1 & & h_1 & h_3 & h_3 & h_1 & & \\ & f_0 & f_2 & f_2 & f_0 & -h_0 & -h_2 & -h_4 & -h_2 & -h_0 & \\ & & f_1 & f_3 & f_1 & & h_1 & h_3 & h_3 & h_1 & \\ & & f_0 & f_2 & f_2 & f_0 & -h_0 & -h_2 & -h_4 & -h_2 & -h_0 \\ & & & f_1 & f_3 & f_1 & & h_1 & h_3 & h_3 & h_1 \\ f_0 & & & f_0 & f_2 & f_2 & -h_0 & & -h_0 & -h_2 & -h_4 & -h_2 \\ f_1 & & & & f_1 & f_3 & h_1 & & & h_1 & h_3 & h_3 \\ f_2 & f_0 & & & f_0 & f_2 & -h_2 & -h_0 & & -h_0 & -h_2 & -h_4 \end{bmatrix}. \quad (4.30)$$

Процесс расчета гарантирует, что сумма коэффициентов будет равна 1. Однако необходимо учитывать и энергию коэффициентов, которая может не быть равной 0.5. Например, можно уменьшить число нулевых моментов (а значит и число уравнений) и добавить уравнение для энергии коэффициентов.

4.4. Критерий оптимизации блоков фильтров, используемых при кодировании изображения

Обычно при разработке фильтров и блоков фильтров стараются обеспечить максимально большое затухание в полосе задерживания, минимальную полосу перехода и хорошую частотную избирательность. Эти требования к фильтрам возникли в основном из потребностей кодирования речи. Фильтры для кодирования изображений должны конструироваться исходя из других соображений.

Целью преобразования сигнала блоком фильтров является перераспределение его энергии. Для кодирования отсчетов на выходе блока фильтров должно тратиться меньше бит, чем для кодирования исходного сигнала. Поэтому естественно было бы ввести параметр, показывающий выигрыш от кодирования сигнала блоком фильтров. В данном разделе вводится такой параметр – выигрыш от субполосного кодирования. Введение данного понятия позволяет сравнивать различные блоки фильтров, а также осуществлять их оптимизацию.

4.4.1. Выигрыш от субполосного кодирования

Пусть имеется входной сигнал $x(n)$. Коэффициенты его вейвлет-преобразования $u_i(n)$ подвергаются квантованию:

$$v_i(n) = u_i(n) + q_i(n), \quad (4.31)$$

где $q_i(n)$ - шум квантования. Ошибка реконструкции

$$r(n) = x(n) - \hat{x}(n). \quad (4.32)$$

Для осуществления оптимизации необходимо предположить, что все сигналы являются случайными:

$$x(n), \sigma_x^2, u_i(n), \sigma_{u_i}^2, v_i(n), \sigma_{v_i}^2, q_i(n), \sigma_{q_i}^2, r(n), \sigma_r^2, \quad (4.33)$$

где σ^2 - дисперсия соответствующих сигналов. Введем параметры A_i и B_i , определяемые как

$$A_i = h_i^T \mathbf{R}_{xx} h_i, \quad B_i = \frac{1}{\alpha_i g_i^T g_i}, \quad (4.34)$$

где \mathbf{R}_{xx} - автокорреляционная матрица входного сигнала $x(n)$, h_i, g_i - фильтры анализа и синтеза i -го канала. Параметры A_i и B_i удовлетворяют следующим условиям:

$$\sigma_{u_i}^2 = A_i \sigma_x^2, \quad \sigma_r^2 = \sum_{i=0}^{M-1} B_i \sigma_{q_i}^2. \quad (4.35)$$

Так как блок вейвлет-фильтров относится к классу блоков фильтров с критической дискретизацией, то $\sum_{i=0}^{M-1} \frac{1}{\alpha_i} = 1$ (см. также формулу (1.16)). Тогда

общий ресурс бит запишется в виде $\sum_{i=0}^{M-1} \frac{1}{\alpha_i} R_i = R$. Целью оптимизации является минимизация дисперсии ошибки реконструкции σ_r^2 (4.35) при ограничении на общую скорость. Дисперсия шума квантования определяется следующим образом:

$$\sigma_{q_i}^2 \approx \varepsilon_x^2 2^{-2R_i} \sigma_{u_i}^2, \quad (4.36)$$

где ε_x - постоянная, зависящая от характеристик входного сигнала. Минимальная дисперсия ошибки реконструкции находится с использованием метода множителей Лагранжа:

$$\min \{ \sigma_{q_i}^2 \} = \prod_{i=0}^{M-1} (\alpha_i A_i B_i)^{\frac{1}{\alpha_i}} * \varepsilon_x^2 2^{-2R} \sigma_x^2. \quad (4.37)$$

Выигрыш от субполосного кодирования определяется как

$$G_{SBC} = \frac{\sigma_{r_{шик}}^2}{\sigma_{r_{opt}}^2} = \frac{1}{\prod_{i=0}^{M-1} (\alpha_i A_i B_i)^{\frac{1}{\alpha_i}}}. \quad (4.38)$$

4.4.2. Оптимальное распределение бит

Обозначим число уровней реконструкции для квантования n -го элемента вектора через $L_n = 2^{R_n}$. Тогда

$$R = \sum_{i=0}^{M-1} \frac{\log_2(L_i)}{\alpha_i} = \log_2 \left(\prod_{i=0}^{M-1} L_i^{\frac{1}{\alpha_i}} \right). \quad (4.39)$$

Далее,

$$2^R = \prod_{i=0}^{M-1} L_i^{\frac{1}{\alpha_i}} = L_{g_\alpha}^M, \quad (4.40)$$

где

$$L_{g_\alpha} = \left(\prod_{i=0}^{M-1} L_i^{\frac{1}{\alpha_i}} \right)^{\frac{1}{M}}. \quad (4.41)$$

Геометрическое среднее

$$L_g = \left(\prod_{i=0}^{M-1} L_i \right)^{\frac{1}{M}}. \quad (4.42)$$

Для достижения оптимального распределения бит по уровням квантования предположим, что

$$\sigma_{q_n}^2 = \varepsilon_{u_n} \frac{\sigma_{u_n}^2}{L_n^2}, \quad (4.43)$$

где ε_{u_i} - коэффициент, зависящий от квантователя. Для минимизации среднего искажения воспользуемся методом множителей Лагранжа. Потребуем выполнения следующего условия:

$$\frac{\partial}{\partial L_i} \left\{ \frac{1}{M} \sum_{n=0}^{M-1} \varepsilon_{u_n} \frac{\sigma_{u_n}^2}{L_n^2} + \lambda \prod_{n=0}^{M-1} L_n^{\alpha_n} \right\} = 0. \quad (4.44)$$

После дифференцирования и некоторых преобразований получим

$$\sigma_{q_i}^2 = \varepsilon_{u_i} \frac{\sigma_{u_i}^2}{L_i^2} = \frac{\lambda M L_g^M}{2 \alpha_i}. \quad (4.45)$$

Умножив обе стороны равенства на α_i , получим

$$\hat{\sigma}_{q_i}^2 = \alpha_i \sigma_{q_i}^2 = \frac{\lambda M L_g^M}{2}. \quad (4.46)$$

Так как правая часть этого выражения - константа, оптимальное распределение бит будет достигнуто в случае равномерного распределения шума по субполосам. Формулу (4.46) можно переписать в виде

$$\hat{\sigma}_{q_i}^2 = \left(\prod_{n=0}^{M-1} \hat{\sigma}_{q_n}^2 \right)^{\frac{1}{M}} = \frac{\left(\prod_{n=0}^{M-1} \varepsilon_{u_n} \right)^{\frac{1}{M}} \left(\prod_{n=0}^{M-1} \alpha_n \right)^{\frac{1}{M}} \left(\prod_{n=0}^{M-1} \sigma_{u_n}^2 \right)^{\frac{1}{M}}}{\left(\prod_{n=0}^{M-1} L_n^2 \right)^{\frac{1}{M}}}. \quad (4.47)$$

Введем следующие обозначения:

$$\sigma_{u_g}^2 = \left(\prod_{i=0}^{M-1} \sigma_{u_i}^2 \right)^{\frac{1}{M}}; \varepsilon_{u_g}^2 = \left(\prod_{i=0}^{M-1} \varepsilon_{u_i}^2 \right)^{\frac{1}{M}}; \alpha_g^2 = \left(\prod_{i=0}^{M-1} \alpha_i^2 \right)^{\frac{1}{M}}; L_g^2 = \left(\prod_{i=0}^{M-1} L_i^2 \right)^{\frac{1}{M}}. \quad (4.48)$$

Тогда из (4.46) и (4.47) следует, что

$$\frac{\alpha_g \varepsilon_{u_g} \sigma_{u_g}^2}{L_g^2} = \frac{\alpha_i \varepsilon_{u_i} \sigma_{u_i}^2}{L_i^2}, \quad i = 0, \dots, M-1. \quad (4.49)$$

Отсюда вытекает, что количество уровней квантования для i -го элемента

$$L_i = L_g \sqrt{\frac{\alpha_i \varepsilon_{u_i} \sigma_{u_i}^2}{\alpha_g \varepsilon_{u_g} \sigma_{u_g}^2}}. \quad (4.50)$$

Формула оптимального распределения бит принимает вид

$$R_i = \log_2(L_i) = \frac{R}{M} + \frac{1}{2} \log_2 \left(\frac{\alpha_i \varepsilon_{u_i} \sigma_{u_i}^2}{\alpha_g \varepsilon_{u_g} \sigma_{u_g}^2} \right). \quad (4.51)$$

Если положить $\alpha_i = M$, то из (4.51) следует, что

$$R_i = \frac{R}{M} + \frac{1}{2} \log_2 \left(\frac{\varepsilon_{u_i} \sigma_{u_i}^2}{\varepsilon_{u_g} \sigma_{u_g}^2} \right). \quad (4.52)$$

Из вышеприведенных формул может быть получено выражение, показывающее уменьшение энтропии входного сигнала за счет его разбиения банком фильтров:

$$\Delta H = H(X) - H_M(U) = \frac{1}{2} \log_2 \frac{\eta_x}{\eta_{u_g}} + \frac{1}{2} \log_2 \frac{\sigma_x^2}{\left[\prod_{i=0}^{M-1} \sigma_{u_i}^2 \right]^{\frac{1}{M}}}, \quad (4.53)$$

где η_x - константа, зависящая от функции плотности распределения и дисперсии x ; η_{u_g} - геометрическое среднее постоянных, зависящих от функции

плотности распределения и дисперсии u ; X и U - векторы отсчетов x и u , соответственно.

Статистика изображений может быть аппроксимирована авторегрессионным процессом первого порядка с корреляцией между пикселями $\rho = 0.95$. Автокорреляционная матрица будет иметь следующие элементы:

$$\mathbf{R}_{xx}(i, j) = \rho^{|i-j|} \sigma_x^2, \quad i, j \in [0, N_k - 1], \quad (4.54)$$

где ρ означает коэффициент корреляции и N_k - длина фильтра анализа. Коэффициенты этого фильтра будут получены путем максимизации выигрыша от субполосного кодирования (4.38) при ограничении (4.39). Нетрудно получить выражение для максимально возможного выигрыша от субполосного кодирования:

$$G_{SBC_{\max}} = \frac{1}{1 - \rho^2}. \quad (4.55)$$

При $\rho = 0.95$ $G_{SBC} = 10.11$ дБ.

Интересно отметить, что коэффициенты вейвлет-преобразования после первого уровня разбиения имеют $\rho \approx 0.85$. Следовательно, и выигрыш от субполосного кодирования увеличивается. Изменение значения коэффициента корреляции означает, что оптимальный блок должен иметь различные коэффициенты вейвлет-фильтров на разных ступенях преобразования.

4.5. Сравнение характеристик обычных и вейвлет-фильтров

Итак, вейвлет-фильтры и обычные фильтры различаются во многих аспектах, как мы показали в предыдущих разделах. Различия сведены в табл.4.2. Кратко прокомментируем эту таблицу.

Длина фильтра важна в силу двух причин. Во-первых, в таких приложениях, как сегментация изображения, длинный фильтр приведет к неверной локализации контуров, так как на протяжении одного фильтра могут встретиться два контура. Во-вторых, короткие фильтры вычислительно экономнее. К симметричным четным фильтрам применимо полифазное построение, что снижает вычислительную сложность в два раза.

В теории субполосного кодирования допускаются различные типы схем разбиения и любое число каналов у блоков фильтров. В теории вейвлет-анализа предполагается двухканальная схема с рекурсивным разбиением НЧ субполосы. Однако идеи и терминология вейвлетов используются и для многоканальных блоков фильтров, и для схем с произвольным разбиением частотной области (вейвлет-пакеты – см. главу 5).

**Сравнение фильтров, применяющихся
при субполосном кодировании, и вейвлет-фильтров**

Свойство	Фильтры		
	Джонстона	Добеши	Биортогональные
Реконструкция	Нет полного восстановления	Полное восстановление	Полное восстановление
Базисные функции	Неортогональные	Ортогональные	Биортогональные
Гладкость отн. длины фильтров	Низкая	Максимальная	Высокая
АЧХ	Хорошее A_p, A_s и $\Delta\omega$	Плоская на половине частоты дискретизации	Плоская на половине частоты дискретизации
Фаза	Линейная	Нелинейная	Линейная
Продолжение сигнала	Любое (симметричное)	Периодическое	Любое (симметричное)
Длина фильтров	Относительно длинные	Относительно короткие	Относительно короткие
Сложность	~длина фильтра/2	~ длина фильтра	~ длина фильтра

Ортогональность вейвлет-фильтров приводит к описанию сигнала взвешенной суммой базисных функций, где весами являются коэффициенты.

В основе теории субполосного кодирования и вейвлетов лежит двухканальная схема анализа-синтеза. По нашему мнению, вклад теории вейвлетов в кодирование изображений двояк. Во-первых, она показала связь дискретной фильтрации с теорией непрерывных функциональных пространств. Эта связь появляется при итерировании фильтров и рассмотрении их предельных функций, масштабирующей и вейвлета. Во-вторых, из теории вейвлетов вытекает рассмотрение гладкости фильтра как критерия для его разработки. Гладкость фильтра связана с плоскостью характеристики на частоте, равной половине частоты дискретизации. В противоположность этому, при субполосном кодировании основным критерием является аппроксимация прямоугольной характеристики. Правда, важность критерия гладкости для кодирования изображений многими авторами ставится под сомнение. По всей видимости, гладкость важна при решении, какой фильтр из биортогональной пары ставить в секцию синтеза. Применение более гладкого фильтра при синтезе улучшает характеристики кодирования.

Многими авторами сравнивались различные типы фильтров для тех или иных задач. В частности, для кодирования изображений рассмотренная биортогональная пара фильтров длиной 9 и 7 считается одной из лучших.

В заключение отметим, что разница между вейвлет-фильтрами и обычными становится все более несущественной. Недавние публикации показывают, что современные методы разработки фильтров используют различные смешанные критерии.

АДАПТИВНЫЕ ОРТОГОНАЛЬНЫЕ ПРЕОБРАЗОВАНИЯ

Вейвлет-преобразование сигнала является сигнально-независимым. Октавополосное разбиение спектра, производимое им, подходит для большинства, но не для всех реальных сигналов. Желательно было бы иметь преобразование, адаптированное к сигналу, подобно ПКЛ, но имеющее быстрый алгоритм выполнения. Это эквивалентно тому, что преобразование было бы способно произвольно менять структуру разбиения частотно-временной плоскости в зависимости от сигнала. Каскадно соединенные блоки вейвлет-фильтров позволяют достичь этого. В данной главе будут рассмотрены адаптивные преобразования, которые на основе введенной функции стоимости реализуют произвольное разбиение частотно-временной плоскости сигнала.

В разделе 5.1 рассмотрены так называемые пакеты вейвлетов, или адаптация в частотной области. В разделе 5.2 рассмотрен так называемый алгоритм двойного дерева, или адаптация базиса разложения как в частотной, так и в пространственной областях. Дальнейшее развитие этих идей приведено в разделе 5.3. В разделе 5.4 обсуждаются вопросы размерности библиотеки базисов для всех преобразований и их вычислительная сложность. Раздел 5.5 содержит выводы по данной главе.

5.1. Пакеты вейвлетов (алгоритм одиночного дерева)

Итак, вейвлет-преобразование сигнала выполняется путем его пропуска через каскадно соединенные двухканальные схемы А-С (см. рис.3.2). При этом каскадирование производится по низкочастотной области. Причина этого в неявном предположении, что эта область содержит больше информации об исходном сигнале. В результате получается «однобокое» дерево (рис. 5.1(а)). Данное предположение оправдано для многих реальных сигналов. В самом деле, оно означает, что наш сигнал является низкочастотным на большом интервале времени, а высокочастотные составляющие появляются на коротком интервале. Однако для некоторых сигналов это предположение не выполняется. Метод пакетов вейвлетов основан на определении того, по какой области на данном уровне выгоднее производить каскадирование. Для этого вначале производится каскадирование по обоим субполосам. В результате получается так называемое «полное», «сбалансированное» дерево (рис.5.1(б)), напоминающее дерево, присущее кратковременному преобразованию Фурье. Далее, на основе введенной функции стоимости определяется наилучший путь по этому дереву (рис. 5.1(в)). Если исходный блок вейвлет-

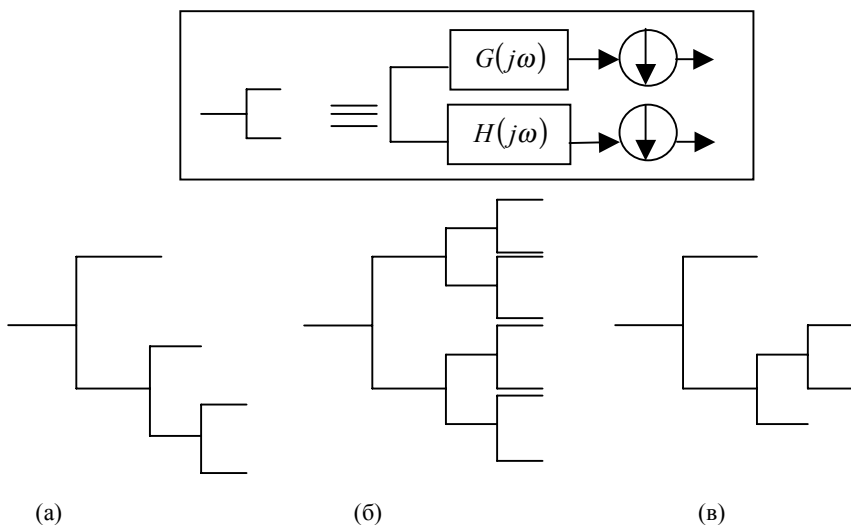


Рис. 5.1. Разбиение частотно-временной плоскости при помощи пакетов вейвлетов: (а) вейвлет-декомпозиция; (б) полная, аналогичная STFT декомпозиция; (в) пример декомпозиции при помощи пакетов вейвлетов

фильтров был ортогональным, то и схема, соответствующая любой конфигурации дерева, будет ортогональной, так как она есть не что иное, как каскадное соединение ортогональных блоков.

Таким образом, получается базис, адаптированный к сигналу. Отметим, что эта адаптация не требует обучения или знания статистических свойств сигнала. Вейвлет-преобразование (DWT), как и STFT, является частным случаем этого базиса. Адаптивность достигается за счет увеличения вычислительной стоимости. К счастью, разработан быстрый алгоритм поиска наилучшего базиса.

Пакеты вейвлетов были разработаны и исследованы Р.Койфманом и М.Викерхаузером. В качестве функции стоимости они использовали энтропию, понимаемую ими, как «концентрацию» числа коэффициентов M , требующихся для описания сигнала. Данная функция будет большой, если коэффициенты примерно одной величины, и малой, если все, кроме нескольких коэффициентов, близки к нулю. Таким образом, любое усреднение приводит к увеличению энтропии. Функция стоимости должна быть аддитивной. Это означает, что

$$M(0) = 0 \quad \text{и} \quad M(\{x_i\}) = \sum_i M(x_i). \quad (5.1)$$

Под энтропией в данном контексте понимается величина

$$M = e^{-\sum_n p_n \log p_n}, \quad (5.2)$$

где $p_n = |x_n|^2 / \|x\|^2$.

Эта энтропия вычисляется для каждого узла полного дерева пакета вейвлетов. Далее сравнивается сумма энтропии двух потомков и энтропия их предка на дереве. Если энтропия предка оказалась меньше, отказываемся от его декомпозиции, то есть «обрезаем» дерево. Алгоритм рекурсивно продолжается до достижения вершины дерева. Доказано, что данный алгоритм приводит к наилучшему базису относительно M .

Для решения задачи сжатия сигнала выбор энтропии в качестве функции стоимости, возможно, не является лучшим. В работах, посвященных сжатию изображений, в качестве функции стоимости используется функционал Лагранжа $J = D + \lambda R$. Здесь D - искажение (средний квадрат ошибки), вносимое за счет непередачи коэффициента узла, R - количество бит, требуемых для описания коэффициента на этом узле и λ - множитель Лагранжа. Эта функция стоимости включает в себя два частных случая: только искажение ($\lambda = 0$) и только скорость ($\lambda = -\infty$). Алгоритм выполняется так же, как и в случае выбора в качестве функции стоимости энтропии. Принятие решения для одного из узлов дерева показано на рис.5.2.

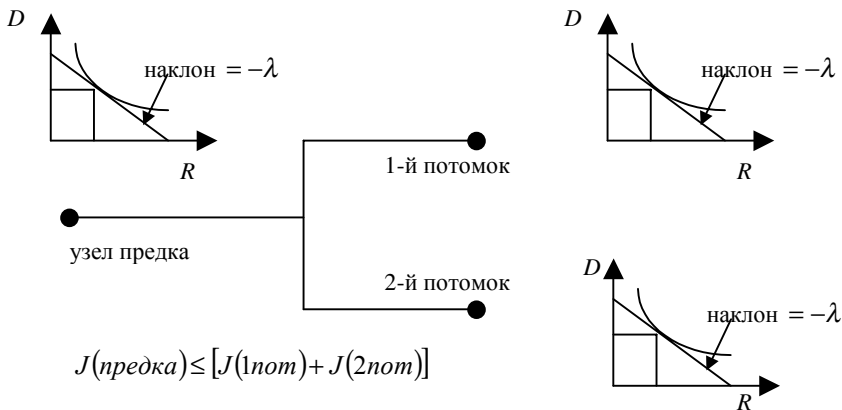


Рис. 5.2. Принятие решения в алгоритме одиночного дерева

Данный алгоритм получил название алгоритма одиночного (частотного) дерева. Он был применен для кодирования изображений. При этом на каждом этапе изображение делилось на четыре субполосы (структура, называемая квадродеревом вейвлет-коэффициентов).

При применении этого алгоритма для целей сжатия нельзя забывать о необходимости передавать декодеру информацию о структуре дерева. Одним из методов может быть посылка декодеру одного бита, указывающего, производилась или нет декомпозиция исходного изображения. Если – да, то посылает еще четыре бита, указывающих решение по разбиению каждой из субполос. Легко показать, что для дерева максимальной глубины d число

дополнительных бит не превышает $N = \sum_{k=1}^{k=d} 4^{k-1} = \frac{1}{3}(4^d - 1)$. Скажем, при 4-

уровневом разбиении изображения размером 512x512 потребуется 85 бит или примерно 0.000324 бит/пиксел, что совершенно незначительно.

5.2. Алгоритм двойного дерева

Хотя вейвлет-пакеты являются более гибким средством декомпозиции сигналов, чем вейвлет-преобразование, они не изменяются, а значит, и не адаптируются во времени (пространстве). Ряд важных классов сигналов (речь, изображения) являются нестационарными во времени и требуют более гибкого разложения. Например, для изображения адаптация может быть достигнута путем выполнения пространственной сегментации и применения алгоритма одиночного дерева к каждому сегменту.

Это приводит к пространственно изменяющимся вейвлет-пакетам. Быстрый алгоритм, позволяющий достигнуть подобного разбиения, получил название алгоритма двойного дерева.

Алгоритм двойного дерева основан на совместном поиске наилучшей (бинарной) пространственной сегментации и частотного разбиения для каждого сегмента сигнала. Данный алгоритм базируется на теории пространственно изменяющихся блоков фильтров. Дадим краткое пояснение работы алгоритма на примере одномерной декомпозиции (рис.5.3), имея в виду, что переход к двумерному случаю элементарен. Предположим, что сигнал содержит четыре субполосы – А,В,С,Д. Мы можем построить одиночное дерево для всего сигнала (ABCD), для двух его половинок после выполнения сегментации (AB или CD) или для каждой из четвертей (А,В,С,Д). В конце концов, мы получим избыточное представление сигнала – структуру двойного дерева – древовидную сегментацию во времени и частоте.

Для совместного поиска наилучшей сегментации во времени и лучшего базиса вейвлет-пакетов для каждого сегмента выполняется следующее. Для каждого возможного временного сегмента длиной, кратной степени двойки, выполняется разложение посредством вейвлет-пакетов. Найденные значения

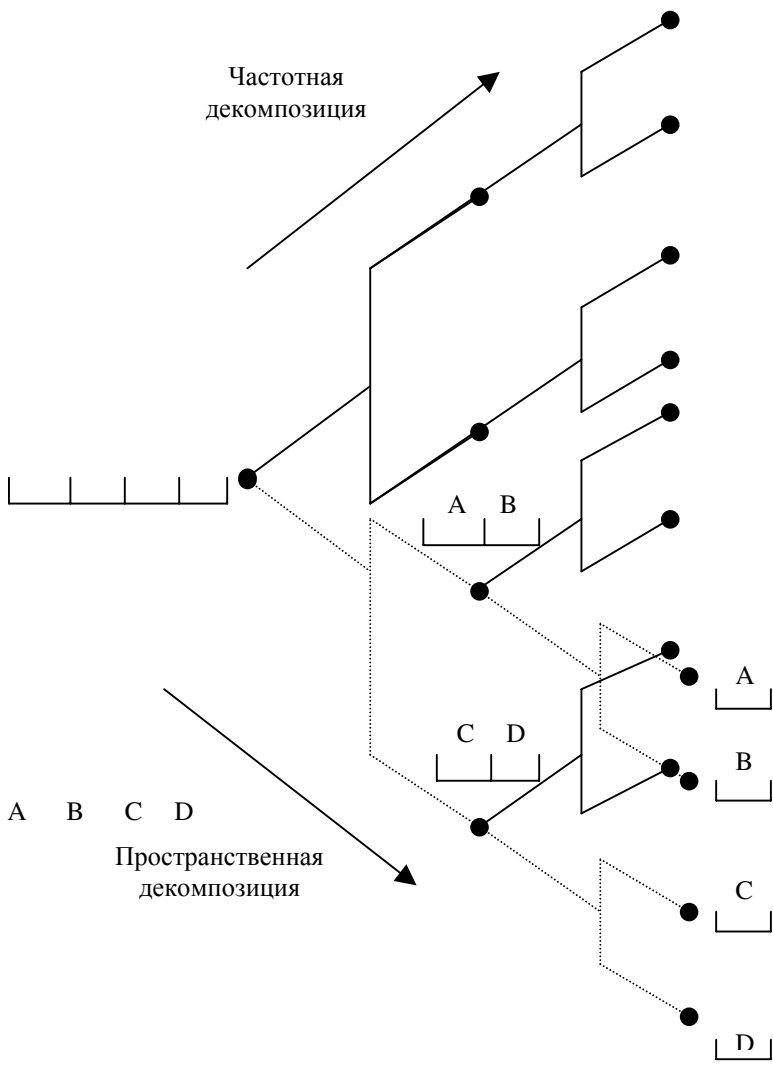


Рис. 5.3. Полное двойное дерево глубины 2 для одномерного сигнала. Сплошные линии показывают частотное дерево, штриховые – пространственное

стоимостей Лагранжа сегментов записываются в виде бинарного дерева. Далее к получившемуся дереву применяют алгоритм одиночного дерева для нахождения наилучшей сегментации.

Может быть показано, что число дополнительных бит, которое необходимо послать декодеру для дерева максимальной глубины d , определяется по формуле $N = \sum_{k=1}^{k=d-1} 4^k$. Для изображения размером 512x512 и дерева глубиной 5 число бит равно 341 или 0.0013 бит/пиксел.

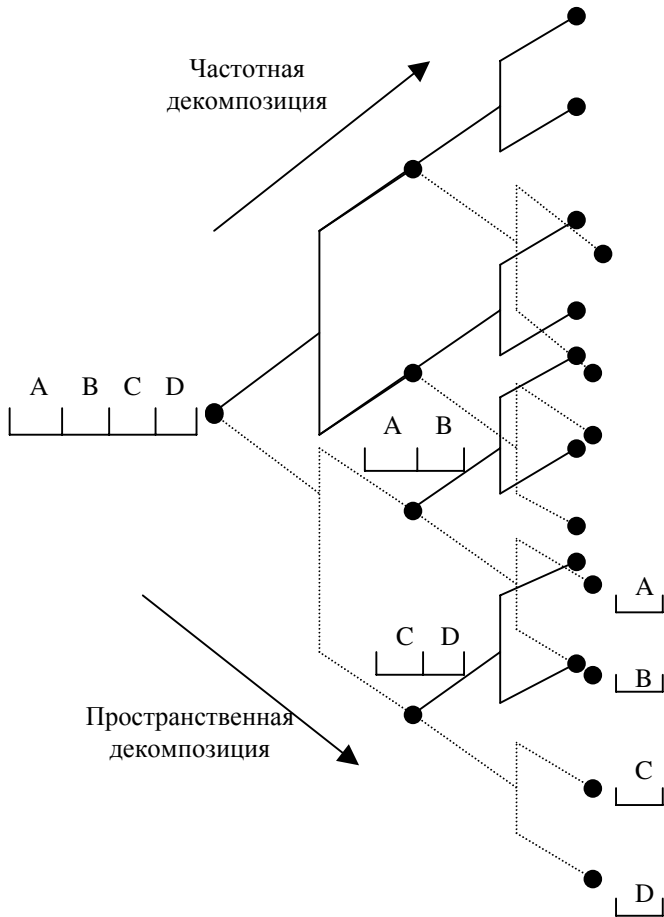


Рис. 5.4. Полное частотно-временное дерево глубиной 2 для одномерного сигнала. Сплошные линии показывают частотное дерево, штриховые – пространственное

5.3. Частотно-временное дерево

Алгоритм двойного дерева (см. раздел 5.2) обладает некоторой асимметрией. В самом деле, деревья в частотной области строятся над временными сегментами, но не наоборот. Этот недостаток можно устранить, построив дерево, приведенное на рис.5.4, где кандидатом на дальнейшее разбиение является субсигнал, как во временной, так и в частотной области. Такое дерево еще называют сбалансированным. Ясно, что дерево, представленное на рис.5.3, является частным случаем этого частотно-временного дерева. Частотно-временное дерево имеет структуру квадродерева. Мы видим, что каждый родительский узел имеет две пары потомков: временные и частотные сегменты.

Обрезание этого дерева осуществляется путем сравнения значений функции стоимости Лагранжа. Сравнение выполняется для пространственной и частотной пары на каждом узле в направлении от листьев дерева к его вершине. В результате выполнения алгоритма получается оптимальное двоичное дерево разбиения по частоте и по времени полной глубины. В этом его отличие от алгоритма одиночного дерева, где обрезанное дерево имеет, как правило, неполную глубину.

Разница между алгоритмами одиночного, двойного и частотно-временного дерева может быть легко уяснена, если взглянуть на разбиение частотно-временной плоскости, производимое ими (рис.5.5).

Отметим, что на рис.5.5(а) каждое деление по частоте относится ко всему сигналу, так как структура дерева не меняется во времени. Разбиение, показанное на рис.5.5(б), невозможно получить при помощи алгоритма одиночного дерева. Вертикальная линия посередине соответствует сегментации во

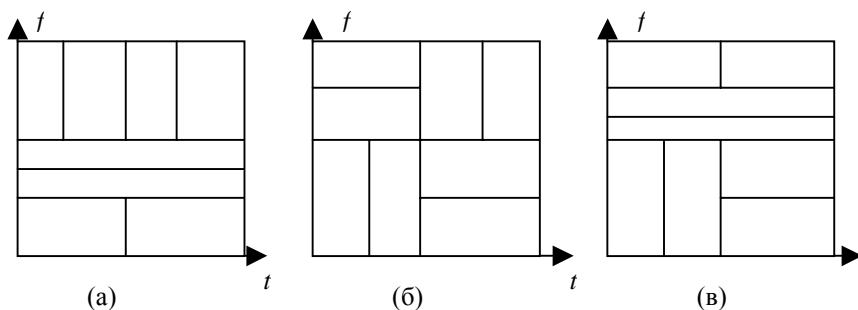


Рис. 5.5. Примеры разбиения, достигаемые различными алгоритмами:
(а) алгоритм одиночного дерева; (б) алгоритм двойного дерева;
(в) алгоритм частотно-временного дерева

временной области. Отметим, что двум половинкам сигнала соответствуют разные одиночные деревья. Разбиения, представленного на рис.5.5(в), можно достичь только с использованием алгоритма частотно-временного дерева.

Для кодирования изображений алгоритм частотно-временного дерева несложно перенести на двумерный случай. Тогда получается пространственно-частотное дерево.

Недостатком рассмотренных алгоритмов является принципиальное ограничение двоичной сегментацией во времени. Отсюда вытекает также чувствительность временной сегментации к сдвигам исходного сигнала. Для ликвидации этой чувствительности известен алгоритм, основанный на динамическом программировании и названный деревом гибкой пространственной сегментации. При этом разбиение в частотной области остается бинарным, так как используется двухканальный блок фильтров. Данный алгоритм включает в себя как частные случаи все выше рассмотренные алгоритмы. Основным недостатком его является невозможность простого перенесения алгоритма на двумерный случай для кодирования изображений.

5.4. Сравнение обсуждаемых алгоритмов

Сравнение алгоритмов произведем по следующим параметрам: 1) размерность библиотеки базисных функций, среди которых осуществляется поиск наилучшей; 2) вычислительная сложность; 3) эффективность кодирования реальных изображений.

5.4.1. Размерность библиотеки базисов

Может быть показано, что для одномерного сигнала длиной N при применении двухканального блока фильтров число базисов $S(N)$, перебираемых алгоритмом одиночного дерева, вычисляется рекурсивно:

$$S(N) = [S(N/2)]^2 + 1, \quad (5.3)$$

с $S(2) = 2$. Это вытекает из следующих соображений. Любое двоичное дерево может быть представлено в виде суммы двух субдеревьев высотой на 1 меньше. Если количество базисов в этих субдеревьях $S(N/2)$, то количество базисов во всем дереве $S(N) = [S(N/2)]^2 + 1$.

Для упрощения анализа алгоритма двойного дерева предположим, что используются фильтры Хаара, так как в этом случае не требуется применение граничных фильтров. Аналогично предыдущему случаю может быть показано, что количество перебираемых базисов

$$D(N) = [D(N/2)]^2 + S(N) - S(N/2), \quad (5.4)$$

с $D(2) = 2$, а $S(N) - S(N/2)$ - количество «новых» базисов одиночного дерева, когда нет пространственной сегментации.

Количество базисов для частотно-временного дерева может быть вычислено аналогично (для фильтров Хаара):

$$B(N) = 2[B(N/2)]^2 - [B(N/4)]^4, \quad (5.5)$$

с $B(2) = 2$. В случае использования других фильтров количество базисов увеличивается за счет применения граничных фильтров либо периодического расширения сигнала и становится равным $B(N) = 2[B(N/2)]^2$.

Для алгоритма гибкой временной сегментации количество базисов

$$F(N) = \sum_{i=1}^{\lfloor \log N \rfloor} (S(2^i) - S(2^{i-1}))F(N - 2^i), \quad (5.6)$$

с $F(2) = 2$ и $S(1) = 0$. В табл.5.1 подытожено количество базисов, перебираемых различными алгоритмами. Например, при $N = 8$ $S(8) = 26$, $D(8) = 70$, $B(8) = 82$ и $F(8) = 94$. При $N = 64$ $S(64) = 2.10 \times 10^{11}$, $D(64) = 9.78 \times 10^{14}$, $B(64) = 6.41 \times 10^{16}$ и $F(64) = 1.06 \times 10^{17}$. Конечно, для двумерного случая количество базисов значительно больше.

Таблица 5.1

Сравнение количества базисов, перебираемых различными алгоритмами

одиночное дерево	$S(N) = [S(N/2)]^2 + 1, S(2) = 2$
двойное дерево	$D(N) = [D(N/2)]^2 + S(N) - S(N/2)$
частотно-временное дерево	$B(N) = 2[B(N/2)]^2 - [B(N/4)]^4$
дерево гибкой временной сегментации	$F(N) = \sum_{i=1}^{\lfloor \log N \rfloor} (S(2^i) - S(2^{i-1}))F(N - 2^i)$

5.4.2. Вычислительная сложность алгоритмов

Для одномерного сигнала длиной N и дерева максимальной высотой d вычислительная сложность алгоритма одиночного, двойного и частотно-временного дерева будет $O(Nd)$, $O(Nd^2)$, $O(N2^d)$, соответственно. Вычислительная сложность алгоритма гибкой сегментации - $O(NM^2d)$, где M - максимальное число сегментов ($N = ML$). Например, выполнение алгоритма одиночного дерева для изображения 512×512 (поиск среди 4.9×10^{19} базисов) занимает 5.65 секунд на машине Pentium-100. Вычисление вейвлет-преобразования этого изображения занимает на той же машине 1.1 секунду. Выполнение алгоритма двойного дерева (поиск среди 5.6×10^{78} базисов) занимает 21.18 секунд.

5.4.3. Эффективность кодирования изображений

Приведем табл.5.2, показывающую эффективность кодирования двух тестовых изображений рассмотренными выше алгоритмами. Как следует из таблицы, применение адаптивных алгоритмов дает выигрыш при кодировании до 2.6 дБ при низких скоростях кодирования по сравнению с вейвлет-преобразованием. Платой за это является дополнительная вычислительная сложность.

Таблица 5.2

Сравнение результатов кодирования тестовых изображений

		Lena	Barbara		Lena	Barbara
ДЕРЕВЬЯ:	Скорость (бит/пикс)	ПОСШ (дБ)	ПОСШ (дБ)	Скорость (бит/пикс)	ПОСШ (дБ)	ПОСШ (дБ)
Вейвлет	0.5	36.2	29.7	0.25	33.1	26.1
Одиночное	0.5	36.4	31.8	0.25	33.4	28.2
Двойное	0.5	36.4	31.8	0.25	33.4	28.2
Частотно-временное	0.5	36.9	32.3	0.25	33.8	28.7

Итак, нами были рассмотрены адаптивные ортогональные преобразования, построенные на базе вейвлет-преобразований. Под адаптивностью здесь понимается автоматический выбор базиса для сигналов как в частотной, так и в пространственной областях. Рассмотрены алгоритмы, позволяющие осуществлять адаптацию в частотной области (вейвлет-пакеты – алгоритм одиночного дерева), сначала во временной, потом – в частотной (алгоритм двойного дерева), одновременно в обеих областях (алгоритм частотно-временного дерева). Недостатком этих алгоритмов является ограничение на

бинарное разбиение во временной области. От этого недостатка свободен алгоритм гибкой сегментации, основанный на динамическом программировании. Этот алгоритм подробно не рассматривался, так как его недостатком является невозможность перенесения на двумерный случай для кодирования изображений.

В разделе 5.4 показано количество базисов, перебираемых каждым алгоритмом, вычислительная сложность и эффективность применения для сжатия изображений. Общая тенденция такова, как и следовало ожидать: чем сложнее алгоритм вычислительно, тем выше его эффективность. Таким образом, перспективы применения того или иного алгоритма зависят от конкретного приложения. Кроме того, вероятно, лучшие результаты могут быть достигнуты, если отделить процесс сегментации от преобразования при помощи пакетов вейвлетов. В настоящее время разработаны эффективные алгоритмы сегментации, которые могут быть с успехом применены. После сегментации каждый сегмент приводится к прямоугольному виду, и над ним выполняется преобразование с использованием пакетов вейвлетов.

ЛИФТИНГОВАЯ СХЕМА

В данной главе представлены основные идеи и концепции, лежащие в основе конструирования вейвлетов во временной области, то есть вне зависимости от преобразования Фурье. Это позволяет создавать вейвлеты второго поколения, уже не являющиеся растяжениями и сдвигами одной функции и обладающие рядом дополнительных свойств. Кроме того, лифтинговая схема позволяет конструировать биортогональные вейвлеты и имеет ряд преимуществ перед классической схемой вейвлет-преобразования.

Основная идея лифтинговой схемы весьма проста. Как показано на рис.6.1, преобразование включает в себя три этапа: разбиение (S), предсказание (P) и обновление (U).

Предположим, имеется сигнал $f(t)$. Обозначим его отсчеты через $\lambda_{0,k} = f(k)$, $k \in Z$. Требуется «декоррелировать» этот сигнал. Другими словами, задачей является поиск представления сигнала меньшим числом коэффициентов, что эквивалентно увеличению интервала дискретизации. Возможно, не удастся точно представить сигнал, но лишь аппроксимировать его с допустимой мерой погрешности. Значит, необходимо управлять количеством теряемой при аппроксимации информации. Очевидно, что эта информация, то есть разница между исходным сигналом и его аппроксимацией, должна быть как можно меньше.

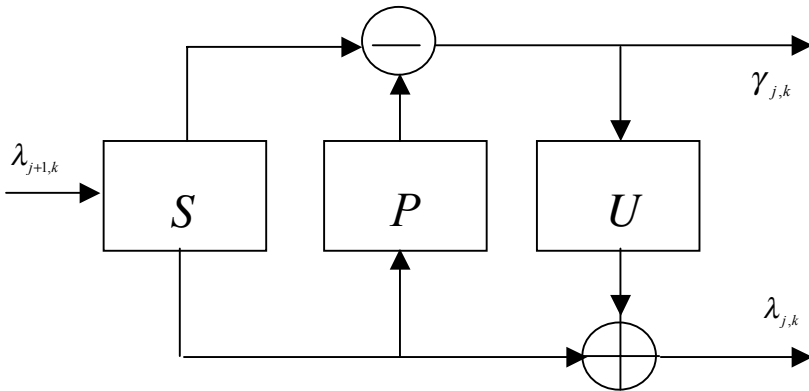


Рис. 6.1. Лифтинговая схема: разбиение, предсказание и обновление

6.1. Этап разбиения

Можно уменьшить число коэффициентов, просто оставив лишь четные отсчеты. В результате получается новая последовательность:

$$\lambda_{-1,k} = \lambda_{0,2k}, \quad k \in Z, \quad (6.1)$$

где отрицательный индекс используется для обозначения последовательности меньшей длины.

Необходимо оценить количество потерянной информации. Другими словами, что (если требуется) должно быть добавлено к последовательности $\{\lambda_{-1,k}\}$ для восстановления исходной последовательности $\{\lambda_{0,k}\}$. Обозначим эту добавку через $\{\gamma_{-1,k}\}$ и назовем ее вейвлет-коэффициентами. В зависимости от статистики исходного сигнала здесь возможны различные выборы, один лучше другого. Лучше означает меньшее значение вейвлет-коэффициентов.

Можно решить, что потерянная информация просто содержится в нечетных коэффициентах, $\gamma_{-1,k} = \lambda_{0,2k+1}$, $k \in Z$. Этот выбор соответствует так называемому вейвлету Лэйзи. В сущности, мы просто разделили сигнал на четные и нечетные отсчеты. Конечно, никакой декорреляции не произошло. Вейвлет-коэффициенты будут малы лишь в случае, если нечетные отсчеты будут малы.

Важно отметить, что ни на способ разбиения последовательности данных, ни на размеры субпоследовательностей не налагается никаких ограничений. Единственным требованием является наличие процедуры, позволяющей восстановить $\{\lambda_{0,k}\}$ по $\{\lambda_{-1,k}\}$ и $\{\gamma_{-1,k}\}$. Простейшей возможностью разбиения может быть деление отрезка сигнала пополам. Однако деление на четную и нечетную части предпочтительнее, так как эти последовательности более коррелированы между собой. Таким образом, получаем вейвлет Лэйзи.

Следующий этап лифтинговой схемы, предсказание, помогает конструировать более сложные и эффективные вейвлеты.

6.2. Этап предсказания

Итак, нашей целью является получение полностью обратимого компактного представления $\{\lambda_{0,k}\}$ через $\{\lambda_{-1,k}\}$ и $\{\gamma_{-1,k}\}$. Ясно, что четные отсчеты непосредственно находятся, как $\lambda_{0,2k} = \lambda_{-1,k}$. Попытаемся найти или, по крайней мере, предсказать нечетные отсчеты, основываясь на корреляции исходных данных. Необходимо найти не зависимый от данных оператор предсказания P , такой что

$$\gamma_{-1,k} = P(\lambda_{-1,k}). \quad (6.2)$$

Вид оператора предсказания зависит от используемой модели сигнала и отражает его корреляционные связи.

Как правило, не существует возможности точного предсказания $\{\gamma_{-1,k}\}$, основанного на $\{\lambda_{-1,k}\}$. Однако $P(\lambda_{-1,k})$ может быть очень близок к $\{\gamma_{-1,k}\}$. Тогда можно заменить $\{\gamma_{-1,k}\}$ разностью между этой последовательностью и $P(\lambda_{-1,k})$. Обозначим абстрактный оператор разности через «-» и получим

$$\gamma_{-1,k} = \lambda_{0,2k+1} - P(\lambda_{-1,k}). \quad (6.3)$$

Вейвлет-коэффициенты теперь показывают, насколько исходный сигнал не соответствует модели, на основе которой построен оператор предсказания P . Если сигнал коррелирован, то большинство вейвлет-коэффициентов будет мало.

При соответствующем выборе оператора P , заменив исходный сигнал меньшими последовательностями $\{\lambda_{-1,k}\}$ и $\{\gamma_{-1,k}\}$, получаем компактное представление сигнала.

Для поиска хорошего оператора предсказания предположим, что соседние отсчеты сигнала сильно коррелированы. Тогда для предсказания нечетных отсчетов $\lambda_{0,2k+1}$ можно просто взять среднее их (четных) соседей: $\{\lambda_{-1,k}\}$ и $\{\lambda_{-1,k+1}\}$. Вейвлет-коэффициенты тогда находятся, как

$$\gamma_{-1,k} = \lambda_{0,2k+1} - 1/2(\lambda_{-1,k} + \lambda_{-1,k+1}). \quad (6.4)$$

Модель, используемая в данном случае для нахождения оператора P , есть кусочно-линейная функция на интервалах длиной 2. Если исходный сигнал совпадает с этой моделью, то все вейвлет-коэффициенты будут равны нулю. Другими словами, вейвлет-коэффициенты показывают, насколько сигнал не является линейным. В терминах частотного наполнения вейвлет-коэффициенты отражают высокочастотные составляющие, присутствующие в сигнале.

Последовательность $\{\lambda_{-1,k}\}$ отражает низкие частоты, имеющиеся в сигнале. Далее производятся итерации этой схемы. Последовательность $\{\lambda_{-1,k}\}$ разбивается на $\{\lambda_{-2,k}\}$ и $\{\gamma_{-2,k}\}$, затем $\{\gamma_{-2,k}\}$ заменяется разностью между $\{\gamma_{-2,k}\}$ и $P(\lambda_{-2,k})$. После выполнения n итераций исходный сигнал оказыва-

ется разложенным в вейвлет-базис $\{\lambda_{-n,k}, \gamma_{-n,k}, \dots, \gamma_{-1,k}\}$. Так как вейвлет-коэффициенты кодируют отличие сигнала от некоторой выбранной модели, это приводит к компактному представлению сигнала.

6.3. Различные операторы предсказания

Предсказание вовсе необязательно должно быть линейным. В качестве модели можно использовать полином любого порядка, что приведет к концепции интерполяционного подразделения. При этом осуществляется аппроксимация исходного сигнала функцией, определенной на всей вещественной оси. В предыдущем разделе было показано использование рекурсивной процедуры для нахождения значений этой предсказывающей (интерполирующей) функции на каждом уровне.

Обозначим через N порядок схемы подразделения (интерполяции). Например, для кусочно-линейной аппроксимации $N = 2$, для кубической аппроксимации $N = 4$ и т.д. N показывает степень гладкости интерполяционной функции, применяемой для вычисления вейвлет-коэффициентов. Будем называть эту функцию дуальный вейвлет, и N - число дуальных нулевых моментов.

Рассмотрим схемы интерполяции более высокого порядка, чем линейная. Например, рассмотрим кубическую интерполяцию. При этом новое значение определяется четырьмя отсчетами сигнала, которые уникальным образом определяют кубический интерполяционный полином:

$$\lambda_{j,k-1} = p(x_{j,k-1}), \lambda_{j,k} = p(x_{j,k}), \lambda_{j,k+1} = p(x_{j,k+1}), \lambda_{j,k+2} = p(x_{j,k+2}). \quad (6.5)$$

Новое значение (с нечетным индексом) будет равно значению, принимаемому этим полиномом на середине интервала. Нечетные отсчеты с четным индексом остаются без изменения:

$$\lambda_{j+1,2k} = \lambda_{j,k} \lambda_{j+1,2k+1} = p(x_{j+1,2k+1}). \quad (6.6)$$

На рис.6.2 показан этот процесс для линейной и кубической интерполяций. Несмотря на то, что на каждом шаге схемы используется кубический полином, предельная функция в общем случае не будет полиномиальной. Однако она способна воспроизвести кубический полином. Предположим, что исходный сигнал – отсчеты кубического полинома. В этом случае интерполяционный полином, проходящий через четыре точки, будет тем же самым полиномом, и все вновь порождаемые отсчеты будут принадлежать ему, в пре-

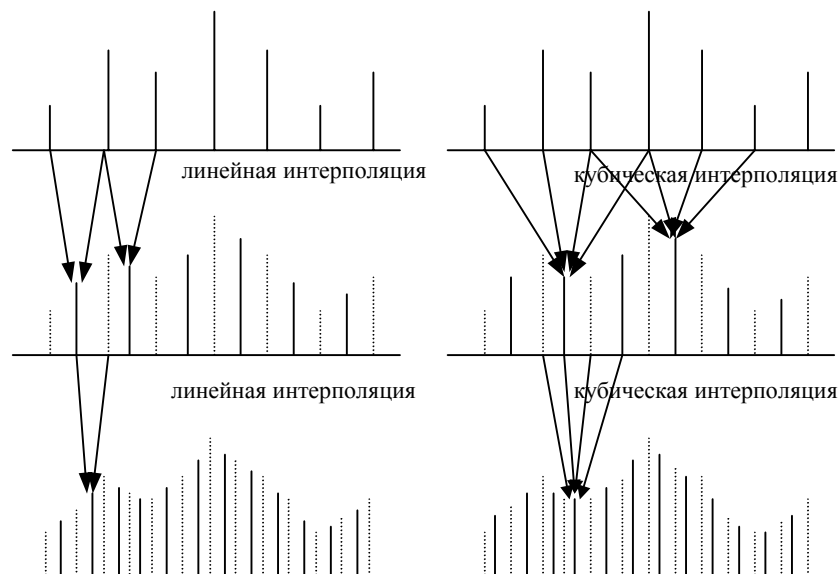


Рис. 6.2. Примеры интерполяционного подразделения.

Линейная и кубическая интерполяции

деле воспроизводя его. Таким образом, используя N отсчетов (N - четное), можно строить полином степени $N-1$. Будем говорить, что схема подразделения имеет порядок N .

Итак, функция предсказания P использует полиномиальную интерполяцию порядка $N-1$ для нахождения предсказываемых значений. Чем выше порядок этой функции, тем лучше аппроксимация коэффициентов γ на основе коэффициентов λ . Это хорошо, если известно, что исходный сигнал может быть представлен полиномом степени $N-1$, так как в этом случае коэффициенты γ будут малы, то есть предсказание будет точным.

Схема интерполяционного подразделения весьма привлекательна с практической точки зрения. В самом деле, нам требуется всего лишь программа, которая бы строила интерполяционный полином для заданных чисел и местоположений. Значение нового отсчета есть просто значение полинома в новой точке. Наиболее подходящим алгоритмом вычисления является алгоритм Невилля. Из процедуры интерполяционного подразделения вовсе не следует, что отсчеты исходного сигнала должны быть равноотстоящими. Это свойство можно использовать для определения масштабирующих функций при неравномерной дискретизации.

Данная интерполяционная схема позволяет легко решить проблему границы для сигналов конечной длины. Например, для кубического полинома у левой границы сигнала можно взять один отсчет слева и три справа. Аналогично и у правой границы. При вычислении новых значений γ около правой границы берется меньше коэффициентов λ справа и больше слева. Если коэффициент γ находится на правой границе, то коэффициентов λ справа не берется вообще. Представим все возможные случаи для $N = 4$.

Случай 1. Возле левой границы: 1 коэффициент λ слева и 3 λ справа.

Случай 2. Вдали от границ: 2 λ слева и 2 λ справа.

Случай 3. Возле правой границы: 3 λ слева и 1 λ справа либо 4 λ слева и 0 λ справа.

На рис.6.3 показан случай, когда коэффициент γ вычисляется возле левой границы для кубического интерполяционного подразделения ($N = 4$).

На рис.6.3(а) и 6.3(б) граница не влияет на вычисление новых значений коэффициентов. На рис.6.3(в) слева имеется лишь один отсчет, поэтому справа берется три отсчета. Отметим, что получается такой же полином, как и на среднем рисунке.

Используя данную интерполяционную схему и алгоритм Невилля, вычисляются коэффициенты, с помощью которых находится аппроксимация функции порядка $N - 1$. Например, если $N = 2$, необходимо два коэффициента для каждого из двух возможных случаев (по одному коэффициенту слева и справа либо 2 слева и 0 справа). Если $N = 4$, требуется четыре коэффициента для каждого из четырех вышеперечисленных случаев. Эти коэффициенты называются коэффициентами фильтра.

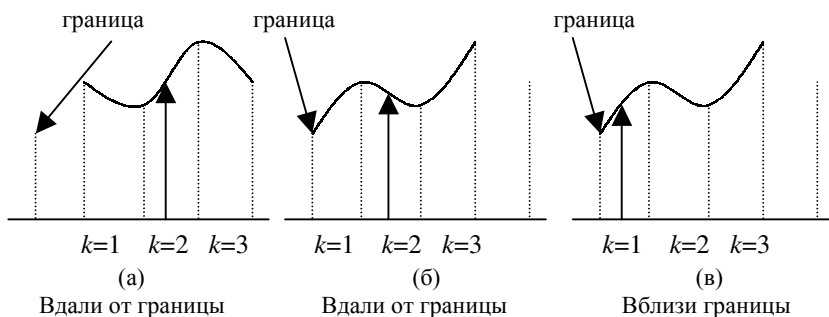


Рис. 6.3. Поведение схемы кубического интерполяционного подразделения (а) вдали от границы; (б) вдали от границы; (в) вблизи границы

Коэффициенты фильтра, вычисляемые для левой границы, равны коэффициентам для правой границы, но записываются в обратном порядке. Так что всего существует $N/2 + 1$ различных случаев (один для середины и $N/2$ для границ интервала).

Пример вычисления коэффициентов кубической интерполяции показан на рис.6.4.

Основная идея вычисления коэффициентов заключается в следующем: $N = 4$, поэтому имеется четыре коэффициента в любом случае. Если мы хотим вычислить, например, c_1 , приравниваем его к единице, а три остальных (c_2, c_3, c_4) – к нулю. Получается полином (в данном случае, кубический). Далее вычисляется функция в интересующих нас точках. Для случая, представленного на рис.6.4(а) это будет «два слева и два справа», а для случая на рис.6.4(б) – «один слева и три справа». В табл.6.1 приведен список коэффициентов фильтров, требующихся для интерполяции при $N = 2, 4$. Одним из свойств коэффициентов фильтров является то, что их сумма равна 1 для любого N .

Этап предсказания, таким образом, реализуется путем поиска в табл.6.2 соответствующих значений для вычисления вейвлет-коэффициентов. Например, если надо предсказать γ для $N = 4$, трех λ слева и одного λ справа, выполняются следующие действия:

$$\gamma_{-j,k} = \lambda_{-j+1,k} - (0.0625 * \lambda_{-j,k-3} - 0.3125 * \lambda_{-j,k-2} + 0.938 * \lambda_{-j,k-1} + 0.3125 * \lambda_{-j,k+1}).$$

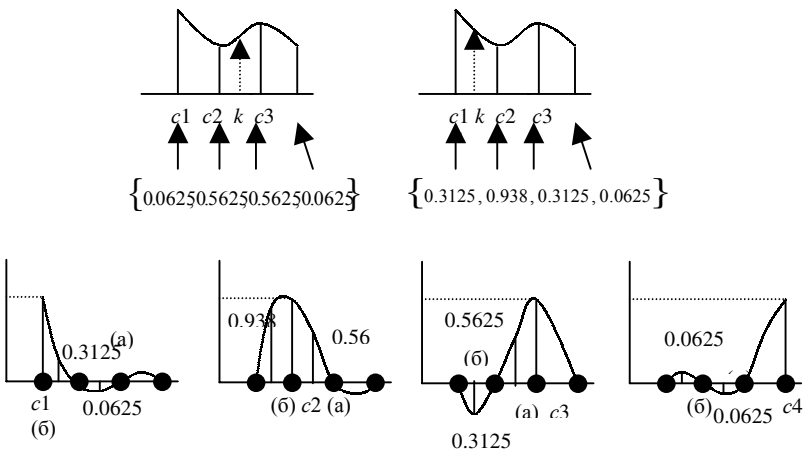


Рис. 6.4. Вычисление коэффициентов для $N = 4$: (а)- в середине интервала; (б) – вблизи границы

Предсказание других производится аналогично, только используются соответствующие значения λ и коэффициенты фильтра.

Таблица 6.1

Коэффициенты фильтра при $N = 2$

Случаи	Коэффициенты				
# λ слева	# λ справа	$k - 3$	$k - 1$	$k + 1$	$k + 3$
0	2			-0.5	1.5
1	1		0.5	0.5	
2	0	1.5	-0.5		

Таблица 6.2

Коэффициенты фильтра при $N = 4$

# слева	# справа	$k - 7$	$k - 5$	$k - 3$	$k - 1$	$k + 1$	$k + 3$	$k + 5$	$k + 7$
0	4					2.1875	-2.1875	1.3125	-0.3125
1	3				0.3125	0.9375	-0.3125	0.0625	
2	2			-0.0625	0.5625	0.5625	-0.0625		
3	1		0.0625	-0.3125	0.9375	0.3125			
4	0	0.3125	1.3125	-2.1875	2.1875				

Итак, мы научились вычислять вейвлет-коэффициенты. Однако нас не устраивает выбор $\{\lambda_{-1,k}\}$. Причина в следующем. Предположим, имеется $2^n + 1$ отсчетов сигнала $\{\lambda_{0,k} \mid 0 \leq k < 2^n\}$. Применим нашу схему (разбиение и предсказание) n раз, в результате чего получим вейвлет-коэффициенты $\{\lambda_{j,k} \mid -n \leq j \leq -1, 0 \leq k < 2^{n+1}\}$ и два коэффициента $\lambda_{-n,0}$ и $\lambda_{-n,1}$. Это - первый и последний далеко удаленные друг от друга отсчеты исходного сигнала. Поэтому возникает значительный элайзинг. Было бы желательно, чтобы некоторые глобальные свойства исходного сигнала сохранялись в уменьшенной версии $\{\lambda_{-j,k}\}$. Например, в случае изображения, желательно, чтобы меньшие изображения $\{\lambda_{-j,k}\}$ имели ту же яркость, что и исходное, то есть то же сред-

нее значение пикселей. Эта проблема решается на третьем этапе - обновлении.

6.4. Этап обновления

На этапе обновления коэффициенты $\lambda_{-1,k}$ «поднимаются» с помощью вейвлет-коэффициентов $\gamma_{-1,k}$. Слово «подъем» по-английски – «lift», отсюда и название схемы – лифтинговая. Идея заключается в том, чтобы найти $\lambda_{-1,k}$, которая сохраняла бы некоторую скалярную характеристику $Q(\cdot)$, например среднее значение:

$$Q(\lambda_{-1,k}) = Q(\lambda_{0,k}). \quad (6.7)$$

Мы могли бы выполнить это, осуществив поиск оператора вычисления $\lambda_{-1,k}$ непосредственно из $\lambda_{0,k}$. Однако в этом случае крайне трудно получить схему обратного преобразования. Поэтому в лифтинговой схеме используются уже вычисленные вейвлет-коэффициенты для обновления последовательности $\lambda_{-1,k}$, с тем чтобы сохранялось $Q(\cdot)$. Другими словами, вводится оператор обновления U , такой что

$$\lambda_{-1,k} = \lambda_{-1,k} + U(\gamma_{-1,k}). \quad (6.8)$$

Таким образом, с использованием уже вычисленных вейвлет-коэффициентов необходимо найти масштабирующую функцию, обеспечивающую сохранение некоторых свойств λ на всех уровнях декомпозиции. Один из возможных путей следующий. Приравняем все $\{\lambda_{0,k}\}$ к нулю, кроме $\lambda_{0,0} = 1$. Затем выполняем интерполяционное подразделение бесконечное число раз. В результате получится масштабирующая функция $\phi(x)$, с помощью которой можно создать вещественный вейвлет, который сохраняет необходимые свойства исходного сигнала. Эта функция будет иметь порядок, зависящий от некоторого (четного) числа \tilde{N} , которое необязательно равно N . \tilde{N} называется числом действительных нулевых моментов. Чем выше порядок этой функции, тем меньше будет в получившемся преобразовании эффект элайзинга.

Основными свойствами, которые мы хотим сохранить на каждом уровне, являются моменты вейвлет-функции ψ . Из свойств вейвлет-функций из-

вестно, что интеграл от этой функции, взятый по всей числовой оси, равен нулю. Это истинно и для высших моментов. Так что

$$\int \psi(x) dx = 0, \quad \int x \psi(x) dx = 0, \quad \int x^2 \psi(x) dx = 0, \dots, \int x^{\tilde{N}-1} \psi(x) dx = 0. \quad (6.9)$$

Нам необходимо сохранить до $\tilde{N} - 1$ моментов последовательности λ на каждом уровне. Далее эта информация используется для определения того, какая часть каждого коэффициента γ требуется для обновления каждого значения λ . Эти обновленные значения называются коэффициентами лифтинга. Перед началом вычисления лифтинговых коэффициентов производится инициализация информацией о моментах коэффициентов на первом уровне. Интеграл приравнивается к единице для всех коэффициентов (так как сумма коэффициентов фильтра равна единице).

После инициализации моментов выполняются следующие шаги.

1. Проверяется вклад λ в предсказание каждого γ . (На основе коэффициентов фильтра, найденных на предыдущем этапе).

2. Для каждого λ на текущем уровне обновляются моменты согласно следующему выражению:

$$m_{j,k} = m_{j,k} + f_j * m_{l,k}, \quad (6.10)$$

где j - индекс коэффициента λ ; $f(j)$ - соответствующий коэффициент фильтра ($0 < j \leq N$); k - обновляемый момент ($0 < k \leq \tilde{N}$) и l - индекс коэффициента γ .

3. Так как моменты на всех уровнях должны быть равны нулю, можно построить линейную систему уравнений для нахождения лифтинговых коэффициентов для каждого γ . Система строится следующим образом:

а) текущий γ приравнивается к 1, остальные γ - к нулю;

б) выполняется один шаг обратного преобразования для определения вклада данного γ в λ , который обновляет его. Получается следующая система уравнений:

$$\begin{aligned} c_1 * m_{l_{0,0}} + c_2 * m_{l_{1,0}} + c_3 * m_{l_{2,0}} + \dots + c_{\tilde{N}} * m_{l_{N,0}} &= m_{g_j,0} \\ c_1 * m_{l_{0,1}} + c_2 * m_{l_{1,1}} + c_3 * m_{l_{2,1}} + \dots + c_{\tilde{N}} * m_{l_{N,1}} &= m_{g_j,1} \\ c_1 * m_{l_{0,2}} + c_2 * m_{l_{1,2}} + c_3 * m_{l_{2,2}} + \dots + c_{\tilde{N}} * m_{l_{N,2}} &= m_{g_j,2} \\ &\dots \dots \dots \\ c_1 * m_{l_{0,\tilde{N}}} + c_2 * m_{l_{1,\tilde{N}}} + c_3 * m_{l_{2,\tilde{N}}} + \dots + c_{\tilde{N}} * m_{l_{N,\tilde{N}}} &= m_{g_j,\tilde{N}}, \end{aligned} \quad (6.11)$$

где c_i ($0 < i \leq \tilde{N}$) - искомые коэффициенты лифтинга, l_i ($0 < i \leq N$) - индекс коэффициента λ на текущем уровне; g_j ($0 < j < \text{длина_сигнала}$) - индекс коэффициента γ , приравненного к единице;

в) в результате решения получившейся $\tilde{N} \times \tilde{N}$ системы находятся коэффициенты лифтинга.

Подобная линейная система строится для каждого коэффициента γ , чтобы найти соответствующие коэффициенты лифтинга (\tilde{N} для каждого γ на каждом уровне). Коэффициенты лифтинга используются в операторе обновления λ U до перехода к следующему уровню. Обновление λ происходит следующим образом. Для данного вейвлет-коэффициента γ находим коэффициенты лифтинга, например (a, b) при $\tilde{N} = 2$. Находим λ , на которую влияет этот γ , например $\lambda_{-j, k-1}$ и $\lambda_{-j, k+1}$. Теперь

$$\begin{aligned}\lambda_{-j, k-1} &= \lambda_{-j, k-1} + a * \gamma_{-j, k}, \\ \lambda_{-j, k+1} &= \lambda_{-j, k+1} + b * \gamma_{-j, k}.\end{aligned}\tag{6.12}$$

Далее переходим к следующему γ , и процесс повторяется.

Рассмотрим этапы разбиения, предсказания и обновления на примере одномерного сигнала длиной $L = 8, N = 2$ и $\tilde{N} = 2$. Вначале рассмотрим разбиение и предсказание:

$$\begin{array}{ccccccccc}\lambda_1 & \lambda_2 & \lambda_3 & \lambda_4 & \lambda_5 & \lambda_6 & \lambda_7 & \lambda_8 \\ \gamma_1 & & & \gamma_2 & & \gamma_3 & & \gamma_4\end{array},$$

где γ_1 использует λ_1 и λ_3 для предсказания. Аналогично и для остальных γ . Второй этап, обновление, выполняется согласно выражениям (6.10) и (6.11). В этом примере получаются следующие пары (a, b) лифтинговых коэффициентов:

$$\begin{array}{cccc}\gamma_1 & \gamma_2 & \gamma_3 & \gamma_4 \\ (2/5, 1/5) & (0, 2/3) & (4/15, 1/5) & (-2/15, 2/5), \\ \lambda_1 & \lambda_3 & \lambda_5 & \lambda_7.\end{array}$$

Коэффициент λ_1 использует a от γ_1 для обновления. Аналогично λ_3 использует b от γ_1 и a от γ_2 и т.д.

После разбиения и предсказания на следующем уровне получаем коэффициенты

$$\begin{array}{cccc} \lambda_1 & \lambda_3 & \lambda_5 & \lambda_7 \\ & \gamma_1 & & \gamma_2 \end{array} .$$

Обновление происходит следующим образом:

$$\begin{array}{cc} \gamma_1 & \gamma_2 \\ (1/2, 0.214286) & (-1/3, 0.47619) \\ \lambda_1 & \lambda_5 \end{array} .$$

Важно отметить, что для достаточно длинных сигналов лифтинговые коэффициенты становятся равными $1/4, 1/4$ для всех λ вдали от границ. Используя эти значения, можно обновлять коэффициенты следующим образом:

$$\lambda_{-1,k} = \lambda_{-1,k} + 1/4 * \gamma_{-1,k-1} + 1/4 * \gamma_{-1,k} . \quad (6.13)$$

Объединение трех этапов лифтинга, представленных на диаграмме рис. 6.1, дает нам алгоритм одномерного быстрого лифтингового вейвлет-преобразования:

$$\text{от } j = -1 \text{ до } -n : \left\{ \begin{array}{l} \{\lambda_{j,k}, \gamma_{j,k}\} = S(\lambda_{j+1,k}) \\ \gamma_{j,k} - = P(\lambda_{j,k}) \\ \lambda_{j,k} + = U(\gamma_{j,k}) \end{array} \right. \quad (6.14)$$

Теперь можно показать одно замечательное свойство лифтинга: для реализации обратного преобразования достаточно в алгоритме прямого преобразования поменять местами знаки «+» и «-». Таким образом, алгоритм обратного преобразования можно записать в виде

$$\text{от } j = -n \text{ до } -1 : \left\{ \begin{array}{l} \lambda_{j,k} - = U(\gamma_{j,k}) \\ \gamma_{j,k} + = P(\lambda_{j,k}) \\ \{\lambda_{j+1,k}\} = \text{Объединить}(\lambda_{j,k}, \gamma_{j,k}) \end{array} \right. \quad (6.15)$$

Для подсчета числа операций, требующихся для данного преобразования, должны учитываться три фактора: длина сигнала (L), число дуальных нулевых моментов (N) и число первичных нулевых моментов (\tilde{N}). Может быть показано, что всего требуется

$$n = \left\lceil \log_2 \left(\frac{L-1}{N_{\max} - 1} \right) \right\rceil \quad (6.16)$$

итераций, где $N_{\max} = \max(N, \tilde{N})$. Из (6.16) видно, что длина сигнала не обязательно должна быть кратной степени двойки. Схема интерполяционного подразделения гарантирует правильную обработку границ сигнала в любом случае.

Перенос алгоритма лифтинга на случай двумерных сигналов заключается в простом выполнении преобразования по строкам и столбцам, так как преобразование разделимое. Равенство (6.16) при этом изменяется лишь в том, что $L = \max(L_x, L_y)$.

Если коэффициенты фильтра равны $(1/2, 1/2)$, а коэффициенты лифтинга - $(1/4, 1/4)$, то мы получаем биортогональное вейвлет-преобразование Коэна-Добеши-Фово ($N = 2, \tilde{N} = 2$). Этот простой пример уже показывает, как лифтинг может ускорить выполнение вычислений. Классически коэффициенты $\{\lambda_{-1,k}\}$ находятся как свертка коэффициентов $\{\lambda_{0,k}\}$ с фильтром $\tilde{h} = \{-1/8, 1/4, 3/4, 1/4, -1/8\}$. Для этого требуется шесть операций на коэффициент, тогда как в случае лифтинга затраты составляют три операции.

Варьируя все три этапа лифтинга, можно получить семейство биортогональных вейвлетов:

1. Разбиение. В качестве начального разбиения возможен другой выбор, чем вейвлет Лэйзи. Классическим примером является вейвлет Хаара.

2. Предсказание. Этап предсказания определяет число нулевых моментов дуального вейвлета (N). Другими словами, если исходный сигнал описывается полиномом степени меньше, чем N , все вейвлет-коэффициенты будут равны нулю. Как было показано, схемы с порядком выше $N = 2$ легко получить, учитывая больше соседних отсчетов.

3. Обновление. Этап обновления определяет число нулевых моментов первичного вейвлета (\tilde{N}). Иначе говоря, преобразование сохраняет первые \tilde{N} моментов последовательностей $\lambda_{j,k}$. И снова схемы с порядком выше $N = 2$ получаются за счет учета большего количества соседних отсчетов. В

некоторых случаях этап обновления может быть опущен, например в случае если на этапе разбиения уже получается вейвлет с нулевыми моментами (например, Хаара). Путем выполнения более одного этапа обновления могут создаваться разные семейства вейвлетов, уже не биортогональные.

Преимуществом лифтинговой схемы является также и выполнение вычислений без расходования дополнительной памяти (рис.6.5). Предположим, что исходный сигнал сохранен в массиве $v[k]$. Каждый коэффициент $\lambda_{j,k}$ или $\gamma_{j,k}$ сохраняется в позиции $v[2^{-j}k]$. Затем применяется вейвлет Лэйзи. Все остальные шаги заключаются в выполнении операций $+=$ или $-=$. Иначе говоря, после предсказания коэффициентов γ коэффициенты λ могут заменяться ими по тому же адресу памяти. На этапе обновления обновленные λ заменяют собой прежние значения.

Таким образом, лифтинговая схема является эффективным алгоритмом вычисления вейвлет-преобразования. Кроме того, она позволяет получать так называемые вейвлеты второго поколения, обладающие рядом дополнительных свойств. В силу ограниченности места в настоящей книге теория вейвлетов второго поколения не описывается. Интересующимся можно порекомендовать статьи В.Свелденса (см. список интернет-ссылок).

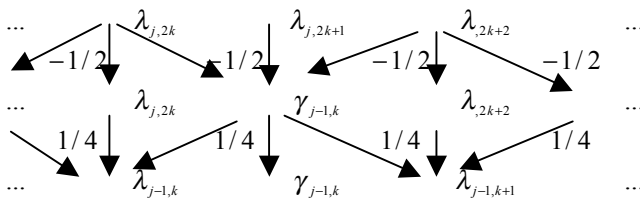


Рис. 6.5. Лифтинговая схема ($N = 2, \tilde{N} = 2$). Разбиение, вычисление вейвлет-коэффициентов $\gamma_{-1,m}$, как степени отклонения сигнала от линейного и использование их для обновления $\lambda_{-j-1,k}$

ЦЕЛОЧИСЛЕННОЕ ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЕ

В данной главе будут рассмотрены методы получения целочисленных вейвлет-коэффициентов изображения. Эти методы могут быть применены для сжатия изображения как без потерь, так и с потерями. В основе рассматриваемых методов лежит некоторая модификация вейвлет-преобразования, позволяющая производить все вычисления в целочисленном виде. Полученное преобразование не является, строго говоря, вейвлет-преобразованием, но обладает всеми его свойствами. Теоретически при вейвлет-преобразовании потери информации не происходит. Однако при реализации возникают неизбежные ошибки округления вейвлет-коэффициентов. Вместе с тем, в некоторых приложениях обработки изображений полная обратимость преобразования является важной. Целочисленное вейвлет-преобразование позволяет достичь полного контроля над точностью вычислений. Поэтому оно получило название обратимого вейвлет-преобразования. Кроме того, целочисленность вычислений ускоряет выполнение алгоритмов на компьютерах.

7.1. Целочисленные вейвлет-преобразования

Рассмотрим два примера, поясняющие обсуждаемые далее методы. Для простоты все выкладки производятся для одного уровня разложения и для одномерного сигнала четной длины. Пусть $\{c_n^0\}_{n=0}^{N-1}$ - исходный сигнал, где верхний индекс показывает уровень разложения (0), нижний – конкретную точку сигнала. Пусть $\{c_n^1\}_{n=0}^{N_1-1}$ и $\{d_n^1\}_{n=0}^{N_1-1}$ - составляющие его разложения на первом уровне (низкочастотная и высокочастотная части, соответственно). Здесь $N_1 = N/2$.

Пример1. Целочисленное вычисление вейвлет-преобразования (2,2).

Это преобразование эквивалентно вейвлет-преобразованию Хаара, использующему следующие фильтры декомпозиции:

$$\tilde{h}_0 = \tilde{h}_1 = \tilde{g}_0 = -\tilde{g}_1 = 1/2.$$

Вычисление ведется следующим образом:

$$d_k^1 = c_{2k}^0 - c_{2k+1}^0, \quad k = 0, \dots, N_1 - 1, \quad (7.1)$$

$$c_k^1 = \text{int}\left(\frac{d_k^1}{2}\right) + c_{2k+1}^0, \quad k = 0, \dots, N_1 - 2, \quad (7.2)$$

$$c_{N_1-1}^1 = \text{int}\left(\frac{d_{N_1-1}^1}{2} + c_{N_1-1}^0\right). \quad (7.3)$$

В выражениях (7.2), (7.3) int означает операцию округления. Таким образом, все элементы $\{c_n^1\}_{n=0}^{N_1-1}$ и $\{d_n^1\}_{n=0}^{N_1-1}$ будут целыми числами. Из (7.1)-(7.3) легко получить алгоритм реконструкции:

$$c_{2k+1}^0 = c_k^1 - \text{int}\left(\frac{d_k^1}{2}\right), \quad k = 0, \dots, N_1 - 1, \quad (7.4)$$

$$c_{2k}^0 = d_k^1 + c_{2k+1}^0, \quad k = 0, \dots, N_1 - 1. \quad (7.5)$$

Пример 2. Вейвлет-преобразование Лэйзи.

Вейвлет-преобразование Лэйзи заключается в простом разбиении входного сигнала на четную и нечетную части. На этапах декомпозиции и реконструкции используются одни и те же формулы:

$$\begin{aligned} c_k^1 &= c_{2k}^0, \quad k = 0, \dots, N_1 - 1, \\ d_k^1 &= c_{2k+1}^0, \quad k = 0, \dots, N_1 - 1. \end{aligned} \quad (7.6)$$

Преобразования, используемые в этих примерах, не подходят для кодирования изображений. Однако на их основе могут быть получены значительно более эффективные преобразования. Они могут рассматриваться как стартовая точка для получения алгоритмов целочисленного обратимого вейвлет-преобразования.

Отметим интересное свойство вышеприведенных преобразований. Оно заключается в том, что если пиксели изображения представляются некоторым числом бит, то такое же число бит может быть использовано в компьютере для представления значений вейвлет-коэффициентов. Это следует из особенностей дополнительного кода, в котором представляются числа в компьютере. Данное свойство преобразования получило название свойства сохранения точности.

Известно, что значения высокочастотных вейвлет-коэффициентов малы. Это позволяет сохранять точность во время вычисления коэффициентов. Рассмотрим целочисленные вычисления, выполняемые компьютером. Большинство компьютеров использует дополнительный код. Пусть необходимо найти разность двух целых чисел $c = b - a$ и выполнить обратное вычисление $a = b - c$. Вычисления на компьютере выполняются следующим образом:

$$c_m = \begin{cases} b-a, & \text{при } -2^{q-1} \leq b-a < 2^{q-1} - 1, \\ -2^q + b-a, & \text{при } b-a \geq 2^{q-1}, \\ 2^q + b-a, & \text{при } b-a < -2^{q-1}, \end{cases}$$

и обратная величина находится, как

$$a_m = \begin{cases} b-c_m, & \text{при } -2^{q-1} \leq b-c_m < 2^{q-1} - 1, \\ -2^q + b-c_m, & \text{при } b-c_m \geq 2^{q-1}, \\ 2^q + b-c_m, & \text{при } b-c_m < -2^{q-1}, \end{cases}$$

где индекс m означает внутреннее представление, а числа a, b, c находятся в пределах $[-2^{q-1}, 2^{q-1} - 1]$. Если c_m оказывается вне диапазона, его внутреннее представление может не совпадать с внешним значением c . Например, пусть $b = 2$ (0000010) и $a = -127$ (100000001). Тогда c_m имеет следующее внутреннее представление: (100000001) при $q = 8$. С другой стороны, при $c_m = -127$ a_m будет равно a .

Пример 3. Целочисленное вычисление вейвлет-преобразования (2,6).

Данное преобразование эквивалентно использованию следующих фильтров анализа:

$$\tilde{h}_n = \{0, 0, 1/2, 1/2, 0, 0\}, \quad \tilde{g}_n = \{-1/16, -1/16, 1/2, -1/2, 1/16, 1/16\}.$$

Декомпозиция выполняется аналогично примеру 1 с добавлением еще одного шага. Вначале производятся вычисления по формулам (7.1)-(7.3). Вместо d_0^1 в данных формулах теперь используется обозначение $d_0^{1,0}$. Затем производится изменение высокочастотных коэффициентов по формулам:

$$\begin{cases} d_0^1 = \text{int}\left(\frac{c_0^1 - c_1^1}{4}\right) - d_0^{1,0}, \\ d_k^1 = \text{int}\left(\frac{c_{k-1}^1 - c_{k+1}^1}{4}\right) - d_k^{1,0}, \quad k = 1, \dots, N_1 - 2, \end{cases} \quad (7.7)$$

$$d_{N_1-1}^1 = \text{int}\left(\frac{c_{N_1-2}^1 - c_{N_1-1}^1}{4}\right) - d_{N_1-1}^{1,0}. \quad (7.8)$$

Алгоритм реконструкции аналогичен алгоритму декомпозиции. Он выполняется в «обратном» порядке:

$$\begin{cases} d_0^{1,0} = \text{int}\left(\frac{c_0^1 - c_1^1}{4}\right) - d_0^1, \\ d_k^{1,0} = \text{int}\left(\frac{c_{k-1}^1 - c_{k+1}^1}{4}\right) - d_k^1, \quad k = 1, \dots, N_1 - 2, \end{cases} \quad (7.9)$$

$$d_{N_1-1}^{1,0} = \text{int}\left(\frac{c_{N_1-2}^1 - c_{N_1-1}^1}{4}\right) - d_{N_1-1}^1 \quad (7.10)$$

и, далее, по формулам (7.4)-(7.5) с заменой в них d_0^1 на $d_0^{1,0}$.

Пример 4. Целочисленное вычисление вейвлет –преобразования (1,3).

Это нелинейное преобразование является разновидностью преобразования, использующего биортогональную пару фильтров: $\tilde{h}_n = \{1, 0, 0\}$, $\tilde{g}_n = \{1/4, -1/2, 1/4\}$. Вычисления начинаются с вейвлета Лэйзи (7.6) с последующим изменением высокочастотных коэффициентов:

$$\begin{cases} d_k^1 = \text{int}\left(\frac{c_k^1 + c_{k+1}^1}{2}\right) - d_k^{1,0}, \quad k = 0, \dots, N_1 - 2, \\ d_{N_1-1}^1 = c_{N_1-1}^1 - d_{N_1-1}^{1,0}. \end{cases} \quad (7.11)$$

Реконструкция выполняется следующим образом:

$$c_{2k}^0 = c_k^1, \quad k = 0, \dots, N_1 - 1, \quad (7.12)$$

$$\begin{cases} c_{2k+1}^0 = \text{int}\left(\frac{c_{2k}^0 + c_{2k+2}^0}{2}\right) - d_k^1, \quad k = 0, \dots, N_1 - 2, \\ c_{N_1-1}^0 = c_{N_1-2}^0 - d_{N_1-1}^1. \end{cases} \quad (7.13)$$

Пример 5. Целочисленное вычисление вейвлет –преобразования (5,3).

Это преобразование также является разновидностью биортогонального преобразования и использует следующую пару фильтров:

$$\tilde{h}_n = \{-1/8, 1/4, 3/4, 1/4, -1/8\}, \quad \tilde{g}_n = \{1/4, -1/2, 1/4, 0, 0\}.$$

Декомпозиция производится следующим образом:

$$c_k^{1,0} = c_{2k}^0, \quad k = 0, \dots, N_1 - 1, \quad (7.14)$$

$$\begin{cases} d_k^1 = \text{int}\left(\frac{c_{2k}^0 + c_{2k+2}^0}{2}\right) - c_{2k+1}^0, & k = 0, \dots, N_1 - 2, \\ d_{N_1-1}^1 = c_{N_1-2}^0 - c_{N_1-1}^0, \end{cases} \quad (7.15)$$

$$\begin{cases} c_0^1 = c_0^{1,0} - \text{int}\left(\frac{d_0^1}{2}\right), \\ c_k^1 = c_k^{1,0} - \text{int}\left(\frac{d_{k-1}^1 + d_k^1}{4}\right), & k = 0, \dots, N_1 - 2, \\ c_{N_1-1}^1 = c_{N_1-2}^{1,0} - \text{int}\left(\frac{d_{N_1-2}^1 + d_{N_1-1}^1}{4}\right) \end{cases} \quad (7.16)$$

Реконструкция осуществляется по следующим формулам :

$$c_0^0 = c_0^1 + \text{int}\left(\frac{d_0^1}{2}\right), \quad (7.17)$$

$$c_{2k}^0 = c_k^1 + \text{int}\left(\frac{d_{k-1}^1 + d_k^1}{4}\right), \quad k = 1, \dots, N_1 - 2, \quad (7.18)$$

$$c_{N_1-2}^0 = c_{N_1-1}^1 + \text{int}\left(\frac{d_{N_1-2}^1 + d_{N_1-1}^1}{4}\right), \quad (7.19)$$

$$c_{2k+1}^0 = \text{int}\left(\frac{c_{2k}^0 + c_{2k+2}^0}{2}\right) - d_k^1, \quad k = 0, \dots, N_1 - 2, \quad (7.20)$$

$$c_{N_1-1}^0 = c_{N_1-2}^0 - d_{N_1-2}^1.$$

Из трех последних примеров видно, что целочисленное вейвлет-преобразование можно выполнять следующим образом. Вначале выбрать некоторое простое преобразование. Затем определенным образом изменить высокочастотные коэффициенты для получения целочисленного преобразования, эффективного для кодирования изображений. Вопрос заключается в том, каким образом модифицировать коэффициенты. Ответ на этот вопрос дается в следующих двух разделах.

7.2. Лифтинговая схема и целочисленная биортогональная фильтрация

Лифтинговая схема, рассмотренная в главе 6, позволяет конструировать биортогональные вейвлеты с компактным носителем. С небольшими изменениями она может быть использована для построения целочисленного нелинейного квазибиортогонального вейвлет-преобразования.

Набор фильтров $h, \tilde{h}, g, \tilde{g}$ называется биортогональным, если удовлетворяется следующее условие:

$$\tilde{m}(\omega)\overline{m^T(\omega)}=1,$$

где

$$m(\omega)=\begin{bmatrix} h(\omega) & h(\omega+\pi) \\ g(\omega) & g(\omega+\pi) \end{bmatrix}$$

и

$$h(\omega)=\sum_k h_k e^{-k\omega}, \quad g(\omega)=\sum_k g_k e^{-k\omega}.$$

Аналогично и для $\tilde{m}(\omega), \tilde{h}(\omega), \tilde{g}(\omega)$.

В.Свелденсом доказано следующее утверждение. Пусть имеется набор биортогональных КИХ-фильтров $\{h, \tilde{h}^0, g^0, \tilde{g}\}$. Новый набор $\{h, \tilde{h}, g, \tilde{g}\}$ можно получить по формулам:

$$\begin{aligned} \tilde{h}(\omega) &= \tilde{h}^0(\omega) + \tilde{g}(\omega)\overline{s(2\omega)}, \\ g(\omega) &= g^0(\omega) - h(\omega)s(2\omega). \end{aligned} \tag{7.21}$$

Аналогично, если имеется набор фильтров $\{h^0, \tilde{h}, g, \tilde{g}^0\}$, новый набор можно получить по формулам:

$$\begin{aligned} h(\omega) &= h^0(\omega) + g(\omega)\overline{\tilde{s}(2\omega)}, \\ \tilde{g}(\omega) &= \tilde{g}^0(\omega) - h(\omega)\tilde{s}(2\omega), \end{aligned} \tag{7.22}$$

где $s(\omega)$ - тригонометрический полином, а соответствующий фильтр S является КИХ-фильтром. Выражения (7.21)-(7.22) можно переписать следующим образом:

$$\begin{aligned} \tilde{h}_k &= \tilde{h}_k^0 + \sum_l \tilde{g}_{k+2l} s_l & g_k &= g_k^0 - \sum_l h_{k-2l} s_l, \\ h_k &= h_k^0 + \sum_l g_{k+2l} \tilde{s}_l, & \tilde{g}_k &= \tilde{g}_k^0 - \sum_l \tilde{h}_{k-2l} \tilde{s}_l. \end{aligned} \quad (7.23)$$

Пусть $\{c_n^0\}$ - исходный сигнал, $\{c_n^1\}$ и $\{d_n^1\}$ - низкочастотная и высокочастотная составляющие его разложения, получаемые с использованием фильтров $\{h, \tilde{h}, g, \tilde{g}\}$. Тогда алгоритм декомпозиции можно записать в виде

$$\begin{cases} c_k^1 = \alpha_c \sum_n c_n^0 \tilde{h}_{n-2k}, \\ d_k^1 = \alpha_d \sum_n c_n^0 \tilde{g}_{n-2k}. \end{cases} \quad (7.24)$$

Соответствующий алгоритм реконструкции:

$$c_n^0 = 2 \sum_k \left(\frac{c_k^1 h_{n-2k}}{\alpha_c} + \frac{d_k^1 g_{n-2k}}{\alpha_d} \right), \quad (7.25)$$

где параметры α_c и α_d - положительные константы, такие что $\alpha_c * \alpha_d = 2$. Например, для биортогональной декомпозиции $\alpha_c = \alpha_d = \sqrt{2}$; для примеров 1-5 $\alpha_c = 1$, $\alpha_d = 2$.

Если набор фильтров $\{h, \tilde{h}, g, \tilde{g}\}$ был получен из $\{h, \tilde{h}^0, g^0, \tilde{g}\}$ по формулам (7.23), декомпозиция представима в виде

$$\begin{cases} c_k^{1,0} = \alpha_c \sum_n c_n^0 \tilde{h}_{n-2k}^0, \\ d_k^1 = \alpha_d \sum_n c_n^0 \tilde{g}_{n-2k}, \end{cases} \quad (7.26)$$

$$c_k^1 = c_k^{1,0} + \frac{\alpha_c}{\alpha_d} \sum_l d_{k-l}^1 s_l. \quad (7.27)$$

Соответствующий алгоритм реконструкции записывается в виде

$$c_k^{1,0} = c_k^1 - \frac{\alpha_c}{\alpha_d} \sum_l d_{k+l}^1 s_l, \quad (7.28)$$

$$c_n^0 = 2 \sum_k \left(\frac{c_k^{1,0} h_{n-2k}^0}{\alpha_c} + \frac{d_k^1 g_{n-2k}^0}{\alpha_d} \right). \quad (7.29)$$

Здесь равенства (7.26) и (7.29) – прямое и обратное биортогональные вейвлет-преобразования. Выражения (7.27) и (7.28) есть дополнительные формулы, обеспечивающие целочисленность преобразования.

Аналогично, если набор фильтров $\{h, \tilde{h}, g, \tilde{g}\}$ был получен из $\{h^0, \tilde{h}^0, g^0, \tilde{g}^0\}$ по формулам (7.23), декомпозиция принимает вид

$$\begin{cases} c_k^1 = \alpha_c \sum_n c_n^0 \tilde{h}_{n-2k}^0, \\ d_k^{1,0} = \alpha_d \sum_n c_n^0 \tilde{g}_{n-2k}^0, \end{cases} \quad (7.30)$$

$$d_k^1 = d_k^{1,0} - \frac{\alpha_c}{\alpha_d} \sum_l c_{k+l}^1 s_l. \quad (7.31)$$

Соответствующая ей реконструкция:

$$d_k^{1,0} = d_k^1 + \frac{\alpha_c}{\alpha_d} \sum_l c_{k+l}^1 s_l, \quad (7.32)$$

$$c_n^0 = 2 \sum_k \left(\frac{c_k^{1,0} h_{n-2k}^0}{\alpha_c} + \frac{d_k^1 g_{n-2k}^0}{\alpha_d} \right). \quad (7.33)$$

Возникающая проблема обработки границы изображения была рассмотрена в разделе 6.4.

Предположим, что биортогональные фильтры $\{h, \tilde{h}, g, \tilde{g}\}$ получены из первоначальных фильтров $\{h, \tilde{h}^0, g^0, \tilde{g}\}$ по формулам (7.21) и (7.22). Пусть декомпозиция первоначальными фильтрами выполнялась целочисленно. Тогда можно доказать, что имеется возможность конструирования целочисленного преобразования фильтрами $\{h, \tilde{h}, g, \tilde{g}\}$. Данное преобразование будет

«близко» первоначальному. Близость означает, что разница между двумя схемами декомпозиции состоит лишь в наличии некоторой ошибки округления. Эта ошибка устраняется за счет целочисленной схемы реконструкции.

Если $\{c_k^{1,0}\}$ и $\{d_k^1\}$ являются целыми после (7.26), $\{c_k^1\}$ может быть вычислена вместо (7.27) следующим образом:

$$c_k^1 = c_k^{1,0} + \text{int} \left(\frac{\alpha_c}{\alpha_d} \sum_l d_{k-l}^1 s_l \right). \quad (7.34)$$

Видно, что (7.34) «близко» к (7.27), и точная реконструкция может быть обеспечена применением формулы (7.33) и следующей формулы:

$$c_k^{1,0} = c_k^1 - \text{int} \left(\frac{\alpha_c}{\alpha_d} \sum_l d_{k-l}^1 s_l \right). \quad (7.35)$$

Аналогичным образом целочисленное преобразование выполняется, если набор фильтров $\{h, \tilde{h}, g, \tilde{g}\}$ получен из первоначального набора $\{h^0, \tilde{h}^0, g, \tilde{g}^0\}$.

Отметим, что большинство стандартных биортогональных вейвлет-преобразований не может быть выполнено непосредственно в целочисленном виде. Однако за счет соответствующего выбора параметров α_c и α_d и некоторого изменения алгоритма получаются различные вариации исходного преобразования. Кроме того, необходимо правильно выбирать параметр $\{s_l\}$ для того, чтобы алгоритм выполнялся только посредством сдвигов и сложений.

Отметим также следующее. Если набор фильтров $\{h, \tilde{h}, g, \tilde{g}\}$ получен из набора $\{h^0, \tilde{h}^0, g, \tilde{g}^0\}$ посредством лифтинговой схемы, а тот, в свою очередь, из набора $\{h^0, \tilde{h}^0, g^0, \tilde{g}^0\}$, то имеется возможность получения целочисленного преобразования, «близкого» к исходному.

7.3. Метод коррекции ошибок для получения целочисленного вейвлет-преобразования

В данном разделе рассмотрим другой метод целочисленного вейвлет-преобразования. Он основан на так называемом $S+P$ преобразовании, предложенном А.Саидом и В.Перельманом. Конструирование биортогональных вейвлетов при помощи лифтинговой схемы может рассматриваться как частный случай метода коррекции.

Предположим, имеется некоторое простое целочисленное вейвлет-преобразование, например рассмотренное в примерах 1-3. Декомпозиция и реконструкция запишется в общем случае следующим образом:

$$\begin{aligned} \text{декомпозиция:} \quad c_k^{1,0} &= df_c(\{c_n^0\}) \\ d_k^{1,0} &= df_d(\{c_n^0\}), \end{aligned} \quad (7.36)$$

$$\text{реконструкция:} \quad c_n^0 = rf(\{c_k^{1,0}\} \{d_k^{1,0}\}). \quad (7.37)$$

В общем случае вышеприведенная декомпозиция может не дать хорошего результата. Например, возможно появление эффектов наложения спектра (элайзинга) либо преобразование не приведет к локализации энергии в низкочастотной субполосе. Поэтому нужна определенная коррекция высокочастотных либо низкочастотных коэффициентов. Существует несколько путей ее реализации. Однако с учетом требования целочисленной реализации необходимо использовать метод коррекции.

Например, выполним коррекцию высокочастотных коэффициентов:

$$d_k^1 = d_k^{1,0} - \text{int}(dc_k^1), \quad k = \dots, 0, 1, 2, \dots, \quad (7.38)$$

где dc_k^1 есть коэффициент коррекции для d_k^1 :

$$dc_k^1 = \sum_{i=S_0}^{S_1} \sigma_i c_{k+i}^{1,0} + \sum_{j=1}^T \tau_j d_{k+j}^{1,0}, \quad k = \dots, 0, 1, 2, \dots, \quad (7.39)$$

здесь $\{\sigma_i\}_{i=S_0}^{S_1}$ и $\{\tau_j\}_{j=1}^T$ - варьируемые параметры, выбираемые для достижения требуемых характеристик преобразования. Разработаны методы нахождения этих параметров, которые не будут рассматриваться в данной книге. Скажем только, что они должны быть рациональными числами, кратными степени 2, для обеспечения целочисленности преобразования. Из (7.36), (7.38)-(7.39) видно, что алгоритм декомпозиции, обеспечивающий полное восстановление после реконструкции (7.37), представим в виде:

$$d_k^{1,0} = d_k^1 + \text{int}(dc_k^1), \quad k = \dots, m, m-1, m-2, \dots \quad (7.40)$$

Как отмечалось выше, лифтинговая схема есть частный случай метода коррекции. Примеры 3-5 могут также рассматриваться как реализации этого метода. Рассмотрим теперь пример, в котором не используется лифтинговая

схема и который не имеет отношения к биортогональным фильтрам с компактной областью определения.

Пример 6. $S + P$ преобразование.

Данное преобразование аналогично использованию следующих фильтров анализа:

$$\tilde{h}_n = \{0, 0, 1/2, 1/2, 0, 0\}, \quad \tilde{g}_n = \{-1/16, -1/16, 15/32, -17/32, 7/32, -1/32\}.$$

При этом фильтры синтеза не имеют компактной области определения. Тем не менее $S + P$ преобразование может быть выполнено следующим образом:

1. Выполним декомпозицию, как в примере 1 (формулы (7.1)-(7.3)). При этом вместо d_k^1 используем $d_k^{1,0}$.

2. Шаг коррекции. Определим:

$$S_0 = -1, \quad S_1 = 1, \quad T = 1, \quad \sigma_{-1} = -1/4, \quad \sigma_0 = -1/8, \quad \sigma_1 = 1/8, \quad \tau_1 = 1/4$$

и вычислим:

$$\left\{ \begin{array}{l} d_0^1 = d_0^{1,0} - \text{int} \left(\frac{c_0^1 - c_1^1}{4} \right) \\ d_k^1 = d_k^{1,0} - \text{int} \left(\frac{2c_{k-1}^1 + c_k^1 - 3c_{k+1}^1 - 2d_{k+1}^{1,0}}{4} \right), \quad k = 1, \dots, N_1 - 2, \\ d_{N_1-1}^1 = d_{N_1-1}^{1,0} - \text{int} \left(\frac{c_{N_1-2}^1 + c_{N_1-1}^1}{4} \right) \end{array} \right. \quad (7.41)$$

Реконструкция выполняется следующим образом:

$$\left\{ \begin{array}{l} d_{N_1-1}^{1,0} = d_{N_1-1}^1 + \text{int} \left(\frac{c_{N_1-2}^1 - c_{N_1-1}^1}{4} \right) \\ d_k^{1,0} = d_k^1 + \text{int} \left(\frac{2c_{k-1}^1 + c_k^1 - c_{k+1}^1 - 2d_{k+1}^{1,0}}{8} \right), \quad k = N_1 - 2, \dots, 1, \\ d_0^{1,0} = d_0^1 + \text{int} \left(\frac{c_0^1 - c_1^1}{4} \right) \end{array} \right. \quad (7.42)$$

Далее вычисляется

$$c_{2k+1}^0 = c_k^1 - \text{int}\left(\frac{d_k^1}{2}\right), \quad k = 0, \dots, N_1 - 1 \quad (7.43)$$

и

$$c_{2k}^0 = d_k^1 + c_{2k+1}^0, \quad k = 0, \dots, N_1 - 1. \quad (7.44)$$

Таким образом, мы рассмотрели некоторые возможные алгоритмы получения целочисленных вейвлет-коэффициентов. Данная область исследования является перспективной и привлекает внимание многих исследователей.

МУЛЬТИВЕЙВЛЕТЫ

Как было показано в главе 2, вейвлеты тесно связаны со схемами субполосного кодирования. Свойства соответствующих фильтров хорошо изучены. В частности, известно, что невозможно построить ортогональный линейно-фазовый блок КИХ фильтров. Следовательно, не существует ортогональных симметричных вейвлетов с компактной областью определения. Вместе с тем, во многих приложениях обработки сигналов наличие такого базиса было бы желательно.

Одной из причин интереса к мультивейвлетам является возможность получения симметричного ортогонального базиса. Мультивейвлеты получаются за счет отказа от времянезависимости характеристик фильтров. Как будет показано, такие конструкции приводят к матричным уравнениям масштабирования, аналогичным (2.18). Кроме того, мультивейвлеты обладают хорошими аппроксимационными свойствами, что важно во многих приложениях обработки сигналов.

В настоящей главе даны основы теории мультифильтров, итерирования мультифильтров, показана их связь с мультивейвлетами. Также рассмотрены основные проблемы практического применения мультивейвлетов при обработке сигналов и некоторые пути их разрешения.

8.1. Блоки мультифильтров

8.1.1. Основы теории блоков фильтров, изменяющихся во времени

Под изменяющимися во времени блоками фильтров понимаются такие блоки, в которых характеристика фильтра периодически изменяется во времени. Вначале рассмотрим блок фильтров синтеза, то есть интерполятор со следующим за ним фильтром. Для простоты предположим, что характеристика фильтра состоит всего из двух характеристик. Тогда во временной области результирующий оператор запишется в виде

В целях упрощения записи объединим два НЧ фильтра $c[n]$ и $d[n]$ в единую матрицу коэффициентов мультифильтра:

$$\mathbf{M}[k] = \begin{pmatrix} c[2k] & c[2k+1] \\ d[2k] & d[2k+1] \end{pmatrix}. \quad (8.8)$$

Тогда z -преобразование НЧ мультифильтра анализа можно представить в виде

$$\mathbf{H}_0(z) = \mathbf{T}'(z) = \sum_k \mathbf{M}[k] z^{-k}. \quad (8.9)$$

Аналогично выписываются выражения для $\mathbf{H}_1(z)$, $\mathbf{G}_0(z)$ и $\mathbf{G}_1(z)$, то есть, соответственно, для ВЧ мультифильтра анализа, НЧ и ВЧ мультифильтров синтеза. Далее, определив входной сигнал как $\mathbf{X}(z) = [X_0(z), X_1(z)]^T$, получим известное равенство для выходного сигнала блока фильтров:

$$\begin{aligned} \widehat{\mathbf{X}}(z) = & \frac{1}{2} \{ [\mathbf{G}_0(z)\mathbf{H}_0(z) + \mathbf{G}_1(z)\mathbf{H}_1(z)]\mathbf{X}(z) + \\ & + [\mathbf{G}_0(z)\mathbf{H}_0(-z) + \mathbf{G}_1(z)\mathbf{H}_1(-z)]\mathbf{X}(-z) \} \end{aligned} \quad (8.10)$$

Отметим, что в отличие от скалярного случая, порядок следования сомножителей является важным, так как матричное произведение не обладает свойством коммутативности.

Из верхней строки (8.10) получим условие полного восстановления

$$[\mathbf{G}_0(z)\mathbf{H}_0(z) + \mathbf{G}_1(z)\mathbf{H}_1(z)] = 2\mathbf{I}_2, \quad (8.11)$$

а из нижней строки – условие отсутствия эйлазинга:

$$[\mathbf{G}_0(z)\mathbf{H}_0(-z) + \mathbf{G}_1(z)\mathbf{H}_1(-z)] = \mathbf{O}_2. \quad (8.12)$$

Введем понятие модуляционной матрицы

$$\mathbf{H}_m(z) = \begin{pmatrix} \mathbf{H}_0(z) & \mathbf{H}_0(-z) \\ \mathbf{H}_1(z) & \mathbf{H}_1(-z) \end{pmatrix}. \quad (8.13)$$

Тогда появляется возможность объединить эти два условия в одно:

$$[\mathbf{G}_0(z)\mathbf{G}_1(z)]\mathbf{H}_m(z) = 2[\mathbf{I}_2, \mathbf{O}_2]. \quad (8.14)$$

Может быть показано, что решением является:

$$\mathbf{G}_0(z) = 2\mathbf{U}^{-1}(z), \quad (8.15)$$

$$\mathbf{G}_1(z) = -2\mathbf{U}^{-1}(z)\mathbf{H}_0(-z)\mathbf{H}_1^{-1}(-z), \quad (8.16)$$

где

$$\mathbf{U}(z) = \mathbf{H}_0(z) - \mathbf{H}_0(-z)\mathbf{H}_1^{-1}(-z)\mathbf{H}_1(z). \quad (8.17)$$

Свойство ортогональности блока фильтров означает, что оператор (8.1) должен быть унитарным, или $\mathbf{T}'\mathbf{T} = \mathbf{I}$. Отсюда следует, что

$$\tilde{\mathbf{H}}_m(z)\mathbf{H}_m(z) = \mathbf{H}_m(z)\tilde{\mathbf{H}}_m(z) = \mathbf{I}_2, \quad (8.18)$$

где $\tilde{\mathbf{H}}(z) = \mathbf{H}'(z^{-1})$ называется парасопряженной матрицей для матрицы $\mathbf{H}(z)$. Тогда получаем уравнения:

$$\mathbf{H}_0(z)\tilde{\mathbf{H}}_0(z) + \mathbf{H}_0(-z)\tilde{\mathbf{H}}_0(-z) = \mathbf{I}_2, \quad (8.19)$$

$$\mathbf{H}_1(z)\tilde{\mathbf{H}}_1(z) + \mathbf{H}_1(-z)\tilde{\mathbf{H}}_1(-z) = \mathbf{I}_2, \quad (8.20)$$

$$\mathbf{H}_0(z)\tilde{\mathbf{H}}_1(z) + \mathbf{H}_0(-z)\tilde{\mathbf{H}}_1(-z) = \mathbf{O}_2, \quad (8.21)$$

$$\mathbf{H}_1(z)\tilde{\mathbf{H}}_0(z) + \mathbf{H}_1(-z)\tilde{\mathbf{H}}_0(-z) = \mathbf{O}_2. \quad (8.22)$$

Отсюда можно получить условия для обеспечения полного восстановления и отсутствия элайзинга:

$$\mathbf{G}_0(z) = \tilde{\mathbf{H}}_0(z), \quad (8.23)$$

$$\mathbf{G}_1(z) = \tilde{\mathbf{H}}_1(z). \quad (8.24)$$

Линейность фазы последовательностей $c[n]$ и $d[n]$ не является достаточным условием того, чтобы мультифильтр $\mathbf{T}(z)$ имел линейную фазу. Этому условия недостаточно также для того, чтобы масштабирующая функция, получающаяся при итерировании матричного произведения, была симметричной или антисимметричной. Поэтому вводят следующее определение линейной фазы мультифильтра.

Мультифильтр $\mathbf{T}(z)$ называется линейно-фазовым, если существуют вещественные числа c_0 и c_1 , такие что

$$\mathbf{T}_j(z) = \alpha_j z^{2c_j} \mathbf{T}_j(z^{-1}), \quad (8.25)$$

где α_j - одна из следующих функций: $\alpha_j = 1$ или $\alpha_j = -1$, или $\alpha_j = (-1)^{i+j}$, или $\alpha_j = -(-1)^{i+j}$.

8.1.2. Построение блоков мультифильтров

С точки зрения математики, двухканальный изменяющийся во времени блок фильтров с двумя фазами эквивалентен четырехканальному блоку фильтров. Это понятно, так как в обоих случаях мы имеем четыре различные импульсные характеристики. Аналогично K -канальный блок мультифильтров с L фазами эквивалентен $K \times L$ блоку фильтров. Таким образом, все известные результаты для N -канальных блоков фильтров могут быть непосредственно перенесены на случай блоков мультифильтров. Например, ясно, что существует линейно-фазовый ортонормальный двухканальный КИХ блок мультифильтров с двумя фазами, так как существует аналогичный ему четырехканальный блок фильтров.

8.1.3. Итерирование блоков мультифильтров

Пусть каскадно включены n блоков мультифильтров интерполяции. Результирующая передаточная матрица запишется в виде

$$\mathbf{T}^{(n)}(z) = \prod_{i=0}^{n-1} \mathbf{T}(z^{2^i}). \quad (8.26)$$

Тогда получим следующий выходной сигнал:

$$\mathbf{Y}^{(n)}(z) = \begin{pmatrix} 1 & z^{-1} \end{pmatrix} \mathbf{T}(n) \begin{pmatrix} X_0(z^2) \\ X_1(z^2) \end{pmatrix}. \quad (8.27)$$

В данном случае мы имеем две импульсные характеристики, определяемые выражениями

$$C^{(n)}(z) = T_{00}(z^2) + z^{-1}T_{10}(z^2), \quad (8.28)$$

$$D^{(n)}(z) = T_{01}(z^2) + z^{-1}T_{11}(z^2). \quad (8.29)$$

Легко показать, что если \mathbf{T} - унитарная матрица, то $\mathbf{T}^{(n)}$ - также унитарна. Также понятно, что свойство линейной фазы мультифильтра сохраняется при проведении итераций. Например, $\mathbf{T}(z^2)\mathbf{T}(z)$ имеет линейную фазу с коэффициентами $3c_0, c_1 - 2c_0$.

8.2. Мультивейвлеты

Так же, как и в случае вейвлетов, масштабирующая функция $\phi(t) = [\phi_0(t), \dots, \phi_{r-1}(t)]$ является решением масштабирующего уравнения

$$\phi(t) = \sum_{k=0}^N \mathbf{M}[k] \phi(2t - k), \quad (8.30)$$

где $\mathbf{M}[k]$ - матрица вещественных коэффициентов размером $r \times r$, называемая маской. Свойства масштабирующей функции тесно связаны с поведением этой маски в области преобразования Фурье. Как и в случае вейвлетов, непрерывная масштабирующая функция получается в пределе при бесконечном числе итераций:

$$\ddot{\mathbf{O}}(\omega) = \lim_{n \rightarrow \infty} \Phi^{(n)}(\omega) = \prod_{i=1}^{\infty} \mathbf{M}(\omega / 2^i). \quad (8.31)$$

Исследованию сходимости этого предела посвящен ряд работ. Основные результаты следующие. Различают безусловную сходимость (для любой частоты ω) и условную. Доказана теорема о том, что необходимыми условиями безусловной сходимости являются

$$\ddot{\mathbf{O}}(0) \neq \mathbf{0}; \mathbf{M}(0) = \mathbf{I}, \mathbf{M}(\pi) = \mathbf{0}. \quad (8.32)$$

Кроме того, маска $\mathbf{M}(\omega)$ должна удовлетворять условию Смита-Барнуэлла:

$$\mathbf{M}(\omega)\mathbf{M}^*(-\omega) + \mathbf{M}(\omega + \pi)\mathbf{M}^*(-\omega + \pi) = \mathbf{I}. \quad (8.33)$$

Условная сходимость подробно рассмотрена в работах П.Массопуста и на страницах нашей книги не обсуждается.

От вида маски зависит, будет ли иметь масштабирующая функция симметрию или асимметрию, а также ее аппроксимационные свойства.

Рассмотрим теперь простейший пример линейных мультивейвлетов, а затем перейдем к вопросу применения мультивейвлетов для обработки сигналов.

Пример. Пусть даны две кусочно-линейные функции:

$$\phi_1(t) = \begin{cases} 1, & 0 \leq t < 1, \\ 0, & \text{в других случаях,} \end{cases} \quad \phi_2(t) = \begin{cases} \sqrt{3}\left(t - \frac{1}{2}\right), & 0 \leq t < 1, \\ 0, & \text{в других случаях.} \end{cases} \quad (8.34)$$

Эти функции изображены на рис.8.1. Их целочисленные сдвиги $\phi_1(\cdot - l), \phi_2(\cdot - l), l \in \mathbf{Z}$ образуют ортонормальный базис замкнутого подпространства $V_0 \subset L_2(\mathfrak{R})$, состоящего из кусочно-линейных на целочисленных интервалах функций. Далее, пусть V_j - подпространство, натянутое на функции $2^{j/2}\phi_1(2^j \cdot - l), 2^{j/2}\phi_2(2^j \cdot - l), l \in \mathbf{Z}$ и содержащее все функции, которые кусочно-линейны на интервалах $[2^{-j}l, 2^{-j}(l+1)]$. Легко показать, что

$$\begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -\sqrt{3} & 2 \end{pmatrix} \begin{pmatrix} \phi_1(2 \cdot) \\ \phi_2(2 \cdot) \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ \sqrt{3} & 2 \end{pmatrix} \begin{pmatrix} \phi_1(2 \cdot - 1) \\ \phi_2(2 \cdot - 1) \end{pmatrix}. \quad (8.35)$$

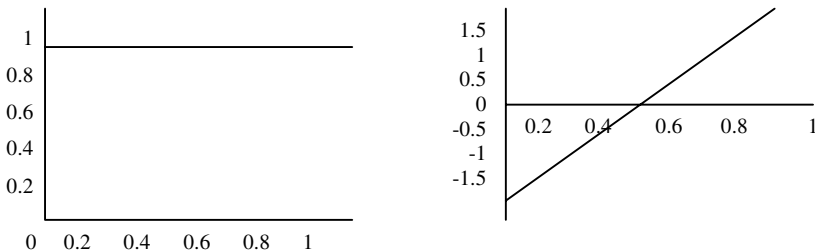


Рис.8.1. Кусочно-линейные ортогональные масштабирующие функции

Из этого выражения следует, что $\phi_1, \phi_2 \in V_0 \subset V_1$. Аналогично $V_j \subset V_{j+1}$. Ортогональные проекции $f_j(t)$ некоторой функции $f \in L^2(\mathfrak{R})$ на подпространства V_j есть не что иное, как последовательные приближения линейными функциями, сходящиеся к $f(t)$ при $j \rightarrow \infty$. Таким образом, мы получаем вложенную структуру подпространств, известную как кратномасштабный анализ (КМА - глава 2):

$$\bigcup_{j=-\infty}^{\infty} V_j = L^2(\mathfrak{R}), \quad \bigcap_{j=-\infty}^{\infty} V_j = \{0\}. \quad (8.36)$$

Однако в нашем случае КМА порождается двумя функциями ϕ_1, ϕ_2 . Подпространства V_j , используемые здесь, не могут быть порождены сдвигами и растяжениями одной функции.

Рассмотрим еще две кусочно-линейные функции (рис.8.2):

$$\psi_1(t) = \begin{cases} 6t-1, & 0 < t < 1/2, \\ 6t-5, & 1/2 < t < 1, \\ 0, & \text{в других случаях,} \end{cases} \quad \psi_2(t) = \begin{cases} 6t-1, & 0 < t < 1/2, \\ 6t-5, & 1/2 < t < 1, \\ 0, & \text{в других случаях.} \end{cases} \quad (8.37)$$

Пусть W_j есть подпространство $L^2(\mathfrak{R})$, натянутое на базисы $2^{j/2}\psi_1(2^j \cdot -l)$ и $2^{j/2}\psi_2(2^j \cdot -l)$. Целочисленные сдвиги $\psi_1(\cdot -l), \psi_2(\cdot -n), l, n \in \mathbf{Z}$ ортогональны друг другу и целочисленным сдвигам ϕ_1, ϕ_2 , что делает подпространство W_0 ортогональным V_0 . Функции ψ_1, ψ_2 кусочно-линейны на половине интервала. Поэтому $W_0 \subset V_1$. В частности,

$$\begin{pmatrix} \psi_1 \\ \psi_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \frac{\sqrt{3}}{2} \begin{pmatrix} \phi_1(2 \cdot) \\ \phi_2(2 \cdot) \end{pmatrix} + \begin{pmatrix} -1 \\ 0 \end{pmatrix} \frac{\sqrt{3}}{2} \begin{pmatrix} \phi_1(2 \cdot -1) \\ \phi_2(2 \cdot -1) \end{pmatrix}. \quad (8.38)$$

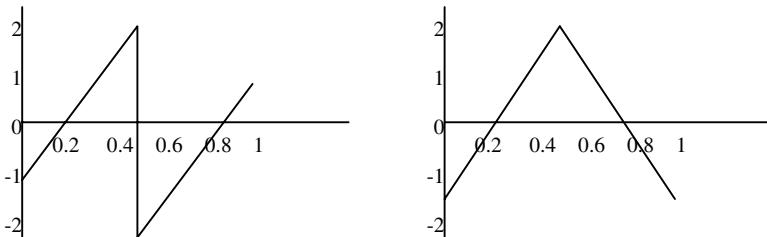


Рис. 8.2. Кусочно-линейные ортогональные вейвлеты ψ_1, ψ_2

Таким образом, базисы пространства V_1 есть линейная комбинация $\phi_1, \phi_2, \psi_1, \psi_2$:

$$\phi_1(2 \cdot) = \frac{1}{2} \phi_1 - \frac{\sqrt{3}}{4} \phi_2 + \frac{1}{4} \psi_1, \quad \phi_1(2 \cdot - 1) = \frac{1}{2} \phi_1 + \frac{\sqrt{3}}{4} \phi_2 - \frac{1}{4} \psi_1, \quad (8.39)$$

$$\phi_2(2 \cdot) = \frac{1}{4} \phi_2 + \frac{\sqrt{3}}{4} \psi_1 + \frac{1}{2} \psi_2, \quad \phi_2(2 \cdot - 1) = \frac{1}{4} \phi_2 + \frac{\sqrt{3}}{4} \psi_1 - \frac{1}{2} \psi_2. \quad (8.40)$$

Следовательно, V_1 есть ортогональная сумма V_0 и W_0 . Аналогично $V_j \oplus W_j = W_{j+1}$ и $L^2(\mathfrak{X}) = \bigoplus_{j \in \mathbb{Z}} W_j$. Значит, сдвиги и растяжения ψ_1, ψ_2 образуют ортогональный базис пространства $L^2(\mathfrak{X})$.

8.3. Обработка сигналов в базисе мультивейвлетов

Как уже отмечалось, мультивейвлеты имеют потенциальные преимущества для обработки сигналов по сравнению с классическими вейвлетами. Особенно эффективным обещает быть их применение для сжатия изображений. Однако для реального применения мультивейвлетов необходимо решить ряд практических задач, одна из которых – предварительная обработка входных данных. Такая обработка необходима в связи с тем, что вход схемы мультивейвлет-преобразования должен быть векторным, а не скалярным, как обычно. Реальные же сигналы, как правило, одномерные или рассматриваются как одномерные. Поэтому возникает вопрос: каким образом "векторизовать" входной сигнал?

Очевидно, существует бесконечно много возможностей. Например, можно взять в качестве второй строки копию первой. Это приводит к увеличению количества отсчетов в два раза и для сжатия сигнала вряд ли применимо. Вместе с тем, для других приложений, например очистки сигнала от шумов, такой подход имеет право на жизнь. Надо заметить, что при обработке двумерных сигналов, количество отсчетов увеличивается уже не в два, а в четыре раза и, соответственно, увеличивается число вычислений.

Другой подход заключается в разделении входных отсчетов на четные и нечетные (в случае одномерного сигнала) или в использовании соседних строк (в случае изображения). При этом не происходит увеличения числа отсчетов. Недостатком такого подхода является введение дополнительных

ограничений на конструкцию мультифильтра, а в случае изображений – необходимость применения нетривиальной двумерной обработки.

Интересный метод, основанный на аппроксимационных свойствах вейвлет-преобразования, был предложен Д.Джеронимо. Он применим к семейству так называемых ГНМ мультивейвлетов, названных так по начальным буквам фамилий их исследователей (Geronimo, Hardin, Massopust). Это – ортогональные симметричные мультивейвлеты. Пример масштабирующих функций приведен на рис.8.3.

Из рис.8.3 видно, что функция $\phi_0(t)$ равна нулю при целом t . Масштабирующая функция $\phi_1(t)$ ненулевая при $t=1$ и равна нулю при других целых t . Пусть функция $f(t)$ принадлежит подпространству V_0 , порождаемому сдвигами масштабируемых функций ГНМ. Это означает, что $f(t)$ может быть записана как линейная комбинация этих сдвигов:

$$f(t) = \sum_n v_{1,n}^{(0)} \phi_0(t-n) + v_{2,n}^{(0)} \phi_1(t-n). \quad (8.41)$$

Предположим, что отсчеты входной последовательности $f[n]$ следуют в два раза чаще, чем отсчеты $f(t)$:

$$f[2n] = f(n), \quad f[2n+1] = f(n+1/2). \quad (8.42)$$

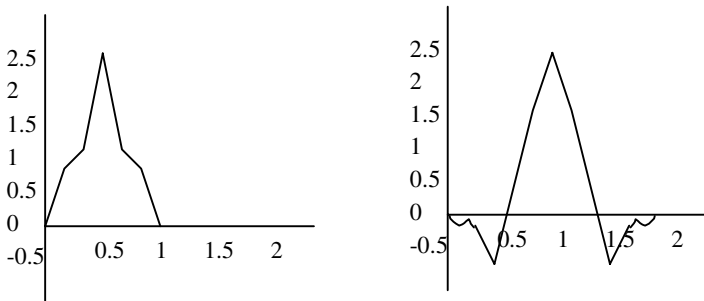


Рис.8.3. Масштабирующие функции мультивейвлетов: $\phi_0(t), \phi_1(t)$

Из рис.8.3 и равенств (8.41) и (8.42) следует, что

$$f[2n] = \phi_1(1)v_{2,n}, \quad (8.43)$$

$$f[2n+1] = \phi_1(3/2)v_{2,n} + \phi_0(1/2)v_{1,n+1} + \phi_1(1/2)v_{2,n+1}. \quad (8.44)$$

Отсюда могут быть получены коэффициенты $v_{1,n}, v_{2,n}$:

$$v_{1,n} = \frac{\phi_1(1)f[2n-1] - \phi_1(1/2)f[2n-2] - \phi_1(3/2)f[2n]}{\phi_1(1)\phi_0(1/2)}, \quad (8.45)$$

$$v_{2,n} = \frac{f[2n]}{\phi_1(1)}. \quad (8.46)$$

Эти выражения могут быть несколько упрощены с учетом симметрии $\phi_1(t)$ ($\phi_1(1/2) = \phi_1(3/2)$). Таким образом, любой входной сигнал длины N может быть разделен на две последовательности $v_{1,n}, v_{2,n}$ длиной $N/2$.

Дополнительным преимуществом аппроксимационного метода является то, что он хорошо согласуется с методом симметричного продолжения сигнала на границах. То есть последовательности $v_{1,n}, v_{2,n}$ также будут обладать свойством симметрии.

Пусть имеются две входные последовательности (одна четной длины и одна – нечетной):

$$\begin{matrix} s_0^1 & s_1^1 & s_2^1 & \dots & s_{N-1}^1 \\ s_0^2 & s_1^2 & s_2^2 & \dots & s_{N-1}^2 & s_N^2 \end{matrix}. \quad (8.47)$$

В результате симметричного отражения сигнала получаем

$$\begin{matrix} \dots & s_2^1 & s_1^1 & s_0^1 & s_0^1 & s_1^1 & \dots & s_{N-2}^1 & s_{N-1}^1 & s_{N-1}^1 & \dots \\ \dots & s_2^2 & s_1^2 & s_0^2 & s_0^2 & s_1^2 & \dots & s_{N-1}^2 & s_N^2 & s_{N-1}^2 & \dots \end{matrix}. \quad (8.48)$$

После одного шага каскадного алгоритма получится четыре симметричные последовательности (две на выходе НЧ фильтра и две на выходе ВЧ фильтра).

8.4. Сбалансированные мультивейвлеты

На пути практического применения мультивейвлетов мы сталкиваемся с еще одной трудностью. Масштабирующие функции и, соответственно, компоненты НЧ мультифильтра имеют различные спектральные характеристики. По выражению М.Веттерли, они являются "разбалансированными". Это приводит к перекрытию в области спектра НЧ и ВЧ субполос, вызывая искажения в виде колебаний на сжатом изображении.

Пусть входной сигнал $f[n]=1$. После аппроксимации и фильтрации мы получим две разные последовательности. В случае ГНМ мультифильтров одна последовательность будет 1, а другая равна $\sqrt{2}$. Таким образом, вместо постоянного сигнала получается "пилообразный" (рис.8.4). Для предотвращения этого, в случае применения аппроксимационного метода, необходимо выполнить деаппроксимацию по формулам

$$f(2n) = \phi_1(1)v_{2,n} \quad (8.49)$$

$$f(2n-1) = \phi_1(1/2)(v_{2,n} + v_{2,n-1}) + \phi_0(1/2)v_{1,n} \quad (8.50)$$

Таким образом, аппроксимационный метод можно рассматривать, как введение некоторой пре/постфильтрации входного сигнала.

В общем случае, мы нуждаемся в некотором правиле, согласно которому можно было бы конструировать сбалансированные мультивейвлеты, свободные от указанного выше недостатка. В работах М.Веттерли, Г.Стрэнга доказано необходимое условие, которому должна удовлетворять сбалансированная масштабирующая функция: для случая $r=2$ вектор $[1,1]^T$ должен быть правым собственным вектором, соответствующим собственному значению 1 маски $\mathbf{M}(0)$. Это означает, что $\hat{\phi}(0) = [1,1]^T$.

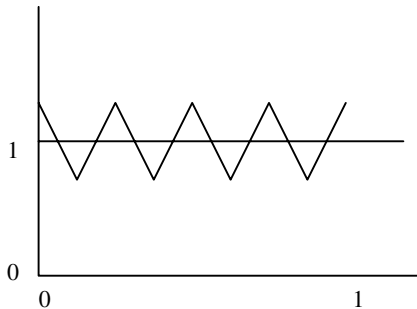


Рис. 8.4. Воспроизведение входного сигнала $f[n]=1$ блоком ГНМ мультифильтров

Известными методами конструирования сбалансированных мультивейвлетов являются:

1) получение сбалансированных мультивейвлетов из комплексных фильтров Добеши;

2) балансировка существующих мультивейвлетов.

Недостатком мультивейвлетов, полученных первым способом, является то, что значение их маски $M(0)$ не удовлетворяет условию (8.32). Следовательно, при итерировании мультифильтров достигается лишь условная сходимость. Использование подобных мультивейвлетов проблематично, так как они неробастны к отбрасыванию коэффициентов: отбрасывание малых коэффициентов приводит не к плавному ухудшению качества сжимаемого сигнала, но к появлению паразитных колебаний.

Сбалансированные мультивейвлеты, полученные вторым способом, свободны от этого недостатка. Как отмечается, они весьма робастны к отбрасыванию коэффициентов без применения какой бы то ни было пре/постфильтрации.

Таким образом, в данной главе рассмотрены основы теории мультивейвлетов: вейвлетов с несколькими масштабирующими функциями. Отмечены положительные стороны подобных конструкций, такие как возможность получения ортогонального симметричного базиса с компактной областью определения, а также хорошие аппроксимационные свойства мультивейвлетов.

В главе рассмотрены также основные проблемы, возникающие при практическом применении мультивейвлетов. Эти проблемы заключаются в определении правила разделения одномерного входного сигнала для получения r -мерного, а также в различном спектральном наполнении масштабирующих функций. Приведены некоторые способы разрешения этих проблем. В целом, необходимо отметить, что исследования в этой области еще только начались, и можно ожидать появления новых многообещающих результатов.

ПОТЕНЦИАЛЬНЫЕ ХАРАКТЕРИСТИКИ КОДИРОВАНИЯ ИЗОБРАЖЕНИЯ С ПРИМЕНЕНИЕМ ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЯ

Коэффициенты вейвлет-преобразования имеют приблизительно гауссовскую плотность распределения. Из теории информации известно, что в случае, если шаг квантователя достаточно мал, среднеквадратичная ошибка кодирования $D(\bar{R})$ пропорциональна $2^{-2\bar{R}}$, где \bar{R} - среднее число бит на пиксел. Под малым шагом квантователя понимается то, что вероятность попадания коэффициента в тот или иной интервал квантователя постоянна и не зависит от значения этого интервала. Коэффициент пропорциональности зависит от используемого базиса и алгоритма распределения бит. В интересующем диапазоне кодирования изображения тратится менее 1 бит на пиксел. В этом случае предположение о малости шага квантователя становится неверным. В данной главе представлены аналитические соотношения для случая вейвлет-кодирования изображений при низких битовых скоростях ($\bar{R} < 1$ бит/пиксел).

В разделе 9.1 кратко рассматриваются основные положения теории информации, касающиеся скалярных квантователей с ограниченной энтропией и кодирования с преобразованием. В разделе 9.2 представлена аналитическая зависимость искажения от скорости для вейвлет-кодеков при низких скоростях кодирования.

9.1. Основные формулы и теоремы теории связи, относящиеся к кодированию с преобразованием при высоких скоростях (>1бит/пиксел)

Изображение, подвергаемое кодированию, представляется случайным вектором Y размерностью N . Хотя оно является двумерным, для простоты обозначений отдельные пикселы обозначаются как $Y[n]$. Кодер с преобразованием декомпозирует эти изображения по ортонормальному базису $G = \{g_m\}_{0 \leq m}$:

$$Y = \sum_{m=0}^{N-1} A[m]g_m.$$

Каждый коэффициент $A[m]$ есть случайная величина, определяемая как

$$A[m] = \langle Y, g_m \rangle = \sum_{n=0}^{N-1} Y[n] g_m^*[n].$$

Для построения конечного кода каждый коэффициент $A[m]$ аппроксимируется квантованной переменной $\hat{A}[m]$. Далее рассматриваются только скалярные квантователи как наиболее часто встречающиеся при кодировании с преобразованием.

9.1.1. Скалярное квантование с ограниченной энтропией

Скалярный квантователь Q аппроксимирует случайную переменную X квантованной переменной $\hat{X} = Q(X)$, которая может принимать конечное множество значений. Предполагается, что X принимает значения в диапазоне $[a, b]$, который может соответствовать всей вещественной оси. Интервал $[a, b]$ декомпозируется на K интервалов $(y_{k-1}, y_k]_{1 \leq k \leq K}$ переменной длины с $y_0 = a$ и $y_k = b$. Если $x \in (y_{k-1}, y_k]$, то $Q(x) = x_k$. Для скалярного квантователя известно, что энтропия

$$H(\hat{X}) = - \sum_{k=1}^K p_k \log_2 p_k$$

есть нижняя граница среднего количества бит на символ, используемого для кодирования значений \hat{X} , где p_k - вероятность того, что $x \in (y_{k-1}, y_k]$. Арифметическое энтропийное кодирование достигает сколь угодно близкого к этой границе среднего числа бит на символ. Поэтому в дальнейшем считается, что эта граница достигнута. Скалярный квантователь с ограниченной энтропией минимизирует $H(\hat{X})$ для фиксированного среднеквадратичного искажения $D = E\{(X - \hat{X})^2\}$.

Пусть $p(x)$ есть плотность распределения вероятности отсчетов случайного источника X . Считается, что квантователь имеет высокое разрешение, если $p(x)$ примерно постоянна в каждом интервале квантования $(y_{k-1}, y_k]$ размером $\Delta_k = y_k - y_{k-1}$. Это будет иметь место в случае, когда размеры Δ_k достаточно малы относительно скорости изменения $p(x)$. Известно, что равномерные квантователи являются оптимальными среди квантователей с высоким разрешением.

Если Q - квантователь с высоким разрешением относительно $p(x)$, то

$$H(\hat{X}) \geq H_d(X) - \frac{1}{2} \log_2(12D). \quad (9.1)$$

Это неравенство превращается в равенство, если и только если Q является равномерным квантователем. Тогда $D = \frac{\Delta^2}{12}$.

Для фиксированного искажения D , при условии соблюдения гипотезы о высоком разрешении, минимальная средняя скорость $R_x = H(X)$ достигается поэтому равномерным квантователем и

$$R_x = H_d(X) - \log_2 \Delta. \quad (9.2)$$

Зависимость искажения от скорости получается из (9.1):

$$D(R_x) = \frac{1}{12} 2^{2H_d(X)} 2^{-2R_x}. \quad (9.3)$$

По-видимому, даже если гипотеза о квантовании с высоким разрешением не выполняется, равномерный квантователь близок к оптимальному для большого класса источников, при условии, что число интервалов квантования достаточно велико.

9.1.2. Зависимость искажения от скорости

Получим зависимость искажения от скорости для коэффициентов $A[m]$ вейвлет-разложения $Y = \sum_{m=0}^{N-1} A[m]g_m$. Средний бюджет бит, необходимый для кодирования $\hat{A}[m] = Q(A[m])$ есть $R_m = H(A[m])$. Для квантования с высоким разрешением ошибка квантования будет минимальна при использовании равномерного скалярного квантователя. Процедура оптимального распределения бит должна минимизировать общее число бит $R = \sum_{m=0}^{N-1} R_m$ для заданной суммарной ошибки $D = \sum_{m=0}^{N-1} D_m$. Пусть $\bar{R} = \frac{R}{N}$ есть среднее число бит на от-

счет. Применяя множители Лагранжа, можно доказать, что \bar{R} будет минимальна в случае, если все D_m равны. Тогда

$$D(\bar{R}) = \frac{N}{12} 2^{2\bar{H}_d} 2^{-2\bar{R}}, \quad (9.4)$$

где \bar{H}_d есть средняя дифференциальная энтропия:

$$\bar{H}_d = \frac{1}{N} \sum_{m=0}^{N-1} H_d(A[m]).$$

Искажение (9.4) зависит от базиса вейвлетов G через \bar{H}_d . В общем случае трудно найти G , минимизирующий \bar{H}_d , так как плотность распределения вероятности $A[m] = \langle Y, g_m \rangle$ может зависеть от g_m сложным образом. Если Y распределен по гауссовскому закону, то коэффициенты $A[m]$ будут гауссовскими случайными переменными в любом базисе. В этом случае плотность распределения вероятности $A[m]$ зависит только от дисперсии σ_m^2 и

$$H_d(A[m]) = \log_2 \sigma_m + \log_2 \sqrt{2\pi e}.$$

Данное выражение подставляется в (9.4):

$$D(\bar{R}) = N \frac{\pi e}{6} \left(\prod_{m=0}^{N-1} \sigma_m^2 \right)^{1/N} 2^{-2\bar{R}}. \quad (9.5)$$

Известно, что $\prod_{m=0}^{N-1} \sigma_m^2$ минимально, если и только если G является базисом Карунена-Лоэва для Y , то есть G диагонализует ковариационную матрицу Y . Таким образом, оптимальным с точки зрения кодирования с преобразованием базисом для гауссовского процесса является базис Карунена-Лоэва. Если Y не является гауссовским (например, в случае изображения), базис Карунена-Лоэва не является априорно оптимальным.

Наиболее популярным при кодировании изображений является разделимый базис вейвлетов. Разделимый вейвлет-базис включает в себя три семейства вейвлетов с горизонтальной, вертикальной и диагональной ориентацией, индексруемые $1 \leq k \leq 3$. При ориентации k и масштабе 2^j вектор вейвлета

$g_m = \Psi_{j,p,q}^k$ примерно центрирован в точке $(2^j p, 2^i q)$ с квадратной областью определения, размер которой пропорционален 2^j . Как было отмечено, при высоких битовых скоростях кодирования минимальное искажение достигается путем равномерного квантования всех коэффициентов декомпозиции. Гладким участкам изображения соответствуют вейвлет-коэффициенты с малым значением амплитуды, которые квантуются в нуль. Для повышения эффективности кодирования вейвлет-коэффициенты сканируются в заранее определенном порядке, и позиции нулевых коэффициентов кодируются кодером длин серий. Амплитуды ненулевых квантованных коэффициентов кодируются кодером Хаффмана или арифметическим кодером.

Из формулы (9.4) следует, что

$$\log_2 D(\bar{R}) = 2\bar{H}_d + \log_2 \left(\frac{N}{12} \right) - 2\bar{R},$$

где \bar{H}_d есть средняя дифференциальная энтропия вейвлет-коэффициентов при всех масштабах и ориентациях. Из данной формулы следует, что $\log_2 D(\bar{R})$ является убывающей функцией с наклоном -2 . Однако из практики известно, что для области $\bar{R} < 1$ функция $\log_2 D(\bar{R})$ убывает значительно быстрее. Для данной области формула (9.4) не выполняется в силу того, что предположение о квантовании с высоким разрешением уже неверно. Сжатие изображения с применением вейвлет-преобразования достигает хороших результатов для скоростей значительно меньших 1 бит/отсчет. Поэтому в следующем разделе исследуется зависимость скорости от искажения для низких скоростей кодирования.

9.2. Сжатие изображения при низких скоростях кодирования

При низких скоростях кодирования имеет место грубое квантование коэффициентов декомпозиции. Так как многие из коэффициентов обращаются в нуль, позиции нулевых и ненулевых коэффициентов сохраняются в бинарной карте значений, которая сжимается при помощи кодера длин серий или более сложного алгоритма нульдеревя (см. главу 10). Теория передачи с погрешностью, описываемая выше, неприменима по следующим двум причинам. Во-первых, не выполняется предположение о квантовании с высоким разрешением в силу того, что интервалы квантователей большие. Во-вторых, невозможно представить общий бюджет бит R в виде суммы битов R_m , независимо распределенных для каждого коэффициента декомпозиции, так как кодирование коэффициентов при помощи бинарной карты значений является, по сути, разновидностью векторного квантования.

Для оценки зависимости искажения от скорости необходима некоторая стохастическая модель изображения. Однако до сих пор не существует модели, которая бы учитывала сложную нестационарную структуру изображения, описывала бы текстуру и контуры. Поэтому далее изображение рассматривается как детерминированный вектор, чьи коэффициенты вейвлет-декомпозиции G имеют параметрический спад. Зависимость скорости от искажения, таким образом, будет находиться не для ансамбля сигналов, а для каждого сигнала f . Основным результатом данного раздела является вывод о том, что функция искажение-скорость зависит от точности аппроксимации f малым числом векторов, выбранных из G .

9.2.1. Функция искажение-скорость

Пусть сигнал f декомпозирован по вейвлет-базису $G = \{g_m\}_{0 \leq m \leq N}$:

$$f = \sum_{m=0}^{N-1} a[m]g_m \quad \text{с} \quad a[m] = \langle f, g_m \rangle.$$

Коэффициенты преобразования квантуются, и декодер восстанавливает

$$\hat{f} = \sum_{m=0}^{N-1} Q(a[m])g_m.$$

Ошибка кодирования

$$D = \|f - \hat{f}\|^2 = \sum_{m=0}^{N-1} |a[m] - Q(a[m])|^2. \quad (9.6)$$

Можно показать, что в случае применения ортонормального базиса, ошибка квантования в области трансформанты будет равна ошибке в области исходного изображения. На этом основано много алгоритмов кодирования. Через $h[x]$ обозначим дискретную гистограмму N коэффициентов $a[m]$, нормализованную так, что $\sum_x h[x] = 1$. Значения этой гистограммы интерполируются, и вводится функция $p(x) \geq 0$ для всех $x \in R$, такая что $\int_{-\infty}^{+\infty} p(x) dx = 1$.

Тогда $p(x)$ есть плотность распределения вероятности случайной переменной X . Предполагается, что N – достаточно большое и, следовательно, гистограмма достаточно регулярна, так что для всех функций $\phi(x)$ справедливо выражение:

$$\frac{1}{N} \sum_{m=0}^{N-1} \phi(a[m]) = \sum_x \phi(x) h[x] \approx \int_{-\infty}^{+\infty} \phi(x) p(x) dx = E\{\phi(X)\}. \quad (9.7)$$

Данная гипотеза выполняется для гистограмм большинства «естественных» изображений. Это эквивалентно тому, что коэффициенты $a[m]$ есть случайная последовательность X . Заменяв $\phi(x) = |x - Q(x)|^2$, из (9.7) получается

$$\frac{D}{N} = \frac{1}{N} \sum_{m=0}^{N-1} |a[m] - Q(a[m])|^2 = E\{X - Q(X)\}^2.$$

Пусть \bar{R} есть среднее число бит на коэффициент для кодирования $Q(a[m])$. Если Q является квантователем с высоким разрешением с шагом Δ , то из (9.3) вытекает формула, аналогичная (9.4):

$$\frac{D(\bar{R})}{N} = \frac{1}{12} 2^{2H_s(x)} 2^{-2\bar{R}}. \quad (9.8)$$

Как правило, многие вейвлет-коэффициенты $a[m] = \langle f, g_m \rangle$ близки к нулю, и малое их количество имеет большую амплитуду. Поэтому $p(x)$ имеет острый пик при $x = 0$. Если шаг квантования Δ велик, то $p(x)$ имеет флюктуирующие значения в интервале $\left[-\frac{\Delta}{2}, \frac{\Delta}{2}\right]$, где коэффициенты квантуются в нуль. Следовательно, в этом интервале не выполняется гипотеза о квантовании с высоким разрешением. Это объясняет то, что функция $\log_2 D(\bar{R})$ имеет спад порядка $-2\bar{R}$ только для $\bar{R} \geq 1$.

На практике для квантования вейвлет-коэффициентов зачастую применяется «почти» равномерный квантователь. Все интервалы квантования, кроме нулевого, имеют равные размеры Δ , а интервал возле нуля $[-T, T]$ - больше, с отношением $\theta = \frac{T}{\Delta}$, которое может быть вычислено (обычно $\theta = 1$). Для $|x| > T$ флюктуации $p(x)$ в каждом интервале квантования относительно малы, и можно считать, что гипотеза о высоком разрешении квантователя выполняется. Эта гипотеза есть всего лишь аппроксимация, но она достаточно точно описывает свойства квантователя для проведения точных вычислений вплоть до очень низких скоростей кодирования. Для $x \in [-T, T]$ гипотеза о квантовании с высоким разрешением не выполняется.

Коэффициенты не квантуются в нуль, если они превышают некоторый порог $|a[m]| > T$. Такие коэффициенты обычно называют значимыми. Кодирование положения ненулевых коэффициентов эквивалентно сохранению бинарной карты значений, определяемой как

$$b[m] = \begin{cases} 0, & \text{если } |a[m]| \leq T \\ 1, & \text{если } |a[m]| > T \end{cases} . \quad (9.9)$$

Вейвлет-кодер изображения может использовать кодер длин серий для кодирования карты значений. Применяются также более эффективные методы кодирования карты, получившие название алгоритмов нульдеревя.

Пусть R_0 - число бит, требуемое для представления карты значений. Пусть M - число значащих коэффициентов. Можно записать пропорцию $p = \frac{M}{N}$ индексов m таких, что $b[m] = 1$. Верхнюю границу для R_0 можно получить в предположении, что в позициях «1» и «0» в карте значений нет избыточности. Среднее количество бит для кодирования позиции одного коэффициента тогда будет равным энтропии бинарного источника, символы которого с вероятностью $p = \frac{M}{N}$ принимают значения 1 и с вероятностью $1 - p$ принимают значение 0:

$$\frac{R_0}{N} \leq -p \log_2 p - (1 - p) \log_2 (1 - p). \quad (9.10)$$

Для $x \in (0, 1]$ справедливо неравенство $-x \log_2 x \leq (1 - x) \log_2 e$, так что среднее количество бит на значащий коэффициент для кодирования карты значений будет

$$r_0 = \frac{R_0}{M} \leq \log_2 \frac{N}{M} + \log_2 e . \quad (9.11)$$

Для вейвлет-коэффициентов, когда отношение $\frac{M}{N}$ мало, на выходе кодера длин серий средняя битовая скорость $r_0 = \frac{R_0}{M}$ значительно меньше верхней границы (9.11), так как существует значительная избыточность в позициях нулевых коэффициентов. Для большого класса изображений расчеты показывают, что $r_0 = \frac{R_0}{M}$ меняется медленно относительно $\frac{M}{N}$.

Амплитуды M значащих коэффициентов равномерно квантуются с шагом Δ , и эти квантованные значения подвергаются энтропийному кодированию. Пусть R_1 - число бит, необходимое для кодирования этих коэффициентов. Для $M \gg 1$ M значащих коэффициентов $a[m]$ имеют нормализованную гистограмму, то есть

$$p_T(x) = \frac{N}{M} p(x) 1_{\{|x|>T\}}. \quad (9.12)$$

Пусть X_T есть случайная переменная, чья плотность распределения вероятности - $p_T(x)$. Так как к значащим коэффициентам применима гипотеза квантования с высоким разрешением, среднее число бит для кодирования амплитуды каждого значащего коэффициента, обозначаемое r_1 , вычисляется из (9.2):

$$r_1 = \frac{R_1}{M} = H_d(X_T) - \log_2 \Delta. \quad (9.13)$$

В целом кодирование с преобразованием требует $R = R_0 + R_1$ бит.

Для получения оценки ошибки квантования $D = \|f - \hat{f}\|^2$ (9.6) суммарное искажение делится на две части: искажение, возникающее в силу квантования незначащих коэффициентов в нуль (D_0), и искажение, возникающее в силу квантования значащих коэффициентов (D_1): $D = D_0 + D_1$, где

$$D_0 = \sum_{|a[m]| \leq T} |a[m]|^2 \quad (9.14)$$

и

$$D_1 = \sum_{|a[m]| > T} |a[m] - Q(a[m])|^2. \quad (9.15)$$

Средняя ошибка квантования на значащий коэффициент $\frac{D_1}{M}$ вычисляется с учетом гипотезы о квантовании с высоким разрешением:

$$\frac{D_1}{M} = \frac{1}{M} \sum_{|a[m]| > T} |a[m] - Q(a[m])|^2 = E\{X_T - Q(X_T)\}^2 = \frac{\Delta^2}{12}. \quad (9.16)$$

Для вычисления ошибки квантования незначащих коэффициентов рассматривается аппроксимация f посредством M векторов g_m из G , чьи ска-

лярные произведения с f имеют наибольшие амплитуды. Тогда искажение D_0 будет

$$D_0\left(\frac{M}{N}\right) = \|f - f_M\|^2 = \sum_{|a[m]| \leq T} |a[m]|^2. \quad (9.17)$$

Величина D_0 называется нелинейной ошибкой аппроксимации, так как векторов выбираются в зависимости от f . Оценка скорости убывания D_0 при увеличении M изучается в теории аппроксимации функций.

Если провести сортировку $a[m]$ по амплитуде, то k -й коэффициент может быть записан в виде

$$x\left(\frac{k}{N}\right) = |a[m_k]| \leq x\left(\frac{k+1}{N}\right) = |a[m_{k+1}]| \quad \text{для } 1 \leq k \leq N. \quad (9.18)$$

Ошибка аппроксимации D_0 есть сумма $N - M$ квадратов коэффициентов меньшей амплитуды:

$$D_0\left(\frac{M}{N}\right) = \sum_{k=M+1}^N \left|x\left(\frac{k}{N}\right)\right|^2. \quad (9.19)$$

Ошибка $D_0\left(\frac{M}{N}\right)$ убывает быстро при увеличении $\frac{M}{N}$, если $x(z)$ убывает быстро при увеличении z . Таким образом, можно рассматривать функцию $x(z)$ при $z \in [0,1]$, которая интерполирует значения $x\left(\frac{k}{N}\right)$.

Для оценки убывания $D_0(z)$ при увеличении z в теории аппроксимации функций $x(z)$ сравнивается с $z^{-\gamma}$ для некоторого $\gamma > 0$. Для функций, декомпозированных в вейвлет-базис, экспонента γ характеризует функциональные пространства Бесова. Однако предположение, что $x(z) = Cz^{-\gamma}$, не подходит для моделирования интересующего нас класса сигналов. Вместо этого вводится предположение о том, что эта экспонента медленно изменяется. Тогда можно определить

$$\gamma(z) = -\frac{d \log_2 x(z)}{d \log_2 z}. \quad (9.20)$$

Зависимость скорости от искажения при кодировании с низкими скоростями была получена С.Маллатом и Ф.Фальзоном. Для этого он предположил, что вторая производная не превосходит некоторого $\varepsilon > 0$:

$$\left| \frac{d^2 \log_2 x(z)}{(d \log_2 z)^2} \right| \leq \varepsilon \quad \text{для } z \in \left(0, \frac{1}{2} \right]. \quad (9.21)$$

Также предполагается, что

$$\gamma(z) > 0, \quad (9.22)$$

$$\frac{d^2 \log_2 x(z)}{(d \log_2 z)^2} \leq 0 \quad \text{для } z \in (0, 1), \quad (9.23)$$

и что $p(x)$ симметрична:

$$p(x) = p(-x). \quad (9.24)$$

Большинство реальных изображений удовлетворяют условиям (9.21)-(9.24). При выполнении этих условий справедлива следующая теорема.

Теорема 1. Предположим, что $x(z)$ удовлетворяет (9.21)-(9.24). Пусть $\gamma_M = \gamma\left(\frac{M}{N}\right) > \frac{1}{2}$. Если $\frac{M}{N} \leq \varepsilon$ и $M \geq \frac{1}{\varepsilon}$, то

$$D(\bar{R}) = (1 + K)D_0 \left(\frac{\bar{R}}{r_1 + r_0} \right), \quad (9.25)$$

где

$$K = \frac{D_1}{D_0} = \frac{2\gamma_M - 1}{12\theta^2} \left[1 + O(\varepsilon |\log_2 \varepsilon|^2 + \varepsilon^{2\gamma_M - 1}) \right] \quad (9.26)$$

и

$$r_1 = \frac{R_1}{M} = 1 + (1 + \gamma_M) \log_2 e + \log_2 \gamma_M + \log_2 \theta + O(\varepsilon). \quad (9.27)$$

Для практического применения данной теоремы формулы могут быть упрощены за счет пренебрежения коэффициентом ε . Так как вторая производная (9.21) мала, наклон

$$\gamma_M = \frac{-\log_2 x \left(\frac{M}{N} \right)}{d \log_2 z} \quad (9.28)$$

изменяется медленно, как функция от $\log_2 \frac{M}{N}$. В диапазоне сжатия изображений его можно считать постоянным: $\gamma_M \approx \gamma$. Из (9.26) и (9.27) следует, что $K = \frac{D_1}{D_0}$ и $r_1 = \frac{R_1}{M}$ также постоянны. Как уже было отмечено, в этом случае $r_0 = \frac{R_0}{M}$. Следовательно, $D(\bar{R})$ вычисляется в (9.25) путем масштабирования и умножения нелинейной ошибки аппроксимации $D_0(z)$ на постоянные множители:

$$D(\bar{R}) = (1 + K) D_0 \left(\frac{\bar{R}}{r_1 + r_0} \right). \quad (9.29)$$

Так как $\gamma_M \approx \gamma$, из (9.28) следует, что $D_0(z) \sim z^{2\gamma-1}$ и $D(\bar{R}) \sim \bar{R}^{1-2\gamma}$.

Получившаяся зависимость скорости от искажения значительно отличается от известной формулы для гипотезы квантования с высоким разрешением, где $D(\bar{R}) \sim 2^{-2\bar{R}}$.

Искажение D в (9.29) существенно зависит от ошибки аппроксимации D_0 сигнала f M векторами ($M = \frac{\bar{R}}{r_1 + r_0}$), выбранными в базисе G . Для оптимального кодирования с преобразованием базис G должен точно аппроксимировать каждый из сигналов f малым числом базисных векторов. Если рассматривать f как реализацию случайного вектора Y , желательно было бы найти базис, минимизирующий $E \left\{ D_0 \left(\frac{M}{N} \right) \right\}$ по всем реализациям. Из теории известно, что оптимальным для аппроксимации сигнала Y с использованием M векторов является базис Карунена-Лозва, но в данном случае это свойство оптимальности теряется, так как число векторов меняется в зависимости от реализации. В некоторых случаях известно, как найти базис, который минимизирует максимальную ошибку $D_0 \left(\frac{M}{N} \right)$ для целого класса сигналов. Например, базис вейвлетов является оптимальным в этом минимаксном смысле

для кусочно-регулярных сигналов, которые описываются пространствами Бесова.

9.2.2. Оптимальный относительный размер интервала квантования

Для оптимизации квантования необходимо выбрать размер интервала возле нуля $[-T, T]$ относительно других интервалов квантования Δ . Для минимизации искажения D необходимо найти $\theta = \frac{T}{\Delta}$, такое что для фиксированной \bar{R}

$$\frac{\partial D(\bar{R}, \theta)}{\partial \theta} = 0. \quad (9.30)$$

Можно показать, что справедлива следующая теорема.

Теорема 2. Предположим, что $x(z)$ удовлетворяет условиям (9.21)-(9.24) и что $r_0 = \frac{R_0}{M}$ - постоянная, не зависящая от M . Пусть $\gamma_M = \gamma\left(\frac{M}{N}\right) > \frac{1}{2}$. Если $\frac{M}{N} \leq \varepsilon$ и $M \geq \frac{1}{\varepsilon}$, то оптимальное отношение

$$\theta = \sqrt{\frac{r_1 + r_0}{6 \log_2 e} - \frac{2\gamma_M - 1}{12} [1 + O(\varepsilon)]}. \quad (9.31)$$

9.2.3. Практическая проверка точности аналитических выражений

Известно, что вейвлет-базис эффективно аппроксимирует кусочно-регулярные функции малым числом ненулевых коэффициентов. Так как изображения часто имеют кусочно-регулярные структуры, вейвлет-преобразование с успехом применяется в вейвлет-кодеках изображения. Основным положением теоремы 1 является то, что сортированные коэффициенты разложения $x(z)$ сигнала f в базис G имеют рациональный спад.

Известна модель изображения, основанная на пространствах Бесова. Пусть $f \in L^2[0, N]^2$. Если существует $C > 0$ и $\gamma > \frac{1}{2}$ такие, что для всех $k \geq 0$ вейвлет-коэффициент f ранга k ограничен $Ck^{-\gamma}$, то f относится к семейству пространств Бесова, чьи индексы зависят от γ . Рассмотрим кусочно-регулярное изображение f , которое является регулярным внутри регионов $\{\Omega_i\}_{i \leq k \leq K}$, делящих $[0, N]^2$. Это изображение имеет неоднородности вдоль

границ Ω_i , которые имеют конечную длину. Можно доказать, что сортированные вейвлет-коэффициенты убывают как $Ck^{-\gamma}$, с $\gamma = 1$. Эта модель применима к достаточно низкочастотным изображениям (типа всем известного изображения LENA), тогда как неоднородности высокочастотных изображений приводят к изменению экспоненты γ . Для изображений $k \leq N$ поэтому возможна нормировка $z = \frac{k}{N}$ и сравнение убывания $x(z)$ с $z^{-\gamma}$ при увеличении z в диапазоне $[0,1]$. Условие (9.21) предполагает медленное изменение γ , как функции от $\log_2 z$.

Можно выполнить практическую проверку точности аналитической формулы, даваемой теоремой 1. Для проверки используется вейвлет-кодек с равномерным квантователем, интервал возле нуля $[-T, T]$ которого вдвое больше остальных интервалов, то есть $\theta = \frac{T}{\Delta} = 1$. Это является стандартным выбором для большинства вейвлет-кодексов изображения. Карта значений сжимается кодером длин серий. Отношение $\frac{D_1}{D_0}$ было вычислено для кодирования тестового изображения LENA. На рис.9.1 это значение сравнивается с теоретической оценкой (9.26) без учета коэффициента ε : $K = \frac{2\gamma_M - 1}{12\theta^2}$. Наклон γ_M в (9.28) вычисляется из сортированных вейвлет-коэффициентов $x(z)$. Как видно из рис.9.1, K точно аппроксимирует истинное значение $\frac{D_1}{D_0}$ для $\frac{M}{N} \leq 2^{-4}$. Возрастание ошибки аппроксимации при $\frac{M}{N} > 2^{-4}$ объясняется невыполнением гипотезы $\frac{M}{N} \leq \varepsilon$.

На рис.9.2 показано значение $\frac{R_1}{M}$, вычисленное из энтропии квантованных вейвлет-коэффициентов. Теоретическая оценка (9.27) также показана на рисунке. Кривые близки друг к другу, что подтверждает точность вычислений.

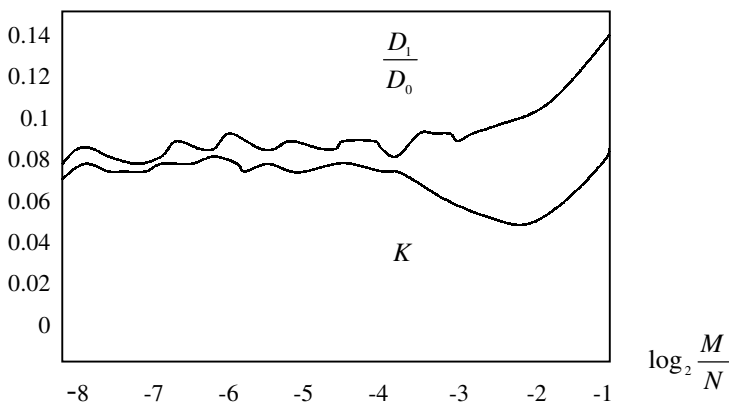


Рис.9.1. Сравнение $\frac{D_1}{D_0}$ с теоретической оценкой для вейвлет-кодека

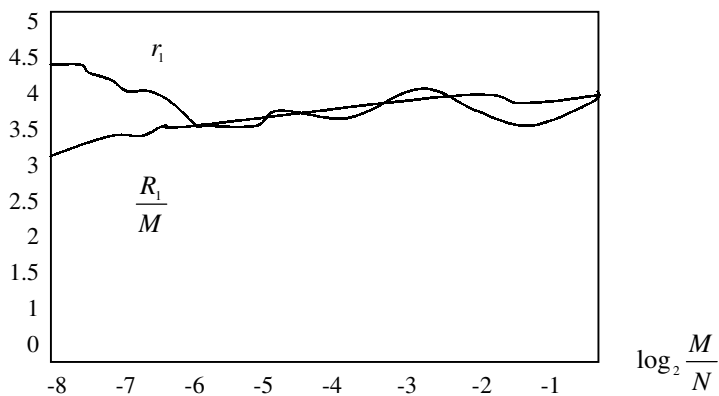


Рис.9.2. Сравнение $\frac{R_1}{M}$ с теоретической оценкой для вейвлет-кодека

Для упрощения вычисления зависимости скорости от искажения наклон γ_M аппроксимируется константой $\gamma_M \approx 1$, что соответствует кусочно-регулярной модели изображения. Хотя наклон отличен от единицы для высокочастотных изображений, эта аппроксимация оказывается достаточно точной в силу малой чувствительности K и $r_1 + r_0$ к флуктуациям γ_M . Так как $\theta = 1$, $K \approx \frac{1}{12}$ и $r_1 \approx 3.9$. На рис.9.3 показано $\frac{R}{M} = \frac{R_0}{M} + \frac{R_1}{M} = r_0 + r_1$, которое

было вычислено для различных тестовых изображений. Для $\frac{M}{N} \in [2^{-7}, 2^{-1}]$ отношение $\frac{R}{M}$ может быть аппроксимировано константой $r_0 + r_1 \approx 6.5$. Искажение D , вычисляемое в (9.25), приблизительно равно

$$\hat{D}(\bar{R}) = \left(1 + \frac{1}{12}\right) D_0\left(\frac{\bar{R}}{6.5}\right). \quad (9.32)$$

Отметим, что $\hat{D}(\bar{R})$ дает весьма точную оценку, несмотря на то, что наклон $\gamma_M \neq 1$.

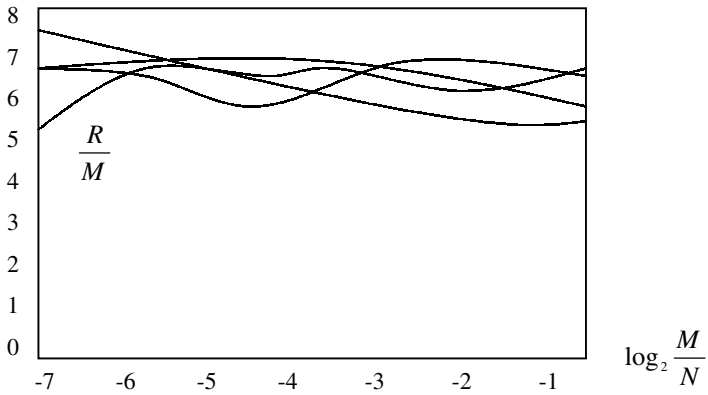


Рис.9.3. Флюктуации $\frac{R}{M}$ вейвлет-кодирования изображений.

Теорема 2 дает аналитическое выражение для θ , минимизирующего D . При $r_0 + r_1 \approx 6.5$ и $\gamma_M \approx 1$ $\theta = 0.81 \approx 1$. Таким образом, теоретически доказывается правильность выбора исследователями параметра θ .

Итак, при кодировании с низкими скоростями функция скорость-искажение для вейвлет-кодексов может быть вычислена путем отделения коэффициентов, квантуемых в нуль, от других. Полученная функция скорость-искажение зависит, прежде всего, от точности нелинейной аппроксимации изображения малым числом базисных коэффициентов. Для вейвлет-базиса при $\bar{R} \leq 1$ $D(\bar{R})$ убывает как $C\bar{R}^{1-2\gamma}$, где экспонента γ имеет порядок 1.

Таким образом, полученные выражения значительно отличаются от известных из теории информации. Это объясняется тем, что при достаточно сильном сжатии интервалы квантования становятся большими, и не выполняется предположение о квантовании с высоким разрешением.

ПРИМЕНЕНИЕ ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЯ ДЛЯ СЖАТИЯ ИЗОБРАЖЕНИЯ

В последнее десятилетие в мире наблюдается значительный интерес к сжатию изображений. Это вызвано стремительным развитием вычислительной техники, графических мониторов, цветных принтеров, а также цифровой техники связи. Изображение представляется в цифровом виде достаточно большим количеством бит. Так, цветная картинка размером 512x512 требует для своего хранения 768 кБайт. Если передавать видеопоследовательность таких картинок со скоростью 25 кадров в секунду, требуемая скорость составит 188.7 Мбит/с.

Различают сжатие изображений без потерь и с потерями. Первое характеризуется незначительными коэффициентами сжатия (от 3 до 5 раз) и находит применение в телевидении, медицине, аэрофотосъемке и других приложениях. При сжатии изображения с допустимыми потерями коэффициент сжатия может достигать сотен раз. Популярность вейвлет-преобразования (ВП) во многом объясняется тем, что оно успешно может использоваться для сжатия изображения как без потерь, так и с потерями. Так, коэффициент сжатия видеосигнала в видеокодеках семейства ADV6xx варьируется от 3 до 350 и больше раз.

Причин успешного применения несколько.

1. Известно, что вейвлет-преобразование хорошо аппроксимирует преобразование Карунена-Лоэва для фрактальных сигналов, к которым относятся и изображения.

2. Дисперсии коэффициентов субполос ортонормального вейвлет-преобразования распределены в широком диапазоне значений. Пусть дисперсии кодируются простым энтропийным кодером. Тогда стоимость кодирования всего изображения есть сумма кодирования субполос. Различные энтропии субполос приведут к стоимости кодирования значительно меньшей, чем при непосредственном кодировании изображения.

3. В результате этого перераспределения дисперсий коэффициенты вейвлет-преобразования имеют существенно негауссовскую статистику и, таким образом, меньшую энтропию, чем гауссовский сигнал той же дисперсии.

4. Наконец, коэффициенты вейвлет-декомпозиции имеют регулярные пространственно-частотные зависимости, которые с успехом используются в ряде алгоритмов кодирования.

Рассмотрим основные проблемы, возникающие при сжатии изображения при помощи вейвлет-преобразования, и возможные пути их решения.

10.1. Базовый вейвлет-кодер изображения

Вейвлет-кодер изображения устроен так же, как и любой другой кодер с преобразованием. Назовем такой кодер базовым. Он состоит из трех основных частей: декоррелирующее преобразование, процедура квантования и энтропийное кодирование. В настоящее время во всем мире проводятся исследования по усовершенствованию всех трех компонент базового кодера. В следующих разделах будут рассмотрены современные, более сложные и эффективные кодеры. Так как в их основе лежит базовый кодер, рассмотрим подробнее его составляющие части.

10.1.1. Выбор вейвлетов для сжатия изображения

Выбор оптимального базиса вейвлетов для кодирования изображения является трудной и вряд ли решаемой задачей. Известен ряд критериев построения «хороших» вейвлетов, среди которых наиболее важными являются: гладкость, точность аппроксимации, величина области определения, частотная избирательность фильтра. Тем не менее, наилучшая комбинация этих свойств неизвестна.

Простейшим видом вейвлет-базиса для изображений является разделимый базис, получаемый сжатием и растяжением одномерных вейвлетов. Использование разделимого преобразования сводит проблему поиска эффективного базиса к одномерному случаю, и почти все известные на сегодняшний день кодеры используют его. Однако неразделимые базисы могут быть более эффективными, чем разделимые.

Прототипами базисных функций для разделимого преобразования являются функции $\phi(x)\phi(y)$, $\phi(x)\psi(y)$, $\psi(x)\phi(y)$ и $\psi(x)\psi(y)$. На каждом шаге преобразования выполняется два разбиения по частоте, а не одно. Предположим, имеем изображение размером $N \times N$. Сначала каждая из N строк изображения делится на низкочастотную и высокочастотную половины. Получается два изображения размерами $N \times N/2$. Далее, каждый столбец делится аналогичным образом. В результате получается четыре изображения размерами

$N/2 \times N/2$: низкочастотное по горизонтали и вертикали, высокочастотное по горизонтали и вертикали, низкочастотное по горизонтали и высокочастотное по вертикали и высокочастотное по горизонтали и низкочастотное по вертикали. Первое из вышеназванных изображений делится аналогичным образом на следующем шаге преобразования, как показано на рис.10.1.

Известно, что для кодирования изображений хорошо подходят сплайновые вейвлеты. Эксперименты, проведенные рядом исследователей, показывают важность гладкости базисных функций для сжатия. Практически столь же большое значение имеет число нулевых моментов вейвлетов, которое тесно связано с гладкостью. Несмотря на это, некоторые исследователи считают, что важность гладкости для приложений цифровой обработки сигналов остается открытым вопросом. Наиболее широко на практике используют базисы, имеющие от одной до двух непрерывных производных. Увеличение гладкости не приводит к увеличению эффективности кодирования.

HCH_2	VCH_2	VCH_1
HCV_2	VCV_2	
HCV_1		VCV_1

Рис.10.1. Два уровня вейвлет-преобразования изображения

Д.Вилласенор систематически протестировал все биортогональные блоки фильтров минимального порядка с длиной фильтров ≤ 36 . В дополнение к вышеперечисленным критериям учитывалась также чувствительность аппроксимации с низким разрешением к сдвигам функции $f(x)$. Наилучшим фильтром, найденным в этих экспериментах, оказался сплайновый фильтр

7/9. Этот фильтр наиболее часто используется в вейвлет-кодерах изображений. В частности, в видеокодеках семейства ADVbxx применяются именно эти фильтры.

Необходимо сделать одно замечание относительно этих результатов. Д.Вилласенор сравнивал пиковое отношение сигнал/шум, получаемое при использовании различных фильтров в простой схеме кодирования. Алгоритм размещения бит, применяемый им, хорошо работает с ортогональными базами. В случае биортогональных фильтров должен применяться другой, более эффективный алгоритм. В силу этой причины некоторые заслуживающие внимания биортогональные фильтры были им обойдены.

Для биортогонального преобразования квадрат ошибки в области преобразования не равен квадрату ошибки в восстановленном изображении. В результате проблема минимизации ошибки становится намного более трудной, чем в ортогональном случае. Можно уменьшить ошибку в области изображения путем применения схемы взвешенного распределения бит, рассматриваемой в пункте 10.1.3. Тогда целый ряд фильтров по своей эффективности становится равным фильтру 7/9. Один из таких базисов – интерполирующий вейвлет Деслаури-Дубук порядка 4, преимуществом которого является то, что коэффициенты фильтра – рациональные числа, кратные степени 2. Оба этих вейвлета имеют 4 нулевых момента и две непрерывные производные.

Семейство многообещающих фильтров было разработано И.Баласингамом и Т.Рамстадом. Процедура разработки заключалась в комбинировании классических методов разработки фильтров с идеями теории вейвлетов. Получившиеся фильтры значительно превосходят популярные фильтры 7/9.

10.1.2. Осуществление преобразования на границах изображения

Для эффективного сжатия необходимо тщательно обрабатывать границы изображения. Методы осуществления вейвлет-преобразования, учитывающие границы, были описаны в главе 3. Альтернативным методом является конструирование граничных фильтров, сохраняющих ортогональность преобразования вблизи границы. Проблеме конструирования граничных фильтров посвящен ряд статей Е.Ковачевич. Как было показано в главе 6, при применении лифтинговой схемы границы учитываются автоматически.

10.1.3. Квантование

В большинстве вейвлет-кодеров применяется скалярное квантование. Существуют две основные стратегии выполнения скалярного квантования. Если заранее известно распределение коэффициентов в каждой полосе, оптимальным будет использование квантователей Ллойда-Макса с ограниченной энтропией для каждой субполосы. В общем случае подобным знанием мы не обладаем, но можем передать параметрическое описание коэффициентов путем посылки декодеру дополнительных бит. Априорно известно, что коэффициенты высокочастотных полос имеют обобщенное гауссовское распределение с нулевым матожиданием.

На практике обычно применяется намного более простой равномерный квантователь с «мертвой» зоной. Как показано на рис.10.2, интервалы квантования квантования имеют размер Δ , кроме центрального интервала (возле нуля), чей размер обычно выбирается 2Δ .

Коэффициенту, попавшему в некоторый интервал, ставится в соответствие значение центра этого интервала. В случае асимптотически высоких скоростей кодирования равномерное квантование является оптимальным. Хотя в практических режимах работы квантователи с «мертвой» зоной субоптимальны, они работают почти так же хорошо, как квантователи Ллойда-Макса, будучи намного проще в исполнении. Кроме того, они робастны к изменениям распределения коэффициентов в субполосе. Дополнительным их преимуществом является то, что они могут быть вложены друг в друга для получения вложенного битового потока.

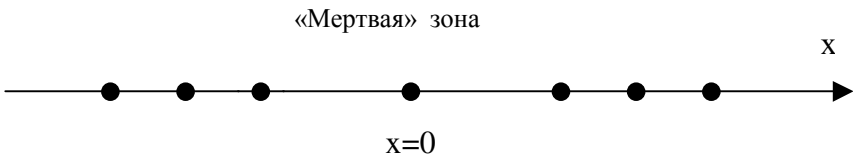


Рис. 10.2. Равномерный квантователь с «мертвой» зоной

10.1.4. Энтропийное кодирование

Субоптимальное энтропийное кодирование коэффициентов можно осуществить при помощи алгоритма арифметического кодирования. Кодеру требуется оценить распределение квантованных коэффициентов. Эта оценка

получается путем аппроксимации распределения коэффициентов гауссовской или лапласовской плотностью и вычисления параметров распределения. Оценка параметров может также производиться и в процессе работы, «на ходу». Такой подход имеет то преимущество, что кодер учитывает локальные изменения статистики изображения. Известны эффективные адаптивные процедуры оценивания.

Так как изображение не является случайным гауссовским процессом, коэффициенты преобразования, хотя и некоррелированные, обладают определенной структурой. Энтропийный кодер может использовать эту структуру, осуществляя некоторое предсказание. В ряде работ отмечено, что применение предсказания приводит к незначительному повышению эффективности.

На практике зачастую вместо арифметического кодера используют кодер Хаффмана. Причина этого заключается в меньшем требуемом объеме вычислений, а также в том, что алгоритмы арифметического кодирования запатентованы. Так, только фирма IBM обладает более чем 90 патентами различных вариаций этого кодера. В силу этого в видеокодеках ADVbxx применен кодер Хаффмана.

10.1.5. Распределение бит

Итак, последним вопросом, на который необходимо ответить при создании алгоритма сжатия, является следующий: как точно квантовать каждую из субполос? На этот вопрос дает ответ алгоритм распределения бит. Общая идея заключается в определении такого числа бит R_j , отводимых для кодирования j субполосы, при котором суммарное искажение $\sum_j D_j(R_j)$ было бы минимальным с учетом ограничения $\sum_j R_j \leq R$. Если известен точный вид функции $D_j(R)$, проблема решается с использованием условий Куна-Тукера. Одно из решений заключается в аппроксимации функции $D_j(R)$ функцией скорость-искажение для гауссовского источника. Однако при низких скоростях кодирования эта аппроксимация будет неточна. Лучшие результаты могут быть получены путем измерения $D_j(R)$ в диапазоне изменения R и решения проблемы ограниченной оптимизации с применением метода целочисленного программирования. Данная задача была решена И.Шохамом и А.Гершо.

В случае применения биортогональных вейвлетов возникает дополнительная трудность, заключающаяся в неравенстве среднеквадратической

ошибки (СКО) области изображения и области трансформанты. П.Мулином был сформулирован алгоритм кратномасштабного ослабления, который дает приближенное решение этой проблемы. Данный алгоритм значительно эффективнее обычной минимизации СКО в области трансформанты. В главе 10 будет описан метод распределения бит, примененный в ADV6xx.

Более простым методом является аппроксимация СКО изображения взвешенной суммой СКО субполос. Вес ω_j для субполосы j находится следующим образом: устанавливаем один коэффициент этой полосы в 1, а остальные – в 0. Затем выполняем обратное преобразование. Вес ω_j равен сумме квадратов получившихся значений. Распределение бит производится с целью минимизации взвешенной суммы $\sum_i \omega_j D_j(R_j)$. Процедура взвешивания дает хорошие результаты, когда используются неортогональные вейвлеты, например вейвлеты Деслари-Дубук, ставшие популярными благодаря лифтинговой схеме (глава 6). Для фильтра 7/9 веса ω_j близки к 1, поэтому взвешивание в данном случае нецелесообразно.

10.1.6. Меры искажения, взвешенные с учетом восприятия человеком

СКО не всегда хорошо согласуется с визуально наблюдаемой ошибкой. Рассмотрим, например, два изображения, которые полностью одинаковы, кроме небольшой области. Хотя визуально разность между этими изображениями хорошо заметна, СКО будет примерно одинаковой. Учет системы человеческого зрения в схеме сжатия является трудной задачей. Было проведено множество исследований, но в силу трудностей с математическим описанием системы зрения человека подходящей меры найдено не было.

Известно, что в человеческом глазу выполняется операция многомасштабного представления изображений. Глаз более чувствителен к искажениям в низкочастотной области. Отсюда существует возможность улучшения визуального качества реконструированного изображения путем взвешивания СКО субполос в соответствии с чувствительностью глаза в различных частотных диапазонах. Веса для наиболее часто используемого фильтра 7/9 были вычислены А.Ватсоном.

10.2. Новые идеи в области сжатия изображений, связанные с вейвлет-преобразованием

Базовый вейвлет-кодер, описанный в разделе 10.1, использует общие принципы кодера с преобразованием, то есть основан на эффектах декорреляции и перераспределения энергии. Математическая теория вейвлет-преобразования позволяет создавать совершенно новые и эффективные методы сжатия. Эти методы лежат в основе алгоритмов, описываемых в разделах 10.3 и 10.5. В данном разделе покажем главные идеи этих методов.

Кодирование с преобразованием основано на том, что большая часть энергии сосредоточивается в малом количестве коэффициентов, которые квантуются в соответствии с их значением. Эта парадигма, являясь достаточно мощной, основывается на нескольких предположениях, которые не всегда верны. В частности, предполагается, что изображение порождается гауссовским источником, что не соответствует действительности. С.Маллат и Ф.Фальзон показали, как это несоответствие приводит к неверным результатам при кодировании с низкими скоростями.

Пусть $Y[n]$ - случайный вектор длиной N , определенный как

$$Y[n] = \begin{cases} X, & \text{если } n = P, \\ X, & \text{если } n = P + 1 \pmod{N}, \\ 0, & \text{в остальных случаях.} \end{cases} \quad (10.1)$$

Здесь P есть случайная целая величина, равномерно распределенная от 0 до $N - 1$, а X - случайная величина, с равной вероятностью принимающая значения 1 и -1. X и P - независимы. Вектор Y имеет нулевое среднее и ковариационную матрицу с элементами

$$E\{Y[n]Y[m]\} = \begin{cases} \frac{2}{N} & \text{для } n = m, \\ \frac{1}{N} & \text{для } |n - m| \in \{1, N - 1\}, \\ 0 & \text{в остальных случаях.} \end{cases} \quad (10.2)$$

Ковариационная матрица является циркулянтной, так что преобразованием Карунена-Лоэва для нее является просто преобразование Фурье. Однако преобразование Фурье вектора Y очень неэффективно с точки зрения коди-

рования. Энергия на частоте k будет равна $\left|1 + e^{2\pi \frac{k}{N}}\right|$. Это означает, что энергия Y распределена по всей низкочастотной половине базиса Фурье и частично – по высокочастотной половине. Таким образом, преобразование Карунена-Лозва «упаковало» энергию двух ненулевых коэффициентов в примерно $\frac{N}{2}$ коэффициентов. Конечно, было бы выгоднее кодировать Y в исходном виде, без всякого преобразования.

Как видно из этого примера, традиционное кодирование с преобразованием может быть улучшено путем введения операторов выбора. Вместо квантования коэффициентов трансформанты в заранее определенном порядке вейвлет-преобразование позволяет выбирать нужные для кодирования элементы. Это становится возможным главным образом благодаря тому, что базис вейвлетов компактен в частотной и пространственной областях. В вышеприведенном примере энергия сигнала была пространственно, но не частотно компактна. Значит, необходимо использовать соответствующий оператор выбора вейвлет-коэффициентов, наиболее эффективно представляющих сигнал. Наиболее значительным результатом этого подхода является создание алгоритма нульдеревя и его разновидностей (раздел 10.3).

Вообще говоря, развитие идей кодирования с преобразованием заключается в снятии ограничения на линейную аппроксимацию изображения, так как оператор выбора является нелинейным. В работах Р.Девора, С.Маллата и Ф.Фальзона показано, что проблема кодирования изображения может быть эффективно решена в рамках теории нелинейной аппроксимации. Отсюда возникает и ряд различий в алгоритмах работы традиционных и вейвлет-кодеров. В случае линейной аппроксимации изображение представляется фиксированным числом базисных векторов Карунена-Лозва. Далее, какое-то число малых коэффициентов трансформанты приравнивается к нулю. Идея нелинейной аппроксимации заключается в аппроксимации изображения путем адаптивного выбора самих базисных функций. Информация о выбранных базисных функциях хранится в бинарной карте значений и передается декодеру, как дополнительная информация. В разделе 10.3 будут описаны нульдеревья, являющиеся исключительно важной структурой данных для кодирования карты значений.

Рассмотренный выше пример показал что изображение неправомерно считать порожденным одиночным гауссовским источником. Для получения большей компактности энергии необходимо адаптировать преобразование к какому-то конкретному, а не к целому классу изображений. В случае если

источник описывается смесью различных распределений, преобразование Карунена-Лоэва не является больше эффективным. В главе 5 были описаны частотно-адаптивные и пространственно-частотно-адаптивные кодеры, в которых происходит разложение изображения в большое количество базисов и выбор из них оптимального по некоторому критерию.

Решетчатое квантование коэффициентов, рассматриваемое в разделе 10.5, гораздо ближе по своей сути к векторному квантованию, чем к кодированию с преобразованием.

Таблица 10.1

Сравнение кодеров, описываемых в главе по отношению сигнал/шум

Тип кодера	LENA (бит/пиксел)			BARBARA (бит/пиксел)		
	1.0	0.5	0.25	1.0	0.5	0.25
JPEG	37.9	34.9	31.6	33.1	29.3	25.2
Оптимизированный JPEG	39.6	35.9	32.3	35.9	30.6	26.7
Базовый вейвлет-кодер	39.4	36.2	33.2	34.6	29.5	26.6
Нульдереве (Шапиро)	39.6	36.3	33.2	35.1	30.5	26.8
Нульдереве (Саид и Перельман)	40.5	37.2	34.1	36.9	31.7	27.8
Нульдереве (R/D оптимизирован)	40.5	37.4	34.3	37.0	31.3	27.2
Частотно-адаптивный	39.3	36.4	33.4	36.4	31.8	29.2
Пространственно- частотно-адаптивный	40.1	36.9	33.8	37.0	32.3	29.7
Частотно-адаптивный + нульдереве	40.6	37.4	34.4	37.7	33.1	29.3
Решетчатое квантование	41.1	37.7	34.3	-	-	-
Обратное оценивание	40.9	37.7	34.6	-	-	-

Итак, развитие идей кодирования с преобразованием заключается в основном во введении некоторого оператора выбора. Информация о выборе должна быть передана декодеру, как дополнительная информация. Она может быть в виде нульдеревьев или в виде обобщенных классов энергии. Метод «обратного оценивания распределения», предложенный К.Рамчандраном, основан на другом подходе. Считается, что дополнительная информация является избыточной и может быть получена декодером непосредственно из данных. Использование данного метода приводит к хорошим показателям кодирования.

В табл.10.1 (см. стр.155) представлены сравнения пикового отношения сигнал/шум для кодеров, которые будут обсуждаться далее. В качестве тестовых использовались полутонные портретные изображения размером 512x512.

Визуальное сравнение восстановленных изображений показывает, что лучшие результаты дают методы, использующие нульдеревья для кодирования коэффициентов. В частности, в этих изображениях лучше выражены контуры и отсутствует размытость мелких деталей.

10.3. Кодирование посредством нульдеревья

Из теории кодирования с погрешностью известно, что оптимальное распределение бит достигается в случае, если сигнал поделен на субполосы, содержащие «белый» шум. Для реальных сигналов это достигается в случае неравномерной ширины субполос: в области НЧ они более узки, чем в области ВЧ. Вот почему вейвлет-преобразование обеспечивает компактность энергии.

Эта компактность энергии ведет к эффективному применению скалярных квантователей. Однако они не учитывают остаточную структуру, сохраняющуюся в вейвлет-коэффициентах, в особенности ВЧ субполос. Современные алгоритмы сжатия все тем или иным образом используют эту структуру для повышения эффективности сжатия.

Одним из наиболее естественных способов является учет взаимосвязей между коэффициентами из различных субполос. В высокочастотных субполосах имеются обычно большие области с нулевой или малой энергией. Области с высокой энергией повторяют от субполосы к субполосе свои очертания и местоположение. И это неудивительно – ведь они появляются вокруг контуров в исходном изображении – там, где вейвлет-преобразование не может адекватно представить сигнал, что приводит к «утечке» части энергии в ВЧ субполосы. Медленно изменяющиеся, гладкие области исходного изображения хорошо описывают НЧ вейвлет-базисы, что приводит к «упаковке» энергии в малом числе коэффициентов НЧ области. Этот процесс примерно повторяется на всех уровнях декомпозиции, что и приводит к визуальной «похожести» различных субполос.

Итак, априорное знание о том, что изображение состоит из гладких областей, текстур и контуров, помогает учитывать эту межполосную структуру. Кодеры, использующие структуру нульдеревя, сочетают учет структуры коэффициентов с совместным кодированием нулей, в результате чего получается очень эффективный алгоритм сжатия.

10.3.1. Алгоритм Льюиса и Ноулеса

Впервые идея нульдеревя была предложена А.Льюисом и Г.Ноулесом. В их алгоритме применялась древовидная структура данных для описания вейвлет-коэффициентов (рис.10.3). Такая структура получается в результате применения двухканального разделимого вейвлет-преобразования. Корневой узел дерева представляет коэффициент масштабирующей функции в самой НЧ области и имеет три отпрыска. Узлы дерева соответствуют вейвлет-коэффициентам масштаба, равного их высоте в дереве. Каждый из узлов имеет четыре отпрыска, соответствующих вейвлет-коэффициентам следующего уровня и того же пространственного расположения. Низом дерева являются листовые узлы, не имеющие отпрысков.

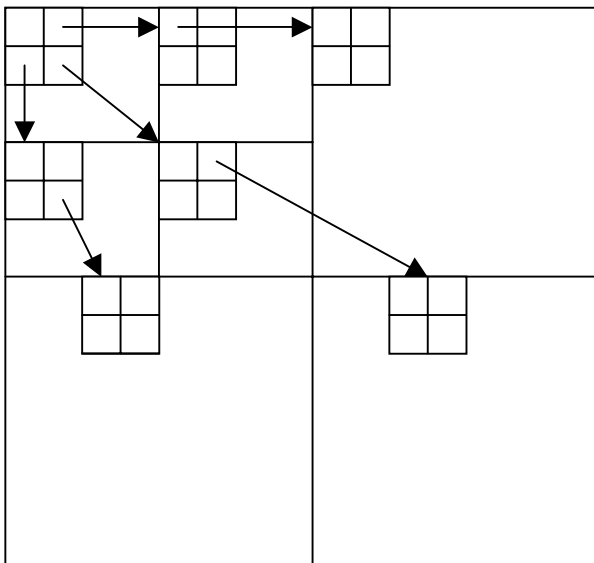


Рис.10.3. Зависимости между коэффициентами вейвлет-преобразования изображения, используемые в алгоритме нульдеревя

Для каждого из коэффициентов самой НЧ области существует три таких дерева, соответствующих трем порядкам фильтрации, как описано в пункте 10.1.1.

Квантование нульдеревом основано на наблюдении, что если коэффициент мал, его отпрыски на дереве зачастую тоже малы. Это объясняется тем, что значимые коэффициенты возникают вблизи контуров и текстур, которые локальны. Нетрудно увидеть, что это является разновидностью предсказания. А.Льюис и Г.Ноулес свели это предсказание к минимуму, предположив, что если какой-либо коэффициент незначимый, то все его потомки также будут незначимыми. Дерево или субдерево, которое содержит (по крайней мере, так предполагается) только незначимые коэффициенты, называется нульдеревом.

А.Льюис и Г.Ноулес использовали следующий алгоритм квантования вейвлет-коэффициентов. Вначале каждый узел квантуется квантователем, оптимальным для плотности распределения Лапласа. Если значение узла меньше некоторого порога, его потомки игнорируются. Эти потомки будут восстановлены декодером как нули. Иначе осуществляется переход к четырем отпрыскам узла, и процедура повторяется. Если узел не имеет отпрысков (является листом), начинает обрабатываться следующий корневой узел и т.д.

Данный алгоритм является эффективным в силу двух причин. Во-первых, в силу хорошей «упаковки» энергии вейвлет-преобразованием и, во-вторых, за счет совместного кодирования нулей. Для кодирования нулей обычно применяется кодер длин серий. Для повышения эффективности на вход этого кодера коэффициенты должны подаваться в определенном порядке. Например, в JPEG применено зигзагообразное сканирование. Наверное, наиболее важным вкладом этой работы была демонстрация того, что область вейвлет-коэффициентов прекрасно приспособлена для работы кодера длин серий. В самом деле, генерируются большие серии нулей и не надо передавать их длину, так как высота дерева известна. Аналогично JPEG, данный алгоритм является разновидностью скалярного/векторного квантования. Каждый (значимый) коэффициент квантуется отдельно, а символы, соответствующие малым коэффициентам, образуют вектор. Этот вектор состоит из символа нульдеревом и последовательности нулей длиной до конца дерева.

Характеристики алгоритма Льюиса и Ноулеса незначительно превосходят JPEG, хотя визуальное качество изображений лучше. Недостатком алгоритма является способ порождения и распознавания нульдеревьев. Как было отмечено, если коэффициент мал, то скорее всего его потомки будут малы, но

может быть, и нет. В случае если это не так, обнуляются значимые коэффициенты, и алгоритм Льюиса и Ноулеса ведет к большим искажениям.

Преимуществом этого алгоритма является его простота. Нульдеревья порождаются путем простого сравнения амплитуд коэффициентов, и не требуется дополнительной информации об их местоположении. Однако эта простота дается ценой невысокой эффективности. Детальный анализ этого взаимодействия привел к появлению следующего поколения кодеров с применением нульдеревьев.

10.3.2. Алгоритмы Шапиро и Саида-Перельмана

Идеи, лежащие в основе алгоритма Льюиса и Ноулеса, легли в основу многих современных кодеров изображения. Основным недостатком данного алгоритма является возможность ошибочного порождения нульдеревя, так как оно генерируется не из реальных данных, а на основе априорных предположений. Если потомки некоторого узла окажутся больше своего родителя (что нечасто, но все-таки бывает), алгоритм Льюиса и Ноулеса приводит к значительным искажениям.

Методы, обсуждаемые ниже, свободны от этого недостатка.

Шапиро разработал элегантный метод, названный алгоритмом вложенного нульдеревя (Embedded Zerotree Wavelet coder - EZW). Данный кодер основан на передаче и ненулевых данных, и карты значений. Биты, отводящиеся для кодирования карты значений, могут преваляировать в общем потоке бит, особенно на низких скоростях. Однако в карте значений, порождаемой изображениями, существует очень большая избыточность, которая и используется для достижения малых скоростей кодирования. Если имеется незначительный родительский узел, то очень вероятно, что потомки также будут незначимы. Так что в большинстве случаев генерируется символ нульдеревя. Если вероятность этого события p близка к 1, то количество информации $-p \log p$, содержащееся в нем, близко к нулю. Значит, среднее число бит, требующихся для кодирования карты значений, мало.

Если один или более потомков незначимого узла является значимым, генерируется символ «изолированного нуля». Вероятность этого события ниже, следовательно, для кодирования требуется большее количество бит. Это плата за то, чтобы не допустить значительного искажения за счет ошибочного порождения нульдеревя.

Алгоритм EZW генерирует вложенный, иерархический код. Подобные кодеры позволяют осуществить прогрессивную передачу изображения с по-

следовательным уточнением его на приеме. При этом изображение вначале аппроксимируется небольшим количеством бит, а потом эта аппроксимация уточняется. Вложенный код имеет то свойство, что при $R_1 > R_2$ код для R_2 будет префиксом кода для R_1 . Такие коды имеют большой практический интерес по следующим причинам:

- 1) возможность точного регулирования скорости передачи;
- 2) возможность восстановления всего изображения при прекращении приема декодером бит в любой точке. При этом изображение будет максимально хорошего качества для данного числа бит. Это применимо для передачи по каналам с потерями, а также для приложений вещания. В этом случае кодер генерирует высокоскоростной высококачественный поток, который передается по каналам различной пропускной способности декодерам различной вычислительной возможности. Последние выделяют из него нужные им субпотoki;
- 3) возможность быстрого просмотра изображений в удаленной базе данных. Для поиска достаточно и грубой копии, а при нахождении нужного изображения оно декодируется полностью.

Алгоритм Шапиро генерирует вложенный код побитовым способом (рис.10.4). В основе метода EZW лежат следующие основные операции.

Вначале выполняется частичное упорядочивание коэффициентов по амплитуде. Оно реализуется путем сравнения величины каждого вейвлет-коэффициента (ВК) с некоторым порогом T . Если $ВК > T$, то выносится решение о том, что коэффициент значимый, в противном случае – незначимый.

5	1	1	0	0	0	0	0	0	0	0	0	0	0
4	→	1	1	0	0	0	0	0	0	0	0	0	0
3	→	→	→	1	1	1	1	0	0	0	0	0	0
2	→	→	→	→	→	→	→	→	1	1	1	1	1
1	→	→	→	→	→	→	→	→	→	→	→	→	→
0	→	→	→	→	→	→	→	→	→	→	→	→	→

Рис.10.4. Битовый план для сканирования упорядоченных вейвлет-коэффициентов

Сканирование производится от низкочастотных полос к высокочастотным. Для кодирования знака и позиции всех коэффициентов используется двухбитный символ. Этот символ может быть: « \pm » - знак ВК; «0» – показывает, что ВК незначущий; «корень нулядеревя» - показывает, что ВК незначущий вместе со всеми ВК данной пространственной области из более высокочастотных полос. Таким образом, используется межполосная, пространственная корреляция ВК. После вычисления и передачи карты значений для значащих коэффициентов должны быть переданы биты, уточняющие их значение («карта данных»). Далее карта данных и карта значений сжимаются арифметическим кодером. В том случае, если не исчерпан ресурс скорости передачи, порог T делится на два и процесс повторяется.

Как видно из рис.10.4, верхние ряды бит содержат много нулей, так как многие коэффициенты имеют значение ниже порога. Роль нулядеревя заключается в предотвращении передачи этих нулей. Символ нулядеревя может снова и снова передаваться для данного коэффициента, пока он не станет больше текущего порога. После этого передается его квантованное значение.

А.Саид и В.Перельман улучшили алгоритм EZW. Их версия кодера называется «установка подразделений в иерархических деревьях» (Set Partition In Hierarchical Trees - SPIHT). Имеется общедоступная программная реализация этого кодера, которая очень быстра. Так, сжатие изображения размером 512×512 в 100 раз занимает на компьютере P-166 порядка 0.1 секунды. При этом качество восстановленного изображения весьма приемлемо. Вложенные кодеры обладают одной интересной особенностью: чем больше коэффициент сжатия, тем меньше время работы кодера. Это объясняется тем, что требуется осуществление меньшего числа уточнений. SPIHT превосходит EZW примерно на 0.3-0.6 дБ за счет кодирования не одиночных, а параллельных нулядеревьев.

Можно показать, что EZW и SPIHT являются членами большого семейства алгоритмов, в которых карта значений имеет древовидную структуру.

10.3.3. Оптимизация нулядеревьев по критерию скорость-искажение

В рассмотренных кодерах нулядеревья порождались только на основе анализируемых данных. Однако рассмотрим следующий гипотетический пример. Пусть изображение имеет большую равномерную область. Соответствующие ей вейвлет-коэффициенты будут малы, будет генерироваться нулядеревя, и на кодирование тратится малое число бит. Предположим теперь, что среди этой области имеется один резко отличающийся по значе-

нию пиксел. Этот пиксел приведет к появлению большого вейвлет-коэффициента, и нульдереву порождаться не будет.

Неточное кодирование одного пиксела не приведет к большому искажению изображения. В нашем примере эффективность кодера может быть существенно повышена путем игнорирования соответствующего коэффициента и построения нульдеревя. Возникает вопрос: каким образом определять, стоит ли отбрасывать коэффициенты, «мешающие» построению нульдеревя.

Введение нульдеревя для группы вейвлет-коэффициентов является, по сути, разновидностью квантования. Значения коэффициентов, которые мы кодируем посредством нульдеревя, не являются в общем случае нулевыми. Значимые коэффициенты также подвергаются квантованию. Если сэкономить часть бит путем порождения больших нульдеревьев, высвободившийся ресурс бит можно направить на более точное квантование значимых коэффициентов. Задачей является оптимальное распределение ограниченного ресурса бит между двумя видами квантователей для достижения меньшего искажения.

Эта задача решена с использованием хорошо известного метода распределения бит. Основным утверждением является то, что для случая оптимального распределения бит наклоны касательных к кривым скорость-искажение для всех квантователей равны. Наклон показывает, насколько искажение увеличивается/уменьшается при обнулении/передаче данного узла. Если один из квантователей имеет меньший наклон, это означает, что при его передаче искажение уменьшится меньше, чем при передаче других узлов. Следовательно, можно передать часть бит от этого квантователя другим. Таким образом, при повторении этой процедуры наклоны всех квантователей будут выровнены.

Ясно, что нульдеревья влияют на уровни квантования ненулевых коэффициентов, так как общий ресурс бит ограничен. Верно и обратное. Поэтому возможен итеративный алгоритм для оптимизации этих двух режимов квантования по критерию скорость-искажение. Вначале фиксируется скалярный квантователь, и ищется оптимальное нульдеревя. Затем оно фиксируется, и ищется оптимальный скалярный квантователь. З.Ксионг было доказано, что эта процедура сходится к локальному оптимуму.

Данный алгоритм незначительно превосходит по эффективности SPIHT, но обладает серьезными недостатками. Во-первых, он намного более сложен. Во-вторых, и, наверное, самое главное, он не порождает иерархический поток бит.

10.4. Частотно, пространственно-частотно-адаптивные кодеры

Данный класс кодеров использует адаптивные ортогональные преобразования, описанные в главе 5. Кодеры на базе вейвлет-пакетов были стандартизированы ФБР США, которое использует их для сжатия изображений отпечатков пальцев. Преимущество адаптивных кодеров проявляется при кодировании сильно нестационарных изображений. Недостатком является относительно высокая вычислительная сложность.

10.5. Использование зависимостей между вейвлет-коэффициентами внутри субполос

Разработка кодера EZW дала импульс к активным исследованиям алгоритмов нульдерев вейвлетов. Естественная простота структуры нульдерев, ее вычислительные преимущества, возможность порождать иерархический код – вот наиболее привлекательные черты этих алгоритмов. Модификация алгоритма нульдерев вейвлетов нашла свое применение и в новом стандарте на сжатие видео: MPEG-4.

Однако не следует думать, что только структура нульдерев, или использование межполосных зависимостей, может привести к созданию эффективного кодера изображений. Некоторые из очень эффективных вейвлет-кодеров построены на совершенно других принципах. В данном разделе мы рассмотрим два метода, учитывающих внутриволосные зависимости между коэффициентами. Первый метод основан на концепции решетчатого квантования. Второй использует одновременно и внутри- и межполосные взаимосвязи между коэффициентами и рекурсивное оценивание дисперсии вейвлет-коэффициентов. Оба метода являются весьма эффективными.

10.5.1. Решетчатое квантование

Решетчатое квантование (РК) является быстрым и эффективным методом квантования случайных переменных. РК использует корреляционные связи между переменными. При этом ячейки квантователя могут быть непрямоугольной формы, что приводит к недостижимой скалярными квантователями эффективности. Главные идеи РК взяты из работы Г. Унгербоэка по решетчатой модуляции. В данном пункте мы на примерах покажем принципы работы РК, а также усовершенствование его идей для кодирования изображений с низкими скоростями.

Основная идея РК заключается в следующем. Предположим, мы хотим выполнить квантование стационарного равномерного источника без памяти на скорости R бит/отсчет. Оптимальный скалярный квантователь будет иметь 2^R уровней квантования (символов). В случае РК, вначале источник квантуется более точно, 2^{R+k} символами. Естественно, это приводит к превышению данного ресурса бит, поэтому мы не имеем возможности произвольного выбора символов в каждый момент времени.

Пусть $k = 1$. Скалярная кодовая книга из 2^{R+1} символов делится на четыре субкниги из 2^{R-1} символов каждая. Пусть $R = 2$ (рис.10.5). Субкниги строятся так, чтобы каждая из них представляла уровни реконструкции более грубого квантователя с $(R - 1)$ бит/отсчет. Четыре субкниги обозначены D_0, D_1, D_2 и D_3 . Также обозначим $S_0 = D_0 \cup D_2$ и $S_1 = D_1 \cup D_3$, где S_0 и S_1 называются суперкнигами.

Как было сказано, ограничение на скорость не позволяет выбирать произвольный символ из 2^{R+1} символов. Однако существует возможность выбирать символы одной из суперкниг. Если известно, какая из суперкниг используется, то для каждого отсчета требуется один бит для обозначения субкниги и $R - 1$ бит для обозначения кодового слова в ней. Выбор субкниги определяется состоянием автомата с конечным числом состояний, описываемого соответствующей решеткой.

Пример решетки с восемью состояниями приведен на рис.10.6. Субкниги $\{D_0, D_1, D_2, D_3\}$ используются для пометки ребер решетки, так что бит, обозначающий субкнигу, определяет следующее состояние решетки. Кодирование осуществляется путем посылки одного бита на отсчет для обозначения пути по решетке и $R - 1$ бит для обозначения кодового слова. Может показаться, что мы вернулись к неоптимальному квантователю со скоростью R . К чему же все старания? Ответ заключается в том, что мы получили большее количество кодовых слов, так как существует некоторая свобода выбора символов из S_0 или S_1 . Конечно, эта свобода неполная: решение по каждому символу принимается с учетом прошлого и будущего решений, то есть допустимых путей по решетке. Однако и эта гибкость приводит к эффективному кодированию. Доступность в каждый момент времени S_0 и S_1 означает, что уровни квантователя являются «скользящими» и настраиваются на данные, с учетом возможных путей по решетке.

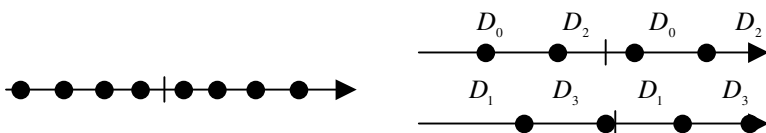


Рис. 10.5. Книги и суперкниги решетчатого квантования

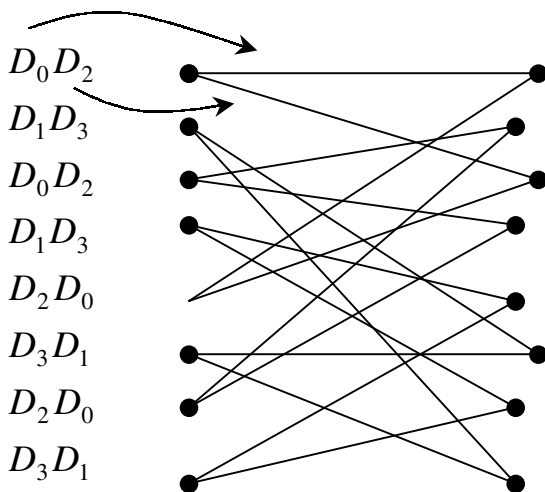


Рис. 10.6. Решетка РК с 8 состояниями

Перед тем как продолжить рассмотрение применения РК к кодированию вейвлет-коэффициентов отметим, что по своей эффективности и вычислительной сложности РК гораздо ближе к векторному, чем к скалярному квантованию. Почему бы не использовать стандартное векторное квантование? Ответ заключается в рекурсивной структуре РК и существовании простого алгоритма динамического программирования, известного как алгоритм Витерби для нахождения кодовых слов РК. Главное отличие заключается в том, что в случае векторного квантования кодовое слово размером $N-1$ не связано с кодовым словом размером N . Алгоритм РК автоматически решает проблему увеличения размерности, увеличивая длину решетки.

Обычное РК мало пригодно для кодирования изображений, особенно на низких скоростях. В самом деле, один бит на отсчет используется только для кодирования самой решетки, тогда как интересующий диапазон скоростей находится ниже одного бита. Для кодирования изображения разработан ал-

горитм РК с ограниченной энтропией (ЕСТСQ). ЕСТСQ имеет две особенности: малое число бит для представления решетки (вводится «энтропия состояния») и используется факт, что вероятность появления нулей на выходе кодера очень велика.

10.5.2. Субполосные кодеры с РК

В работах Международной конференции по обработке изображений в 1996 году были представлены три похожих, но независимо разработанных алгоритма. Основными их составляющими являются: субполосная декомпозиция, классификация и оптимальное распределение бит по различным подмножествам данных и РК с ограниченной энтропией (ЕСТСQ). Известен кодер, объединяющий эти алгоритмы. Кратко обсудим их главные аспекты.

Рассмотрим субполосную декомпозицию изображения и предположим, что субполосы можно представить как нестационарный случайный процесс X , чьи отсчеты взяты из распределений с дисперсией σ_i^2 . Конечно, в этом случае можно вычислить «среднюю дисперсию» всего случайного процесса и выполнить обычное оптимальное квантование. Но лучших результатов можно достичь путем передачи дополнительной информации о дисперсии каждого отсчета и квантования его в соответствии с его плотностью распределения вероятностей.

Эта идея была впервые предложена В.Ченом и К.Смитом для адаптивного квантования коэффициентов ДКП. Они предложили разделять коэффициенты на четыре группы в соответствии с их «уровнем активности», то есть дисперсией, и кодировать коэффициенты оптимальным квантователем для каждой группы. Вопрос о том, каким образом формировать группы, не рассматривался, и количество коэффициентов в группах было одинаковым.

Тем не менее, можно показать, что равное число коэффициентов в группах не всегда является правильным выбором. Предположим, что имеется J групп, и отсчеты, соответствующие одному классу $i \in \{1, \dots, J\}$, считаются порожденными одним источником X_i . Пусть источнику X_i соответствует N_i отсчетов, а общее число отсчетов во всех группах - N . Определим вероятность того, что отсчет принадлежит источнику X_i , как $p_i = N_i / N$. Кодирование источника X_i на скорости R_i приводит к среднеквадратической ошибке вида

$$D_i(R_i) = \varepsilon_i^2 \sigma_i^2 2^{-2R_i}, \quad (10.3)$$

где ε_i - константа, зависящая от вида плотности распределения вероятности. Проблема распределения бит далее решается методом множителей Лагранжа. В результате получаем выражение

$$R_i = \frac{R}{J} + \frac{1}{2} \log_2 \frac{\varepsilon_i^2 \sigma_i^2}{\prod_{j=1}^J (\varepsilon_j^2 \sigma_j^2)^{p_j}}, \quad (10.4)$$

где R - суммарная скорость, а R_i - скорость, соответствующая каждой группе. Выигрыш от классификации определяется как отношение ошибки квантования исходного сигнала X и классифицированного:

$$G_c = \frac{\varepsilon^2 \sigma^2}{\prod_{j=1}^J (\varepsilon_j^2 \sigma_j^2)^{p_j}}. \quad (10.5)$$

Этот выигрыш максимизируется по всем $\{p_i\}$. Неудивительно, что процесс оптимизации зачастую приводит к неравномерным $\{p_i\}$. Это означает неравномерное распределение коэффициентов по группам. Кроме того, неравномерное распределение требует передачи меньшего количества дополнительной информации: любое распределение $\{p_i\}$ имеет меньшую энтропию, чем равномерное.

Мы определили выигрыш от классификации для одной субполосы. Обобщение этого результата привело к объединению этого выигрыша с выигрышем от субполосного кодирования. Также там уточнено выражение для количества требующейся для классификации дополнительной информации. Алгоритм кодирования оптимизирует получившееся выражение, и, затем, применяется решетчатое квантование с ограниченной энтропией (ЕСТСҚ).

Практическая реализация данного алгоритма требует учета намного большего количества деталей. Например, классификация отображает определенные уровни энергии сигнала, которые связаны с расположением контуров на изображении и соответствуют различным субполосам. Для уменьшения дополнительной информации могут быть использованы различные методы (например, алгоритм, обсуждаемый в следующем пункте). Другие исследования состоят в поиске альтернативных критериев для классификации, а также применении РК в сочетании с арифметическим кодированием. Эффективность кодирования ЕСТСҚ - одна из самых высоких из ныне известных,

хотя и вычислительная сложность также значительно превышает сложность рассматриваемых в главе кодеров.

10.5.3. Моделирование и оценивание смеси распределений

Один из подходов к кодированию вейвлет-коэффициентов изображения заключается в том, что они представляются как случайные переменные, порождаемые смесью распределений. Для каждого отсчета необходимо определить, источнику с каким распределением он соответствует, и квантовать его в соответствии с этим распределением. Так как декодер должен знать, какому распределению принадлежит данный коэффициент, многие кодеры посылают ему эту информацию как дополнительную. Уменьшение количества этой информации становится особенно важным при низких скоростях кодирования, так что управление им является чрезвычайно важным для эффективности кодера.

Все алгоритмы, обсуждаемые здесь, так или иначе используют этот подход. Они отличаются в основном ограничениями, накладываемыми на дополнительную информацию для ее эффективного кодирования. Например, нульдеревья отображают не что иное, как дополнительную информацию. В этом случае предполагается, что данные порождаются смесью источников с очень низкой (нулевой) и высокой энергией и что нули имеют древовидную структуру.

Субполосные кодеры на основе РК, обсуждаемые в предыдущем пункте, также используют этот подход. Информация о разбиении множества коэффициентов на группы с разной энергией передается как дополнительная информация. Очевидно, что существуют методы ее сокращения, также основанные на геометрических ограничениях в расположении коэффициентов той или иной структуры.

Полностью отличный подход к управлению количеством дополнительной информации известен, как «квантование через оценивание». Данный класс алгоритмов называется «кодированием с оцениванием смеси по прошлому» (КОСП).

КОСП использует в качестве модели вейвлет-коэффициентов нестационарный обобщенный гауссовский процесс, чья нестационарность выражается в медленно изменяющейся дисперсии коэффициентов в каждой полосе. Так как их энергия меняется медленно, она может быть предсказана на основе соседей. Следовательно, в случае КОСП, в отличие от предыдущих методов, не передается дополнительная информация, но декодер пытается восстано-

вить ее из принятых данных, отсюда и название – «по прошлому». КОСП основано на предположении, что ближайшие соседи коэффициента, в том числе и его родители по дереву, имеют ту же энергию, что и он сам. Оценка энергии производится по принципу максимального правдоподобия на базе соседних коэффициентов.

Так же, как и в других рекурсивных алгоритмах, где есть квантование, в КОСП должны быть учтены проблемы стабильности и смещения. Так как декодер имеет доступ только к квантованным коэффициентам, кодер тоже обязан использовать только квантованные коэффициенты. Иначе, кодер и декодер будут располагать различным контекстом, и возникнет проблема смещения. Таким образом, появляется добавочная трудность – оценивание дисперсии на основе уже квантованных коэффициентов. КОСП объединяет квантование и максимально правдоподобную оценку дисперсии.

Квантование выполняется равномерным квантователем с «мертвой» зоной (рис. 10.2). Такой квантователь хорошо подходит для квантования с ограниченной энтропией обобщенного гауссовского процесса. Ширину «мертвой» зоны и шага квантователя определяет оптимизационная процедура, основанная на методе множителей Лагранжа. Эта процедура выполняется заранее для различных скоростей кодирования и параметров источника. Результаты хранятся в таблице. Поэтому кодер работает очень быстро.

Еще одна проблема, связанная с алгоритмом оценивания по прошлому, заключается в следующем. Возможен случай, когда все соседи коэффициента будут равны нулю. В этом случае он тоже будет квантован в нуль. При повторении этой процедуры окажется, что все коэффициенты будут квантованы в нуль. Для предотвращения этой опасности в КОСП предусмотрен механизм защиты от распространения ошибки. В случае если все соседи равны нулю, декодеру посылается специальный символ. Это происходит на предварительном этапе кодирования, когда алгоритм проверяет возможность возникновения ошибочных серий нулей для всех коэффициентов. Такие коэффициенты группируются вместе, вычисляется их дисперсия и параметр огибающей обобщенного гауссовского распределения, и передаются декодеру. Далее, каждый раз при возникновении угрозы появления ошибочных нулей кодер и декодер работают с этими параметрами, а не с ближайшими соседями коэффициентов.

КОСП является очень быстрым и эффективным алгоритмом. Поэтому, очевидно, необходимо пересмотреть роль дополнительной информации и механизм ее передачи для вейвлет-кодеров.

10.6. Современные направления исследований

Исследования в области сжатия изображений ведутся по разным направлениям. Так, появилась новая интерпретация (описанная в главе 6) вейвлет-преобразования – лифтинговая схема, не основанная на преобразовании Фурье. С использованием этой схемы появилась возможность конструирования новых неразделимых базисов вейвлетов, которые потенциально могут привести к повышению эффективности кодеров. Интересным направлением исследований является изучение нелинейных аналогов вейвлет-преобразования, которые философия лифтинга делает возможным.

Активные исследования проводятся в области кодеров, основанных на классификации и оценивании по прошлому.

Одним из наиболее интересных направлений является разработка кодеров изображения, робастных к ошибкам, возникающим в каналах связи. При этом используется идея совместной оптимизации кодеров источника и канала, а также оптимального сочетания раздельно оптимизированных кодеров.

Особый интерес представляет адаптация вейвлет-кодирования изображения для кодирования видео. Здесь можно сочетать внутрикадровое кодирование с межкадровым предсказанием, как это заложено в стандарте MPEG-4. Можно также рассматривать видеопоследовательность как трехмерный массив и применять трехмерный вейвлет-анализ. Однако этот метод наталкивается на трудности, связанные с фундаментальными особенностями вейвлет-преобразования, как и любого субполосного кодирования. Вейвлет-преобразование не является пространственно-инвариантным в силу присутствия децимации и интерполяции. Эта изменчивость в пространстве мешает компактному представлению видеосигналов. Причина поясняется ниже.

Видеосигналы состоят из кадров. От кадра к кадру информация меняется незначительно. Поэтому существует возможность достичь хороших результатов сжатия, передав одинаковую информацию лишь однажды. Однако вейвлет-преобразование не является инвариантным к сдвигу, следовательно, подобное кодирование невозможно. Аналогичные доводы против трехмерного вейвлет-преобразования приводят и в частотной области.

Итак, в настоящей главе рассмотрено применение вейвлет-преобразования для сжатия изображений. Во всем мире в данном направлении ведутся интенсивные работы. Разработано большое число алгоритмов и кодеров, некоторые из которых стандартизированы.

Современные вейвлет-кодеры основаны на предположении, что изображение порождается источником с флуктуирующей дисперсией. Каждый ко-

дер реализует определенный механизм для отображения локальной дисперсии вейвлет-коэффициентов и квантует их оптимальным или субоптимальным образом в соответствии с дисперсией. Кодеры отличаются друг от друга стратегиями квантования коэффициентов и тем, каким образом происходит оценка и передача значения дисперсии декодеру.

Кодеры, основанные на алгоритме нульдеревя, предполагают у дисперсии наличие двух состояний: нуль или нет. Декодеру передается дополнительная информация о местоположении значимых коэффициентов. Этот процесс приводит к нелинейной аппроксимации изображения. Множества нулевых коэффициентов выражаются в терминах деревьев вейвлетов (Льюис и Ноулес, Шапиро и др.) или комбинаций этих деревьев (Саид и Перельман). Нули передаются декодеру как дополнительная информация, так же как и квантованные данные. Кодеры, основанные на нульдеревьях, учитывают межполосные зависимости вейвлет-коэффициентов.

В частотно-адаптивных кодерах применяются ортогональные адаптивные преобразования – метод вейвлет-пакетов. Локальные флуктуации корреляционных связей используют пространственно-адаптивные кодеры.

Другие вейвлет-кодеры учитывают внутриволновые зависимости между вейвлет-коэффициентами (иногда одновременно и межполосные). Кодеры, основанные на решетчатом квантовании, делят коэффициенты на группы в соответствии с их энергией. Для каждого коэффициента они оценивают и (или) передают информацию о группе и значение квантованного в соответствии с номинальной дисперсией группы коэффициента. Другой новый класс кодеров передает незначительное количество информации о дисперсии. Это показывает, что, возможно, информация о дисперсии имеет большую избыточность, чем считалось раньше.

Многие исследования посвящены разработке алгоритмов кодирования изображения на основе векторного квантования вейвлет-коэффициентов. Эти алгоритмы не рассмотрены в нашей книге. Представление о них вы можете получить, прочтя обзорную статью Р.Грэя (см. Интернет-ссылки).

В заключение можно отметить, что вейвлеты и сопутствующие им идеи внесли неоценимый вклад в теорию и практику кодирования изображений и, по-видимому, будут оставаться основным направлением исследований в этой области в ближайшем будущем.

Глава 11

ВИДЕОКОДЕКИ СЕМЕЙСТВА ADV6XX ПРОИЗВОДСТВА ФИРМЫ ANALOG DEVICES

В настоящей главе описаны микросхемы ADV6xx, разработанные фирмой

Таблица 11.1

Характеристики видеокодеков

	ADV601	ADV601LC	ADV611	ADV612
Разрядность, бит	10	8	8	8
Возможность подкл. ЦПОС через RS- порт	Есть	Нет	Нет	Нет
Число выводов корпуса	160	120	120	120
Диапазон рабочих температур	0 – +70 ⁰ C	0 - +70 ⁰ C	0 - +70 ⁰ C	-25 - +85 ⁰ C
Регулирование частоты кадров	Программно	Программно	Аппаратно	Аппаратно
Режим стоп-кадра	Нет	Нет	Есть	Есть
Возможность выделения части кадра	Нет	Нет	Есть	Есть
Оценочная плата (evaluation board)	VideoLab	VideoPipe	CCTVPIPE	CCTVPIPE
Возможная область применения	Профессиональная	Бытовая техника	Кабельное телевидение	Кабельное телевидение

Analog Devices и предназначенные для сжатия и восстановления видео в реальном масштабе времени. Семейство к настоящему времени включает в себя четыре микросхемы: ADV601, ADV601LC, ADV611, ADV612. Различия между микросхемами приведены в табл.11.1.

Все микросхемы являются однокристалльными дешевыми устройствами, позволяющими достигать высоких степеней сжатия видео при сохранении достаточно высокого качества изображения. В данных микросхемах впервые применено вейвлет-преобразование изображения. Алгоритмы, используемые

в этих микросхемах, примерно одинаковы. Поэтому в дальнейшем описывается микросхема ADV601.

11.1. Принципы работы ADV601

Вейвлет-преобразование в данной микросхеме осуществляется фильтрацией изображения в вертикальном и горизонтальном направлениях при помощи биортогональной пары фильтров 9/7, описанных в главе 4.

ADV601 работает в двух основных режимах: без потерь качества изображения (коэффициент сжатия 4:1) и с допустимыми потерями качества (коэффициент сжатия достигает 350:1). Данный кодек поддерживает все форматы изображения с черезстрочной разверткой.

На рис.11.1 показана структура ADV601. Алгоритм сжатия ADV601 основан на применении биортогонального вейвлет-преобразования, после чего выполняется квантование коэффициентов и энтропийное кодирование.

Все три компоненты цветного видеосигнала (Y , C_r , C_b) независимо друг от друга подвергаются двумерной фильтрации и децимации в два раза на каждом шаге разложения. Всего получается 42 новых изображения (по 14 для каждой компоненты). Каждое из этих изображений несет в себе определенную информацию об исходном изображении. В результате такого преобразования никакого сжатия еще не достигнуто, число пикселей в этих 14 блоках равно числу пикселей в исходном изображении. Но теперь возможны следующие варианты: 1) применить сжатие почти без потерь; 2) применить сжатие с потерями при ограничениях на качество или на скорость передачи; 3) создать высококачественные отмасштабированные изображения без каких-либо дополнительных вычислений; 4) создать помехоустойчивый сжатый поток бит, так как каждый блок несет в себе информацию о всем изображении.

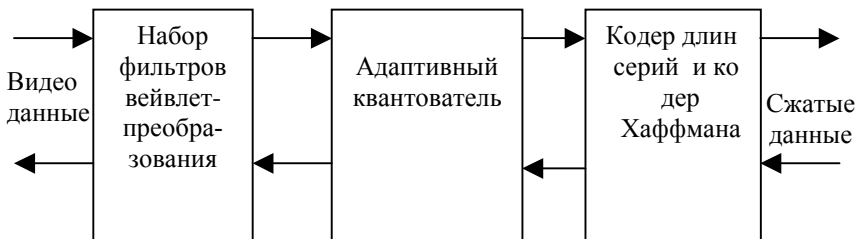


Рис.11.1. Структура ADV601

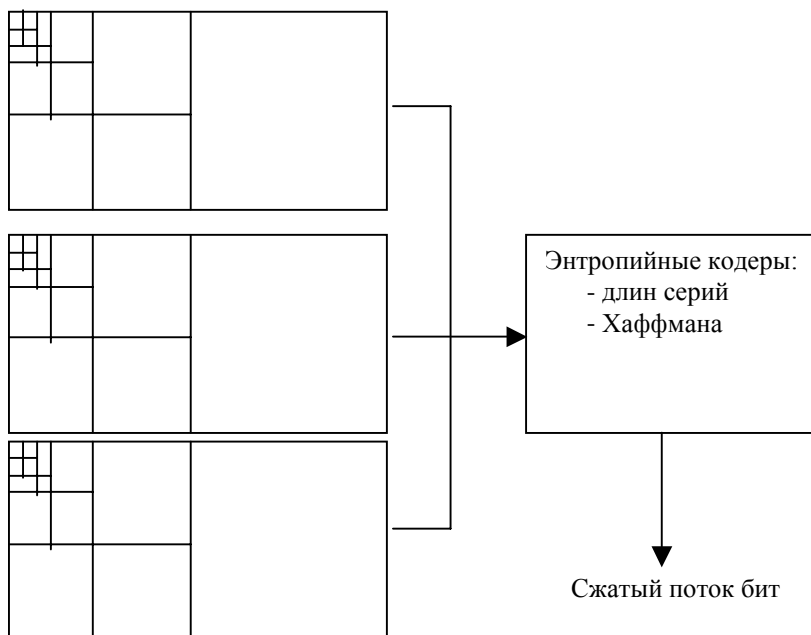


Рис.11.2. Сжатие изображения в режиме почти без потерь

На рис.11.2 показана схема использования ADV601 в режиме сжатия почти без потерь. 42 блока кодируются двумя типами энтропийных кодеров. Коэффициент сжатия зависит от степени сложности, высокочастотности исходного изображения. Для типичного видео сигнала он находится в диапазоне от 2:1 до 5:1, таким образом, скорость цифрового потока будет «плавать» в значительных пределах.

На рис.11.3 показана схема использования ADV601 в режиме сжатия с допустимыми потерями. После преобразования изображения из 42 блоков извлекается ряд статистик: сумма квадратов (энергия), минимальное и максимальное значения пиксела для каждого блока. Эта информация учитывается в квантователе, построенном в соответствии с моделью зрения человека. Квантователь на основе этих данных и требуемой пользователем скорости цифрового потока вычисляет 42 величины для каждого блока, которые могут рассматриваться как бюджет точности на блок. Вычисление этих величин должен выполнять хост-компьютер или внешний DSP, в ADV601 оно не производится.

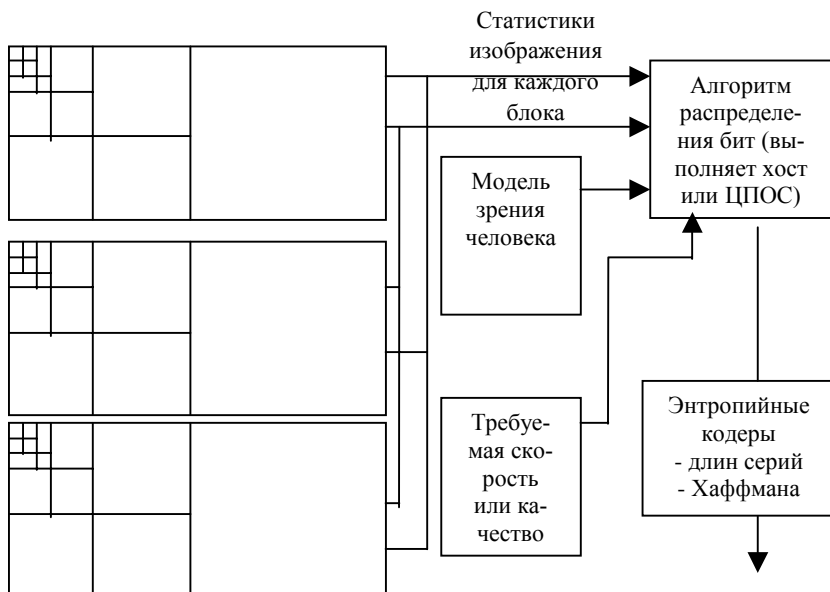


Рис.11.3. Сжатие изображения в режиме с потерями

За счет правильного распределения бюджета бит для различных блоков, в соответствии со моделью зрения человека, и достигается высокая степень сжатия - до 350:1 (в видеокодеках ADV611 и ADV612 сжатие видеопотока может достигать до 7500:1). Конечно, качество при столь сильном сжатии не очень хорошее, однако достаточное для таких приложений, как наблюдение или идентификация.

При больших коэффициентах сжатия проявляются искажения в виде высокочастотного шума и некоторого размывания деталей. Человеческий глаз, сам, по сути, являясь полосовым фильтром, не критичен к подобным искажениям. Блочные искажения, появляющиеся в JPEG и MPEG, гораздо более заметны.

11.2. Использование микросхемы ADV601

На рис.11.4 представлена блок-диаграмма ADV601. Видеоинтерфейс ADV601 разработан для работы со всеми популярными аналоговыми кодеками фирм Analog Devices, Philips, Brooktree, Raytheon. Интерфейс с внешней памятью поддерживает использование динамического ОЗУ 256К x 16. Хост-

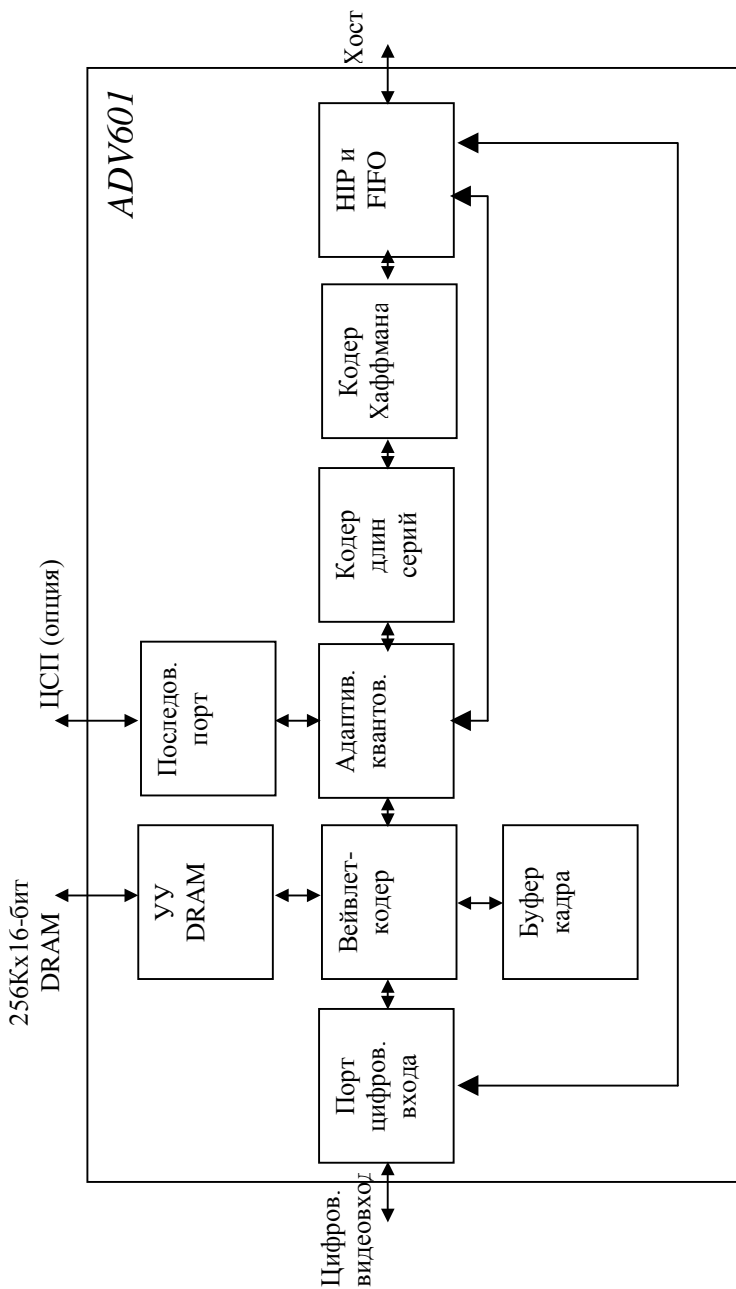


Рис.1.1.4. Функциональная блок-диаграмма ADV601

интерфейс может быть сконфигурирован, как 8-, 16- или 32-битный. Он также имеет 512x32-битный стек FIFO для равномерной передачи сжатого видео. Хост-интерфейс может быть подсоединен к шинам компьютера (как PCI, так и ISA), а также к высокоскоростным цифровым абонентским линиям (HDSL, ADSL).

С микросхемой поставляется драйвер для Windows, позволяющий рассчитывать бюджет бит для каждого блока. Также имеется PCI-плата для Windows95, позволяющая оценивать качество сжатого видеосигнала.

ADV601 может также использоваться как отдельное устройство, при наличии дополнительного ЦПОС для расчета бюджета бит. Микросхема имеет последовательный интерфейс для подключения процессоров типа ADSP21xx.

ADV601JS, размещенная в корпусе со 160 выводами, работает в диапазоне температур от 0 до +70°C.

С ADV601 поставляется программное обеспечение (Feature Software Library) и подробная документация к нему. Оно позволяет перепрограммировать микросхему, например для достижения еще больших коэффициентов сжатия. Рассмотрим этот вопрос подробнее. В ADV601 применен алгоритм сжатия без учета межкадровой корреляции. Программное обеспечение, поставляемое с платой, позволяет применить межкадровое кодирование. Для этого возможны два пути:

- 1) система человеческого зрения более чувствительна к движению в низкочастотных областях, чем в высокочастотных. Для достижения высокого сжатия можно не передавать какое-то время изменение изображения в высокочастотных областях. В этом случае декодер будет «подставлять» вместо текущего предыдущий кадр;

- 2) существуют приложения, где в видеосигнале движение практически всегда отсутствует. (Например, сюда можно отнести системы видеонаблюдения на объектах, где нет людей). Тогда изображение необходимо обновлять достаточно редко либо при обнаружении движения. Субполосы вейвлет-декомпозиции можно использовать для обнаружения наличия движения в кадре и, таким образом, достигать высоких коэффициентов сжатия.

Применение ADV601 не ограничивается только сжатием видео. Например, ее можно использовать для масштабирования изображений. Так как микросхема включает в себя дециматоры, интерполяторы и блоки фильтров, масштабированные изображения получатся намного более высокого качества, чем полученные путем простого отбрасывания пикселей. Масштабирование видео имеет важное значение для его редактирования, а также передачи по каналам с различной скоростью или гетерогенным сетям связи. Кроме того, фирма Analog Devices разработала программный симулятор микросхемы ADV601. В зависимости от производительности процессора компьютера этот симулятор может производить декодирование сжатого изображения с той или иной степенью уменьшения в реальном масштабе времени.

фильтрация изображений. Имеющийся квантователь способен удалить из видеокадра все или некоторые блоки, соответствующие сигналам яркости и цветоразностным. Это может быть применено для повышения контрастности изображения, ослабления или усиления насыщенности цвета и т.д.

Важным различием видеокодеков ADV611 и ADV612 является наличие режима выделения в кадре прямоугольного фрагмента произвольного размера. При этом изображение внутри фрагмента декодируется с максимально высоким качеством, а остальная часть кадра – с пониженным разрешением.

В настоящее время разработаны платы кодирования и обработки видео для персональных компьютеров на базе микросхемы ADV601. Цена подобных плат зависит от производителя и находится в пределах 1000 долларов.

С микросхемой ADV601, работающей в автономном режиме или совместно с ПК, могут быть использованы следующие микросхемы фирмы Analog Devices:

ADV7175 – цифровой конвертер сигнала YUV в композитный сигнал;

AD722 – аналоговый преобразователь сигнала RGB в аналоговый композитный сигнал;

AD1843 – аудиокодек, синхронизированный с видеопоследовательностью;

ADSP-21XX – семейство цифровых процессоров обработки сигнала;

AD8XXX – семейство операционных видеоусилителей.

В заключение авторы хотели бы выразить глубокую признательность фирме Autex за любезно предоставленную информацию о видеокодеках семейства ADV6xx. Более подробную информацию о данной микросхеме и о многих других микросхемах вы можете получить у представителей этой фирмы в Санкт-Петербурге или на сайте фирмы Analog Devices : <http://www.analog.com>

ЗАКЛЮЧЕНИЕ

Итак, в книге была предпринята попытка познакомить читателя с некоторыми важными, на взгляд авторов, положениями теории и практики вейвлет-преобразования. Интенсивность исследований, ведущихся в данной области такова, что для подробного освещения всего обширного круга вопросов, касающихся данной темы, потребовалось бы издание, сопоставимое по масштабам с БСЭ. Однако основы теории в книге представлены.

В качестве практического применения вейвлет-преобразования рассмотрены современные подходы к сжатию изображений. Как отмечалось выше, вейвлет-преобразование легло в основу международного стандарта MPEG-4, стандарта на сжатие отпечатков пальцев ФБР, видеокодексов фирмы Analog Devices. В настоящее время ведется разработка стандарта JPEG-2000, где вейвлет-фильтры, вероятно, также найдут себе применение.

Другие применения вейвлет-преобразования в книге не рассматривались. Между тем, во многих областях можно ожидать существенно лучших результатов за счет использования вейвлетов. Перечислим некоторые из них.

Задачи, связанные с предсказанием. Это - предсказание курса ценных бумаг на рынке, предсказание землетрясений, прогноз погоды.

Вейвлеты успешно применяются в квантовой физике, при изучении строения атома, в лазерной технике.

Задачи анализа нестационарных сигналов. Такого рода задачи возникают в медицине (томография, электрокардиография), гидроакустике и других областях.

Очистка от шума зашумленных сигналов. Так, ученые Стэнфорда с успехом применили вейвлеты для улучшения звучания старых грампластинок.

Задачи, связанные с обнаружением сигнала на фоне помехи, его распознаванием, классификации. Сотрудниками Исследовательской лаборатории ВМС США вейвлеты применялись для обнаружения подводных лодок, для оценки разрушений, произведенных бомбардировками, и для многих других важных военно-прикладных задач.

Наверное, наиболее существенным, особенно с коммерческой стороны, будет применение вейвлетов в цифровой связи. В частности, известны многообещающие результаты по применению вейвлетов в трансмультиплексорах, в системах с широкополосными сигналами. Перспективным является использование вейвлет-пакетов для скрытой связи, а также в системах с многостанционным доступом.

Список возможных и уже состоявшихся успешных применений вейвлетов

можно было бы продолжить и дальше. Теория вейвлетов и вейвлет-преобразований в основном уже оформилась. Сейчас настало время практических приложений.

Однако при этом ни в коем случае нельзя полагать, что вейвлеты являются универсальным средством, с помощью которого можно решить любую задачу. Приведем слова одного из исследователей, работающих в данной области, Б.Торезани: "My dream is to solve problems, with or without wavelets".

**Некоторые вейвлет-преобразования,
находящие применение для сжатия изображений**

Таблица 1

Краткая характеристика преобразований

Условное обозначение	Название преобразования	Описание	Длина фильтров	Число исчезающих моментов
Haar	Хаара	ортогональное, симметричное, интерполирующее	2 / 2	(1,1)
TS	“два-шесть”	биортогональное, симметричное	2 / 6	(3,1)
CDF22	Козна-Добеши-Фово	биортогональное, симметричное, интерполирующее	5 / 3	(2,2)
CDF24	Козна-Добеши-Фово	биортогональное, симметричное, интерполирующее	9 / 3	(2,4)
CDF97	Козна-Добеши-Фово	биортогональное, симметричное	9 / 7	(4,4)
V610	Вилласенора	биортогональное, симметричное	6 / 10	(4,4)
MIT97	MIT9/7	биортогональное, симметричное, интерполирующее	9 / 7	(4,2)
BCW3	Койфмана, биортогональное	биортогональное, симметричное, интерполирующее	13 / 7	(4,4)

Коэффициенты вейвлет-фильтров
(для симметричных фильтров показана только половина коэффициентов)

Haar	$h_0 = 1/\sqrt{2}, 1/\sqrt{2};$ $h_1 = -1/\sqrt{2}, 1/\sqrt{2}.$
TS	$h_0 = 1/\sqrt{2};$ $h_1 = -1/8\sqrt{2}, -1/8\sqrt{2}, 8/8\sqrt{2}.$
CDF22	$h_0 = -1/4\sqrt{2}, 2/4\sqrt{2}, 6/4\sqrt{2};$ $h_1 = 1/2\sqrt{2}, -2/\sqrt{2}.$
CDF24	$h_0 = 3\sqrt{2}/128, -6\sqrt{2}/128, -16\sqrt{2}/128, 38\sqrt{2}/128, 90\sqrt{2}/128;$ $h_1 = \sqrt{2}/4, -2\sqrt{2}/4.$
CDF97	$h_0 = 0.03783, -0.02385, -0.1106, 0.3774, 0.8527;$ $h_1 = 0.06454, -0.04069, -0.4181, 0.7885.$
V610	$h_0 = -0.1291, 0.04770, 0.7885;$ $h_1 = -0.01891, 0.006989, 0.06724, 0.1334, -0.6151.$
MIT97	$h_0 = \sqrt{2}/64, -8\sqrt{2}/64, 16\sqrt{2}/64, 46\sqrt{2}/64;$ $h_1 = -\sqrt{2}/32, 9\sqrt{2}/32, -16\sqrt{2}/32.$
BCW3	$h_0 = -1/256\sqrt{2}, 18/256\sqrt{2}, -1/16\sqrt{2}, -63/256\sqrt{2}, 9/16\sqrt{2}, 348/256\sqrt{2};$ $h_1 = -1/16\sqrt{2}, 9/16\sqrt{2}, -16/16\sqrt{2}.$

**Текст функций, выполняющих прямое и обратное
вейвлет-преобразование изображения**

// Функции для вычисления прямого и обратного вейвлет-преобразования
// изображения с использованием биортогональных 9/7 фильтров. Текст
// программы – модифицированная версия текста G.Davis (см. список
// Интернет-источников). Выполнение вейвлет-преобразования изображения
// размером 512x512 на компьютере P-166 занимает примерно 0.4 с.

```
#include "wavelet.h"
```

```
#define aI0      0.852699
#define aI1      0.377403
#define aI2      -0.110624
#define aI3      -0.023849
#define aI4      0.037829
```

```
#define aH0      0.788485
#define aH1      -0.418092
#define aH2      -0.040690
#define aH3      0.064539
```

```
#define sI0      -0.852699
#define sI1      -0.418092
#define sI2      0.110624
#define sI3      0.064539
#define sI4      -0.037829
```

```
#define sH0      -0.788485
#define sH1      0.377403
#define sH2      0.040690
#define sH3      -0.023849
```

```
#define AnaLowPass(v)  aI0*v[0]+aI1*(v[1]+v[-1])+aI2*(v[2]+v[-\
2])+aI3*(v[3]+v[-3])+aI4*(v[4]+v[-4])
```

```

#define AnaHighPass(v)  ah0*v[0]+ah1*(v[1]+v[-1])+ah2*(v[2]+v[-\
2])+ah3*(v[3]+v[-3])
#define SynLowPass(v)   sl0*v[0]+sl1*(v[1]+v[-1])+sl2*(v[2]+v[-\
2])+sl3*(v[3]+v[-3])+sl4*(v[4]+v[-4])
#define SynHighPass(v)  sh0*v[0]+sh1*(v[1]+v[-1])+sh2*(v[2]+v[-\
2])+sh3*(v[3]+v[-3])

```

```
WaveletTransform::WaveletTransform ()
```

```

{
    // Это необходимо для продолжения изображения при симметричном
    отражении
    npad = 4;
}

```

```
WaveletTransform::~~WaveletTransform ()
```

```

{
}

```

```
void WaveletTransform::Reflection(float * h, float * t)
```

```

{
    for (int i=1; i <= npad; i++)
    {
        h[-i] = h[i];
        t[i] = t[-i];
    }
}

```

```
// -----
```

```
// FWT (int ** input, float ** output, int hSize, int vSize,
//      int trSteps, int symExtension)
//
```

// Функция прямого вейвлет-преобразования. Выполняется разделимое преобразование, как описано в главе 9.

// Входные параметры:

// float *input: входной буфер данных,

// float *output: выходной буфер данных,

// int hSize: горизонтальный размер изображения,

// int vSize: вертикальный размер изображения,

// int trSteps: число шагов декомпозиции,

// int symExtension: вид продолжения на границе, по умолчанию -1.

```
// -----
```

```
BOOL WaveletTransform::FWT (float *coeff, int trSteps, int hSize, int vSize)
```

```

{

```

```

int i, j, lowSize, highSize;
int hTransSize = hSize;
int vTransSize = vSize;

float ** output = new float * [vTransSize];
output[0] = coeff;
for ( i=1; i<vSize; i++ )
    output[i] = output[i-1] + hTransSize;
int *intIn, *intOut;
float *floatIn, *workingVector, *bufferPtr, *bufferEnd;
floatIn = new float [2*npad + Max(hTransSize, vTransSize)];
workingVector = floatIn+npad;
    while (trSteps-- )
{
    // Выполняем свертку каждой строки с НЧ фильтром
    for ( i=0; i<vTransSize; i++ )
    {
        lowSize = (hTransSize+1) >> 1;
        highSize = hTransSize - lowSize;
        // Копируем строку i в массив данных
        memcpy( workingVector, output[i], hTransSize*sizeof(float) );
        Reflection( workingVector, workingVector+hTransSize-1);
        for ( bufferPtr = workingVector, j = 0; j < highSize; j++)
        {
            output[i][j] = AnaLowPass(bufferPtr);
            bufferPtr++;
            output[i][j+lowSize] = AnaHighPass(bufferPtr); bufferPtr++;
        }
        if ( lowSize > highSize )
            output[i][highSize] = AnaLowPass(bufferPtr);
    }
    // Выполнение свертки НЧ фильтра со столбцами
    for ( j=0; j<hTransSize; j++ )
    {
        lowSize      = (vTransSize+1) >> 1;
        highSize     = vTransSize - lowSize;
        // Копируем столбец j в массив данных
        for ( i=0; i<vTransSize; i++ )
            workingVector[i] = output [i][j];
        Reflection( workingVector, workingVector+vTransSize-1);

        for ( bufferPtr = workingVector, i = 0; i < highSize; i++)
        {

```

```

        output[i][j] = AnaLowPass(bufferPtr);
bufferPtr++;
        output[i+lowSize][j] = AnaHighPass(bufferPtr); bufferPtr++;
    }
    if ( lowSize > highSize )
        output[highSize][j] = AnaLowPass(bufferPtr);
    }

    hTransSize = (hTransSize + 1) / 2;
    vTransSize = (vTransSize + 1) / 2;
}
delete output;
delete floatIn;
return TRUE;
}

```

```

// -----
// IWT (float *input, float *output, int hSize, int vSize,
//      int trSteps, int symExtension)
//
// Обратное вейвлет-преобразование
// Входные параметры:
// float *input: входной буфер данных,
// float *output: выходной буфер данных,
// int hSize: размер изображения по горизонтали,
// int vSize: размер по вертикали,
// int maxOrig: максимальное значение пиксела в исх. изображении,
// int minOrig: минимальное значение пиксела в исх. изображении,
// int trSteps: число шагов декомпозиции,
// int symExtension: продолжение на границе, по умолчанию -1.
// -----

```

```

BOOL WaveletTransform::IWT( float *coeff, int trSteps, int hSize,
                           int vSize, int maxOrig, int minOrig)

```

```

{
    int i, j;
    float ** input = new float * [vSize];
    input[0] = coeff;
    for ( i=1; i<vSize; i++ )
        input[i] = input[i-1] + hSize;
    int * hTransSize = new int [trSteps];
    int * vTransSize = new int [trSteps];
    int hCrtSize, vCrtSize, normalSize, lowSize, highSize;

```

```

hTransSize[0] = hSize;
vTransSize[0] = vSize;
for (i = 1; i < trSteps; i++)
    {
        hTransSize[i] = (hTransSize[i-1] + 1) / 2;
        vTransSize[i] = (vTransSize[i-1] + 1) / 2;
    }

int *intIn, *intOut;
float *floatIn, *workingVector, *bufferPtr, *bufferEnd;
floatIn = new float [2*npad + Max(hSize, vSize)];
workingVector = floatIn+npad;

while (trSteps--)
    {
        hCrtSize = hTransSize[trSteps];
        vCrtSize = vTransSize[trSteps];
        // Выполняем реконструкцию для столбцов
        for (j = 0; j < hCrtSize; j++)
            {
                lowSize          = (vCrtSize + 1) >> 1;
                highSize = vCrtSize - lowSize;
                normalSize      = highSize*2;
                bufferPtr       = workingVector;
                for ( i = 0; i < highSize; i ++ )
                    {
                        *bufferPtr = input[i][j];
                        bufferPtr++;
                        *bufferPtr = input[i+lowSize][j];
                        bufferPtr++;
                    }
                if ( lowSize > highSize )
                    workingVector[normalSize] = input[highSize][j];
                Reflection(workingVector, workingVector + vCrtSize - 1);
                bufferPtr= workingVector;
                for ( i = 0; i < normalSize; )
                    {
                        input[i++][j] = SynHighPass(bufferPtr);
                        bufferPtr++;
                        input[i++][j] = SynLowPass(bufferPtr);
                        bufferPtr++;
                    }
            }
    }

```

```

        if ( lowSize > highSize )
            input[normalSize][j] = SynHighPass(bufferPtr);
    }
    // Выполняем реконструкцию для строк
    for ( i = 0; i < vCrtSize; i ++ )
    {
        lowSize          = (hCrtSize + 1) >> 1;
        highSize = hCrtSize - lowSize;
        normalSize       = highSize*2;
        bufferPtr        = workingVector;
        for ( j = 0; j < highSize; j ++ )
        {
            *bufferPtr = input[i][j];
            bufferPtr++;
            *bufferPtr = input[i][j+lowSize];
            bufferPtr++;
        }
        if ( lowSize > highSize )
            workingVector[normalSize] = input[i][highSize];
        Reflection(workingVector, workingVector + hCrtSize - 1);
        bufferPtr        = workingVector;
        for ( j = 0; j < normalSize; )
        {
            input[i][j++] = SynHighPass(bufferPtr);
            bufferPtr++;
            input[i][j++] = SynLowPass(bufferPtr);
            bufferPtr++;
        }
        if ( lowSize > highSize )
            input[i][normalSize] = SynHighPass(bufferPtr);
    }
}

```

```

int imgSize = vSize*hSize;
bufferPtr=coeff, bufferEnd = coeff+imgSize;

```

```

while ( bufferPtr < bufferEnd )
{
    *bufferPtr = Round(*bufferPtr);
    if (*bufferPtr > maxOrig)
        *bufferPtr = maxOrig;
    else
        if (*bufferPtr < minOrig)

```

```

                *bufferPtr = minOrig;
                bufferPtr ++;
            }
            delete floatIn;

delete hTransSize;
delete vTransSize;
delete input;
return TRUE;
}

void WaveletTransform::SP_TransformStep(int *input, int *output, int size)
{
    int i, k, d1, d2;

    int lowSize = (size+1)/2;
    int highSize = size - lowSize;
    int *l = output;
    int *h = output + lowSize;
    int *in = input;

    for (i = k = 0; i < highSize; i++, k += 2)
    {
        l[i] = (in[k] + in[k+1]) >> 1;
        h[i] = in[k] - in[k+1];
    }
    d2 = l[0] - l[1];
    h[0] -= d2 >> 2;
    for (i = 1; i < highSize-1; i++)
    {
        d1 = d2;
        d2 = l[i] - l[i+1];
        h[i] -= (((d1 + d2 - h[i+1]) << 1) + d2 + 3) >> 3;
    }
    h[i] -= d2 >> 2;

    if ( lowSize > highSize )
        l[highSize] = in[2*highSize];
}

```



```

void WaveletTransform::SP_InvertStep(int *input, int *output, int size)
{
    int i, k, d1, d2, t;
    int lowSize = (size+1) >> 1;
    int highSize = size - lowSize;
    int *l = input;
    int *h = input + lowSize;
    int *out = output;

    t = (h[highSize-1] += (d1 = l[highSize-2] - l[highSize-1]) >> 2);
    for (i = highSize - 2; i > 0; i--)
    {
        d2 = d1;
        d1 = l[i-1] - l[i];
        t = (h[i] += (((d1 + d2 - t) << 1) + d2 + 3) >> 3);
    }
    h[0] += d1 >> 2;

    for (i = k = 0; i < highSize; i++, k += 2)
    {
        out[k] = l[i] + ((h[i] + 1) >> 1);
        out[k+1] = out[k] - h[i];
    }

    if ( lowSize > highSize )
        output[2*highSize] = l[highSize];
}

//
// Файл wavelet.h
//
#ifndef WAVELET_H
#define WAVELET_H

#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <memory.h>
#include <string.h>

```

```

class WaveletTransform
{
public:

    WaveletTransform ();
    ~WaveletTransform ();

    BOOL FWT (float *coeff, int trSteps, int hSize, int vSize);
    BOOL IWT (float *coeff, int trSteps, int hSize, int vSize,
              int maxOrig, int minOrig);

    int npad;

protected:

    void Reflection(float * h, float * t);
    void SP_TransformStep(int *input, int *output, int size);
    void SP_InvertStep(int *input, int *output, int size);
};
#endif

```

БИБЛИОГРАФИЯ

Глава 1

1. Вайдьянатхан П.П. Цифровые фильтры, блоки фильтров и полифазные цепи с многочастотной дискретизацией: Методический обзор //ТИИЭР, 1990, №3. С.77-120.
2. Харатишвили Н.Н. Пирамидальное кодирование. М.: Мысль, 1997.160с.
3. Burt P., Adelson H.. The Laplacian pyramid as a compact image code //IEEE Transactions on Communications, 1983, №4. P.23-38.
4. Malvar H.S. Lapped transforms for efficient transform /subband coding // IEEE Trans. Acoust. Speech Signal Process., 1990, №2. P.969-978.
5. Simoncelli E., Adelson E.. Subband transforms / Woods J., editor. Subband Image Coding. Kluwer Academic Publishers, Norwell, MA, 1990.
6. Vaidyanathan P.P. Theory of optimal orthonormal subband coders //IEEE Trans. on Signal Processing, 1998, №4. P.1528-1543.

Глава 2

1. Астафьева Н.М. Вейвлет-анализ:основы теории и некоторые приложения //Успехи физических наук, 1996, №11. С.1145-1170.
2. Желудев В.А. О вейвлетах на базе периодических сплайнов //Докл. РАН, 1994, №1. С. 9- 13.
3. Петухов А. П. Периодические дискретные всплески //Алгебра и анализ, 1996, №3. С. 151–183.
4. Кравченко В. Ф., Рвачев В. А., Пустовойт В. И. Ортонормированные системы типа wavelet на основе атомарных функций //Докл. РАН, 1996, №1.С. 16–18.
5. Daubechies I. Orthonormal Bases of Compactly Supported Wavelets //Comm. Pure Appl. Math., 1988, vol.41.P. 906-966.
6. Daubechies I. The wavelet transform, time-frequency localization and signal analysis // IEEE Trans. Inform. Theory, 1990, №5. P.961-1005.
7. Jawerth B., Sweldens W. An overview of wavelet based multiresolution analysis //SIAM Rev., 1994, №3. P.377-412.
8. Mallat S. A theory for multiresolution signal decomposition:the wavelet representation //IEEE Trans. Pattern Analysis and Machine Intelligence, 1989, №7. P.674-693.

9. Rioul O., Vetterli M. Wavelets and signal processing //IEEE Signal Processing Magazine, 1991, №10. P. 14-38.

10. Schumaker L., Webb G., editors. Recent Advances in Wavelet Analysis. Academic Press, New York, 1993.480p.

11. Strang G. Wavelets and dilation equations: A brief introduction //SIAM Rev., 1989, №4. P.614-627.

12. Vetterli M., Herley C. Wavelets and Filter Banks: Theory and Design // IEEE Transactions on Signal Processing, 1992, v.40. P. 2207-2232.

Глава 3

1. Balasingham I. On Optimal Perfect Reconstruction Filter Banks for Image Compression. PhD Thesis. NUST, Trondheim, Norway, 1998r.

2. Johnston J. A filter family designed for use in quadrature mirror filter banks //Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, 1980. P.291-294.

3. Vetterli M. Splitting a signal into subband channels allowing perfect reconstruction //Proc. IASTED Conf. Appl. Signal Proc., 1985, Paris.

Глава 4

1. Strang G., Nguyen T. Wavelets and Filter Banks. Wellesley-Cambridge Press, Wellesley, MA, 1996.

2. Majani E. Biorthogonal wavelets for image compression //Proc. SPIE Visual Communications and Image Processing, 1994.

3. Smith M., Barnwell T. Exact reconstruction techniques for tree - structured subband coders // IEEE Transactions On Acoustic, Speech, Signal Processing, 1986, №6.P. 434-441.

4. Soman A., Vaidyanathan P. On orthonormal wavelets and paraunitary filter banks. // IEEE Trans. Signal Processing, 1993,№3. P. 1170-1183.

5. Vetterli M. Filter banks allowing perfect reconstruction //IEEE Transactions On Signal Processing. 1986, №2. P. 219-244.

6. Vetterli M. Wavelet and filter banks for discrete-time signal processing //Ruskai M. , ed. Wavelets and Their Applications, 1992, Boston. P.17-52.

Глава 5

1. Herley C., Kovacevic J., Ramchandran K., Vetterli M. Tilings of the time-frequency plane: Construction of arbitrary orthogonal bases and fast tiling algorithms //IEEE Transactions on Signal Processing, 1993, №12. P. 3341-3359.

2. Herley C., Xiong Z., Ramchandran K., Orchard M. Joint space-frequency segmentation using balanced wavelet packet trees for least-cost image representation //IEEE Transactions on Image Processing, 1997, №9.

3. Ramchandran K., Vetterli M. Best wavelet packet bases in a rate-distortion sense //IEEE Transactions on Image Processing, 1992, №2.P.160-175.
4. Saito N. Local Feature Extraction and Its Applications Using a Library of Bases. Ph.D Dissertation, Yale Univ., 1994.
5. Xiong Z. Image Coding Using Wavelet Transform. PhD Thesis, Illinois Univ., 1996.
6. Wickerhauser M. Acoustic signal compression with wavelet packets. In Chui C. ed. Wavelets :A Tutorial in Theory and Applications, 1992, Academic Press, Boston. P. 679-700.

Глава 6

1. Daubechies I., Sweldens W. Factoring wavelet transforms into lifting steps. Technical report, Bell Laboratories, Lucent Technologies, 1996.
2. Sweldens W., Schreder P. Building your own wavelets at home. Technical Report, Industrial Mathematics Initiative, Mathematics Department, University of South Carolina, 1995.
3. Sweldens W. The lifting scheme: A construction of second generation wavelets. Technical Report, Industrial Mathematics Initiative, Mathematics Department, University of South Carolina, 1995.
4. Sweldens W. The lifting scheme: A custom-design construction of biorthogonal wavelets //Applied and Computational Harmonic Analysis, 1996, №2. P.186-200.

Глава 7

1. Adams M., Antoniou A. Design of reversible subband transforms using lifting //Proc. of IEEE Pacific Rim Conference, 1997, v.1. P. 489-492.
2. Adams M. Reversible Wavelet Transforms and Their Application to Embedded Image Compression. Master Thesis. University of Victoria, 1998.
3. Balasingham I., Lervik J., Ramstad T. Lossless subband image compression using integer coefficient filter banks and classwise arithmetic coding //Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, Seattle, 1998.
4. Calderbank A., Daubechies I., Sweldens W., Yeo B. Wavelet transforms that map integers to integers. Tech. report, Department of Mathematics, Princeton University, 1996.
5. Jung H., Prost R. Rounding transform based approach for lossless subband coding //Proc. IEEE Int. Conf. on Image Processing, 1997, v.2. P.274-277.
6. Zandi A., Allen J., Schwartz E., Boliek M. CREW: Compression with reversible wavelets //Proc. of IEEE Data Compression Conf., 1995.P.212-221.

Глава 8

1. Lebrun J., Vetterli M. Balanced multiwavelets: theory and design. IEEE Trans. Signal Proc., №4, 1998.
2. Geronomo J., Hardin D., Massopust P.. Fractal functions and wavelet expansions based on several scaling functions //Journal Approx.Theory, 1993, №3.
3. Strang G., Strela V. Short wavelets and matrix dilation equations //IEEE Trans.Signal Proc., 1995, v.3. P.108-115.
4. Strela V. Multiwavelets: Theory and applications. PhD Thesis. Massachusetts Institute Technology, 1996.
5. Park H. A computational theory of Laurent polynomial rings and multidimensional FIR systems. PhD Thesis. UC Berkeley, 1995.
6. Strela V., Heller P., Strang G., Topiwala G., Heil C. The application of multiwavelet filter banks to image processing //IEEE Trans. Sig.Proc., 1996.
7. Plonka G., Strela V. From wavelets to multiwavelets / Mathematical Methods for Curves and Surfaces M.Dahlen, T.Lyche, L.Schumaker (eds.), Vanderbilt University Press, Nashville, 1998. P. 375-399.

Глава 9

1. Галлагер Р. Теория информации и надежная связь. М.:Мир, 1974.
2. Макхоул Д., Рукос С., Гиш Г. Векторное квантование при кодировании речи //ТИИЭР, т.73, №11, 1985. С.19-61.
3. Шеннон К. Математическая теория связи / Шеннон К. Работы по теории информации и кибернетике.М.:1963. С.243-333.
4. Mallat S. Analysis of low bit rate image transform coding //IEEE Trans. Signal Proc., 1998, №4.P.1127-1142.

Глава 10

1. Antonini M., Barlaud M., Mathieu P. Image Coding Using Wavelet Transform //IEEE Trans. Image Proc., 1992, №2. P.205-220.
2. Criminal Justice Information Services. WSQ Grayscale Fingerprint Image Compression Specification. Federal Bureau of Investigation, Feb.1993.
3. Davis G., Chawla S. Image coding using optimized significance tree quantization //Proc. Data Compression Conference, 1997. P. 387-396.
4. Lewis A., Knowles G. Image compression using the 2-d wavelet transform //IEEE Transactions on Image Processing, 1992, №2. P.244-250.
5. LoPresto S., Ramchandran K., Orchard M.. Image coding based on mixture modeling of wavelet coefficients and a fast estimation-quantization framework //Proc. Data Compression Conference, Snowbird, 1997. P. 221-230.
6. Mallat S., Falzon F. Understanding image transform codes //Proc. SPIE Aerospace Conf., Orlando, 1997.

7. Said A., Pearlman W. A new, fast, and efficient image codec based on set partitioning in hierarchical trees //IEEE Trans. on Circuits and Systems for Video Technology, 1996, №3. P.243-250.

8. Shapiro J. Embedded image coding using zerotrees of wavelet coefficients //IEEE Trans. on Signal Processing, 1993, №12. P.3445-3462.

9. Shoham Y., Gersho A. Efficient bit allocation for an arbitrary set of quantizers //IEEE Trans. Acoustics, Speech, and Signal Processing, 1988, №9. P.1445-1453.

10. Villasenor J., Belzer B., Liao J. Wavelet filter evaluation for image compression //IEEE Trans. on image processing, 1995, №8. P.1053-1060.

11. Yoo Y., Ortega A., Yu B.. Progressive classification and adaptive quantization of image subbands. Preprint, 1997.

Глава 11

Техническая информация фирмы Analog Devices, которую можно получить на сайте: <http://www.analog.com>

Интернет-адреса

Адрес сервера фирмы Mathsoft, на котором имеется много ссылок на статьи по вейвлетам, сгруппированных по категориям:

<http://www.mathsoft.com/wavelet.html>

Очень хороший список групп, исследующих вейвлеты, находится на сервере корейского университета:

<http://dali.korea.ac.kr/research/Wavelet/wavelets.html>

В.Свелденс поддерживает в Сети общедоступный вейвлет-дайджест – ежемесячный реферативный журнал. Адрес:

<http://www.wavelet.org>

Адрес сервера В.Свелденса, на котором размещены его статьи и диссертация «Вейвлеты и численный анализ»:

<http://cm.bell-labs.com/who/wim/papers/papers.html>

Статьи посвящены различным математическим аспектам теории вейвлетов.

Адрес Е.Ковачевич, руководителя подразделения лаборатории Bell, занимающегося вейвлетами:

<http://cm.bell-labs.com/who/jelena/papers/papers.html>

Здесь вы найдете много интересных статей, посвященных локальным базисам, произвольному разбиению частотно-временной плоскости.

На сервере Дортмутского колледжа имеется ряд работ Д.Дэвиса, в том числе и его диссертация, посвященная адаптивной нелинейной аппроксимации (алгоритм Matching Pursuit). Здесь же находится общедоступный C++ код вейвлет-кодера изображений.

<http://www.cs.dartmouth.edu/~Davis>

Большое количество работ ученых лаборатории вычислительной математики Райсовского университета размещено на сервере

<http://cml.rice.edu>

Особый интерес этот адрес должен вызвать у людей, занимающихся частотно-временным анализом сигналов, обнаружением локальных особенностей и т.д. Советуем обратить внимание на многочисленные работы Р.Баранюка, которые он любезно сделал доступными в он-лайнном режиме.

Читателям, интересующимся вопросом распараллеливания алгоритма вейвлет-преобразования, рекомендуем ознакомиться с работами А. Ухла:

<http://www.mat.sbg.ac.at/~uhl/pubs.html>

Те, кого в первую очередь интересуют вопросы применения многоскоростной цифровой фильтрации для кодирования изображений, не должны пройти мимо работ норвежских ученых:

анализ артефактов, возникающих при кодировании, выполнен в диссертации А.Аасе:

<http://www.hsr.no/~aase/publicatons.html>

оптимизация блоков фильтров, в т.ч. и вейвлет-фильтров, проведена И.Баласингамом:

<http://www.tele.ntnu.no/~ilangko>

обработка ЭКГ:

<http://www.hsr.no/SIGPROC/www/programs.html>

(здесь же имеется программный код).

Вопросы целочисленного вейвлет-преобразования исследовал в своей диссертации М.Адамс:

<http://www.ece.ubc.ca/~mdadams/papers>

Наверное, наиболее популярным алгоритмом вейвлет-кодирования изображения является SPIHT. Полное описание алгоритма, а также EXE-модули можете найти по адресу:

<http://ipl.rpi.edu/RESEARCH/SPIHT>

Много интересных статей, а также C-код алгоритма сжатия EPWIC (не рассмотренного на страницах нашей книги) предлагает Е.Симонселли по адресу:

<http://www.cns.nyu.edu/~eero/publications.html>

С работами Д.Вилласенора можете ознакомиться по адресу

<http://www.icsl.ucla.edu>

Особо рекомендуем прочесть статью “Wavelet filter evaluation for image compression”.

Труды М.Веттерли можно найти по адресу

<http://www.eecs.berkeley.edu>.

К сожалению, они недоступны в он-лайновом режиме.

Вопросам кодирования изображений, видео и одновременного обеспечения их передачи по каналам связи посвящено много трудов. Проблемы передачи сжатой видеоинформации по сетям исследуют по адресу

<http://sipi.usc.edu/~ORTEGA/>

Здесь вы найдете диссертацию и много работ А.Ортега, а также работы его учеников.

С методами вейвлет-компрессии изображений и передачи их по радиоканалу можно ознакомиться, например, в диссертации Х.Джавархани по адресу:

http://www.isr.umd.edu/TechReports/ISR/1997/Ph_D_97_2

В настоящей книге не были рассмотрены вопросы векторного квантования коэффициентов вейвлет-преобразования. Однако это вовсе не означает, что данные методы являются неэффективными. Многие исследователи занимаются этими вопросами. Для первоначального знакомства с проблемой мы бы советовали прочесть обзорную статью

Cosman P., Gray R., and Vetterli M. Vector quantization of image subbands: a survey //IEEE Transactions on Image Processing, 1996, №2. P.202-225. (166 ссылок на источники (!)).

Интересные тексты кодов вейвлет-преобразования можно получить на сервере черноголовского отделения РАН:

<http://netserv2.chg.ru/ftp/prog/wavelets/programming>

Вообще говоря, список Интернет-адресов можно было бы продолжать до бесконечности. Но все-таки поиск информации в Сети – личное дело

каждого. Для осуществления поиска мы, например, использовали такие серверы, как Yahoo, Altavista, Hotbot, Infoseek и другие. Для получения качественного результата поиска необходимо предельно точно набирать запрос. Например, мы набрали: “image compression”. Ответом будут сотни тысяч ссылок, среди которых преобладает рекламная информация. Если же наберем нечто вроде “wavelet zerotree” или “optimal quantization”, ссылок будет значительно меньше, и почти все они будут информативными.

Так что ЖЕЛАЕМ УДАЧИ!

ОГЛАВЛЕНИЕ

Предисловие	3
Введение	5
Глава 1. Субполосное кодирование	7
1.1. Требования, предъявляемые к преобразованиям	8
1.2. Линейные преобразования конечных сигналов	9
1.2.1. Система фильтров анализа-синтеза	9
1.2.2. Каскадное соединение систем А-С	11
1.2.3. Представление субполосного кодирования при помощи аппарата матриц	13
1.2.4. Обратное преобразование	15
1.2.5. Ортогональное преобразование	16
1.3. Некоторые примеры преобразований	17
1.3.1. Преобразование Габора	17
1.3.2. Дискретное косинусное и перекрывающееся ортогональное преобразования	18
1.3.3. Пирамида Лапласа	19
1.4. Квадратурно-зеркальные фильтры	21
1.4.1. Построение КЗФ	23
1.4.2. Асимметричная система	23
1.5. О преимуществе преобразования при помощи блоков фильтров перед преобразованием Фурье	24
Глава 2. Основы теории вейвлет-преобразования	28
2.1. Непрерывное вейвлет-преобразование	28
2.2. Кратномасштабное представление функций	31
2.2.1. Представление функций при помощи вейвлетов	36
2.3. Вейвлет-ряды дискретного времени	39
2.4. Дискретное вейвлет-преобразование	42
2.4.1. Матричное описание DWT	42
2.4.2. Описание DWT посредством блоков фильтров	44
2.5. Гладкость базисных функций	47
Глава 3. Вейвлет-декомпозиция сигналов произвольной длины	50
3.1. Условия полного восстановления сигнала	50

3.2. Методика расчета фильтров, позволяющих осуществить полное восстановление сигнала	52
3.3. Продолжения сигналов, сохраняющие свойство полного восстановления	54
3.3.1. Периодическое продолжение	55
3.3.2. Симметричное продолжение	56
3.4. Эффективный метод продолжения для декомпозиции сигнала произвольной длины	58
3.5. Симметрично-периодическое продолжение сигнала	59
Глава 4. Сравнение вейвлет-фильтров с фильтрами, применяемыми при субполосном кодировании	61
4.1. Критерии для расчета фильтров	61
4.2. Построение обычных фильтров: фильтры Джонстона	64
4.3. Расчет вейвлет-фильтров	66
4.3.1. Расчет фильтров Добеши	66
4.3.2. Расчет пары биортогональных фильтров	67
4.4. Критерий оптимизации блоков фильтров, используемых при кодировании изображений	71
4.4.1. Выигрыш от субполосного кодирования	72
4.4.2. Оптимальное распределение бит	73
4.5. Сравнение характеристик обычных и вейвлет-фильтров	76
Глава 5. Адаптивные ортогональные преобразования	79
5.1. Пакеты вейвлетов (алгоритм одиночного дерева)	79
5.2. Алгоритм двойного дерева	82
5.3. Частотно-временное дерево	85
5.4. Сравнение обсуждаемых алгоритмов	86
5.4.1. Размерность библиотеки базисов	86
5.4.2. Вычислительная сложность алгоритмов	88
5.4.3. Эффективность кодирования изображений	88
Глава 6. Лифтинговая схема	90
6.1. Этап разбиения	91
6.2. Этап предсказания	91
6.3. Различные операторы предсказания	93
6.4. Этап обновления	98

Глава 7.	Целочисленное вейвлет-преобразование	105
7.1.	Целочисленные вейвлет-преобразования	105
7.2.	Лифтинговая схема и целочисленная биортогональная фильтрация	110
7.3.	Метод коррекции ошибок для получения целочисленного вейвлет-преобразования	113
Глава 8.	Мультивейвлеты	117
8.1.	Блоки мультифильтров	117
8.1.1.	Основы теории блоков фильтров, изменяющихся во времени	117
8.1.2.	Построение блоков мультифильтров	121
8.1.3.	Итерирование блоков мультифильтров	121
8.2.	Мультивейвлеты	122
8.3.	Обработка сигналов в базисе мультивейвлетов	125
8.4.	Сбалансированные мультивейвлеты	128
Глава 9.	Потенциальные характеристики кодирования изображения с применением вейвлет-преобразования	130
9.1.	Основные формулы и теоремы теории связи, относящиеся к кодированию с преобразованием при высоких скоростях (>1 бит/пиксел)	130
9.1.1.	Скалярное квантование с ограниченной энтропией	131
9.1.2.	Зависимость искажения от скорости	132
9.2.	Сжатие изображения при низких скоростях кодирования	134
9.2.1.	Функция искажение-скорость	135
9.2.2.	Оптимальный относительный размер интервала квантования	142
9.2.3.	Практическая проверка точности аналитических выражений	142
Глава 10.	Применение вейвлет-преобразования для сжатия изображения	147
10.1.	Базовый вейвлет-кодер изображения	148
10.1.1.	Выбор вейвлетов для сжатия изображения	148
10.1.2.	Осуществление преобразования на границах изображения	150
10.1.3.	Квантование	150
10.1.4.	Энтропийное кодирование	151
10.1.5.	Распределение бит	152
10.1.6.	Меры искажения, взвешенные с учетом восприятия человеком	153
10.2.	Новые идеи в области сжатия изображений, связанные с вейвлет-преобразованием	153
10.3.	Кодирование посредством нульдеревя	156

10.3.1.	Алгоритм Льюиса и Ноулеса	157
10.3.2.	Алгоритмы Шапиро и Саида-Перельмана	159
10.3.3.	Оптимизация нульдеревьев по критерию скорость-искажение ...	161
10.4.	Частотно, пространственно-частотно адаптивные кодеры	162
10.5.	Использование зависимостей между вейвлет-коэффициентами внутри субполос	163
10.5.1.	Решетчатое квантование	163
10.5.2.	Субполосные кодеры с РК	165
10.5.3.	Моделирование и оценивание смеси распределений	167
10.6.	Современные направления исследований	169
Глава 11.	Видекодеки семейства ADV6XX производства фирмы Analog Devices	172
11.1.	Принципы работы ADV601	173
11.2.	Использование микросхемы ADV601	175
Заключение		179
Приложение 1		181
Приложение 2		183
Библиография		192