



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ,
ПАТЕНТАМ И ТОВАРНЫМ ЗНАКАМ

(51) МПК
G06T 9/00 (2006.01)
H04N 7/30 (2006.01)

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21), (22) Заявка: **2005135425/09**, **14.11.2005**

(24) Дата начала отсчета срока действия патента:
14.11.2005

(30) Конвенционный приоритет:
17.12.2004 US 11/015,148

(43) Дата публикации заявки: **27.05.2007**

(45) Опубликовано: **27.12.2009** Бюл. № 36

(56) Список документов, цитированных в отчете о поиске: **TRAC D. TRAN, Lapped Transform via Time-Domain Pre- and Post-Processing, 2001 Conference on Information Sciences and Systems, The Johns Hopkins University, Mar.21-23, 2001. US 6771828 B1, 03.08.2004. EP 1202219 A1, 02.05.2002. US 5859788 A, 12.01.1999. RU 2003108435 A, 10.10.2004. RU 2202826 C2, 20.04.2004. HENRIQUE S. MALVAR, Fast Progressive (см. прод.)**

Адрес для переписки:
**129090, Москва, ул. Б.Спасская, 25, стр.3,
ООО "Юридическая фирма Городиский и
Партнеры", пат.пов. Ю.Д.Кузнецову,
рег.№ 595**

(72) Автор(ы):

**ТУ Чэнцзе (US),
МАЛЬВАР Энрике Сарменто (US),
СРИНИВАСАН Сридхар (US)**

(73) Патентообладатель(и):

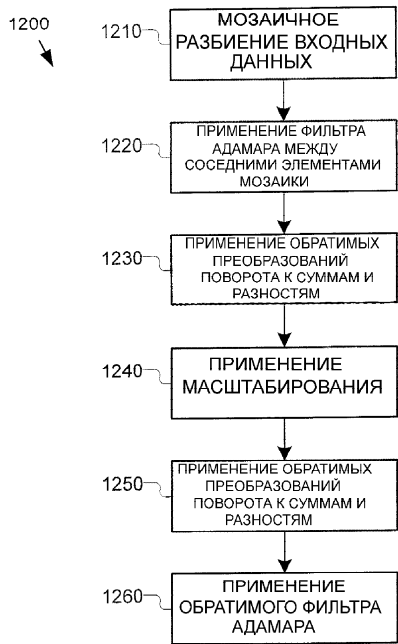
МАЙКРОСОФТ КОРПОРЕЙШН (US)

(54) ОПЕРАТОР ОБРАТИМОГО ПЕРЕКРЫТИЯ ДЛЯ ЭФФЕКТИВНОГО СЖАТИЯ ДАННЫХ БЕЗ ПОТЕРЬ

(57) Реферат:

Изобретение относится к сжатию цифрового мультимедиа с помощью преобразования с перекрытием. Техническим результатом является повышение эффективности сжатие/восстановление данных без потерь. Предложено эффективное преобразование с перекрытием, реализуемое с помощью предварительных фильтров и постфильтров (или операторов обратимого перекрытия), которые составлены из компонентных матриц с единичным детерминантом. Предварительные фильтры и

постфильтры реализованы как последовательность плоских преобразований поворота и плоских преобразований масштабирования с единичным детерминантом. Плоские преобразования масштабирования могут быть реализованы с помощью планарных сдвигов или этапов лифтинга. Дополнительно плоские повороты и плоские сдвиги имеют реализацию как обратимые операции без потерь, предоставляя в качестве результата оператор обратимого перекрытия. 3 н. и 15 з.п. ф-лы, 15 ил.



ФИГ. 12

(56) (продолжение):

Image Coding without Wavelets, IEEE DATA COMPRESSION CONFERENCE - SNOWBIRD, March 2000.

RU 2 3 7 7 6 5 3 C 2

RU 2 3 7 7 6 5 3 C 2



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY,
PATENTS AND TRADEMARKS

(51) Int. Cl.
G06T 9/00 (2006.01)
H04N 7/30 (2006.01)

(12) ABSTRACT OF INVENTION

(21), (22) Application: **2005135425/09, 14.11.2005**

(24) Effective date for property rights:
14.11.2005

(30) Priority:
17.12.2004 US 11/015,148

(43) Application published: **27.05.2007**

(45) Date of publication: **27.12.2009 Bull. 36**

Mail address:
**129090, Moskva, ul. B.Spasskaja, 25, str.3, OOO
"Juridicheskaja firma Gorodisskij i Partnery",
pat.pov. Ju.D.Kuznetsovu, reg.№ 595**

(72) Inventor(s):
**TU Chehntsze (US),
MAL'VAR Ehnrike Sarmeto (US),
SRINIVASAN Sridkhar (US)**

(73) Proprietor(s):
MAJKROSOFT KORPOREJShN (US)

(54) REVERSIBLE OVERLAP OPERATOR FOR EFFICIENT LOSSLESS DATA COMPRESSION

(57) Abstract:

FIELD: physics; computer engineering.

SUBSTANCE: invention relates to compressing digital multimedia through lapped transform. Efficient lapped transform realised using pre- and post-filters (or reversible overlap operators) composed of unit determinant component matrices, is proposed. Pre- and post-filters are realised as a series of planar rotational transforms and unit determinant planar scaling transforms. The planar scaling transforms can be implemented using planar shears or lifting steps. Further, the planar rotations and planar shears have an implementation as reversible lossless operations, as a result, providing a reversible overlap operator.

EFFECT: more efficient lossless

compression/retrieval of data.

18 cl, 15 dwg



RU 2 377 653 C2

RU 2 377 653 C2

Область техники, к которой относится изобретение

Изобретение относится к сжатию цифрового мультимедиа (к примеру, видео и изображений) с помощью преобразования с перекрытием.

Предшествующий уровень техники**Преобразования с перекрытием**

Преобразование с перекрытием - это мощная методика обработки сигналов, которая используется при сжатии данных. См., к примеру, H. S. Malvar, "Signal Processing with Lapped Transforms", Boston, MA: Artech House, 1992 г. Тем не менее, до настоящего времени эффективные преобразования с перекрытием с линейной фазой не были сформулированы и никогда не применялись для сжатия данных без потерь (обратимого).

Как описано более подробно ниже, известно, что преобразование с перекрытием может быть сформулировано как предварительный фильтр, за которым следует преобразование данных (и его инверсия как обратное преобразование данных, за которым следует постфильтр). См., к примеру, H. S. Malvar, "A pre- and post-filtering technique for the reduction of blocking effects", на Proc. Picture Coding Symposium, Стокгольм, Швеция, июнь 1987 г.; и T.D. Tran, J. Liang и C. Tu, "Lapped Transform via Time-Domain Pre- and Post-Filtering", IEEE Trans. on Signal Processing, том 51, номер 6, июнь 2003 г. Преобразование данных без потерь может быть использовано в данной формулировке, чтобы достичь хорошего показателя обратимости. До настоящего времени считалось, что только определенное ограниченное множество предварительных фильтров и постфильтров может быть выбрано для обратимости. Этот ограниченный набор очень ограничен в своей производительности сжатия (скорость по сравнению с искажением, или R-D). В новой статье (W. Dai и T. Tran, "Regularity-constrained pre- and post-filtering for block DCT-based systems", IEEE Trans. on Signal Processing, том 51, стр.2568-2581, октябрь 2003 г.), была представлена структура, в которой большинство элементов являются обратимыми и которая имеет хорошие свойства сжатия.

В сжатии звука было внедрено несколько структур для обратимых преобразований с перекрытием. См., к примеру, R. Geiger, J. Herre, J. Koller и K. Brandenburg, "IntMDCT - A link between perceptual and lossless audio coding", на Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, Орландо, штат Флорида, май 2002 г.; и J. Li, "Reversible FFT and MDCT via matrix lifting", на Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, Монреаль, Канада, май 2004 г. Тем не менее, эти структуры применимы только к модулированному преобразованию с перекрытием (MLT), также известному как модифицированное дискретное косинусное преобразование (MDCT), базисные функции которого являются ортогональными и не являются симметричными (т.е. базисные функции не соответствуют линейной фазе). Эти преобразования не применимы к приложениям сжатия данных, в которых требуются функции линейной фазы (симметричные), такие как цифровое сжатие изображений.

Для сжатия рисунков (изображений) одна из наиболее производительных методик преобразования с точки зрения производительности R-D - это биортогональное преобразование с перекрытием (LBT). См. S. Malvar, "Biorthogonal and nonuniform lapped transforms for transform coding with reduced blocking and ringing artifacts", IEEE Trans. on Signal Processing, том 46, стр.1043-1053, апрель 1998 г. В отличие от MLT, базисные функции LBT являются симметричными и не являются точно ортогональными (в LBT базисные функции анализа являются ортогональными для базисных функций синтеза,

следовательно, условно биортогональными). ЛВТ успешно использовались в приложениях сжатия изображений, но они до сих пор не использовались в сжатии изображений без потерь, поскольку численно-целообратимые структуры были неизвестны.

5 Обзор блочного основанного на преобразовании кодирования

Кодирование с преобразованием - это методика сжатия, используемая во многих системах сжатия звука, изображений и видео. Несжатое цифровое изображение или видео в типичном случае представляется или захватывается как выборки элементов изображений или цветов в местах в изображении или видеокадре, организованные в 10 двумерную (2D) сетку. Это упоминается как представление в пространственной области изображения или видео. Например, типичный формат для изображений состоит из потока 24-битных выборок элементов цветных изображений, организованных как сетка. Каждая выборка - это число, представляющее цветные 15 компоненты в месте пикселя сетки в рамках цветового пространства, такого как RGB или YIQ, помимо прочего. Различные системы изображений и видео могут использовать различные цветовые, пространственные и временные разрешения выборки. Также цифровое аудио в типичном случае представляется как 20 дискретизируемый по времени поток звуковых сигналов. Например, типичный формат аудио состоит из потока 16-битных выборок по амплитуде звукового сигнала, взятых с постоянным интервалом времени.

Несжатые цифровые сигналы звука, изображений и видео могут потреблять 25 значительную емкость при хранении и передаче. Кодирование с преобразованием уменьшает размер цифрового аудио, изображений и видео посредством преобразования представления сигнала в пространственной области в представление в частотной области (или другой аналогичной области преобразования) и последующего уменьшения разрешения определенных, в общем, менее 30 воспринимаемых частотных компонентов представления в области преобразования. Это обычно дает в результате гораздо менее воспринимаемое ослабление цифрового сигнала по сравнению с уменьшением цветового или пространственного разрешения изображений или видео в пространственной области или звука во временной области.

Более конкретно, типичный блочный основанный на преобразовании кодек 100, 35 показанный на фиг.1, делит пиксели несжатого цифрового изображения на двумерные блоки фиксированного размера (X_1, \dots, X_n), при этом каждый блок, возможно, перекрывается другими блоками. Линейное преобразование 120-121, которое выполняет анализ пространственных частот, применяется к каждому блоку, что 40 преобразует разнесенные выборки в рамках блока к набору коэффициентов частот (или преобразования), обычно представляющих интенсивность цифрового сигнала в соответствующих полосах частот по блочному интервалу. Для сжатия коэффициенты преобразования могут быть выборочно квантованы 130 (т.е. уменьшено их разрешение, например, посредством отбрасывания самых младших битов значений коэффициентов или иного отображения значений в числовом множестве с более 45 высоким разрешением на меньшее разрешение), а также закодированы 130 энтропийным кодированием или кодированием с переменной длиной поля в сжатый поток данных. При декодировании коэффициенты преобразования обратно 50 преобразуются 170-171, чтобы приблизительно восстановить исходный дискретизированный по цвету/пространству сигнал изображения/видео (восстановленные блоки $\hat{X}_1 \dots \hat{X}_n$).

Блочное преобразование 120-121 может быть задано как математическая операция

над вектором x размера N . Более часто операцией является линейное умножение, дающее выход области преобразования $y=Mx$, где M является матрицей преобразования. Когда входные данные имеют произвольную длину, они сегментируются на N векторов определенной длины, и блочное преобразование

5 применяется к каждому сегменту. Для целей сжатия данных выбираются обратимые блочные преобразования. Другими словами, матрица M является обратимой. В нескольких измерениях (к примеру, для изображения и видео) блочные преобразования в типичном случае реализуются как отдельные операции.

10 Матричное умножение применяется раздельно вдоль каждого измерения данных (т.е. и строк, и столбцов).

Для сжатия коэффициенты преобразования (компоненты вектора y) могут быть выборочно квантованы (т.е. уменьшено их разрешение, например, посредством отбрасывания самых младших битов значений коэффициентов или иного отображения значений в числовом множестве с более высоким разрешением на меньшее разрешение), а также закодированы энтропийным кодированием или кодированием с

15 переменной длиной в сжатый поток данных.

При декодировании в декодере 150 инверсия этих операций (деквантованию/энтропийное декодирование 160 и обратное блочное преобразование 170-171) применяется на стороне декодера 150, как показано на фиг.1. При восстановлении данных обратная матрица M^{-1} (обратное преобразование 170-171) применяется в качестве множителя к данным области преобразования. При

20 применении к данным области преобразования обратное преобразование приблизительно восстанавливает исходное цифровое мультимедиа временной области или пространственной области.

Во многих приложениях кодирования на основе блочного преобразования преобразование, желательно, является обратимым, чтобы поддерживать кодирование, как с потерями, так и без потерь в зависимости от коэффициента квантования. При

30 отсутствии квантования (обычно представляемого как коэффициент квантования 1), например, кодек, использующий обратимое преобразование, может точно воспроизводить входные данные при декодировании. Тем не менее, требование обратимости в этих приложениях ограничивает выбор преобразований, на основе

35 которых может быть разработан кодек.

Многие системы сжатия изображений и видео, такие как MPEG и Windows Media, помимо прочих, используют преобразования на основе дискретного косинусного преобразования (DCT). DCT, как известно, имеет подходящие энергетические свойства

40 сжатия, что приводит к практически оптимальному сжатию данных. В этих системах сжатия обратное DCT (IDCT) используется в циклах восстановления, как в кодере, так и в декодере системы сжатия для восстановления отдельных блоков изображений. DCT описано в публикации N. Ahmed, T. Natarajan и K.R. Rao "Discrete Cosine Transform", IEEE Transactions on Computers, C-23 (январь 1974), на стр.90-93. Примерная реализация IDCT

45 описана в "IEEE Standard Specification for the Implementations of 8x8 Inverse Discrete Cosine Transform", IEEE Std. 1180-1990, 6 декабря 1990 г.

При сжатии статического изображения (или внутренне закодированного кадра видеопоследовательности) самые распространенные стандарты, такие как

50 MPEG-2, MPEG-4 и Windows Media, разбивают изображение на квадратные мозаичные элементы и применяют блочное преобразование к каждому мозаичному элементу изображения. На коэффициенты преобразования в заданном разделе (общеизвестном как блок) влияют только компоненты необработанных данных в этот блоке.

Необратимые операции или операции с потерями на стороне кодера, такие как квантование, обуславливают появления артефактов в закодированном изображении. Эти артефакты не зависят от блоков и создают визуально раздражающий эффект, называемый эффектом блокирования. Также для звуковых данных, когда
5 неперекрывающиеся блоки независимо кодируются с преобразованием, ошибки квантования генерируют разрывы в сигнале на границах блоков при восстановлении звукового сигнала в декодере. Для звука слышен эффект периодических щелчков.

Несколько методик используются, чтобы бороться с эффектом блокирования.

10 Наиболее популярные среди них - это фильтр деблокирования, который сглаживает края внутренних блоков, и пространственная экстраполяция, которая кодирует различия между необработанными входными данными и прогнозированием от краев соседних блоков. Эти методики имеют свои недостатки. Например, подход фильтра деблокирования - это "разомкнутый контур", т.е. процесс прямого преобразования не
15 принимает во внимание тот факт, что деблокирование должно выполняться до восстановления на стороне декодера. Кроме того, обе эти методики с большим объемом вычислений.

Чтобы минимизировать эффект блокирования, могут быть использованы
20 межблочные корреляции. Один способ достижения межблочной корреляции заключается в использовании преобразования с перекрытием, как описано в Н. Malvar, "Signal Processing with Lapped Transforms", Artech House, Норвуд, штат Массачусетс, 1992 г. Перекрывающееся преобразование - это преобразование, вход которого распространяется, помимо элементов данных в текущем блоке, на
25 несколько соседних элементов в соседних блоках. Также на стороне восстановления обратное преобразование влияет на все точки данных в текущем блоке, а также на несколько точек данных в соседних блоках.

В случае двумерных (2D) данных перекрывающееся двумерное преобразование -
30 это функция текущего блока, вместе с выбранными элементами блоков слева, сверху, справа, внизу и, возможно, сверху слева, сверху справа, внизу слева и внизу справа. Количество точек данных в соседних блоках, которые используются, чтобы вычислять текущее преобразование, упоминается как перекрытие.

35 **Обзор преобразования с перекрытием, соответствующего пространственной области**

Преобразование с перекрытием может быть реализовано в области преобразования в качестве этапа, который объединяет величины областей преобразования после традиционного блочного преобразования. Кроме того, это может быть реализовано в
40 пространственной области посредством стадии предварительной обработки, которая применяется к пикселям в диапазоне перекрытия. Эти две реализации математически связаны и поэтому эквивалентны.

Фиг.2 показывает пример традиционного преобразования с перекрытием соответствующей пространственной области. В показанном примере перекрытие
45 составляет два пикселя, и два пикселя, каждый из двух показанных соседних блоков, предварительно обрабатываются на стадии 210 предварительной обработки. Два предварительно обработанных вывода отправляются каждому из блоков для блочного основанного на преобразовании кодирования кодеком 100, как показано на
50 фиг.1. Инверсия стадии предварительной обработки применяется на стадии последующей обработки (постобработки) 220 после декодирования. С разумным выбором предварительной обработки и блочного преобразования широкий диапазон преобразований с перекрытием может быть реализован.

Ключевое преимущество реализации в пространственной области преобразования с перекрытием состоит в том, что используемый основанный на блочном преобразовании кодек может быть модернизирован с помощью стадии предварительной или последующей обработки, чтобы получать преимущества преобразования с перекрытием, т.е. уменьшенный эффект блокирования и лучшее сжатие, используя структуру используемого кодека. Предварительная обработка 210 и постобработка могут быть представлены как матричное умножение, как показано на фиг.3. Традиционно матрицы предварительной и постобработки являются инверсиями друг друга, т.е. перемноженные матрица предварительной обработки (P_f) и инверсия или матрица постобработки (P_i) равны единичной матрице I .

Определения

В общем, длина N преобразования - это число коэффициентов преобразования в конкретном блоке преобразования.

Основание K преобразования - это число точек входных данных, которые оказывают влияние на коэффициенты блока преобразования, также - это число точек выходных данных, которые находятся под влиянием каждого коэффициента преобразования посредством процесса обратного преобразования.

Для типичных блочных преобразований, таких как дискретное косинусное преобразование (DCT), длина и основание идентичны. Тем не менее, преобразования с перекрытием (LT) являются важным классом преобразований, для которых основание K больше, чем длина N . Обозначение $K \times N$ используется, чтобы отмечать основание и длину преобразования с перекрытием. (Преобразования, для которых $K < N$, являются расширяющимися и поэтому не используются в сжатии данных.)

В качестве примера 300, LT 310 6×4 , показанное на фиг.3, является преобразованием с шестью входами и четырьмя выходами. Поскольку преобразование является обратимым, два входа совместно используются соседними блоками преобразования. Обратное преобразование с перекрытием (ILT) 320 генерирует шесть выходов из своих четырех входов. Точки входных данных рядом с границей блоков (в данном случае одна точка в каждом конце блока) восстанавливаются посредством суммирования соответствующих откликов от двух соседних блоков обратного преобразования.

Ограничения на преобразования с перекрытием, используемые в системах сжатия

В математическом смысле преобразования с перекрытием - это обратимые структуры, когда мы рассматриваем входные и выходные сигналы, а также результаты промежуточных вычислений как действительные числа. Если бесконечная точность может быть достигнута, входные данные могут быть полностью восстановлены из своих коэффициентов преобразования с перекрытием. Тем не менее, бесконечная точность невозможна на практике; для сжатия данных без потерь требование состоит в том, чтобы спроектировать платформу, которая осуществляет операции в отношении целых чисел или с заданной точностью, тем не менее, полностью восстанавливает данные при заданном целочисленном представлении коэффициентов преобразования. Это более строгое условие, чем математическая обратимость, и такое преобразование, упоминаемое в данном документе как преобразование "без потерь". Более того, требуется, чтобы преобразование без потерь было так же эффективным и для сжатия данных (как без потерь, так и с потерями). Эта эффективность может быть измерена посредством энтропии преобразованных данных; чем ниже эта энтропия, тем больше преобразованные данные могут быть сжаты посредством стандартных методик энтропийного кодирования, таких как основанное на контексте арифметическое кодирование или адаптивное кодирование

по длинам серий.

Сущность изобретения

В данном документе описаны различные реализации эффективного преобразования с перекрытием, которое обратимо в целочисленной арифметике и может быть
5 использовано в качестве основы системы эффективного сжатия/восстановления данных без потерь.

Может быть показано, что наиболее эффективные структуры преобразования без потерь (т.е. структуры с минимальной энтропией преобразованных данных) требуют,
10 чтобы матрица преобразования была с единичным детерминантом (т.е. детерминант матрицы преобразования равнялся ± 1). В последующем описании предполагается, что преобразование может быть представлено как матричное умножение, хотя очевидно, что могут быть незначительные нелинейные явления, такие как округление данных. Таким образом, когда мы ссылаемся на детерминант, аспекты усечения и округления
15 не рассматриваются.

Эффективное преобразование с перекрытием реализуется с помощью предварительных фильтров и постфильтров, которые упоминаются в данном документе как "операторы перекрытия". Эта реализация является обратимой и
20 по-прежнему очень эффективной по R-D. Среди других вариантов применения, эти новые операторы перекрытия разрешают реализацию обратимых LBT, которые могут быть использованы для сжатия изображений без потерь. Предварительные фильтры и постфильтры используют обратимые операции. Дополнительно описанные операторы перекрытия включают в себя упрощения для вычислительной эффективности.

Одна реализация операций предварительной и последующей постфильтрации заключается в последовательности плоских преобразований поворота и плоских преобразований с единичным детерминантом масштабирования. Дополнительно плоские повороты и плоские сдвиги имеют реализацию как обратимые операции без
30 потерь, предоставляя в качестве результата оператор обратимого перекрытия.

Типичное применение - это одномерное преобразование с перекрытием 8x4, реализованное с помощью вычислительно эффективных аппроксимаций операторов обратимого перекрытия.

Дополнительные признаки и преимущества станут более явными из последующего
35 подробного описания вариантов осуществления, которое осуществляется со ссылками на прилагаемые чертежи.

Перечень чертежей

Фиг.1 - блок-схема традиционного блочного основанного на преобразовании
40 кодека согласно предшествующему уровню техники.

Фиг.2 - блок-схема преобразования с перекрытием, соответствующего пространственной области, реализованного в качестве операций предварительной и последующей фильтрации в сочетании с блочным основанным на преобразовании кодеком по фиг.1, также согласно предшествующему уровню техники.

Фиг.3 - блок-схема, иллюстрирующая пару преобразования с перекрытием и
45 обратного преобразования с перекрытием на одномерных данных.

Фиг.4 - блок-схема алгоритма кодера, основанного на преобразовании с перекрытием, использующем оператор обратимого перекрытия.

Фиг.5 - блок-схема декодера, основанного на преобразовании с перекрытием.
50

Фиг.6 - блок-схема, иллюстрирующая пару преобразования с перекрытием и обратного преобразования с перекрытием на одномерных данных, использующую операции предварительной и последующей фильтрации (или оператор обратимого

перекрытия) совместно с блочным преобразованием.

Фиг.7 - сигнальный ориентированный граф, иллюстрирующий структуру предварительного фильтра или постфильтра линейной фазы для использования в качестве оператора обратимого перекрытия в преобразовании с перекрытием по фиг.6.

Фиг.8 - сигнальный ориентированный граф масштабирования без потерь в качестве четырех этапов лифтинга для использования в операторе обратимого перекрытия.

Фиг.9 - сигнальный ориентированный граф масштабирования без потерь в качестве пяти этапов лифтинга для использования в операторе обратимого перекрытия.

Фиг.10 - сигнальный ориентированный граф каскадного двухточечного масштабирования, применяемого к матрице большей возможности, чтобы реализовать масштабирование без потерь с единичным детерминантом.

Фиг.11 - сигнальный ориентированный граф оператора обратимого перекрытия (или предварительного фильтра/постфильтра), имеющего структуру, показанную на фиг.7, и использующего масштабирование без потерь с единичным детерминантом по фиг.10.

Фиг.12 - блок-схема алгоритма работы оператора обратимого перекрытия по фиг.11.

Фиг.13 - сигнальный ориентированный граф, иллюстрирующий пример реализации обратимого преобразования с перекрытием, использующего оператор обратимого перекрытия по фиг.11.

Фиг.14 - график импульсной характеристики коэффициента постоянного тока (ДС коэффициента) примерного преобразования с перекрытием по фиг.13.

Фиг.15 - блок-схема подходящей вычислительной среды для реализации блочного основанного на преобразовании кодека с помощью усовершенствованного преобразования с перекрытием в пространственной области по фиг.4 и 5.

Подробное описание изобретения

Последующее описание относится к системе сжатия или кодеку цифрового мультимедиа, который использует оператор обратимого перекрытия для преобразования с перекрытием. В целях иллюстрации, вариантом осуществления системы сжатия, содержащей оператор обратимого перекрытия, является система сжатия изображений или видео. Альтернативно оператор обратимого перекрытия также может быть включен в системы сжатия или кодеки для других двумерных данных. Оператор обратимого перекрытия не требует, чтобы система сжатия цифрового мультимедиа кодировала сжатые цифровые мультимедийные данные в конкретном формате кодирования.

1. Кодер/декодер

Фиг.4 и 5 - это обобщенная диаграмма процессов, используемых в репрезентативном двумерном (2D) кодере 400 и декодере 500 данных, основанных на преобразовании с перекрытием, использующем оператор обратимого перекрытия. Схемы представляют обобщенную или упрощенную иллюстрацию использования и применения данного оператора обратимого перекрытия в системе сжатия, включающей кодер и декодер двумерных данных. В альтернативных кодерах, основанных на этом операторе обратимого перекрытия, дополнительное или меньшее число процессов, чем проиллюстрировано в репрезентативном кодере и декодере, может быть использовано для сжатия двумерных данных. Например, некоторые кодеры/декодеры могут также включать в себя цветное преобразование, форматы цветов, масштабируемое кодирование, кодирование без потерь, режимы макроблоков и т.д. Система сжатия (кодер и декодер) может предоставлять кодирование двумерных

данных без потерь и/или с потерями, в зависимости от квантования, которое может быть основано на параметре квантования, варьирующемся от соответствующего режима без потерь до соответствующего режима с потерями.

5 Кодер 400 двумерных данных генерирует сжатый поток 420 битов, который является более компактным представлением (для типичного ввода) двумерных
данных 410, представляемых в качестве входа кодери. Например, вводом двумерных
данных может быть изображение, кадр видеопоследовательности или другие данные,
имеющие два измерения. Кодер двумерных данных разбивает 430 входные данные на
10 макроблоки, которые имеют размер 16x16 пикселей в данном репрезентативном
кодере. Кодер двумерных данных дополнительно разбивает макроблок на блоки 432
4x4. Оператор 440 "прямого перекрытия" применяется к каждому краю между
блоками, после чего каждый блок 4x4 преобразуется с помощью блочного
15 преобразования 450. Этим блочным преобразованием 450 может быть обратимое
безразмерное двумерное преобразование, описанное Srinivasan в Патентной заявке
США, озаглавленной "Improved Reversible Transform For Lossy And Lossless 2-D Data
Compression", поданной одновременно с данной заявкой, раскрытие которой
20 включено в данный документ посредством ссылки. Альтернативно дискретное
косинусное преобразование или другие блочные преобразования могут быть
использованы с оператором обратимого перекрытия, описанным в данном документе.
Вслед за упомянутым преобразованием ДС коэффициент 460 каждого блока
преобразования 4x4 подвергается аналогичной цепочке обработки (разбиение, прямое
25 перекрытие, за которым следует блочное преобразование 4x4). Результирующие ДС
коэффициенты преобразования тока и АС коэффициенты (коэффициенты переменного
тока) преобразования квантуются 470, кодируются 480 энтропийным кодированием и
пакетируются 490.

Декодер выполняет обратный процесс. На стороне декодера биты коэффициентов
30 преобразования извлекаются 510 из своих соответствующих пакетов, из которых сами
коэффициенты декодируются 520 и деквантуются 530. ДС коэффициенты 540
регенерируются посредством применения обратного преобразования, и в отношении
плоскости ДС коэффициентов выполняется «обращенное перекрытие» с помощью
35 подходящего сглаживающего оператора, применяемого по краям блока постоянного
тока. В дальнейшем все данные регенерируются посредством применения обратного
преобразования 550 4x4 к ДС коэффициентам, и АС коэффициенты 542 декодируются
из потока битов. Наконец, края блока в плоскостях результирующих изображений
фильтруются 560 на основе обращенного перекрытия. Это генерирует
40 восстановленный вывод двумерных данных.

2. Преобразование с перекрытием, реализованное с помощью операторов перекрытия

Более обобщенно, оператор 440 перекрытия и блочное преобразование 450
кодера 400 (фиг.4) является примером обширного класса преобразований 600 с
45 перекрытием, которые могут быть разложены на операцию 610 предварительной
фильтрации, за которой следует блочное преобразование 620 данных, как
проиллюстрировано на фиг.6. Фиг.6 иллюстрирует обобщенный пример такого
разложения преобразований с перекрытием. В проиллюстрированном случае
50 преобразование с перекрытием 310 6x4, показанное на фиг.3, разлагается на стадии
операции 610 предварительной фильтрации и блочного преобразования 620.
Операция 610 предварительной фильтрации и блочное преобразование 620
равномерно применяются относительно точек данных. В данном

проиллюстрированном примере преобразования 600 с перекрытием 6×4 каждый предварительный фильтр - это преобразование с длиной 2 точек данных, переходящих на соседние блоки. На стороне декодера постфильтр 640 применяется после обратного блочного преобразования вдоль границ блоков. Также для общего случая $K \times N$ предварительный фильтр применяется к $(K-N)/2$ точкам данных каждого блока, соседствующего с границей блока.

Для обратимости предварительный фильтр 610 и постфильтр 640 являются инверсиями друг друга. Для реализации преобразования с перекрытием без потерь, тем не менее, этого условия недостаточно. Это дополнительно ограничивает предварительные фильтры и постфильтры 610, 640, которые также должны быть преобразованиями без потерь, помимо блочного (базового) преобразования 620, которое должно быть реализовано способом без потерь. DCT может быть реализовано способом без потерь, используя лестничные, решетчатые или основанные на лифтинге (преобразования, основывающемся на сокращении числа выборок за счет удаления коррелирующих значений при сокращении определенных скалярных характеристик) способы, помимо прочего. См., к примеру, A. A. M. L. Brackens и A. W. M. van den Enden, "New networks for perfect inversion and perfect reconstruction", IEEE J. Selected Areas Communications, том 10, номер 1, 1992 г.; и I. Daubechies и W. Sweldens, "Factoring wavelet transform into lifting steps", J. Fourier Anal. Appl., т.4, стр.247-269, 1998. Обратимое безразмерное двумерное преобразование также описано Srinivasan в Патентной заявке США, озаглавленной "Improved Reversible Transform For Lossy And Lossless 2-D Data Compression", поданной одновременно с данной заявкой и включенной в данный документ посредством ссылки. Также известны основанные на лифтинге обратимые аппроксимации к DCT в одном измерении. См., к примеру, J. Liang и T. D. Tran, "Fast multiplierless approximations of the DCT with the lifting scheme", IEEE Trans. Signal Processing, т.49, стр.3032-3044, декабрь 2001 г. Эффективная обратимость дополнительно требует, чтобы оба этапа, т.е. предварительная/последующая фильтрации и блочное преобразование, были с единичным детерминантом.

3. Оператор обратимого перекрытия

Оператор эффективного обратимого перекрытия для использования в качестве предварительного фильтра 610 (фиг.6) преобразования 600 с перекрытием без потерь, на котором основан кодер 400/декодер 500 (фиг.4 и 5), может быть реализован в качестве предварительного фильтра линейной фазы, который представлен в виде в структуры 700, показанной на фиг.7. Инверсия данного предварительного фильтра (т.е. постфильтр 640) также имеет такую же структуру, но с другими коэффициентами.

Данная структура 700 фильтра линейной фазы имеет несколько ортогональных компонентов, в том числе перекрестную схему 710 Адамара на своем входе и выходе. Внутренние стрелки в проиллюстрированной схеме 710 Адамара означают отрицание на данной схеме. Структура 700 дополнительно включает в себя ортогональные матрицы U_1 , U_2 , V_1 и V_2 . Эти компоненты могут быть реализованы способом без потерь посредством использования решетчатых/основанных на лифтинге способов.

Помимо этого, структура 700 имеет ненулевые масштабные коэффициенты $S_1 - S_M$. Ограничение по единичности детерминанта подразумевает, что $\prod_i s_i = \pm 1$. Когда все

масштабные коэффициенты равны ± 1 , предварительные фильтры/постфильтры могут быть реализованы в качестве преобразования без потерь, где матрицы компонентов U_1 , U_2 , V_1 и V_2 реализованы как этапы решетки/лифтинга без потерь.

Тем не менее, когда масштабные коэффициенты не равны все ± 1 , реализация без потерь остается сложной задачей, которая решается и обсуждается подробнее далее.

С помощью данной структуры 700 предварительного фильтра линейной фазы проблема реализации пары предварительного фильтра/постфильтра сокращается до

трех следующих этапов:

1. Разложение фильтра F в следующую форму для ортогональных матриц U_1 , U_2 , V_1 и V_2 :

$$F = \begin{pmatrix} 1 & J \\ J/2 & I/2 \end{pmatrix} \begin{pmatrix} U_1 & 0 \\ 0 & V_1 \end{pmatrix} \begin{pmatrix} s_1 & & & \\ & s_2 & & \\ & & \ddots & \\ & & & s_M \end{pmatrix} \begin{pmatrix} U_2 & 0 \\ 0 & V_2 \end{pmatrix} \begin{pmatrix} I/2 & J \\ J/2 & -1 \end{pmatrix} \quad (1)$$

где I - это единичная матрица,

$$J = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & \ddots & \\ 1 & 0 & 0 \end{pmatrix}$$

2. Получение реализаций без потерь для U_1 , U_2 , V_1 и V_2 ; и
3. Получение реализаций без потерь для матрицы масштабирования.

Что касается этапа 1, первая и последняя матрицы в правой части, которые задают двухточечные преобразования Адамара, включают коэффициент $1/2$ в некоторых членах, чтобы сделать эти стадии с единичным детерминантом. Прочее реорганизуется в блочную диагональную форму с помощью двух блоков, каждый из половины линейных измерений F . Разложение по сингулярным значениям (или SVD) каждого блока предоставляет ортогональные матрицы U_1 , U_2 , V_1 и V_2 , а также масштабы.

Реализации без потерь матриц компонентов могут быть получены на этапе 2 с помощью стандартных основанных на лифтинге методик, таких как описанные А. А. М. L. Bruekens и А. W. M. van den Enden, "New networks for perfect inversion and perfect reconstruction", IEEE J. Selected Areas Communications, том 10, номер 1, 1992 г.

Реализация без потерь матрицы масштабирования на этапе 3 разрешается следующим образом. Для простоты допустим, что мы имеем конкретный компонент с двумя входами и двумя выходами, который является (а) без потерь и (б) реализует масштабирование посредством s ($0 < s < 1$) для первого компонента и посредством $1/s$ для второго компонента (другие случаи могут быть получены посредством перемены знака одного или обоих выходных сигналов). Другими словами, мы имеем отношение входа-выхода, заданное как:

$$y = \begin{pmatrix} s & 0 \\ 0 & 1/s \end{pmatrix} x \quad (2)$$

Детерминант матрицы преобразования в (2) - это $s/s=1$. Данная матрица может быть реализована в процедуре 800 с четырьмя этапами лифтинга или процедуре 900 с пятью этапами лифтинга, показанных на фиг.8 и 9. Мы обычно аппроксимируем все этапы лифтинга в форму $y=(a \cdot x + r) \gg b$, где x - это вход, y - это выход, a , b и r - это целые числа, и r используется для контроля ошибок округления, чтобы получить уменьшенную на деление целочисленную реализацию. Преобразование, заданное уравнением (2), упоминается в данном документе как с единичным детерминантом преобразование масштабирования, сокращенно преобразование масштабирования.

Интересно, что преобразование масштабирования тесно связано с операцией

сдвига, как задано ниже:

$$y = \begin{pmatrix} a & b \\ b & a \end{pmatrix} x \quad (3)$$

5 При ограничении $a^2 - b^2 = 1$ ($a > 0, b \geq 0$) операция сдвига имеет единичный детерминант и может быть реализована с помощью трех этапов лифтинга:

$$10 \begin{pmatrix} a & b \\ b & a \end{pmatrix} = \begin{pmatrix} 1 & \sqrt{a-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ b & 1 \end{pmatrix} \begin{pmatrix} 1 & \sqrt{a-1} \\ 0 & 1 \end{pmatrix} \quad (4)$$

Следовательно,

$$15 \begin{bmatrix} a+b & 0 \\ 0 & a-b \end{bmatrix} = \begin{bmatrix} 1 & 1/2 \\ -1 & 1/2 \end{bmatrix} \begin{bmatrix} 1/2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a & b \\ b & a \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1/2 & -1/2 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & \frac{1}{2}(\sqrt{a-1}-1) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad (5)$$

При этом коэффициенты масштабирования $1/2$ и 2 в матрицах, размещенных слева и справа относительно матрицы сдвига, распространяются на этапы лифтинга со сдвигом, и последний этап лифтинга первой матрицы объединяется с первым этапом лифтинга со сдвигом, тогда как первый этап лифтинга последней матрицы объединяется с первым этапом лифтинга со сдвигом. Реализация в пять этапов в качестве процедуры 900 преобразования масштабирования, показанная на фиг.9, основана на уравнении (5). Упрощения в структуре могут быть возможны посредством отмены операций инверсии, при возможности, между тремя группами в уравнении, т.е. схемами Адамара, ортогональными матрицами и операциями масштабирования (которые, в свою очередь, могут быть разложены в операции Адамара и сдвига).

30 Более конкретно, эффективная матрица преобразования данной реализации с четырьмя этапами лифтинга масштабирования без потерь в качестве процедуры - это

$$T = \begin{pmatrix} c^{-2} & 0 \\ 0 & c^2 \end{pmatrix}$$

35 где $c^2 = 1 - s^2$. С другой стороны, эффективная матрица преобразования реализации с пятью этапами лифтинга в процедуре 900 - это

$$T = \begin{pmatrix} \frac{1+s}{c} & 0 \\ 0 & \frac{1-s}{c} \end{pmatrix}$$

40 где $c^2 = 1 - s^2$.

Хотя процедура 800 масштабирования, показанная на фиг.8, имеет меньше на один этап лифтинга, чем на фиг.9, вторая процедура 900 имеет только три этапа нетривиального лифтинга, в отличие от четырех в первой. По причинам, указанным в параграфе выше, первый или последний этап тривиального лифтинга на фиг.9 могут быть объединены с предшествующим или последующими этапами преобразования (например, со схемой 710 Адамара на любом конце фиг.7) при определенных условиях (например, когда $U1, U2$ и $V1$ - единичные).

50 Процедура масштабирования легко может быть распространена на более крупные матрицы. Это проиллюстрировано на фиг.10, где M , возможно, различных масштабных коэффициентов $s_1 - s_M$ применяются к M путем данных в качестве каскада 1000 преобразований масштабирования. Чтобы достичь этого обратимым

способом, в общем, требуется $M-1$ обратимых преобразований масштабирования.

Один полезный особый случай - это когда M масштабных коэффициентов $s_1 \dots s_M$ группируются в $M/2$ групп формы $(s, 1/s)$. В этом случае требуется только $M/2$ обратимых преобразований масштабирования. Один пример - это $s_1 = s_2 = \dots = s_{M/2} = s$, а $s_{M/2+1} = s_{M/2+2} = \dots = s_M = 1/s$. Предпочтительный способ группирования - сохранять симметрию вдоль центральной оси, другими словами, каждая группа масштабирует коэффициенты s_i и s_{M+1-i} . Если M нечетная, один не сгруппированный масштабный коэффициент - это 1, соответствующий пути данных вдоль оси.

На границах сигналов, где предварительные фильтры /постфильтры должны распространяться за пределы сигнала, одно решение - распространить сигнал симметрично и затем применить предварительные фильтры /постфильтры. Это, в общем, не операция без потерь из-за масштабирования. Другое решение - пропустить предварительную/последующую фильтрацию на границах. Нет заметной разницы между двумя решениями в показателях производительности R-D, как и в качестве восприятия (например, при использовании для сжатия изображений/видео с потерями).

Обращаясь теперь к фиг.11, оператор обратимого перекрытия, имеющий требуемое эффективное по R-D (т.е. с единичным детерминантом) свойство, затем реализуется в качестве структуры 700 предварительного фильтра линейной фазы (фиг.7), которая включает в себя обратимые схемы 710 Адамара с единичным детерминантом, обратимые ортогональные преобразования поворота 1110 (для матриц компонентов $U1, U2, V1$ и $V2$) и обратимое масштабирование 1120 с единичным детерминантом (к примеру, с помощью процедур 800, 900 с этапами лифтинга или каскада 1100). Постфильтр аналогичен предварительному фильтру и создан с помощью той же структуры, хотя и с этапами обратного лифтинга в обратном порядке. Это проиллюстрировано на фиг.7, где количество значений данных M в блоке, в общем, является любым натуральным числом. Хотя иллюстрация приведена для M с четными значениями, нечетные значения также допустимы посредством замечания того, что одноточечное преобразование Адамара центрального значения данных является самим собой. Эта процедура может быть обобщена до данных с большим числом измерений.

Обобщая, работа оператора обратимого перекрытия проиллюстрирована на фиг.12. На первом этапе 1210 входные двумерные цифровые мультимедийные данные мозаично разбиваются на блоки (как также показано для кодера 400 на фиг.4). Оператор обратимого перекрытия применяет схему 710 Адамара к соседним и элементам мозаики на этапе 1220. Затем оператор применяет обратимое преобразования поворота к суммам и разностям на этапе 1230, после чего следует обратимый оператор масштабирования на этапе 1240. Затем выполняется еще одно обратимое блочное преобразование поворота (этап 1250) и обратная инверсная схемы Адамара (этап 1260).

Ссылаясь теперь на фиг.13, матричные представления обратимых блочных преобразований поворота и операторов масштабирования зависят от требуемого преобразования с перекрытием оператора, используя, например, арифметику, описанную в уравнении (1). Фиг.13 показывает пример постфильтра, имеющего структуру 700, показанную на фиг.7 и 11, которая предваряется обратимым блочным преобразованием (в данном случае четырехточечным преобразованием Адамара). Функция преобразования постфильтра следующая:

$$T = \begin{pmatrix} 0,9885 & 0,1553 & -0,1553 & 0,2183 \\ -0,1499 & 0,9885 & 0,2183 & 0,1499 \\ 0,1502 & 0,2167 & 0,9884 & -0,1502 \\ 0,2167 & -0,1556 & 0,1556 & 0,9884 \end{pmatrix} \quad (6)$$

Низкочастотный компонент процедур Адамара формирует импульсную характеристику, показанную на графике по фиг.14.

4. Вычислительная среда

Вышеописанный кодек, основанный на преобразовании с перекрытием, использующем оператор обратимого преобразования, может быть применен в любом из множества устройств, в котором выполняется обработка цифровых мультимедийных сигналов, в том числе (помимо других примеров) на вычислительных машинах; на оборудовании для записи, передачи и приема изображений и видео; на портативных видеопроигрывателях; в устройствах видеоконференцсвязи и т.д. Методики кодирования цифрового мультимедиа могут быть реализованы в аппаратной схемотехнике, а также в программном обеспечении обработки цифрового мультимедиа, исполняемом на вычислительной машине или в другом вычислительном окружении, таком как показанное на фиг.15.

Фиг.15 иллюстрирует обобщенный пример подходящей вычислительной среды 1500, в которой могут быть реализованы описанные варианты осуществления. Вычислительная среда 1500 не предназначена, чтобы налагать какое-либо ограничение на область использования или функциональность изобретения, поскольку настоящее изобретение может быть реализовано в различных вычислительных средах общего или специального назначения.

Согласно фиг.15, вычислительная среда 1500 включает в себя, по меньшей мере, один модуль 1510 обработки и память 1520. На фиг.15 эта наиболее общая конфигурация 1530 показана пунктирной линией. Модуль 1510 обработки данных исполняет машиноисполняемые инструкции и может быть реальным или виртуальным процессором. В многопроцессорной системе несколько модулей обработки данных исполняют машиноисполняемые инструкции, чтобы повысить вычислительную мощность. Памятью 1520 может быть энергозависимая память (к примеру, регистры, кэш, ОЗУ), энергонезависимая память (к примеру, ПЗУ, ЭСППЗУ, флэш-память и т.д.) или какое-либо их сочетание. Память 1520 хранит программное обеспечение 1580, реализующее описанный кодер/декодер и преобразования.

Вычислительная среда может иметь дополнительные признаки. Например, вычислительная среда 1500 включает в себя хранилище 1540, устройства 1550 ввода, одно или более устройств 1560 вывода и одно или более коммуникационных соединений 1570. Механизм межкомпонентного соединения (не показан), такой как шина, контроллер или схема соединяет между собой компоненты вычислительной среды 1500. В типичном случае, программное обеспечение операционной системы (не показано) предоставляет рабочую среду для другого программного обеспечения, исполняемого в вычислительной среде 1500, и координирует действия компонентов вычислительной среды 1500.

Хранилище 1540 может быть съемным или стационарным и включает в себя магнитные диски, магнитные ленты или кассеты, CD-ROM, CD-RW, DVD или любой другой носитель, который может быть использован, чтобы сохранять информацию, и к которому можно осуществлять доступ в рамках вычислительной среды 1500. Хранилище 1540 сохраняет инструкции для программного обеспечения 1580,

реализующего кодек, основанный на преобразовании с перекрытием, использующем оператор обратимого перекрытия.

Устройствами 1550 ввода могут быть устройства сенсорного ввода, такие как клавиатура, мышь, перо или шаровой манипулятор, устройство голосового ввода, устройство сканирования или другое устройство, которое предоставляет ввод в вычислительную среду 1500. Для аудио устройствами 1550 ввода могут быть звуковая плата или аналогичное устройство, которое принимает ввод звука в аналоговой или цифровой форме, либо устройство считывания CD-ROM, которое предоставляет аудиовыборки вычислительной среде. Устройствами 1560 вывода может быть дисплей, принтер, динамик, пишущее устройство CD-RW или другое устройство, которое предоставляет вывод из вычислительного окружения 1500.

Коммуникационные соединения 1570 обеспечивают обмен данными по среде обмена данными с другой вычислительной объектной сущностью. Среда обмена данными передает информацию, такую как машиноисполняемые инструкции, сжатую аудио- и видеоинформацию или другие данные в модулированном информационном сигнале. Модулированный информационный сигнал - это сигнал, который имеет одну или более характеристик, установленных или изменяемых таким образом, чтобы кодировать информацию в этом сигнале. В качестве примера, а не ограничения, среда обмена данными включает в себя проводные или беспроводные методики, реализованные с помощью электрического, оптического, радиочастотного, инфракрасного, акустического или другого оборудования связи.

Методики обработки цифрового мультимедиа в данном документе могут быть описаны в общем контексте машиночитаемых носителей. Машиночитаемые носители - это любые доступные носители, к которым можно осуществлять доступ в вычислительной среде. В качестве примера, а не ограничения, в вычислительной среде 1500 машиночитаемые носители включают в себя память 1520, хранилище 1540, среды обмена данными и сочетания любого из вышеуказанного.

Методики обработки цифрового мультимедиа в данном документе могут быть описаны в общем контексте машиноисполняемых инструкций, таких как включенные в программные модули, исполняемые в вычислительной среде на целевом реальном или виртуальном процессоре. В общем, программные модули включают в себя процедуры, программы, библиотеки, объекты, классы, компоненты, структуры данных и т.д., которые выполняют конкретные задачи или реализуют определенные абстрактные типы данных. Функциональность программных модулей может быть сочетаема или разделена между программными модулями, как требуется в различных вариантах осуществления. Машиноисполняемые инструкции для программных модулей могут быть исполняемы в локальной или распределенной вычислительной среде.

Для целей представления подробное описание использует такие термины, как "определить", "сгенерировать", "настроить" и "применить", чтобы описывать операции вычислительной машины в вычислительной среде. Эти термины являются высокоуровневыми абстракциями для операций, выполняемых вычислительной машиной, и не должны путаться с действиями, выполняемыми человеком. Фактические операции вычислительной машины, соответствующие этим терминам, различаются в зависимости от реализации.

5. Вариации и расширения оператора обратимого перекрытия

Различные модификации и расширения вышеописанного оператора обратимого перекрытия могут быть выполнены. Хотя представленные выше описания касаются

одномерных данных, аналогичная процедура может быть применена, отдельно или не отдельно, к нескольким измерениям данных.

Ортогональные преобразования поворота в вышеописанной реализации оператора обратимого перекрытия могут быть заменены их аппроксимацией либо другими преобразованиями, которые могут быть не ортогональными.

Более того, хотя основное внимание вышеприведенного описания было уделено восстановлению входных данных без потерь, то же преобразование может быть использовано также для сжатия данных с потерями. В этом случае потери могут происходить либо в процессе квантования, либо вследствие ограниченной точности/аппроксимации реализации или предварительного фильтра, или постфильтра, либо вследствие других неточностей, либо при сочетании нескольких факторов.

Оператор обратимого перекрытия, описанный в данном документе, может быть применен к областям за пределами сжатия данных. Преобразование с перекрытием, использующее оператор обратимого перекрытия, может само по себе быть расширяющим.

Оператор обратимого перекрытия может быть применен, надлежащим образом, в модифицированной форме, чтобы реализовать многоскоростные банки фильтров с различными скоростями, вейвлеты, преобразования с перекрытием с основанием, охватывающим ширину более чем 2 блока ($K > 2N$).

Оператор обратимого перекрытия может быть применен в пространственно варьирующейся форме, в которой рамки действия и форма фильтра перекрытия может варьироваться вдоль пространственных рамок данных.

Принимая во внимание множество возможных вариантов осуществления, к которым могут быть применены принципы нашего изобретения, мы заявляем в изобретении все такие варианты осуществления, которые могут подпадать под объем, определяемый последующей формулой изобретения и ее эквивалентами.

Формула изобретения

1. Компьютерно-реализуемый способ декодирования данных изображения, разбитых на блоки в сжатом потоке битов, содержащий этапы, на которых извлекают блоки данных изображения из сжатого потока битов; применяют обратимое блочное преобразование к блокам; применяют обратимый постфильтр вдоль, по меньшей мере, некоторых из границ между блоками в целях снижения негативного эффекта, связанного с разбиением на блоки, причем обратимый пост-фильтр реализуется посредством преобразования с единичным детерминантом; и формируют изображение из данных изображения для представления на дисплее.

2. Способ по п.1, в котором обратимый пост-фильтр реализуется посредством множества компонентов, каждый из которых имеет единичный детерминант.

3. Способ по п.2, в котором при применении обратимого постфильтра: применяют обратимую схему Адамара; применяют обратимое блочное преобразование поворота; применяют обратимый оператор масштабирования; применяют еще одно обратимое блочное преобразование поворота; и применяют обратимую инверсную схему Адамара.

4. Способ по п.3, в котором обратимый пост-фильтр реализуется посредством операций, соответствующих четырем шагам лифтинга.

5. Способ по п.3, в котором обратимый пост-фильтр реализуется посредством операций, соответствующих пяти шагам лифтинга.

6. Способ по п.5, в котором при применении обратимого оператора масштабирования масштабируют M путей данных M масштабными коэффициентами s_1-s_M с помощью каскада двухточечного масштабирования.

5 7. Устройство кодирования и/или декодирования изображений, содержащее буфер хранения данных для хранения данных изображения, которые должны быть закодированы и/или декодированы; процессор, запрограммированный, чтобы разбивать данные изображения на блоки; применять обратимый фильтр вдоль, по меньшей мере, некоторых из границ между блоками в целях снижения негативного
10 эффекта, связанного с разбиением на блоки, причем обратимый фильтр реализован посредством преобразования с единичным детерминантом; и применять обратимое блочное преобразование к блокам, при этом применение обратимого фильтра и применение обратимого блочного преобразования инвертируются между
15 кодированием и декодированием данных изображения.

8. Устройство кодирования и/или декодирования изображений по п.7, в котором обратимый фильтр реализован посредством множества компонентных преобразований, каждое из которых является преобразованием с единичным детерминантом.

20 9. Устройство кодирования и/или декодирования изображений по п.8, в котором процессор реализует обратимый фильтр посредством применения обратимой схемы Адамара; применения обратимого блочного преобразования поворота; применения обратимого оператора масштабирования; применения еще одного обратимого блочного преобразования поворота; и применения обратимой инверсной схемы Адамара.
25

10. Устройство кодирования и/или декодирования изображений по п.9, в котором процессор реализует обратимый фильтр посредством операций, соответствующих четырем шагам лифтинга.

30 11. Устройство кодирования и/или декодирования изображений по п.9, в котором процессор реализует обратимый фильтр посредством операций, соответствующих пяти шагам лифтинга.

12. Устройство кодирования и/или декодирования изображений по п.11, в котором процессор реализует обратимый оператор масштабирования посредством
35 масштабирования M путей данных M масштабными коэффициентами s_1-s_M с помощью каскада двухточечного масштабирования.

13. Машиночитаемый носитель записи, несущий на себе машиноисполняемую программу обработки изображений, которая при ее исполнении компьютером
40 предписывает компьютеру осуществлять способ обработки данных изображения, разбитых на блоки, при этом способ содержит этапы, на которых применяют обратимый фильтр вдоль, по меньшей мере, некоторых из границ между блоками в целях снижения негативного эффекта, связанного с разбиением на блоки, причем обратимый фильтр реализуется посредством преобразования с единичным
45 детерминантом; и применяют обратимое блочное преобразование к блокам, при этом применение обратимого фильтра и применение обратимого блочного преобразования инвертируются между кодированием и декодированием данных изображения.

14. Машиночитаемый носитель записи по п.13, в котором обратимый фильтр
50 реализуется посредством множества компонентов, каждый из которых имеет единичный детерминант.

15. Машиночитаемый носитель записи по п.14, в котором при применении обратимого фильтра применяют обратимую схему Адамара; применяют обратимое

блочное преобразование поворота; применяют оператор обратимого масштабирования; применяют еще одно обратимое блочное преобразование поворота; и применяют обратимую инверсную схему Адамара.

5 16. Машиночитаемый носитель записи по п.15, в котором обратимый фильтр реализуется посредством операций, соответствующих четырем шагам лифтинга.

17. Машиночитаемый носитель записи по п.15, в котором обратимый фильтр реализуется посредством операций, соответствующих пяти шагам лифтинга.

10 18. Машиночитаемый носитель записи по п.17, в котором применение обратимого оператора масштабирования содержит масштабирование M путей данных M масштабными коэффициентами s_1-s_M с помощью каскада двухточечного масштабирования.

15

20

25

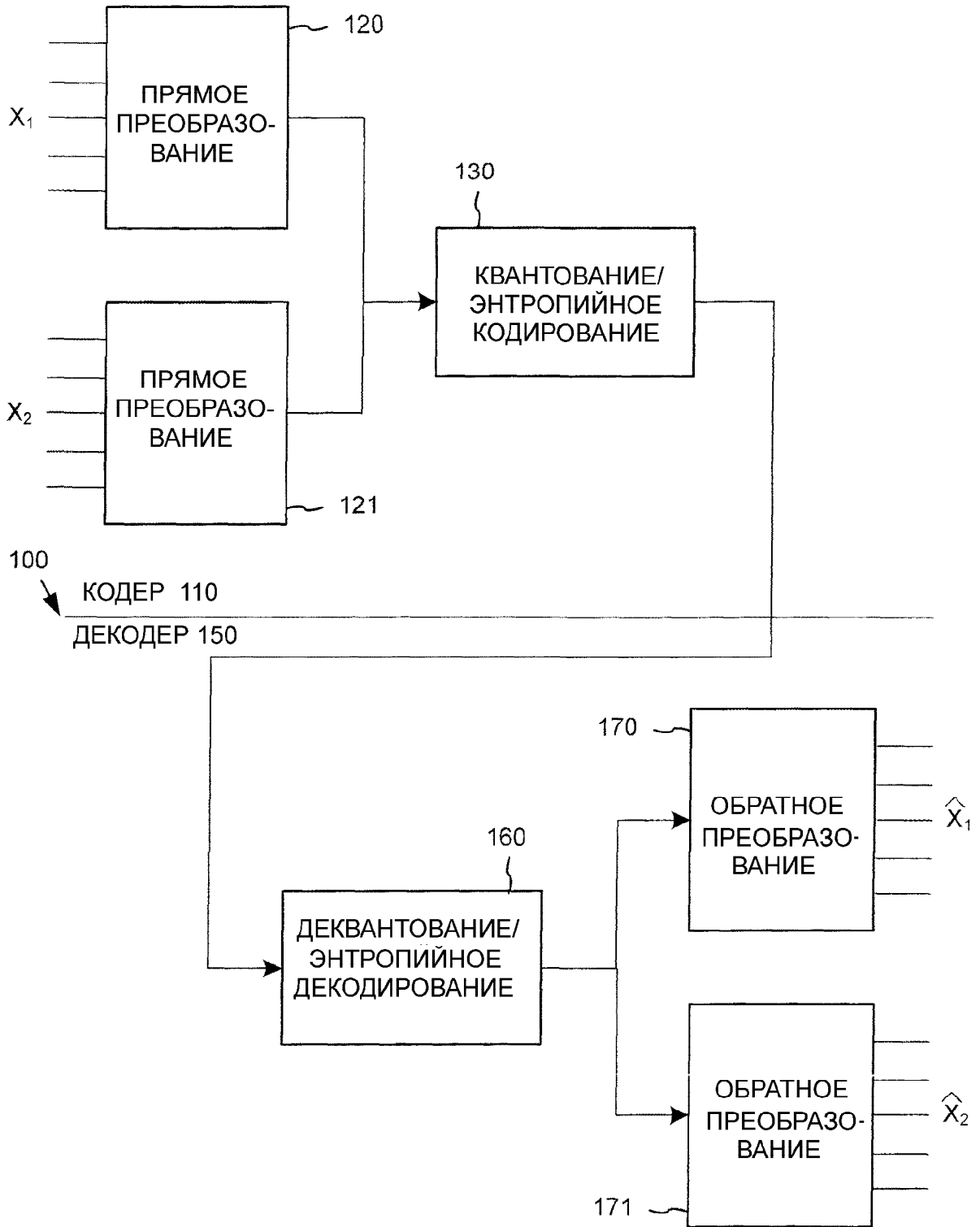
30

35

40

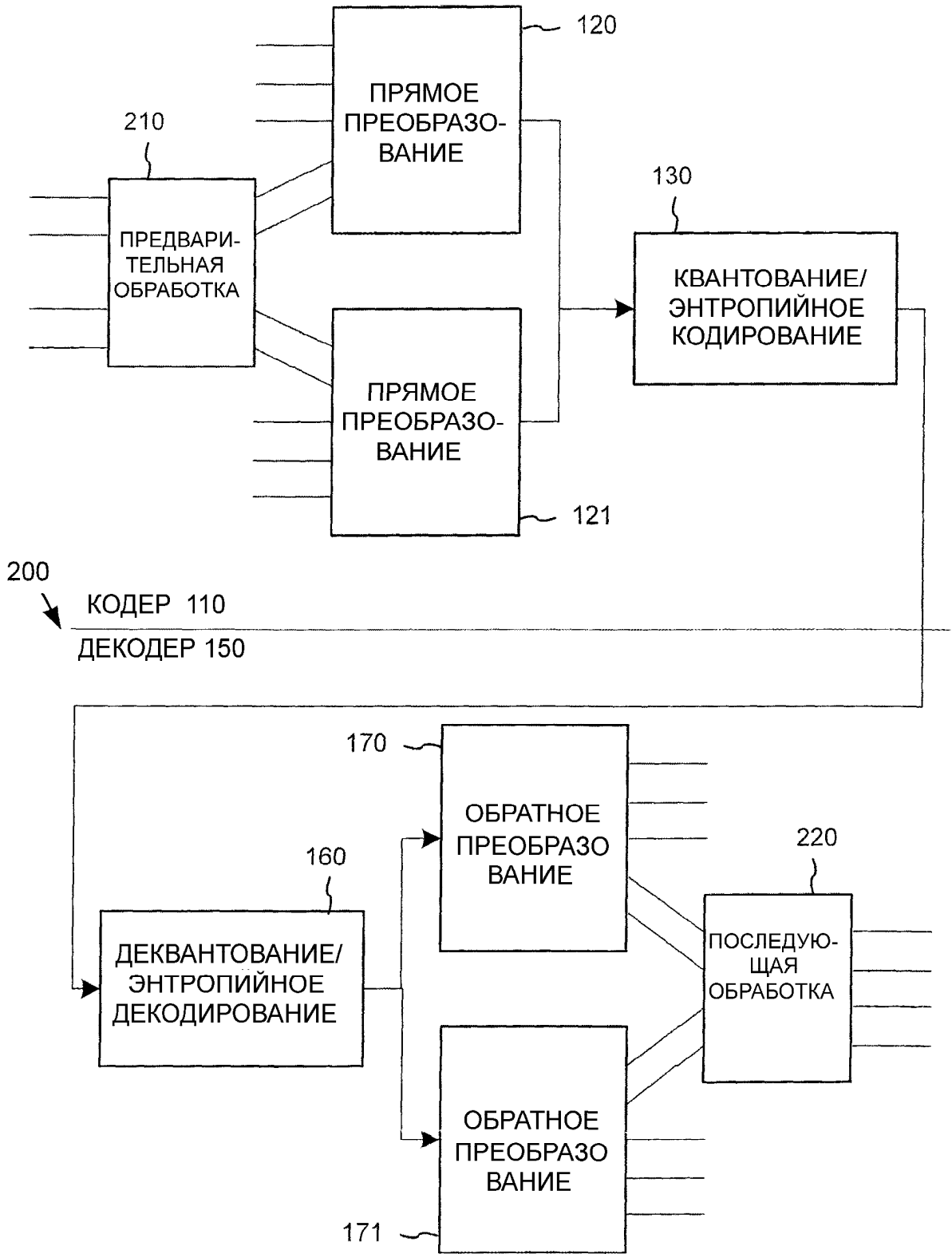
45

50



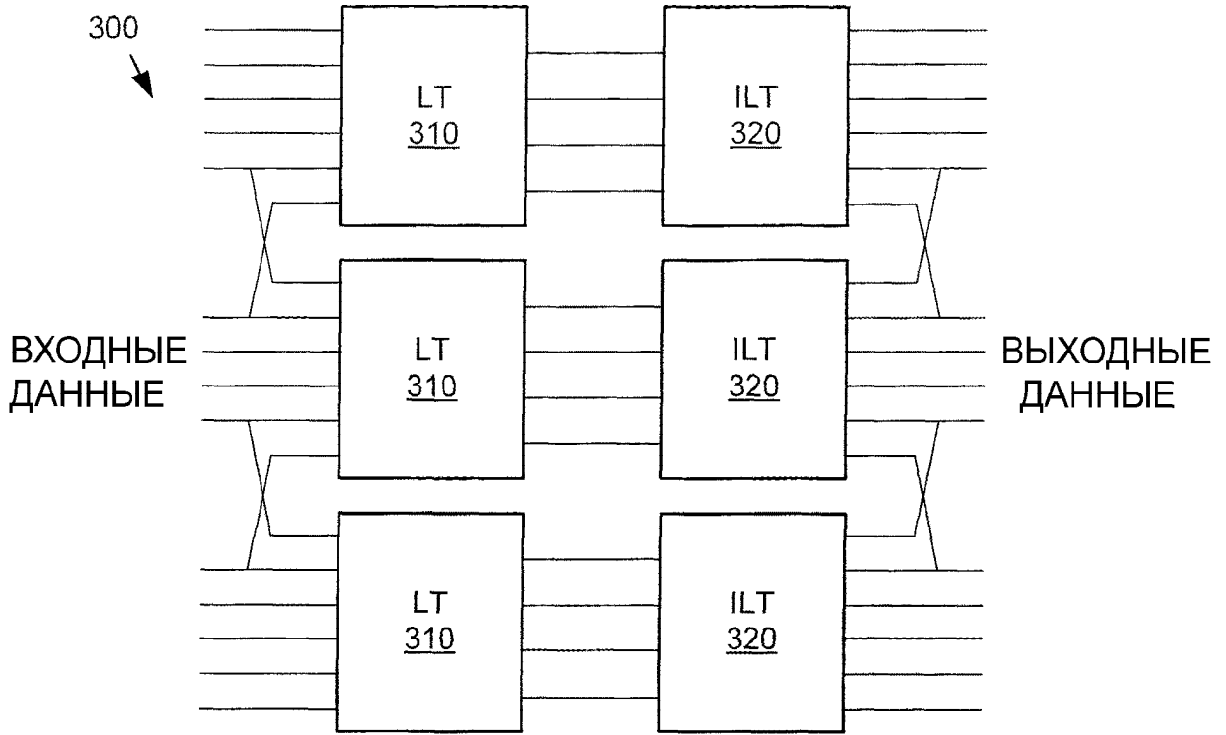
ФИГ.1

ПРЕДШЕСТВУЮЩИЙ УРОВЕНЬ ТЕХНИКИ

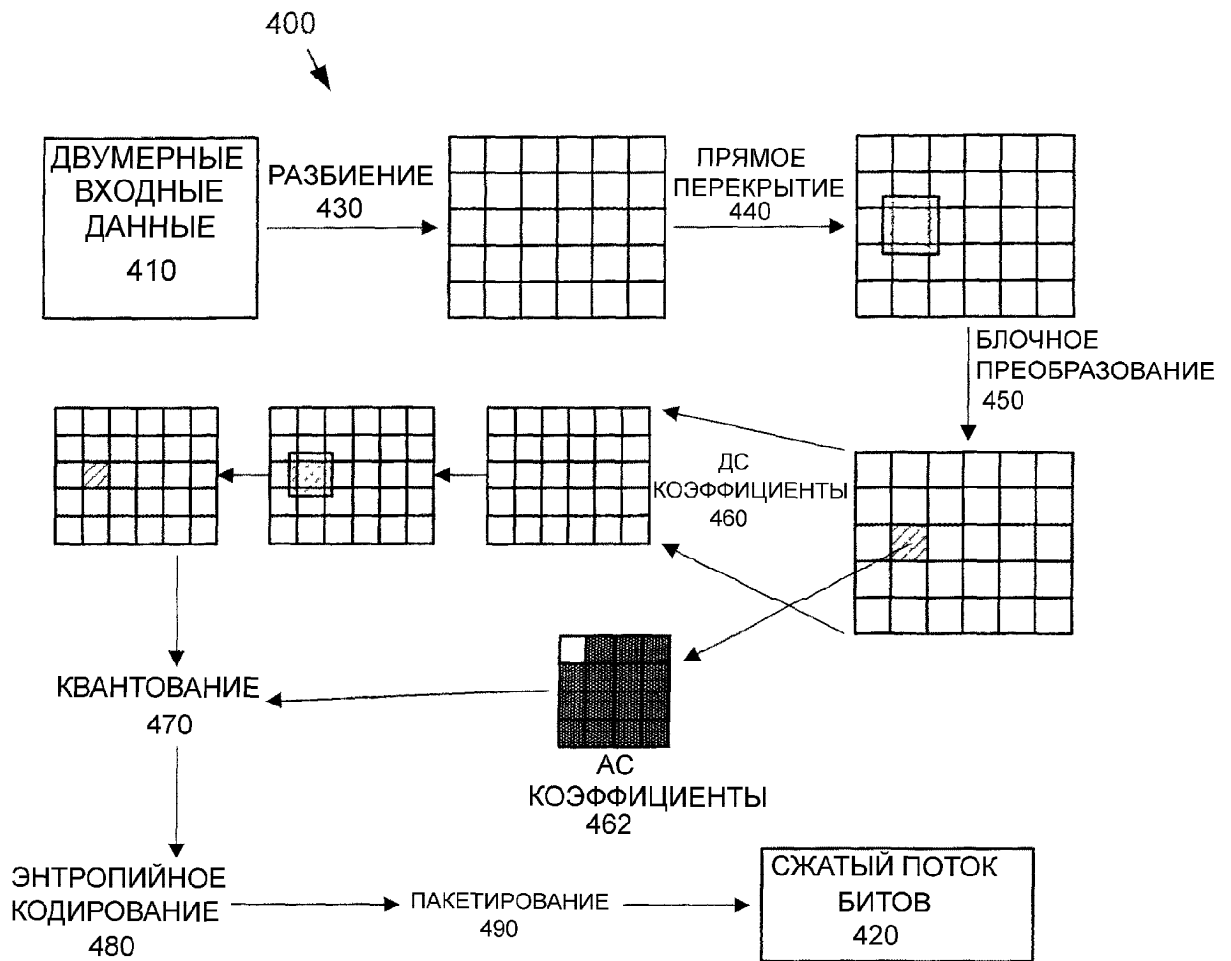


ФИГ.2

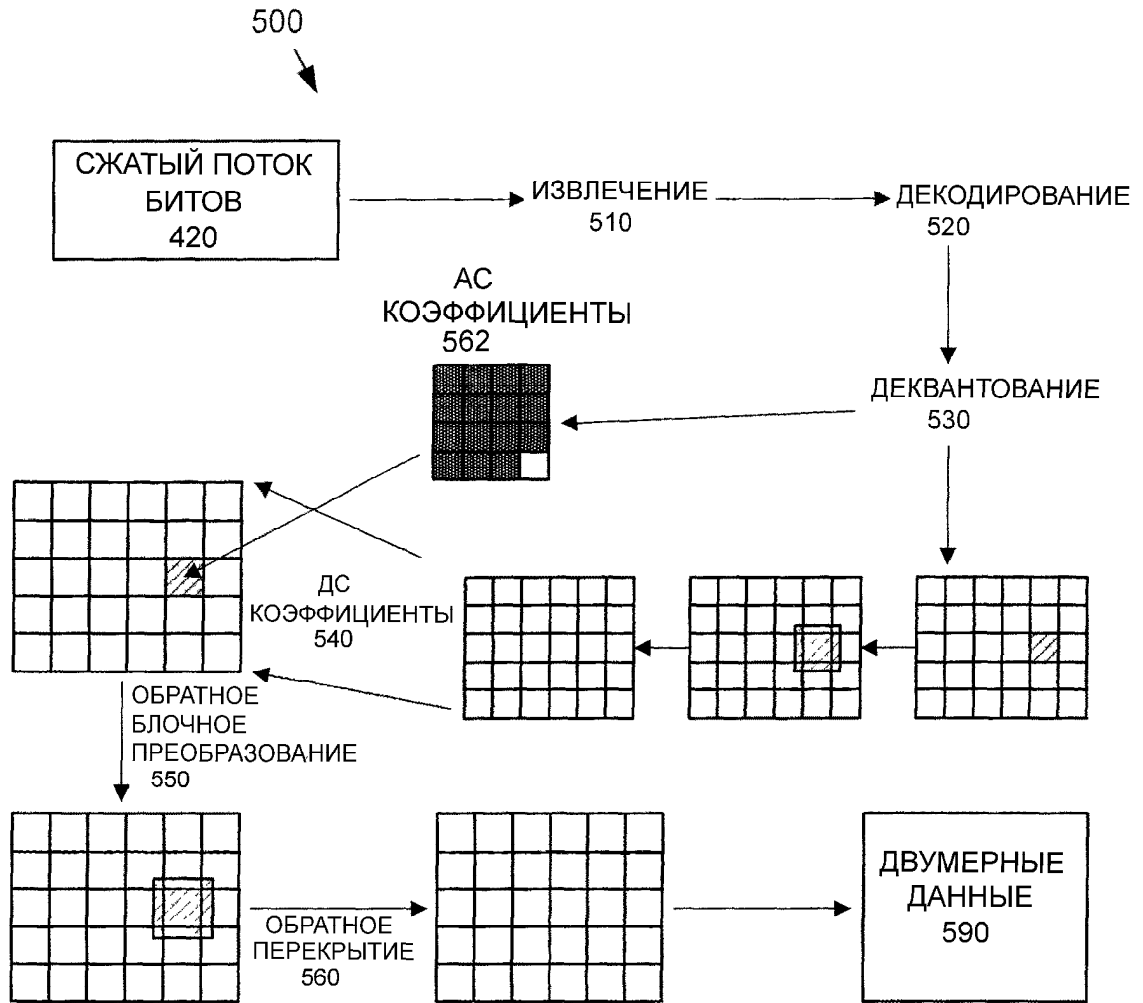
ПРЕДШЕСТВУЮЩИЙ УРОВЕНЬ ТЕХНИКИ



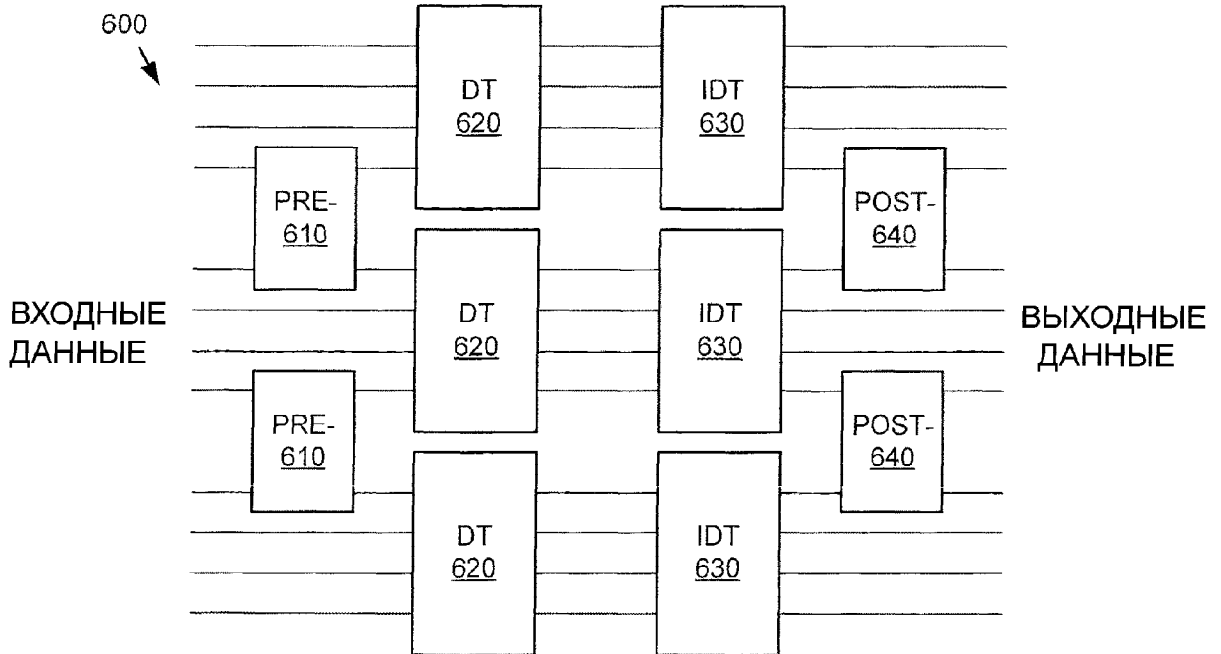
ФИГ.3



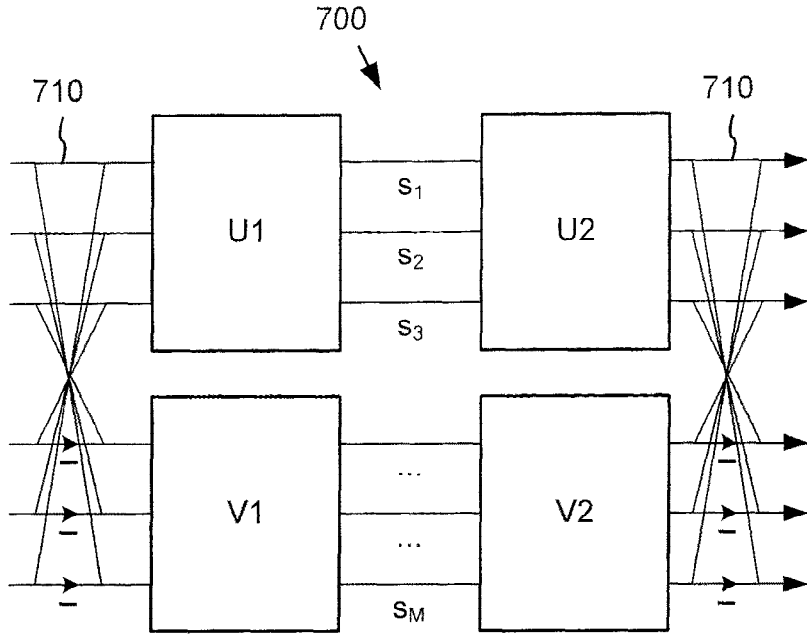
ФИГ.4



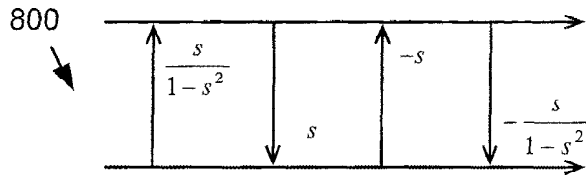
ФИГ.5



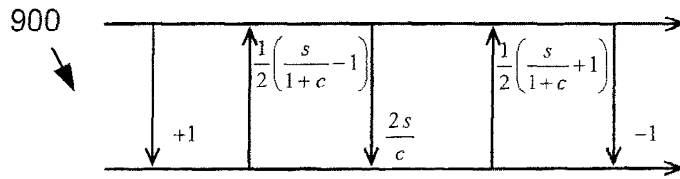
ФИГ.6



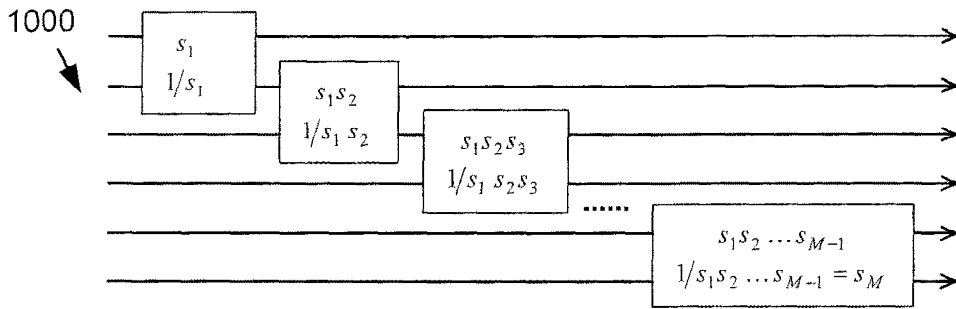
ФИГ.7



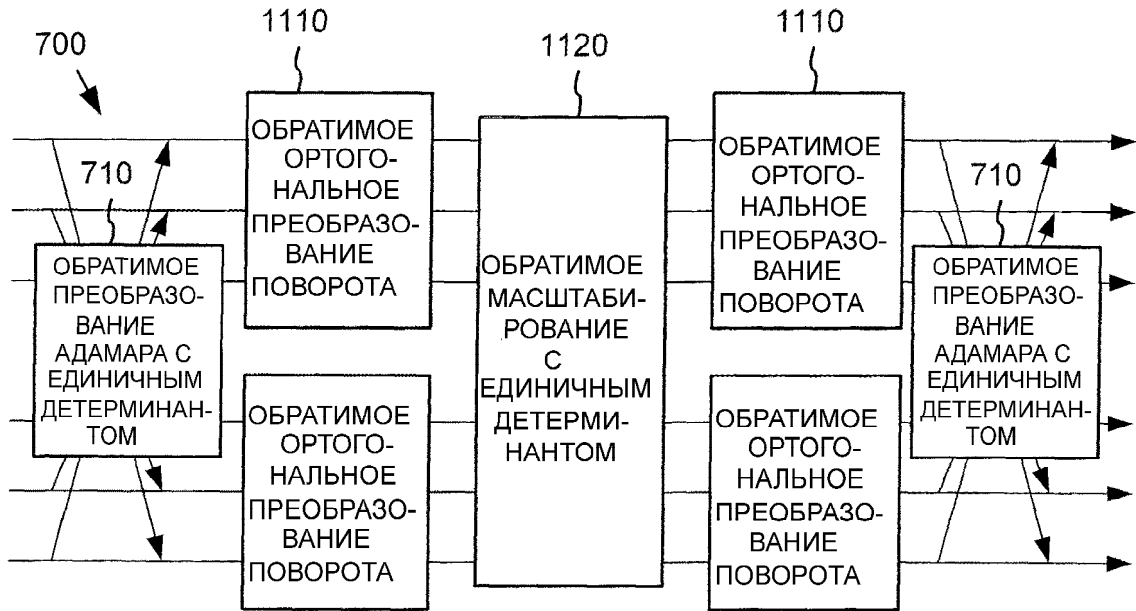
ФИГ.8



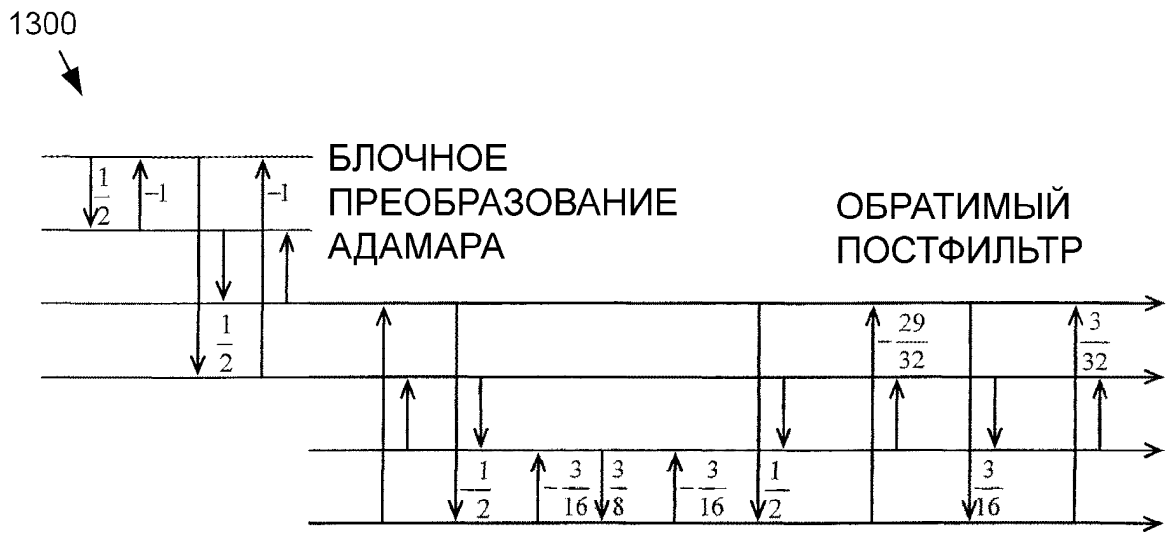
ФИГ.9



ФИГ.10

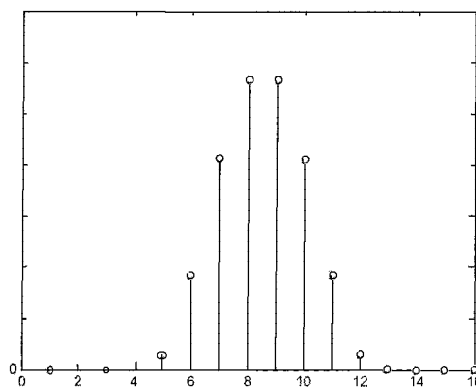


ФИГ.11

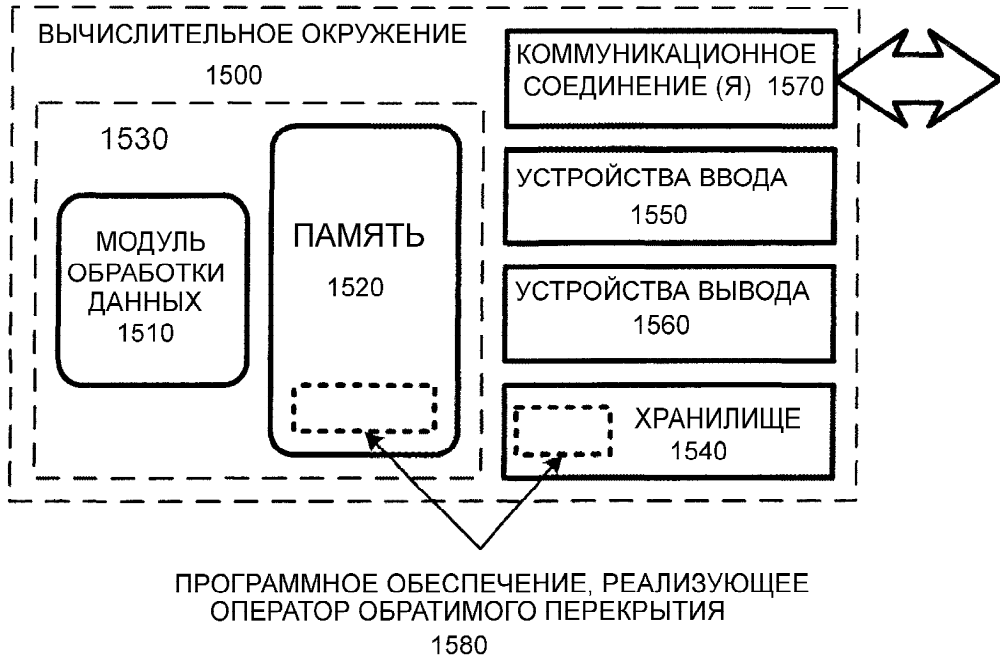


ФИГ.13

1400



ФИГ.14



ФИГ.15