



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: 2013143011/08, 22.02.2012

(24) Дата начала отсчета срока действия патента:
22.02.2012

Приоритет(ы):

(30) Конвенционный приоритет:

23.02.2011 US 61/445,967;

03.03.2011 US 61/448,771;

08.04.2011 US 61/473,713;

16.04.2011 US 61/476,260;

22.04.2011 US 61/478,287;

30.06.2011 US 61/503,426;

30.06.2011 US 61/503,434;

30.06.2011 US 61/503,440;

25.08.2011 US 61/527,463;

06.09.2011 US 61/531,571;

(см. прод.)

(43) Дата публикации заявки: 27.03.2015 Бюл. № 9

(45) Опубликовано: 10.04.2016 Бюл. № 10

(56) Список документов, цитированных в отчете о поиске: WO 2008/084378 A2, 17.07.2008. US 5844613 A, 01.12.1998. US 2007/0165282 A1, 19.07.2007. US 2003/0026495 A1, 06.02.2003. US 2004/0008787 A1, 15.01.2004. WO 2009/087095 A1, 16.07.2009. RU 2387094 C1, 20.04.2010. RU 2377737 C2, 27.12.2009.

(85) Дата начала рассмотрения заявки РСТ на национальной фазе: 23.09.2013

(86) Заявка РСТ:
US 2012/026166 (22.02.2012)(87) Публикация заявки РСТ:
WO 2012/116095 (30.08.2012)

Адрес для переписки:

129090, Москва, ул. Б. Спасская, 25, строение 3,
ООО "Юридическая фирма Городисский и
Партнеры"

(72) Автор(ы):

ЧОНГ Ин Сук (US),

КАРЧЕВИЧ Марта (US)

(73) Патентообладатель(и):

КВЭЛКОММ ИНКОРПОРЕЙТЕД (US)

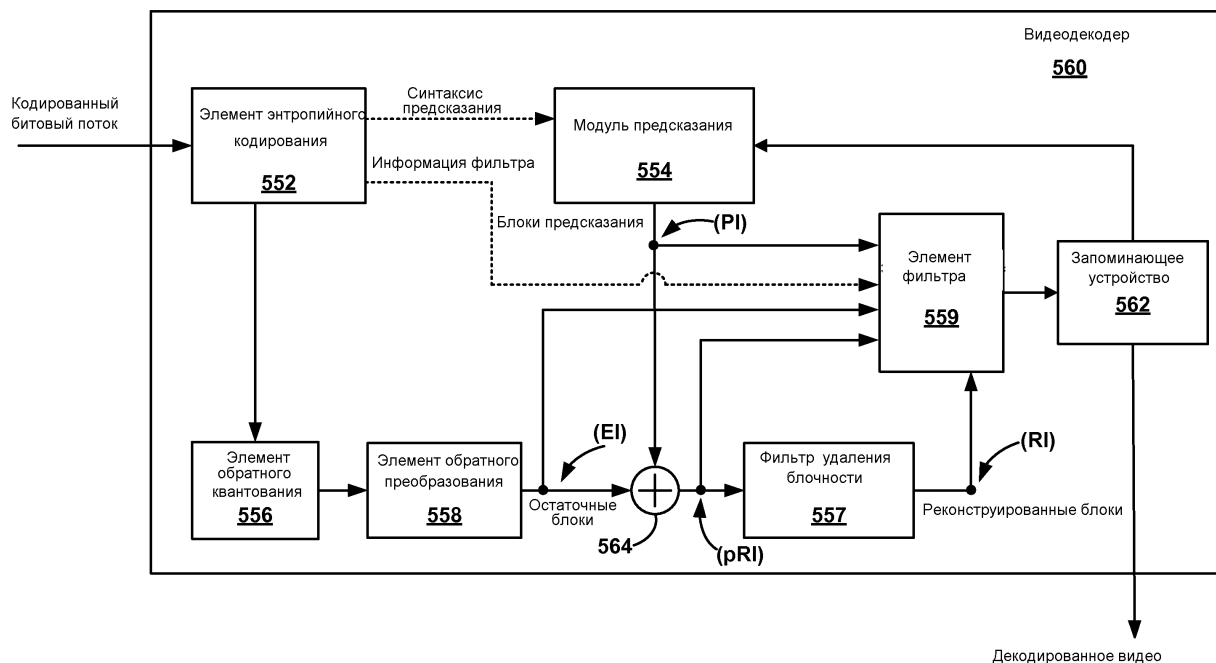
(54) МУЛЬТИМЕТРИЧЕСКАЯ ФИЛЬТРАЦИЯ

(57) Реферат:

Изобретение относится к технологиям кодирования/декодирования видеоданных.

Техническим результатом является сокращение количества битов, необходимых для сигнализации коэффициентов фильтра. Предложен способ декодирования данных видео. Способ включает в себя этап, на котором определяют направление, по меньшей мере, одного пикселя из блока пикселей. Далее, согласно способу, определяют метрику для упомянутого, по меньшей мере, одного пикселя из упомянутого блока пикселей, при этом метрика определяется на основе

сравнения пиксельных значений для поднабора пикселей в этом блоке с упомянутым, по меньшей мере, одним пикселем. А также осуществляют определение, на основе упомянутой метрики и упомянутого направления, фильтра из набора фильтров, сигнализированного в данных видео, для упомянутого по меньшей мере одного пикселя из упомянутого блока пикселей. 4 н. и 36 з.п. ф-лы, 2 табл., 19 ил.



ФИГ.5

(30) (продолжение):

21.02.2012 13/401,685 US;
 23.02.2011 61/445,967 US;
 03.03.2011 61/448,771 US;
 08.04.2011 61/473,713 US;
 16.04.2011 61/476,260 US;
 22.04.2011 61/478,287 US;
 30.06.2011 61/503,426 US;
 30.06.2011 61/503,434 US;
 30.06.2011 61/503,440 US;
 25.08.2011 61/527,463 US;
 06.09.2011 61/531,571 US;
 21.02.2012 13/401,685 US



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(51) Int. Cl.

H04N 19/102 (2014.01)*H04N 19/117* (2014.01)*H04N 19/14* (2014.01)*H04N 19/80* (2014.01)(12) **ABSTRACT OF INVENTION**(21)(22) Application: **2013143011/08, 22.02.2012**(24) Effective date for property rights:
22.02.2012

Priority:

(30) Convention priority:

23.02.2011 US 61/445,967;**03.03.2011 US 61/448,771;****08.04.2011 US 61/473,713;****16.04.2011 US 61/476,260;****22.04.2011 US 61/478,287;****30.06.2011 US 61/503,426;****30.06.2011 US 61/503,434;****30.06.2011 US 61/503,440;****25.08.2011 US 61/527,463;****06.09.2011 US 61/531,571;**

(to be continued)

(43) Application published: **27.03.2015 Bull. № 9**(45) Date of publication: **10.04.2016 Bull. № 10**(85) Commencement of national phase: **23.09.2013**(86) PCT application:
US 2012/026166 (22.02.2012)(87) PCT publication:
WO 2012/116095 (30.08.2012)

Mail address:

**129090, Moskva, ul. B. Spasskaja, 25, stroenie 3,
OOO "Juridicheskaja firma Gorodisskij i Partnery"**

(72) Inventor(s):

CHONG In Suk (US),**KARCHEVICH Marta (US)**

(73) Proprietor(s):

KVELKOMM INKORPOREJTED (US)(54) **MULTIMETRIC FILTRATION**

(57) Abstract:

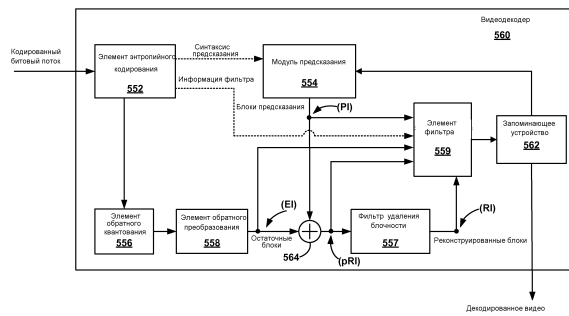
FIELD: electronics.

SUBSTANCE: disclosed is a method of decoding video data. Method includes a step of determining direction of at least one pixel from a block of pixels. Further, method comprises determining first metric for said at least one pixel from said block of pixels, wherein said metric is determined based on comparison of pixel values for a subset of the pixels in said block with said at least one pixel. Method also includes determining, based on said metric and said direction, a filter from a

set of filters, signalled in video data for said at least one pixel from said block of pixels.

EFFECT: reducing number of bits required for signalling of filter coefficients.

40 cl, 2 tbl, 19 dwg



ФИГ.5

(30) Convention priority:

21.02.2012 13/401,685 US;
 23.02.2011 61/445,967 US;
 03.03.2011 61/448,771 US;
 08.04.2011 61/473,713 US;
 16.04.2011 61/476,260 US;
 22.04.2011 61/478,287 US;
 30.06.2011 61/503,426 US;
 30.06.2011 61/503,434 US;
 30.06.2011 61/503,440 US;
 25.08.2011 61/527,463 US;
 06.09.2011 61/531,571 US;
 21.02.2012 13/401,685 US

R U 2 5 7 9 6 8 8 C 2

R U 2 5 7 9 6 8 8 C 2

[0001] По настоящей заявке испрашивается приоритет:
предварительной заявки на патент США № 61/445,967, поданной 23 февраля 2011 года;

предварительной заявки на патент США № 61/448,771, поданной 3 марта 2011 года;
5 предварительной заявки на патент США № 61/473,713, поданной 8 апреля 2011 года;
предварительной заявки на патент США № 61/476,260, поданной 16 апреля 2011 года;

предварительной заявки на патент США № 61/478,287, поданной 22 апреля 2011 года;

10 предварительной заявки на патент США № 61/503,426, поданной 30 июня 2011 года;
предварительной заявки на патент США № 61/503,434, поданной 30 июня 2011 года;
предварительной заявки на патент США № 61/503,440, поданной 30 июня 2011 года;
предварительной заявки на патент США № 61/527,463, поданной 25 августа 2011 года;

15 предварительной заявки на патент США № 61/531,571, поданной 6 сентября 2011 года;

полное содержание которых включено сюда в полном объеме посредством ссылки.

ОБЛАСТЬ ТЕХНИКИ, К КОТОРОЙ ОТНОСИТСЯ ИЗОБРЕТЕНИЕ

[0002] Данное раскрытие относится к цифровому кодированию видео на основе
20 блоков, используемому для сжатия видеоданных, в частности - к методикам фильтрации видеоблоков.

УРОВЕНЬ ТЕХНИКИ

[0003] Возможности цифрового видео могут внедряться для широкого диапазона устройств, включая цифровое телевидение, системы цифрового прямого вещания,
25 устройства беспроводной связи, такие как радиотелефонные гарнитуры, беспроводные системы вещания, персональные цифровые помощники (PDA), портативные компьютеры, стационарные компьютеры, планшетные компьютеры, цифровые камеры, цифровые записывающие устройства, устройства для видеоигр, игровые приставки, и т.п. Цифровые видеоустройства реализуют методики сжатия видео, такие как MPEG-2,
30 MPEG-4, ITU-T H.264/MPEG-4, Часть 10, усовершенствованное кодирование видео (AVC), чтобы более эффективно передавать и принимать цифровое видео. Методики сжатия видео осуществляют пространственное и временное предсказание, чтобы уменьшить избыточность, присущую видеопоследовательностям. Новые видеостандарты, такие как стандарт высокоэффективного кодирования видео (HEVC),
35 разрабатываемый «Совместной объединенной командой кодирования видео» (JCTVC), что является совместной работой MPEG и ITU-T, продолжают появляться и развиваться. Этот новый стандарт HEVC также иногда именуется H.265.

[0004] Методики сжатия видео на основе блоков могут осуществлять пространственное предсказание и/или временное предсказание. Внутреннее (intra-)
40 кодирование опирается на пространственное предсказание, чтобы уменьшить или устранить пространственную избыточность между видеоблоками внутри заданного элемента кодированного видео, который может содержать видеокادر, вырезку видеокадра и т.п. В отличие от этого, внешнее (inter-) кодирование опирается на временное предсказание, чтобы уменьшить или устранить временную избыточность
45 между видеоблоками последовательных элементов кодирования видеопоследовательности. Для внутреннего кодирования видеокодер осуществляет пространственное предсказание, чтобы сжать данные на основе других данных в пределах одного и того же элемента кодированного видео. Для внешнего кодирования

видеокодер осуществляет оценку движения и компенсацию движения, чтобы отследить движение соответствующих видеоблоков двух или более соседних элементов кодированного видео.

[0005] Кодированный видеоблок может быть представлен информацией предсказания, которая может использоваться для создания или идентификации блока предсказания и остаточного блока данных, показывающего различия между блоком, подвергаемым кодированию, и предсказанным блоком. В случае внешнего кодирования один или более векторов движения используются для идентификации блока предсказания данных из предыдущего или последующего элемента кодирования, тогда как в случае внутреннего кодирования режим предсказания может использоваться для генерирования блока предсказания на основе данных в пределах СУ, связанного с кодируемым видеоблоком. И внутреннее кодирование и внешнее кодирование могут определять несколько различных режимов предсказания, которые могут определять различные размеры блоков и/или методики предсказания, используемые при кодировании. Дополнительные типы синтаксических компонентов также могут быть включены как часть кодированных видеоданных с целью управлять методиками кодирования или параметрами, используемыми в процессе кодирования, или определять их.

[0006] После кодирования с предсказанием на основе блоков видеокодер может осуществлять преобразование, квантование и энтропийное кодирование, чтобы дополнительно уменьшить скорость передачи битов, относящуюся к сообщению остаточного блока. Методики преобразования могут содержать дискретные косинусные преобразования (DCT) или концептуально сходные процессы, такие как вейвлетные преобразования, целочисленные преобразования или другие типы преобразований. В процессе дискретного косинусного преобразования, в качестве примера, процесс преобразования преобразует набор значений разности пикселей в коэффициенты преобразования, которые могут представлять энергию значений пикселей в частотной области. Квантование применяется в отношении коэффициентов преобразования и в общем включает в себя процесс, который ограничивает количество битов, связанных с любым заданным коэффициентом преобразования. Энтропийное кодирование содержит один или более процессов, которые совместно сжимают последовательность квантованных коэффициентов преобразования.

[0007] Фильтрация видеоблоков может выполняться как часть циклов кодирования и декодирования или как часть постфильтрационного процесса в отношении реконструированных видеоблоков. Фильтрация обычно используется, например, чтобы уменьшить блочность или другие артефакты, распространенные при кодировании видео на основе блоков. Коэффициенты фильтра (иногда именуемые отводами фильтра) могут определяться или выбираться с целью поддерживать желаемые уровни фильтрации видеоблока, которые могут уменьшить блочность и/или улучшить качество видеоизображения другими способами. Набор коэффициентов фильтра, например, может определять, как фильтрация выполняется по границам видеоблоков или в других точках в пределах видеоблоков. Различные коэффициенты фильтра могут вызывать достижение разных уровней фильтрации по отношению к различным пикселям видеоблоков. Фильтрация, например, может сглаживать или заострять различия в интенсивности значений соседних пикселей, чтобы помочь избежать нежелательных артефактов.

СУЩНОСТЬ ИЗОБРЕТЕНИЯ

[0008] Данное раскрытие описывает методики, связанные с фильтрацией видеоданных в процессе кодирования и/или декодирования видео. В соответствии с этим раскрытием

фильтрация осуществляется в кодере, и информация фильтрации кодируется в битовом потоке, чтобы позволить декодеру идентифицировать фильтрацию, которая выполнялась в кодере. Декодер принимает кодированные видеоданные, которые включают в себя информацию фильтрации, декодирует видеоданные и осуществляет фильтрацию на основе информации фильтрации. Таким образом, декодер осуществляет ту же фильтрацию, которая применялась в кодере. В соответствии с методиками данного раскрытия на покадровой, на основе вырезка за вырезкой или на основе LCU за LCU кодер может выбирать один или более наборов фильтров и на основе кодированный элемент за кодированным элементом кодер может определять, осуществлять ли фильтрацию. Для кодированных элементов (CU), которые подлежат фильтрации, кодер может осуществлять фильтрацию на попиксельной основе или на основе группа за группой, где группа может, например, быть блоком пикселей 2x2 или блоком пикселей 4x4.

[0009] В одном примере способ кодирования видео включает в себя определение первой метрики для блока пикселей, при этом первая метрика определяется на основе сравнения поднабора пикселей в блоке с другими пикселями в блоке; на основе первой метрики определение фильтра для блока пикселей; и генерирование отфильтрованного изображения путем применения фильтра в отношении блока пикселей.

[0010] В другом примере устройство кодирования видео включает в себя элемент фильтра, выполненный с возможностью определять первую метрику для блока пикселей, при этом первая метрика определяется на основе сравнения поднабора пикселей в блоке с другими пикселями в блоке, определять фильтр для блока пикселей на основе первой метрики и генерировать отфильтрованное изображение путем применения фильтра в отношении блока пикселей; и запоминающее устройство, выполненное с возможностью хранить отфильтрованный результат элемент фильтра.

[0011] В другом примере аппарат включает в себя средство для определения первой метрики для блока пикселей, при этом первая метрика определяется на основе сравнения поднабора пикселей в блоке с другими пикселями в блоке, средство для определения фильтра для блока пикселей на основе первой метрики и средство для генерирования отфильтрованного изображения путем применения фильтра в отношении блока пикселей.

[0012] В другом примере машиночитаемый носитель данных хранит команды, которые при выполнении инициируют определение одним или более процессорами первой метрики для блока пикселей, при этом первая метрика определяется на основе сравнения поднабора пикселей в блоке с другими пикселями в блоке, определение фильтра для блока пикселей на основе первой метрики и генерирование отфильтрованного изображения путем применения фильтра в отношении блока пикселей.

[0013] Детали одного или более примеров изложены ниже в прилагаемых чертежах и описании. Другие признаки, цели и преимущества будут понятны из описания и чертежей, а также из формулы изобретения.

КРАТКОЕ ОПИСАНИЕ ЧЕРТЕЖЕЙ

[0014] Фиг. 1 представляет собой блок-схему, иллюстрирующую примерную систему кодирования и декодирования видео.

[0015] Фиг. 2А и фиг. 2В представляют собой концептуальные схемы, иллюстрирующие пример разделения квадродерева, применяемого в отношении самого большого элемента кодирования (LCU).

[0016] Фиг. 2С и фиг. 2D представляют собой концептуальные схемы,

иллюстрирующие пример схемы фильтра для серии видеоблоков, соответствующей примерному разделению квадродерева с фиг. 2А и фиг. 2В.

[0017] Фиг. 3 представляет собой блок-схему, иллюстрирующую примерный видеокодер, совместимый с данным раскрытием.

5 [0018] Фиг. 4А представляет собой концептуальную схему, иллюстрирующую приведение в соответствие диапазонов для двух метрик и фильтров.

[0019] Фиг. 4В представляет собой концептуальную схему, иллюстрирующую приведение в соответствие диапазонов для метрики активности и метрики направления для фильтров.

10 [0020] Фиг. 5 представляет собой блок-схему, иллюстрирующую примерный видеодекoder в соответствии с данным раскрытием.

[0021] Фиг. 6А, фиг. 6В и фиг. 6С демонстрируют концептуальные схемы блока пикселей 4x4.

15 [0022] Фиг. 7 представляет собой блок-схему последовательности операций, иллюстрирующую методики кодирования в соответствии с данным раскрытием.

[0023] Фиг. 8А и фиг. 8В представляют собой блок-схемы последовательности операций, иллюстрирующие методики кодирования в соответствии с данным раскрытием.

20 [0024] Фиг. 9А и фиг. 9В представляют собой блок-схемы последовательности операций, иллюстрирующие методики кодирования в соответствии с данным раскрытием.

[0025] Фиг. 10 представляет собой блок-схему последовательности операций, иллюстрирующую методики кодирования в соответствии с данным раскрытием.

25 [0026] Фиг. 11 представляет собой блок-схему последовательности операций, иллюстрирующую методики кодирования в соответствии с данным раскрытием.

ПОДРОБНОЕ ОПИСАНИЕ

[0027] Данное раскрытие описывает методики, связанные с фильтрацией видеоданных в процессе кодирования и/или декодирования видео. В соответствии с этим раскрытием фильтрация осуществляется в кодере, и информация фильтрации кодируется в битовом потоке, чтобы позволить декодеру идентифицировать фильтрацию, которая выполнялась в кодере. Декодер принимает кодированные видеоданные, которые включают в себя информацию фильтрации, декодирует видеоданные и осуществляет фильтрацию на основе информации фильтрации. Таким образом, декодер осуществляет ту же фильтрацию, которая применялась в кодере. В соответствии с методиками данного раскрытия на покaдровой основе, на основе вырезка за вырезкой или на основе LCU за LCU кодер может выбирать один или более наборов фильтров и на основе кодированный элемент за кодированным элементом кодер может определять, осуществлять ли фильтрацию. Для кодированных элементов (CU), которые подлежат фильтрации, кодер может осуществлять фильтрацию на попиксельной или погрупповой основе, где группа может, например быть блоком пикселей 2x2 или блоком пикселей 4x4.

45 [0028] В соответствии с методиками этого раскрытия видеоданные могут кодироваться в элементах, именуемых кодированными элементами (CU). CU могут разделяться на меньшие CU или подэлементы с использованием схемы деления квадродерева. Синтаксис, идентифицирующий схему деления квадродерева для конкретного CU, может передаваться от кодера к декодеру. Множественные входы, связанные с каждым подэлементом заданного CU, могут фильтроваться во время процесса декодирования и реконструкции кодированных видеоданных. В соответствии

с методиками этого раскрытия синтаксис описания фильтра может описывать набор фильтров, как то - сколько фильтров в наборе или какую форму принимают фильтры. Дополнительный синтаксис в битовом потоке, принимаемом декодером, может идентифицировать фильтры (то есть коэффициенты фильтров), используемые в кодере для конкретного подэлемента. Фильтр, используемый для конкретного ввода, может быть выбран на основе двух или более метрик, где некоторые комбинации значений для двух или более метрик индексируются для конкретных фильтров в пределах набора фильтров. В других примерах две или более метрик могут объединяться для формирования единственной метрики. Соответствие метрик фильтрам может также сигнализироваться в битовом потоке.

[0029] Различные типы фильтрации могут применяться в отношении пикселей или блоков пикселей на основе двух или более метрик, определенных для видеоданных. Фильтр, используемый для конкретного пикселя, может быть выбран на основе двух или более метрик, таких как некоторая комбинация метрики активности и метрики направления. Метрика активности, например, может определять количественно активность, связанную с одним или более блоками пикселей в видеоданных. Метрика активности может содержать метрику дисперсии, показывающую дисперсию пикселя в пределах набора пикселей. Метрика активности может быть как специфичной для направления, так и не специфичной для направления. Например, метрика активности, не специфичная для направления, может включать в себя значение Лапласиана с измененной суммой, как объясняется ниже более подробно.

[0030] Примеры метрик активности, специфичных для направления, включают в себя метрику горизонтальной активности, метрику вертикальной активности, метрику 45-градусной активности и метрику 135-градусной активности. Метрика направления может определять количественно для блока пикселей любую из горизонтальной активности, вертикальной активности или диагональной активности пикселя или группы пикселей или, метрика направления может включать в себя сравнение горизонтальной активности, вертикальной активности и/или диагональной активности, где горизонтальная активность в общем относится к изменениям пиксельных значений в горизонтальном направлении, вертикальная активность в общем относится к изменениям пиксельных значений в вертикальном направлении, а диагональная активность в общем относится к изменениям пиксельных значений в диагональном направлении.

[0031] В соответствии с методиками этого раскрытия при определении фильтра для блока пикселей поднабор пикселей в пределах блока может использоваться, чтобы уменьшить сложность кодирования и декодирования. Например, при определении фильтра для блока пикселей 4x4 может быть необязательно использовать все шестнадцать пикселей из блока 4x4. Кроме того, в соответствии с методиками этого раскрытия поднабор пикселей в пределах текущего кодируемого блока может быть выбран так, что метрики вычисляются с использованием только пиксельных значений текущего блока, но не пиксельных значений соседних блоков. К примеру, метрика для оцениваемого пикселя может вычисляться на основе сравнения пикселя с расположенными поблизости пикселями. В некоторых случаях один или более пикселей, расположенных поблизости от оцениваемого пикселя, могут быть в блоке, отличном от того, в котором оцениваемый пиксель. В других случаях, однако, один или более пикселей, расположенных поблизости от упомянутого пикселя, могут быть в том же блоке, что и упомянутый пиксель. В соответствии с методиками этого раскрытия поднабор пикселей может выбираться так, чтобы включать в себя пиксели, у которых нет расположенных поблизости пикселей в соседних блоках. Дополнительно или в

качестве альтернативы поднабор пикселей может включать в себя пиксели, у которых есть расположенные поблизости пиксели в соседних блоках, но эти расположенные поблизости пиксели в соседних блоках могут не использоваться при определении метрики. Путем базирования определения конкретной метрики на пикселях в пределах
 5 текущего блока, но не на пикселях из соседних блоков, потребность в буферах в кодере и/или декодере может, в некоторых случаях, быть уменьшена или вовсе устранена.

[0032] В некоторых случаях, в соответствии с методиками этого раскрытия, поднабор пикселей из текущего кодируемого блока может быть выбран так, что метрики вычисляются с использованием только значений пикселей текущего блока и левых и
 10 правых соседних блоков, но не значений пикселей из верхних соседних блоков или нижних соседних блоков. В результате порядка растрового сканирования, используемого при кодировании видеоблоков, линейные буферы для верхних и нижних соседних блоков стремятся к сохранению гораздо большего количества пиксельных значений, нежели линейные буферы для хранения пиксельных значений левых и правых соседних блоков.

[0033] В соответствии с методиками этого раскрытия элемент фильтра, такой как адаптивный внутриконтурный фильтр, может быть сконфигурирован с возможностью использовать несколько фильтров, основываясь на мультиметрическом соответствии фильтров. Несколько фильтров могут использоваться вместе с единственным вводом или множественными вводами. Как будет описано более подробно ниже, множественные
 15 вводы, описанные в данном раскрытии, в общем относятся к промежуточным данным видеоблока или данным изображения, которые производятся во время процессов кодирования и декодирования. Множественные вводы, связанные с заданным видеоблоком, могут включать в себя, например, реконструированный блок или изображение (RI), предварительно деблокированный реконструированный блок или
 25 изображение (pRI), блок или изображение предсказания (PI) и/или квантованное изображение ошибки предсказания (EI). В схеме единичного ввода фильтр может применяться только в отношении одного из представленных выше вводов, такого как RI. Также, как будет более подробно объяснено ниже, методики фильтрации могут применяться в отношении CU различных размеров с использованием схемы разделения
 30 квадродерева. При использовании нескольких фильтров с мультиметрическим соответствием фильтров для CU, разделенного с применением схемы разделения квадродерева, производительность кодирования видео, как измерено посредством одним или обоими из коэффициента сжатия и качества восстановленного видео, может быть улучшено.

[0034] Для реализации методик мультиметрической фильтрации, описанных выше, кодер поддерживает определение соответствия комбинаций диапазонов и фильтров путем генерирования, обновления, сохранения и другими способами. В качестве одного
 35 примера комбинация первого диапазона для первой метрики и первого диапазона для второй метрики может приводиться в соответствие первому фильтру. Комбинация первого диапазона для первой метрики и второго диапазона для второй метрики может также приводиться в соответствие первому фильтру или же может приводиться в соответствие второму фильтру. Если первая метрика имеет восемь диапазонов и вторая метрика имеет четыре диапазона, например, тогда первая и вторая метрики могут иметь
 40 тридцать две комбинации диапазонов, и каждая из тридцати двух комбинаций может приводиться в соответствие фильтру. Каждая комбинация, однако, не обязательно приводится в соответствие единственному фильтру. Так, тридцать две комбинации могут приводиться в соответствие четырём фильтрам, восьми фильтрам, десяти фильтрам или некоторому другому количеству фильтров. С целью применить те же

фильтры, что и кодер, декодер может также поддерживать те же соответствия комбинаций диапазонов фильтрам.

[0035] Раскрытие описывает методики для сигнализации от кодера к декодеру в кодированном битовом потоке соответствия комбинаций диапазонов фильтрам.

5 Приведение в соответствие может, например, связывать каждую комбинацию диапазонов с идентификацией фильтра (ID). Один простой способ сигнализировать об этом соответствии - использовать одно кодовое слово для каждого ID фильтра и затем для каждой комбинации диапазонов отправить кодовое слово соответствующего ID фильтра. Эта методика, однако, обычно неэффективна. Методики по настоящему раскрытию
10 могут использовать корреляции в пределах соответствия путем использования способов разностного кодирования. Комбинации диапазонов, которые совместно используют общий диапазон, иногда используют один и тот же фильтр. В качестве одного примера комбинация первого диапазона для первой метрики и первого диапазона для второй метрики и комбинация первого диапазона для первой метрики и второго диапазона
15 для второй метрики совместно используют общий диапазон (первый диапазон первой метрики). Так, эти две комбинации могут в некоторых случаях соответствовать одному и тому же ID фильтра. Используя эту корреляцию, методики данного раскрытия могут уменьшать количество битов, необходимое для сигнализирования соответствия комбинаций диапазонов разным ID фильтров от кодера к декодеру.

20 [0036] В дополнение к сигнализации соответствия комбинаций диапазонов на множество ID фильтров это раскрытие также описывает методики для сигнализации в кодированном битовом потоке коэффициентов фильтра для фильтров. Методики настоящего раскрытия включают в себя использование способов разностного кодирования для сигнализации коэффициентов фильтра от кодера к декодеру. Таким
25 образом, коэффициенты фильтра для второго фильтра могут быть сообщены декодеру как информация разности, где информация разности описывает, как модифицировать коэффициенты фильтра первого фильтра способом, который создает коэффициенты фильтра второго фильтра. Методики разностного кодирования могут быть более эффективными (то есть могут способствовать большему сохранению битов), когда
30 коэффициенты фильтра первого и второго фильтра более сходны, по сравнению со случаем когда коэффициенты фильтра первого и второго фильтра менее сходны. Методики этого раскрытия включают в себя определение порядка последовательности, в котором необходимо сигнализировать коэффициенты фильтра для фильтра. Очередности, определенные с использованием методик, описанных в этом раскрытии,
35 могут давать в результате улучшенное разностное кодирование коэффициентов фильтра и, таким образом, могут в некоторых случаях давать в результате экономию битов при сигнализации коэффициентов фильтра.

[0037] Хотя методики этого раскрытия иногда могут описываться в связи с внутриконтурной фильтрацией, они могут применяться к внутриконтурной фильтрации,
40 пост-контурной фильтрации и другим схемам фильтрации, таким как переключаемая фильтрация. Внутриконтурная фильтрация в общем относится к фильтрации, при которой отфильтрованные данные являются частью контуров кодирования и декодирования так, что отфильтрованные данные используются для внутреннего или внешнего кодирования с предсказанием. Пост-контурная фильтрация относится к
45 фильтрации, которая применяется в отношении реконструированных видеоданных после контура кодирования. При пост-контурной фильтрации неотфильтрованные данные, в противоположность отфильтрованным данным, используются для внутреннего или внешнего кодирования с предсказанием. В некоторых вариантах

осуществления тип фильтрации может переключаться между пост-контурной фильтрацией и внутриконтурной фильтрацией на, к примеру, покадровой основе, основе вырезка за вырезкой или другой такой основе, и решение о том, использовать ли пост-контурную фильтрацию или внутриконтурную фильтрацию, может сигнализироваться от кодера к декодеру для каждого кадра, вырезки и т.д. Методики этого раскрытия не ограничиваются внутриконтурной фильтрацией или постфильтрацией и могут задействовать широкий диапазон фильтрации, применяемой во время кодирования видео.

[0038] В этом раскрытии термин «кодирование» относится к кодированию или декодированию. Подобным образом, термин «кодер» в общем относится к любому видеокодеру, видеодекодеру или комбинированному кодеру/декодеру (кодеку). Соответственно, термин «кодер» используется здесь в отношении специального компьютерного устройства или аппарата, который осуществляет кодирование видео или декодирование видео.

[0039] Дополнительно в этом раскрытии термин «фильтр» в общем относится к набору коэффициентов фильтра. Например, фильтр 3x3 может определяться набором из 9 коэффициентов фильтра, фильтр 5x5 может определяться набором из 25 коэффициентов фильтра, фильтр 9x5 определяться набором из 45 коэффициентов фильтра и так далее. Термин «набор фильтров» в общем относится к набору из более чем одного фильтра. Например, набор из двух фильтров 3x3 может включать в себя первый набор из 9 коэффициентов фильтра и второй набор из 9 коэффициентов фильтра. В соответствии с методиками этого раскрытия, для серии видеоблоков, таких как кадр, вырезка или самый большой элемент кодирования (LCU), информация, идентифицирующая наборы фильтров, сигнализируется от кодера к декодеру в заголовке серии видеоблоков. Термин «форма», иногда называемый «поддержкой фильтра», в общем относится к количеству строк коэффициентов фильтра и количеству столбцов коэффициентов фильтра для конкретного фильтра. Например, 9x9 представляет собой пример первой формы, 9x5 - это пример второй формы, а 5x9 - пример третьей формы. В некоторых случаях фильтры могут принимать непрямоугольные формы, включая ромбические формы, ромбоподобные формы, круглые формы, подобные круглым формы, шестиугольные формы, восьмиугольные формы, крестообразные формы, Х-образные формы, Т-образные формы, другие геометрические формы или многочисленные другие формы или конфигурации.

[0040] Фиг. 1 представляет собой блок-схему, иллюстрирующую примерную систему 110 кодирования и декодирования видео, которая может реализовывать методики этого раскрытия. Как показано на фиг. 1, система 110 включает в себя устройство-источник 112, которое передает кодированные видеоданные к устройству-адресату 116 посредством канала 115 связи. Устройство-источник 112 и устройство-адресат 116 могут содержать любое из широкого диапазона устройств. В некоторых случаях устройство-источник 112 и устройство-адресат 116 могут содержать беспроводные портативные устройства связи, такие как так называемые сотовые или спутниковые радиотелефоны. Методики этого раскрытия, однако, которые в общем больше применяются к фильтрации видеоданных, не обязательно ограничиваются беспроводными приложениями и установками и могут применяться к небеспроводным устройствам, включающим в себя возможности кодирования и/или декодирования видео.

[0041] В примере с фиг. 1 устройство-источник 112 включает в себя видеоисточник 120, видеокодер 122, модулятор/демодулятор (модем) 123 и передатчик 124. Устройство-

адресат 116 включает в себя приемник 126, модем 127, видеodeкодер 128 и устройство 130 отображения. В соответствии с этим раскрытием видеodeкодер 122 устройства-источника 112 может быть выполнен с возможностью выбирать один или более наборов коэффициентов фильтра для множественных вводов в процессе фильтрации видеоблока и затем кодировать выбранный один или более наборов коэффициентов фильтра. Конкретные фильтры из одного или более наборов коэффициентов фильтра могут быть выбраны на основе одной или более метрик для одного или более вводов, и коэффициенты фильтра могут использоваться для фильтрации одного или более вводов. Методики фильтрации этого раскрытия в общем совместимы с любыми методиками кодирования или сигнализации коэффициентов фильтра в кодированном битовом потоке.

[0042] В соответствии с методиками этого раскрытия устройство, включающее в себя видеodeкодер 122, может сигнализировать устройству, включающему в себя видеodeкодер 128, один или более наборов коэффициентов фильтра для серии видеоблоков, как то кадр или вырезка. Для серии видеоблоков видеodeкодер 122 может, например, сигнализировать один набор фильтров, подлежащих использованию со всеми вводами, или может сигнализировать несколько наборов фильтров, подлежащих использованию с множественными вводами (один набор на каждый ввод, например). Каждый видеоблок или CU в пределах серии видеоблоков затем может содержать дополнительный синтаксис, чтобы идентифицировать, какой фильтр или фильтры из набора фильтров подлежат использованию для каждого ввода этого видеоблока, или в соответствии с методиками этого раскрытия, то, какой фильтр или фильтры из набора фильтров подлежат использованию, может быть определено на основе двух или более метрик, связанных с одним или более вводами.

[0043] Точнее, видеodeкодер 122 устройства-источника 112 может выбирать один или более наборов фильтров для серии видеоблоков, применять фильтры из набора (наборов) к пикселям или группам пикселей из вводов, связанных с CU серии видеоблоков, во время процесса кодирования и затем кодировать наборы фильтров (то есть наборы коэффициентов фильтра) для связи с видеodeкодером 128 устройства-адресата 116. Видеodeкодер 122 может определять одну или более метрик, связанных со вводами CU, кодированных с целью выбрать, какой фильтр (какие фильтры) из набора (наборов) фильтров использовать с пикселями или группами пикселей для того конкретного CU. Видеodeкодер 122 может также сигнализировать видеodeкодеру 128, как часть кодированного битового потока, соответствие комбинаций диапазонов фильтрам в наборе фильтров.

[0044] На стороне декодера, видеodeкодер 128 может определять коэффициенты фильтра на основе информации фильтра, принятой в синтаксисе битового потока. Видеodeкодер 128 может декодировать коэффициенты фильтра на основе прямого декодирования или декодирования с предсказанием в зависимости от того, как были кодированы коэффициенты фильтра, что может быть сигнализировано как часть синтаксиса битового потока. Дополнительно битовый поток может включать в себя информацию синтаксиса описания фильтра, чтобы описать фильтры для набора фильтров. На основе синтаксиса описания фильтра декодер 128 может реконструировать коэффициенты фильтра, основываясь на дополнительной информации, принятой от кодера 122. Проиллюстрированная система 110 с фиг. 1 - только примерная. Методики фильтрации этого раскрытия могут реализовываться любыми устройствами кодирования или декодирования. Устройство-источник 112 и устройство-адресат 116 - это только примеры устройств кодирования, которые могут поддерживать подобные методики.

Видеодекодер 128 также может определять соответствие комбинаций диапазонов фильтрам на основе информации фильтра, принятой в синтаксисе битового потока.

[0045] Видеокодер 122 устройства-источника 112 может кодировать видеоданные, принятые от видеоисточника 120, используя методики этого раскрытия. Видеоисточник 120 может содержать устройство захвата видео, такое как видеокамера, видеоархив, содержащий прежде захваченное видеоизображение, или видеосигнал, поданный от провайдера видеоконтента. В качестве дополнительной альтернативы видеоисточник 120 может генерировать данные на основе компьютерной графики как исходное видео или в комбинации с «живым» видео, архивированным видео и сгенерированным компьютером видео. В некоторых случаях, если видеоисточник 120 является видеокамерой, устройство-источник 112 и устройство-адресат 116 могут формировать так называемые камерофоны или видеофоны. В каждом случае, захваченное, предварительно захваченное или сгенерированное компьютером видеоизображение может быть кодировано видеокодером 122.

[0046] Когда видеоданные кодированы видеокодером 122, информация кодированного видео затем может модулироваться модемом 123 в соответствии со стандартом связи, например, таким как множественный доступ с кодовым разделением каналов (CDMA), множественный доступ с частотным разделением каналов (FDMA), мультиплексирование с ортогональным частотным разделением каналов (OFDM), или любым другим стандартом или методикой связи и передаваться к устройству-адресату 116 посредством передатчика 124. Модем 123 может включать в себя различные блоки смешивания, фильтры, усилители и другие компоненты, созданные для модуляции сигнала. Передатчик 124 может включать в себя схемы, созданные для передачи данных, включая усилители, фильтры и одну или более антенн.

[0047] Приемник 126 устройства-адресата 116 принимает информацию по каналу 115, и модем 127 демодулирует эту информацию. Процесс декодирования видео, выполняемый видеодекодером 128, может включать в себя фильтрацию, например, как часть внутриконтурного декодирования или как этап постфильтрации, следующий за контуром декодирования. Так или иначе, набор фильтров, применяемый видеодекодером 128 для конкретной вырезки или кадра, может быть декодирован с использованием методик этого раскрытия. Информация декодированного фильтра может включать в себя идентификацию синтаксиса описания фильтра в кодированном битовом потоке. Если, например, кодирование с предсказанием используется для коэффициентов фильтра, сходства между различными коэффициентами фильтра могут использоваться, чтобы сократить количество информации, переносимой по каналу 115. В частности, фильтр (то есть набор коэффициентов фильтра) может подвергаться кодированию с предсказанием как значения разности относительно другого набора коэффициентов фильтра, связанного с другим фильтром. Другой фильтр может, например, быть связанным с другими вырезкой или кадром. В таком случае видеодекодер 128 может принимать кодированный битовый поток, содержащий видеоблоки и информацию фильтра, которая идентифицирует другой кадр или вырезка, с которым связан другой фильтр. Информация фильтра также включает в себя значения разности, которые определяют текущий фильтр относительно фильтра другого CU. В частности, значения разности могут содержать значения разности коэффициентов фильтра, которые определяют коэффициенты фильтра для текущего фильтра относительно коэффициентов фильтра другого фильтра, используемого для другого CU.

[0048] Видеодекодер 128 декодирует видеоблоки, генерирует коэффициенты фильтра и фильтрует декодированные видеоблоки на основе сгенерированных коэффициентов

фильтра. Видеодекодер 128 может генерировать коэффициенты фильтра на основе синтаксиса описания фильтра, извлеченного из битового потока. Декодированные и отфильтрованные видеоблоки могут быть собраны в видеокадры, чтобы сформировать декодированные видеоданные. Устройство 128 отображения отображает

5 декодированные видеоданные пользователю и может содержать любое из многообразия устройств отображения, таких как электронно-лучевая трубка (CRT), жидкокристаллический дисплей (LCD), плазменный дисплей, дисплей на органических светодиодах (OLED), или устройство отображения другого типа.

[0049] Канал 115 связи может содержать любое беспроводное или проводное средство

10 связи, такое как радиочастотный (RF) спектр или одна или более физических линий передачи или любую комбинацию беспроводных и проводных средств. Канал 115 связи может формировать часть пакетной сети, такой как локальная сеть, глобальная сеть, такая как Интернет. Канал 115 связи в общем представляет собой любую подходящую среду связи или совокупность различных сред связи для передачи видеоданных от

15 устройства-источника 112 или устройства-адресата 116. Снова, фиг. 1 - только пример и методики этого раскрытия могут применяться к установкам кодирования видео (например, кодирование видео или декодирование видео), которые не обязательно включают в себя какое-либо сообщение данных между устройствами кодирования и декодирования. В других примерах данные могли бы быть извлечены из локального

20 запоминающего устройства, направлены по сети и т.п.

[0050] В качестве альтернативы кодированные данные могут выводиться от видеокодера 122 к устройству 132 хранения данных. Подобным образом, кодированные данные могут быть доступны с запоминающего устройства 132 видеодекодеру 128. Запоминающее устройство 132 может включать в себя любое из распределенных или

25 локально доступных средств для хранения данных, таких как жесткий диск, диски Blu-ray, DVD, CD-ROM, флэш-память, энергозависимое или энергонезависимое запоминающее устройство, или любых других подходящих цифровых носителей для хранения кодированных видеоданных. В дополнительном примере устройство 132 хранения данных может соответствовать файловому серверу или другому

30 промежуточному устройству хранения данных, которое может хранить кодированное видео, сгенерированное устройством-источником 112. Устройство-адресат 116 может осуществлять доступ к сохраненным видеоданным из устройства 132 хранения данных посредством потокового режима или загрузки. Файловый сервер может быть сервером любого типа, способным хранить кодированные видеоданные и передавать эти

35 кодированные видеоданные к устройству-адресату 116. Примерные файловые серверы включают в себя веб-сервер (например, для веб-сайта), FTP-сервер, подключаемые к сети устройства-накопители (NAS) или локальный накопитель на дисках. Устройство-адресат 14 может осуществлять доступ к кодированным видеоданным посредством любого стандартного информационного соединения, включая интернет-соединение.

40 Это может включать в себя беспроводной канал (например, Wi-Fi-соединение), проводное соединение (например, DSL, кабельный модем и т.д.) или комбинацию обоих вариантов, которая является подходящей для доступа к кодированным видеоданным, хранимым на файловом сервере. Передача кодированных видеоданных от устройства 132 хранения данных может быть потоковой передачей, передачей загрузки или их

45 комбинацией.

[0051] Методики этого раскрытия не обязательно ограничиваются беспроводными приложениями или установками. Методики могут применяться в отношении кодирования видео при поддержке любого из многообразия мультимедийных

приложений, таких как эфирное телевизионные вещание, кабельные телевизионные передачи, спутниковые телевизионные передачи, передачи потокового видео, например, через Интернет, кодирование цифрового видео для хранения на средстве хранения данных, декодирование цифрового видео, сохраненного на средстве хранения данных, или других приложений. В некоторых примерах система 110 может быть выполнена с возможностью поддерживать одностороннюю или двустороннюю передачу видео, чтобы поддерживать приложения, такие как потоковое видео, воспроизведение видеозаписи, телевизионное вещание и/или видеотелефония.

[0052] Видеокодер 122 и видеodeкодер 128 могут функционировать в соответствии со стандартом сжатия видео, таким как стандарт ITU-T H.264, в качестве альтернативы именуемый MPEG-4, Часть 10, усовершенствованное кодирование видео (AVC), который будет использоваться в частях этого раскрытия в разъяснительных целях. Однако многие из методик этого раскрытия могут легко применяться к любому из многообразия стандартов кодирования видео, включая недавно появившийся стандарт HEVC. В общем, любой стандарт, который допускает фильтрацию в кодере и декодере, может извлекать пользу из различных аспектов идеи этого раскрытия.

[0053] Хотя это и не показано на фиг. 1, в некоторых аспектах видеокодер 122 и видеodeкодер 128 - каждый из них - могут объединяться с аудиокодером и декодером и могут включать в себя соответствующие элементы MUX-DEMUX или другое аппаратное и программное обеспечение, чтобы осуществить кодирование как аудио, так и видео в общем потоке данных или отдельных потоках данных. Если применимо, элементы MUX-DEMUX могут соответствовать протоколу мультиплексирования ITU H.223 или другим протоколам, таким как протокол пользовательских дейтаграмм (UDP).

[0054] Видеокодер 122 и видеodeкодер 128 - каждый из них - могут быть реализованы как один или более микропроцессоров, цифровых процессоров сигналов (DSP), специализированных интегральных схем (ASIC), программируемых пользователем вентильных матриц (FPGA), блоков дискретной логики, программного обеспечения, аппаратного обеспечения, программно-аппаратного обеспечения или любых их комбинаций. Каждый из видеокодера 122 и видеodeкодера 128 может быть включен в один или более кодеров или декодеров, каждый из которых может быть встроен как часть комбинированного кодера/декодера (CODEC) в соответствующее мобильное устройство, абонентское устройство, вещательное устройство, сервер и т.п.

[0055] В некоторых случаях устройства 112, 116 могут функционировать в основном симметрично. Например, каждое из устройств 112, 116 может включать в себя компоненты кодирования и декодирования видео. Следовательно, система 110 может поддерживать одностороннюю или двустороннюю передачу видео между устройствами 112, 116, например, для потокового видео, воспроизведения видеозаписи, телевизионного вещания или видеотелефонии.

[0056] Во время процесса кодирования видеокодер 122 может выполнять некоторое количество методик или этапов кодирования. В общем, видеокодер 122 осуществляет функционирование в отношении на видеоблоков, с отдельными видеокадрами, чтобы кодировать видеоданные. В одном примере видеоблок может соответствовать макроблоку или фрагменту макроблока. Макроблоки представляют собой один тип видеоблока, определенного стандартом ITU H.264 и другими стандартами. Макроблоки обычно относятся к блокам данных 16x16, хотя этот термин также иногда используется в общем для любого видеоблока размера NxN или NxM. Стандарт ITU-T H.264 поддерживает внутреннее предсказание в блоках различных размеров, таких как 16x16,

8x8, 4x4 для компонент яркости и 8x8 для компонент цветности, а также внешнее предсказание в блоках различных размеров, таких как 16x16, 16x8, 8x16, 8x8, 8x4, 4x8 и 4x4 для компонент яркости, и соответствующих масштабированных размеров для компонент цветности. В этом раскрытии «NxN» относится к пиксельным измерениям блока с точки зрения вертикальных и горизонтальных измерений, например, 16x16 пикселей. В общем, блок 16x16 будет содержать 16 пикселей в вертикальном направлении и 16 пикселей в горизонтальном направлении. Подобным образом блок NxN в общем содержит N пикселей в вертикальном направлении и N пикселей в горизонтальном направлении, где N представляет положительное целое значение. Пиксели в блоке могут быть распределены в строки и столбцы.

[0057] Появившийся стандарт HEVC определяет новые термины для видеоблоков. В частности, видеоблоки (или их фрагменты) могут именоваться «элементами кодирования» (или CU). С использованием стандарта HEVC самые большие элементы кодирования (LCU) могут разделяться на меньшие CU в соответствии со схемой разбиения квадродерева, и различные CU, которые определяются в этой схеме, могут дополнительно фрагментироваться на так называемые элементы предсказания (PU). LCU, CU и PU - это все видеоблоки в пределах значения этого раскрытия. Также могут использоваться другие типы видеоблоков, совместимые со стандартом HEVC или другими стандартами кодирования видео. Так, фраза «видеоблоки» относится к видеоблоку любого размера. Отдельные CU могут быть включены в компоненты яркости и масштабированные размеры для компонент цветности для заданного пикселя, хотя другие цветовые пространства также могут использоваться.

[0058] Видеоблоки могут иметь фиксированные или переменные размеры и могут различаться по размеру в соответствии с определенным стандартом кодирования. Каждый видеокادر может включать в себя множество вырезок. Каждая вырезка может включать в себя множество видеоблоков, которые могут быть распределены на фрагменты, также именуемые субблоками. В соответствии со схемой фрагментирования квадродерева, ссылка на которую была сделана выше и которая более подробно будет описана ниже, первый CU $N/2 \times N/2$ может содержать субблок LCU $N \times N$, второй CU $N/4 \times N/4$ может также содержать субблок первого CU. PU $N/8 \times N/8$ может содержать субблок второго CU. Подобным образом, в качестве дополнительного примера размеры блоков, которые меньше, чем 16x16, могут именоваться фрагментами видеоблока 16x16 или субблоками видеоблока 16x16. Подобно этому, для блока NxN, размеры блока меньше, чем NxN, могут именоваться фрагментами NxN или субблоками NxN. Видеоблоки могут содержать блоки данных пикселя в области пикселя или блоки коэффициентов преобразования в области преобразования, например, осуществляя применение преобразования, такой как дискретное конусное преобразование (DCT), целочисленное преобразование, вейвлетное преобразование или концептуально подобное преобразование, к остаточному видеоблоку данных, представляющему разность пикселей между кодированными видеоблоками и предсказанными видеоблоками. В некоторых случаях видеоблок может содержать блоки квантованных коэффициентов преобразования в области преобразования.

[0059] Данные синтаксиса в пределах битового потока могут определять LCU для кадра или вырезки, который является самым большим элементом кодирования с точки зрения количества пикселей для этого кадра или вырезки. В общем, LCU или CU имеет сходное назначение для макроблока, кодированного согласно H.264, за исключением того, что LCU и CU не имеют определенного различия по размеру. Вместо этого размер LCU может определяться на покадровой основе или на основе вырезка за вырезкой, и

LCU может разбиваться на CU. В общем, ссылки в этом раскрытии на CU могут относиться к LCU изображения или к суб-CU LCU. LCU может быть разбит на суб-CU, и каждый суб-CU может быть разбит на суб-CU. Данные синтаксиса для битового потока могут определять максимальное количество раз, на которое LCU может быть разбит, что называется глубиной CU. Соответственно, битовый поток также может определять самый маленький элемент кодирования (SCU). Это раскрытие также использует термины «блок» и «видеоблок», чтобы сослаться на любой из LCU, CU, PU, SCU и TU.

[0060] Как было представлено выше, LCU может быть связан со структурой данных квадродерева. В общем, структура данных квадродерева включает в себя один узел на каждый CU, где корневой узел соответствует LCU. Если CU разбивается на четыре суб-CU, узел, соответствующий CU, включает в себя четыре листовых узла, каждый из которых соответствует одному из суб-CU. Каждый узел структуры данных квадродерева может обеспечивать данные синтаксиса для соответствующего CU. Например, узел в квадродереве может включать в себя флаг разбиения, указывающий, разбивается ли CU, соответствующий узлу, на суб-CU. Компоненты синтаксиса для CU могут определяться рекурсивно и могут зависеть от того, разбивается ли CU на суб-CU.

[0061] CU, который не разбивается, может включать в себя один или более элементов предсказания (PU). В общем, PU представляет все или часть соответствующего CU и включает в себя данные для извлечения опорного образца для PU. Например, когда PU кодирован во внутреннем режиме, он может включать в себя данные, описывающие режим внутреннего предсказания для PU. В качестве другого примера, когда PU, кодирован во внешнем режиме, он может включать в себя данные, определяющие вектор движения для PU. Данные, определяющие вектор движения, могут описывать, например, горизонтальную компоненту вектора движения, вертикальную компоненту вектора движения, разрешение для вектора движения (например, точность до четверти пикселя или точность до одной восьмой пикселя), опорный кадр, на который указывает вектор движения, и/или опорный список (например, список 0 или список 1) для вектора движения. Данные для CU, определяющие PU (или более PU), могут также описывать, к примеру, фрагментирование CU на один или более PU. Режимы фрагментирования могут отличаться в плане того, является ли CU некодированным, кодированным в режиме внутреннего предсказания или кодированным в режиме внешнего предсказания.

[0062] CU, имеющий один или более PU, также может включать в себя один или более элементов преобразования (TU). TU содержат структуру данных, которая включает в себя остаточные коэффициенты преобразования, которые обычно квантуются. В частности, после предсказания с использованием PU видеокодер может вычислять остаточные значения для фрагмента CU, соответствующей PU. Остаточные значения могут преобразовываться, квантоваться, сканироваться и храниться в TU, который может иметь переменные размеры, соответствующие размеру преобразования, которое было осуществлено. Соответственно, TU не обязательно ограничивается размером PU. Так, TU могут быть больше или меньше, чем соответствующие PU для одного и того же CU. В некоторых примерах максимальный размер TU может быть размером соответствующего CU. Еще раз стоит отметить, что TU может содержать структуры данных, которые включают в себя остаточные коэффициенты преобразования, связанные с заданным CU.

[0063] Фиг. 2А и фиг. 2В представляют собой концептуальные схемы, иллюстрирующие примерное квадродерево 250 и соответствующий LCU 272. Фиг. 2А иллюстрирует примерное квадродерево 250, которое включает в себя узлы, распределенные иерархически. Каждый узел в квадродереве, таком как квадродерево

250, может быть концевой вершиной без потомков или иметь узлы-потомки. В примере фиг. 2А квадродерево 250 включает в себя корневой узел 252. Корневой узел 252 имеет четыре узла-потомка 256А-256С (концевые вершины 256) и узел 254. Поскольку узел 254 не является концевой вершиной, он включает в себя четыре узла-потомка, которые

5 в этом примере - концевые вершины 258А-258D (концевые вершины 258).
 [0064] Квадродерево 250 может включать в себя данные, описывающие характеристики соответствующего LCU, такого как LCU 272, например. К примеру, квадродерево 250 по своей структуре может описывать разбиение LCU на суб-CU. Допустим, что LCU 272 имеет размер $2N \times 2N$. LCU 272 в этом примере имеет четыре
 10 суб-CU 276А-276С (суб-CU 276) и 274, каждый размера $N \times N$. Суб-CU 274 дополнительно разбивается на четыре суб-CU 278А-278D (суб-CU 278), каждый размера $N/2 \times N/2$. Структура квадродерева 250 соответствует разбиению LCU 272 в этом примере. То есть корневой узел 252 соответствует LCU 272, концевые вершины 256 соответствуют суб-CU 276, узел 254 соответствует суб-CU 274, а концевые вершины 258 соответствуют
 15 суб-CU 278.

[0065] Данные для узлов квадродерева 250 могут описывать, разбивается ли CU, соответствующий узлу. Если CU разбивается, в квадродереве 250 могут присутствовать четыре дополнительных узла. В некоторых примерах узел квадродерева может быть реализован подобно тому, как в следующем псевдокоде:

```

20  quadtree_node {
    boolean split_flag(1);
    // signaling data
    if (split_flag) {
        quadtree_node child1;
25  quadtree_node child2;
        quadtree_node child3;
        quadtree_node child4;
    }
    }
30  узел_квадродерева {
    булевый флаг_разбиения(1);
    // сообщение данных
    если (флаг_разбиения) {
        потомок1 узла_квадродерева;
35  потомок2 узла_квадродерева;
        потомок3 узла_квадродерева;
        потомок4 узла_квадродерева;
    }
    }

```

40 Значение флага_разбиения может быть однобитным значением, показывающим, разбивается ли CU, соответствующий текущему узлу. Если CU не разбивается, значение флага_разбиения может быть «0», тогда как если CU разбивается, значение флага_разбиения может быть «1». Что касается примера квадродерева 250, массив значений флага разбиения может быть 1010000.

45 [0066] В некоторых примерах каждый из суб-CU 276 и суб-CU 278 может быть подвергшимся кодированию с внутренним предсказанием с использованием одного и того же режима внутреннего предсказания. Соответственно, видеокодер 122 может обеспечивать индикацию режима внутреннего предсказания в корневом узле 252. Более

того, определенные размеры суб-CU могут иметь множественные возможные преобразования для конкретного режима внутреннего предсказания. Видеокодер 122 может обеспечивать индикацию преобразования, подлежащего использованию для таких суб-CU в корневом узле 252. Например, суб-CU размера $N/2 \times N/2$ может иметь множество доступных возможных преобразований. Видеокодер 122 может сигнализировать преобразование, которое нужно использовать в корневом узле 252. Соответственно, видеodeкодер 128 может определять преобразование, которое следует применить к суб-CU 278, на основе режима внутреннего предсказания, сигнализированного в корневом узле 252, и преобразования, сигнализированного в корневом узле 252.

[0067] Как таковой видеокодер 122 не нуждается в сигнализации преобразований, подлежащих применению к суб-CU 276 и суб-CU 278, в концевых вершинах 256 и концевых вершинах 258, но может вместо этого просто сигнализировать режим внутреннего предсказания и, в некоторых примерах, преобразование, подлежащее применению к определенным размерам суб-CU, в корневом узле 252, в соответствии с методиками этого раскрытия. Таким образом, эти методики могут снизить издержки на сигнализирование функций преобразования для каждого суб-CU LCU, такого как LCU 272.

[0068] В некоторых примерах режимы внутреннего предсказания для суб-CU 276 и/или суб-CU 278 могут быть отличными от режимов внутреннего предсказания для LCU 272. Видеокодер 122 и видеodeкодер 130 могут быть сконфигурированы с функциями, которые приводят в соответствие режим внутреннего предсказания, сигнализированный в корневом узле 252, доступному режиму внутреннего предсказания для суб-CU 276 и/или суб-CU 278. Функция может обеспечивать соответствие "множество-к-одному" режимов внутреннего предсказания, доступных для LCU 272, режимам внутреннего предсказания для суб-CU 276 и/или суб-CU 278.

[0069] Вырезка может быть разделен на видеоблоки (или LCU), и каждый видеоблок может быть фрагментирован в соответствии со структурой квадродерева, описанной в отношении фиг. 2А - фиг. 2В. Дополнительно, как показано на фиг. 2С, субблоки квадродерева, отмеченные «включено» («ON») могут быть отфильтрованы циклическими фильтрами, описываемыми здесь, тогда как субблоки квадродерева, отмеченные «выключено» («OFF»), могут не быть отфильтрованы. Решение о том, фильтровать или нет заданный блок или субблок, может быть принято в кодере путем сравнения отфильтрованного результата и неотфильтрованного результата в отношении оригинального блока, подвергаемого кодированию. Фиг. 2D - это дерево решений, представляющее решения о фрагментировании, что приводит к фрагментированию квадродерева, показанному на фиг. 2С. Действительная фильтрация, применяемая к любым пикселям для «включенных» блоков, может определяться на основе метрик, обсуждаемых здесь.

[0070] В частности, фиг. 2С может представлять относительно большой видеоблок, который фрагментируется в соответствии со схемой разбиения квадродерева на меньшие видеоблоки переменных размеров. Каждый видеоблок помечается (включен или выключен) на фиг. 2С, чтобы проиллюстрировать, должна ли фильтрация применяться или миновать данный видеоблок. Видеокодер может определить это соответствие фильтра путем сравнения отфильтрованных и неотфильтрованных версий каждого видеоблока с оригинальным кодируемым видеоблоком.

[0071] Еще раз отметим, что фиг. 2D - это дерево решений, соответствующих решениям о фрагментировании, которые приводят к фрагментированию квадродерева,

показанному на фиг. 2С. На фиг. 2D каждый круг может соответствовать CU. Если круг включает в себя флаг «1», то CU дополнительно фрагментируется на еще четыре CU, но если круг включает в себя флаг «0», то CU больше не фрагментируется. Каждый круг (например, соответствующий CU), также включает в себя соответствующий ромб.

- 5 Если флаг в ромбе для заданного CU установлен на 1, то фильтрация переходит на «включено» для этого CU, не если флаг в ромбе для заданного CU установлен на 0, то фильтрация выключается. Таким образом, фиг. 2С и фиг. 2D могут по отдельности или вместе рассматриваться как схема фильтра, которая может быть сгенерирована в кодере и сообщена к декодеру по меньшей мере один раз за вырезку кодированных
- 10 видеоданных, чтобы сообщить уровень фрагментирования квадродерева для заданного видеоблока (например, LCU), применять ли фильтрацию к каждому фрагментированному видеоблоку (например, каждому CU в пределах LCU).

- [0072] Меньшие видеоблоки могут обеспечивать лучшее разрешение и могут использоваться для расположений видеокадра, которые включают в себя высокий
- 15 уровень деталей. Более большие видеоблоки могут обеспечивать более высокую эффективность кодирования и могут использоваться для точек расположения видеокадра, которые включают в себя низкий уровень деталей. Вырезка может рассматриваться как множество видеоблоков и/или субблоков. Каждая вырезка может быть независимо декодируемой серией видеоблоков видеокадра. В качестве
- 20 альтернативы сами кадры могут быть декодируемыми сериями видеоблоков. Термин «серия видеоблоков» может относиться к любой независимо декодируемой части видеокадра, такой как целый кадр, вырезка кадра, группа изображений (GOP), также именуемая последовательностью, или другому независимо декодируемому элементу, определенному в соответствии с применимыми методиками кодирования. Аспекты
- 25 этого раскрытия могли бы быть описаны со ссылкой на кадры или вырезки, но такие ссылки лишь примерные. Следует понимать, что в общем любая серия видеоблоков может использоваться вместо кадра или вырезки.

- [0073] Данные синтаксиса могут быть определены на основе наличия в каждом кодированном элементе так, что каждый CU включает в себя соответствующие данные
- 30 синтаксиса. Информация фильтра, описанная здесь, может быть частью такого синтаксиса для CU, но более вероятно могла бы быть частью синтаксиса для серии видеоблоков, таких как кадр, вырезка, GOP, LCU или последовательность видеокадров, нежели для CU. Данные синтаксиса могут указывать набор или наборы фильтров, подлежащих использованию с CU вырезки или кадра. Дополнительно не вся информация
- 35 фильтра обязательно должна быть включена в заголовок общей серии видеоблоков. Например, синтаксис описания фильтра может передаваться в заголовке кадра, тогда как другая информация фильтра сигнализируется в заголовке для LCU.

- [0074] Видеокодер 122 может осуществлять кодирование с предсказанием, в котором кодируемый видеоблок сравнивается с предсказанным кадром (или другим CU), чтобы
- 40 идентифицировать предсказанный блок. Различия между текущим кодируемым видеоблоком и предсказанным блоком кодируются как остаточный блок, и синтаксис предсказания используется, чтобы идентифицировать предсказанный блок. Остаточный блок может быть преобразован и квантован. Методики преобразования могут содержать процесс DCT или концептуально подобный процесс, целочисленные преобразования,
- 45 вейвлетные преобразования или другие типы преобразований. В процессе DCT, как пример, процесс преобразования преобразует набор пиксельных значений в коэффициенты преобразования, которые могут представлять энергию пиксельных значений в частотной области. Квантование обычно применяется к коэффициентам

фильтра и в общем включает в себя процесс, который ограничивает количество битов, связанных с любым заданным коэффициентом преобразования.

[0075] После преобразования и квантования к квантованным и преобразованным остаточным видеоблокам может применяться энтропийное кодирование. Синтаксические компоненты, такие как информация фильтра и векторы предсказания, определенные во время кодирования, также могут быть включены в подвергшийся энтропийному кодированию битовый поток для каждого CU. В общем, энтропийное кодирование содержит один или более процессов, которые совместно сжимают последовательность квантованных коэффициентов преобразования и/или другую информацию синтаксиса. Методики сканирования, такие как методики зигзагообразного сканирования, реализуются в отношении квантованных коэффициентов преобразования, например, как часть процесса энтропийного кодирования, чтобы определить один или более преобразованных в последовательную форму одномерных векторов коэффициентов из двухмерных видеоблоков. Другие методики сканирования, включающие в себя порядки сканирования или адаптивные сканы, также могут использоваться и, возможно, сигнализироваться в кодированном битовом потоке. В любом случае сканированные коэффициенты затем подвергаются энтропийному кодированию наряду с информацией синтаксиса, например, посредством контекстно-адаптивного кодирования с переменной длиной (CAVLC) или контекстно-адаптивного двоичного арифметического кодирования (CABAC) или другого процесса энтропийного кодирования.

[0076] Как часть процесса кодирования, кодированные видеоблоки могут декодироваться с целью генерировать видеоданные, используемые для последующего кодирования на основе предсказания в отношении последующих видеоблоков. На этом этапе фильтрация может осуществляться с целью улучшить качество видео и, например, устранить артефакты блочности из декодированного видео. Отфильтрованные данные могут использоваться для предсказания других видеоблоков, в этом случае фильтрация называется «внутриконтурной» фильтрацией. В качестве альтернативы предсказание других видеоблоков может быть основано на неотфильтрованных данных, в этом случае фильтрация называется «пост-фильтрацией».

[0077] На покадровой основе, основе вырезка за вырезкой или на основе LCU-за-LCU видеокoder 122 может выбирать один или более наборов фильтров и на основе кодированный элемент за кодированным элементом видеокoder может определять, осуществлять ли фильтрацию. Для CU, которые подлежат фильтрации, кодер может осуществлять фильтрацию на попиксельной или погрупповой основе, где группа могла бы, например, быть блоком пикселей 2x2 или блоком пикселей 4x4. Эти выборы могут быть сделаны так, чтобы поддерживать качество видео. Такие наборы фильтров могут выбираться из заранее определенных наборов фильтров или могут адаптивно определяться, чтобы поддерживать качество видео. В качестве примера видеокoder 122 может выбирать или определять несколько наборов фильтров для заданного кадра или вырезки так, что различные фильтры используются для разных пикселей или групп пикселей CU этого кадра или вырезки. В частности, для каждого ввода, связанного с CU, может быть определено несколько наборов коэффициентов фильтра, и две или более метрики, связанные с пикселями CU, могут быть использованы, чтобы определить, какой фильтр из набора фильтров применять с такими пикселями или группами пикселей.

[0078] В некоторых случаях видеокoder 122 может применять несколько наборов коэффициентов фильтра и выбирать один или более наборов, которые создают видео лучшего качества с точки зрения количества искажений между кодированным блоком и оригинальным блоком и/или самых высоких уровней сжатия. В любом случае, будучи

однажды выбранным, набор коэффициентов фильтра, применяемый видеокодером 122 для каждого CU, может быть кодирован и сообщен к видеодекодеру 128 устройства-адресата 118 так, что видеодекодер 128 может применять такую же фильтрацию, что и примененная во время процесса кодирования, для каждого заданного CU.

5 [0079] Когда две или более метрики используются для определения того, какой фильтр использовать с конкретным вводом для CU, выбор фильтра для этого конкретного CU не обязательно должен сообщаться видеодекодеру 128. Наоборот, видеодекодер 128 также может вычислять две или более метрики и на основе информации фильтра, прежде предоставленной видеокодером 122, согласовывать комбинацию из
10 двух или более метрик для конкретного фильтра.

[0080] Фиг. 3 представляет собой блок-схему, иллюстрирующую видеокодер 350, в соответствии с данным раскрытием. Видеокодер 350 может соответствовать видеокодеру 122 устройства 120 или видеокодеру другого устройства. Как показано на фиг. 3, видеокодер 350 включает в себя элемент 332 предсказания, сумматоры 348 и 351 и
15 запоминающее устройство 334. Видеокодер 350 также включает в себя элемент 338 преобразования и элемент 340 квантования, как и элемент 342 обратного квантования и элемент 344 обратного преобразования. Видеокодер 350 также включает в себя фильтр 347 удаления блочности и элемент 349 адаптивного фильтра. Видеокодер 350 также включает в себя элемент 346 энтропийного кодирования. Элемент 349 фильтра
20 видеокодера 350 может осуществлять операции фильтрации и также может включать в себя элемент 353 выбора фильтра (FSU) для идентификации желаемого или предпочтительного фильтра или набора фильтров, подлежащих использованию для декодирования. Элемент 349 фильтра также может генерировать информацию фильтра, идентифицирующую выбранные фильтры так, что выбранные фильтры могут
25 эффективно сообщаться как информация фильтра к другому устройству, подлежащему использованию во время операции декодирования.

[0081] Во время процесса кодирования видеокодер 350 принимает видеоблок, такой как LCU, подлежащий кодированию, и модуль 332 предсказания реализует методики кодирования с предсказанием в отношении видеоблока. С использованием схемы
30 фрагментирования квадродрева, рассмотренной выше, модуль 332 предсказания может фрагментировать видеоблок и реализовывать методики кодирования с предсказанием в отношении CU разных размеров. Для внешнего кодирования модуль 332 предсказания сравнивает подлежащий кодированию видеоблок, включающий в себя субблоки видеоблока, с различными блоками в одном или более опорных видеокадрах или
35 вырезок с целью определить предсказанный блок. Для внутреннего кодирования модуль 332 предсказания генерирует блок предсказания на основе смежных данных в пределах того же CU. Модуль 332 предсказания выводит предсказанный блок, и сумматор 348 вычитает предсказанный блок из кодируемого видеоблока, чтобы генерировать остаточный блок.

40 [0082] Для внешнего кодирования модуль 332 предсказания может содержать элементы оценки движения и компенсации движения, которые идентифицируют вектор движения, который указывает на предсказанный блок и генерирует предсказанный блок на основе вектора движения. Обычно оценка движения рассматривается как процесс генерирования вектора движения, который оценивает движение. Например,
45 вектор движения может представлять смещение предсказанного блока в пределах предсказанного кадра относительно текущего кодируемого блока в пределах текущего кадра. Компенсация движения обычно рассматривается как процесс выборки или генерирования предсказанного блока на основе вектора движения, определенного

оценкой движения. Для внутреннего кодирования модуль 332 предсказания генерирует предсказанный блок на основе смежных данных в пределах того же CU. Один или более режимов внутреннего предсказания могут определять, как может быть определен блок внутреннего предсказания.

5 [0083] После этого модуль 332 предсказания выводит блок предсказания и сумматор 348 вычитает блок предсказания из кодируемого видеоблока, чтобы генерировать остаточный блок, элемент 338 преобразования применяет преобразование к остаточному блоку. Преобразование может содержать дискретное косинусное преобразование (DCT) или концептуально подобное преобразование, такое как определенное стандартом
10 кодирования, таким как стандарт HEVC. Вейвлетные преобразования, целочисленные преобразования, субполосные преобразования или другие типы преобразований также могут использоваться. В любом случае, элемент 338 преобразования применяет преобразование к остаточному блоку, производя блок остаточных коэффициентов преобразования. Преобразование может преобразовать остаточную информацию из
15 пиксельной области в частотную область.

[0084] Элемент 340 квантования затем квантует остаточные коэффициенты преобразования, чтобы дополнительно снизить скорость передачи битов. Элемент 340 квантования, например, может ограничивать количество битов, используемых, чтобы кодировать каждый из коэффициентов. После квантования элемент 346 энтропийного
20 кодирования сканирует квантованный блок коэффициента из двухмерного представления в один или более преобразованных в последовательную форму одномерных векторов. Порядок сканирования может быть предварительно запрограммированным так, чтобы реализовываться в определенном порядке (как то зигзагообразное сканирование, горизонтальное сканирование, вертикальное сканирование, комбинации или другой
25 заранее определенный порядок) или, возможно, адаптивно определенным на основе предшествующей статистики кодирования.

[0085] После этого процесса сканирования элемент 346 энтропийного кодирования кодирует квантованные коэффициенты преобразования (наряду с синтаксическими компонентами) в соответствии с методологией энтропийного кодирования, как то
30 CAVLC или CABAC, чтобы дополнительно сжать данные. Синтаксические компоненты, включенные в энтропийно кодированный битовый поток, могут включать в себя синтаксис предсказания из модуля 332 предсказания, такой как векторы движения для внешнего кодирования или режимы предсказания для внутреннего кодирования. Синтаксические компоненты, включенные в энтропийно кодированный битовый поток,
35 могут также включать в себя информацию фильтра от элемента 349 фильтра, которая может быть кодирована способом, описанным здесь.

[0086] CAVLC - это один тип методики энтропийного кодирования, поддерживаемый стандартом ITU H.264/MPEG-4, AVC, который может применяться на векторизованной основе элементом 346 энтропийного кодирования. CAVLC использует таблицы
40 кодирования с переменной длиной (VLC) таким образом, который способствует эффективному сжатию преобразованных в последовательную форму «ранов» коэффициентов преобразования и/или синтаксических компонентов. CABAC - это другой тип методики энтропийного кодирования, поддерживаемый стандартом ITU H.264/MPEG-4, AVC, который может применяться на векторизованной основе элементом
45 346 энтропийного кодирования. CABAC включает в себя несколько этапов, включая преобразование в двоичную форму, выбор модели контекста и двоичное арифметическое кодирование. В этом случае элемент 346 энтропийного кодирования кодирует коэффициенты преобразования и синтаксические компоненты в соответствии с CABAC.

Подобно стандарту ITU H.264/MPEG-4, AVC, появившийся стандарт HEVC также может поддерживать как энтропийное кодирование CAVLC, так и энтропийное кодирование CABAC. Кроме того, существуют и многие другие типы методик энтропийного кодирования, и новые методики энтропийного кодирования, вероятно, появятся в будущем. Это раскрытие не ограничивается какой-либо конкретной методикой энтропийного кодирования.

[0087] После энтропийного кодирования элементом 346 энтропийного кодирования кодированное видео может передаваться к другому устройству или архивироваться для передачи или извлечения позднее. Снова стоит отметить, что кодированное видео может содержать энтропийно кодированные векторы и различный синтаксис, который может быть использован декодером, чтобы надлежащим образом конфигурировать процесс декодирования. Элемент 342 обратного квантования и элемент 344 обратного преобразования осуществляют обратное квантование и обратное преобразование соответственно, чтобы реконструировать остаточный блок в области пикселя. Сумматор 351 прибавляет реконструированный остаточный блок к блоку предсказания, созданному модулем 332 предсказания, чтобы создать предварительно деблокированный реконструированный видеоблок, иногда именуемый предварительно деблокированным реконструированным изображением. Фильтр 347 удаления блочности может применять фильтрацию к предварительно деблокированному реконструированному видеоблоку, чтобы улучшить качество видео путем устранения блочности и других артефактов. Вывод деблокирующего фильтра 347 может именоваться впоследствии деблокированным видеоблоком, реконструированным видеоблоком или реконструированным изображением.

[0088] Модуль 349 фильтра может быть выполнен с возможностью принимать единственный ввод или множественные вводы. В примере с фиг. 3 модуль 349 фильтра принимает в качестве ввода впоследствии деблокированное реконструированное изображение (RI), предварительно деблокированное реконструированное изображение (pRI), изображение предсказания (PI) и реконструированный остаточный блок (EI). Модуль 349 фильтра может использовать любые из этих вводов, как отдельно, так и в сочетании, чтобы создать реконструированное изображение, чтобы хранить в памяти 334. Дополнительно, как будет ниже рассматриваться более подробно, на основе двух или более метрик один или более фильтров могут быть выбраны для применения к вводу (вводам). В одном примере вывод модуля 349 фильтра может быть одним дополнительным фильтром, применяемым к RI. В другом примере вывод модуля 349 фильтра может быть одним дополнительным фильтром, применяемым к pRI. В других примерах, однако, вывод модуля 349 фильтра может быть основан на множественных вводах. Например, модуль 349 фильтра может применять первый фильтр к pRI и затем использовать отфильтрованную версию pRI совместно с отфильтрованными версиями EI и PI, чтобы создать реконструированное изображение. В случаях, когда вывод модуля 349 фильтра - это продукт одного дополнительного фильтра, применяемого к единственному вводу, модуль 349 фильтра может фактически применять фильтры к другим вводам, но те фильтры могли бы иметь все нулевые коэффициенты. Подобным образом, если вывод модуля 349 фильтра - это продукт применения трех фильтров к трем вводам, модуль 349 фильтра может фактически применять фильтр к четвертому вводу, но тот фильтр может иметь все нулевые коэффициенты.

[0089] Модуль 349 фильтра также может быть выполнен с возможностью принимать единственный ввод. Например, хотя фиг. 3 показывает PI, EI, pRI и RI вводимыми в модуль 349 фильтра, в некоторых вариантах осуществления RI может быть

единственным вводом, принятым модулем 349 фильтра. В таком варианте осуществления модуль 349 фильтра может применять фильтр к RI так, что отфильтрованная версия RI более сходна с оригинальным изображением, чем неотфильтрованная версия RI. В других вариантах осуществления модуль 349 фильтра и фильтр 347 удаления блочности могут быть скомбинированы в единственный элемент фильтрации, который применяет фильтрацию к rRI. Методики этого раскрытия, которые в общем относятся к построению соответствия фильтров на мультиметрической основе, совместимы как со схемами фильтрации с единственным вводом, так и со схемами фильтрации с множеством вводов, которые используют множественные фильтры.

[0090] Фильтрация модулем 349 фильтра может улучшить сжатие путем генерирования предсказанных видеоблоков, которые более точно соответствуют кодируемым видеоблокам, нежели неотфильтрованные предсказанные видеоблоки. После фильтрации реконструированный видеоблок может использоваться модулем 332 предсказания как опорный блок, чтобы подвергнуть внешнему кодированию блок в последующем видеокadre или другой CU. Хотя модуль 349 фильтра и показан «внутриконтурным», методики этого раскрытия также могут применяться с пост-фильтрами, в этом случае неотфильтрованные данные (в отличие от отфильтрованных данных) могли бы использоваться в целях предсказания данных в последующих CU.

[0091] Для серии видеоблоков, такой как вырезка или кадр, модуль 349 фильтра может выбирать наборы фильтров для каждого ввода способом, который поддерживает качество видео. Например, модуль 349 фильтра может выбирать наборы фильтров из предварительно заданных наборов коэффициентов или может адаптивно задавать фильтры, чтобы поддерживать качество видео или улучшенное сжатие. Модуль 349 фильтра выбирать или задавать один или более наборов фильтров для заданного CU так, что один и тот же набор (наборы) фильтров используется (используются) для пикселей из разных видеоблоков того CU. Для конкретного кадра вырезка или LCU модуль 349 фильтра может применять несколько наборов фильтров к множественным вводам, и FSU 353 может выбирать набор, который создает видео лучшего качества или самые высокие уровни сжатия. В качестве альтернативы FSU 353 может подготавливать новый фильтр путем анализа автокорреляций и кросс-корреляций между множественными вводами и оригинальным изображением. Новый набор фильтров может, к примеру, быть определен путем решения уравнений Винера-Хопфа на основе авто- и кросс-корреляций. Вне зависимости от того, подготавливается ли новый набор фильтров или выбирается существующий набор фильтров, модуль 349 фильтра генерирует синтаксис для включения в битовый поток, что позволяет декодеру также идентифицировать набор или наборы фильтров, подлежащих использованию для конкретного кадра или вырезки.

[0092] В соответствии с этим раскрытием для каждого пикселя CU в пределах серии видеоблоков модуль 349 фильтра может выбирать, какой фильтры из набора фильтров использовать, на основе двух или более метрик, которые определяют количество свойств, связанных с одним или более наборами пикселей в пределах CU. Таким образом, FSU 353 может определять наборы фильтров для кодированного блока более высокого уровня, как то кадр или вырезка, тогда как модуль 349 фильтра определяет, какой фильтр (какие фильтры) из набора (наборов) подлежит использованию для конкретного пикселя кодированного элемента более низкого уровня, на основе двух или более метрик, связанных с пикселями этого кодированного элемента более низкого уровня.

[0093] Набор M-фильтров может использоваться для каждого ввода. В зависимости от предпочтений исполнения M может, например, быть лишь 2 или почти как 16, или

даже больше. Большое количество фильтров на ввод может улучшить качество видео, но также может увеличить издержки, связанные с наборами сигнализации у фильтров от кодера к декодеру. Набор М-фильтров может определяться FSU 353, как описывается выше, и сигнализироваться к декодеру для каждого кадра или вырезки. Схема сегментации может использоваться для индикации того, как сегментируются CU, и того, 5 подлежит ли фильтрации конкретный подэлемент CU. Схема сегментации может, например, включать в себя массив флагов разбиения для суб-CU, как описывается выше, как и дополнительный бит, сигнализирующий о том, подлежит ли каждый суб-CU фильтрации. Для каждого ввода, связанного с пикселем CU, который подлежит 10 фильтрации, может быть выбран конкретный фильтр из набора фильтров на основе двух или более метрик. Комбинации значений для двух или более метрик могут быть индексированы для конкретных фильтров из набора М-фильтров.

[0094] Фиг. 4А представляет собой концептуальную схему, иллюстрирующую диапазоны значений для двух метрик, индексированных для фильтров из набора 15 фильтров. Конкретный пример с фиг. 4А демонстрирует восемь фильтров (то есть Фильтр 1, Фильтр 2 ... Фильтр 8), но также может использоваться и большее или меньшее количество фильтров. Фиг. 4А демонстрирует две метрики, которые могут использоваться для выбора фильтра в соответствии с методиками этого раскрытия. Две метрики могут, например, определять количество свойств данных пикселя, 20 относящихся к активности, не связанной с направлением (например, значение Лапласиана с измененной суммой), и направлению, активности, связанной с направлением, и обнаружению краев, метрике направления и метрике края, метрике горизонтальной активности и метрике вертикальной активности или двум другим подобным метрикам. В некоторых случаях три или более метрик могло бы 25 использоваться, в этом случае концептуальная схема с фиг. 4А включала бы в себя третье измерение для диапазонов соответствия метрик фильтрам из набора фильтров.

[0095] В примере с фиг. 4А первая метрика (Метрика 1) имеет четыре диапазона (Диапазоны 1-1, 1-2, 1-3 и 1-4) и вторая метрика (Метрика 2) также имеет четыре 30 диапазона (Диапазоны 2-1, 2-2, 2-3 и 2-4). Следовательно, пример с фиг. 4А имеет шестнадцать комбинаций диапазонов для Метрики 1 и Метрики 2. Как можно увидеть с фиг. 4А, однако каждая комбинация не обязательно связана с уникальным фильтром. Комбинация Диапазона 1-1 и Диапазона 2-1, как и комбинации 1-1 и 2-2, и 1-1 и 2-3, к примеру, все приведены в соответствие Фильтру 1, в примере с фиг. 4А. Фильтр 4, напротив, приведен в соответствие только одной комбинации (1-1 и 2-4). Хотя диапазоны 35 с фиг. 4А показана как примерно равные, размеры диапазонов могут варьироваться. Например, в некоторых вариантах осуществления Диапазон 1-1 может охватывать больший диапазон значений, нежели Диапазон 1-2. Дополнительно, хотя фиг. 4А демонстрирует Метрику 1 и Метрику 2 как имеющие одинаковое количество диапазонов, количество диапазонов для первой метрики и количество диапазонов для второй 40 метрики не обязательно должны быть равными. Если, к примеру, Метрика 1 - метрика дисперсии, а Метрика 2 - метрика направления, Метрика 1 могла бы использовать восемь диапазонов, тогда как Метрика 2 использует три диапазона.

[0096] В некоторых примерах диапазоны Метрики 1 и Метрики 2 могут представлять непрерывный спектр значений. Например, если Метрика 1 является значением 45 Лапласиана с измененной суммой, Диапазон 1-2 может соответствовать большей активности, нежели Диапазон 101, но меньшей активности, чем Диапазон 1-3, и Диапазон 1-4 может соответствовать большей активности, чем Диапазон 1-3. В пределах диапазона количество активности, определенное для конкретного пикселя или группы пикселей,

может подобным образом увеличиваться вдоль оси Метрики 1. В других примерах диапазоны Метрики 1 и Метрики 2 могут не представлять действительных диапазонов, но вместо этого могут представлять дискретные определения. Например, если Метрика 2 - метрика направления, Диапазон 1-1 может соответствовать определению отсутствия направления, Диапазон 2-2 может соответствовать определению горизонтального направления, Диапазон 2-3 может соответствовать определению вертикального направления, а Диапазон 2-4 может соответствовать определению диагонального направления. Как будет описано более подробно ниже, отсутствие направления, горизонтальное направление, вертикальное направление и диагональное направление могут быть дискретными определениями, и таким образом, диапазоны для Метрики 2 могли бы не представлять непрерывный спектр значений таким же образом, как делают диапазоны Метрики 1.

[0097] Фиг. 4В представляет собой концептуальную схему, иллюстрирующую диапазоны значений для метрики активности и метрики направления. В примере фиг. 4В метрика направления включает в себя три дискретных определения (Отсутствие Направления, Горизонтальное, Вертикальное). Методики для определения отсутствия направления, горизонтального и вертикального, как и методики определения активности, более подробно будут объяснены ниже. Конкретный пример с фиг. 4В демонстрирует шесть фильтров (то есть Фильтр 1, Фильтр 2 ... Фильтр 6), но большее или меньшее количество фильтров также может использоваться. Как можно увидеть из фиг. 4В, две метрики (активности и направления) создают 15 комбинаций, идентифицированных как комбинации 421-435. В некоторых случаях, однако, дополнительные комбинации, не показанные детально на фиг. 4В, также могут использоваться. Например, комбинация, соответствующая отсутствию активности, может быть 16-й комбинацией, которая также имеет соответствующий фильтр.

[0098] Модуль 349 фильтра может хранить соответствие фильтров комбинациям диапазонов двух или более метрик, как то примерные соответствия с фиг. 4А и фиг. 4В, и использовать построение соответствий, чтобы определить, какой фильтр из набора фильтров применять к конкретному пикселю или группе пикселей в СУ. Соответствие фильтров комбинациям диапазонов двух или более метрик может, к примеру, определяться модулем 349 фильтра как часть процесса выбора фильтра, описанного выше. Вне зависимости от того, как определяется соответствие, модуль 349 фильтра может генерировать информацию, позволяющую декодеру реконструировать соответствие. Эта информация может быть включена в кодированный битовый поток, чтобы сигнализировать соответствие комбинаций диапазонов фильтрам. Сигнализируемое соответствие комбинаций диапазонов фильтра может приводить в соответствие комбинации диапазонов идентификационным ID фильтра. Действительные коэффициенты для конкретного фильтра могут сигнализироваться отдельно.

[0099] Чтобы генерировать такую информацию, модуль 349 фильтра сначала определяет порядок передачи для комбинаций. Порядок передачи в общем относится к порядку, в котором фильтры будут сигнализированы для комбинаций диапазонов. Используя фиг. 4А в качестве примера, модуль 349 фильтра может использовать порядок передачи слева-направо, сверху-вниз, где фильтр для комбинации 401 сигнализируется первым, фильтр для комбинации 402 сигнализируется вторым, и оставшиеся комбинации сигнализируются в порядке 403=>404=>405=>406=>407=>408=>409=>410=>411=>412=>413=>414=>415=>416. Модуль 349 фильтра может также использовать порядок передачи сверху-вниз, зигзагообразно, где фильтры для комбинаций сигнализируются

в порядке 401=>402=>403=>404=>408=> 407=>406=>405=>409=>410=>411=>412=>416=>415=>414=>413. Модуль 349 фильтра может также использовать порядок передачи сверху-вниз, слева-направо, где фильтры для комбинаций сигнализируются в порядке 401=>405=>409=>413=>402=>406=>410=>414=>403=>407=>411=>415=>404=>408=>412=>416. Модуль 349 фильтра может также использовать порядок передачи слева-направо, зигзагообразно, где фильтры для комбинаций сигнализируются в порядке 401=>405=>409=>413=>414=>410=>406=>402=>403=>407=>411=>415=>416=>412=>408=>404. Со ссылкой на фиг. 4В модуль 349 фильтра может использовать порядок передачи слева-направо, снизу-вверх так, что порядок сигнализации представляет собой 421=>422=>423=>424=>425=>426=> 427=>428=>429=>430=>431=>432=>433=>434=>435. Как можно представить, это лишь некоторые из множества порядков передачи, которые являются возможными.

[0100] В соответствии с методикой этого раскрытия модуль 349 фильтра может использовать серию кодовых слов, чтобы сигнализировать соответствие декодеру. Например, модуль 349 фильтра может генерировать первое кодовое слово для индикации того, если текущая декодируемая комбинация соответствует тому же фильтру, что и самая последняя декодированная комбинация, которая совместно использует тот же диапазон для первой метрики. Если текущая декодируемая комбинация соответствует тому же фильтру, что и самая последняя декодированная комбинация, которая совместно использует тот же диапазон для второй метрики, то модуль 349 фильтра может генерировать второе кодовое слово вместо первого кодового слова. Если текущая декодируемая комбинация не соответствует тому же фильтру, что и любая из этих самых последних декодированных комбинаций, то модуль 349 фильтра может генерировать третье кодовое слово вместо первого кодового слова и второго кодового слова, которое указывает фильтр, соответствующий текущей декодируемой комбинации. Первое кодовое слово и второе кодовое слово из настоящего примера могут быть относительно короткими по сравнению с третьим кодовым словом. Например, первое кодовое слово и второе кодовое слово могли бы представлять собой два бита каждое (например, 00 и 01 соответственно), тогда как третье кодовое слово - это больше битов (первый бит 1, плюс дополнительные биты). В этом конкретном контексте текущая декодируемая комбинация или предыдущая декодируемая комбинация относится к части процессов кодирования и декодирования, где построение соответствия фильтров комбинациям диапазонов сигнализируется кодером или реконструируется декодером, и не обязательно к передаче или декодированию самой комбинации.

[0101] Примеры методик, описанные выше, теперь будут представлены со ссылкой на фиг. 4А и порядок передачи сверху-вниз, слева-направо. Если, к примеру, комбинация 407 - это комбинация, декодируемая в текущий момент, то комбинация 406 - это самая последняя декодированная комбинация, которая совместно использует тот же диапазон для Метрики 1, и комбинация 403 - это самая последняя декодированная комбинация, которая совместно использует тот же диапазон для Метрики 2. Если комбинация 407 соответствует тому же фильтру (Фильтр 7 на фиг. 4А), что и самая последняя декодированная комбинация, которая совместно использует тот же диапазон для второй метрики (то есть Диапазон 2-3 для Метрики 2), модуль 349 фильтра может передавать второе кодовое слово (например, 01) для индикации того, что текущая декодируемая комбинация (комбинация 407) соответствует тому же фильтру, что и самая последняя декодированная комбинация, которая совместно использует тот же диапазон для второй метрики (комбинация 403).

[0102] Если, например, комбинация 410 является текущей декодируемой комбинацией,

то комбинация 409 - это самая последняя декодированная комбинация, которая совместно использует тот же диапазон для Метрики 1, и комбинация 406 - это самая последняя декодированная комбинация, которая совместно использует тот же диапазон для Метрики 2. Если комбинация 410 соответствует тому же фильтру (Фильтр 2 на фиг. 4А), что и самая последняя декодированная комбинация, которая совместно использует тот же диапазон для первой метрики (то есть Диапазон 1-2 для Метрики 1), модуль 349 фильтра может передавать первое кодовое слово (например, 00) для индикации того, что текущая декодируемая комбинация (комбинация 410) соответствует тому же фильтру (Фильтр 2), что и самая последняя декодированная комбинация, которая использует тот же диапазон для первой метрики (комбинация 409).

[0103] Если, например, комбинация 411 является текущей декодируемой комбинацией, то комбинация 410 - это самая последняя декодированная комбинация, которая совместно использует тот же диапазон для Метрики 1, и комбинация 407 - это самая последняя декодированная комбинация, которая совместно использует тот же диапазон для Метрики 2. Если комбинация 411 не соответствует тому же фильтру, что и любая из комбинации 410 и комбинации 407, модуль 349 фильтра может передавать третье кодовое слово (например, 1 + дополнительные биты) для индикации того, что текущая декодируемая комбинация (комбинация 411) соответствует другому фильтру (Фильтр 3), нежели обе - и самая последняя декодированная комбинация, которая совместно использует тот же диапазон для Метрики 1, и самая последняя декодированная комбинация, которая совместно использует тот же диапазон для Метрики 2.

[0104] Для тех текущих комбинаций, где комбинация, которая совместно использует тот же диапазон для Метрики 1, или комбинация, которая совместно использует тот же диапазон для Метрики 2, еще не были декодированы, те опции могут или рассматриваться как недоступные или могут быть заменены другой комбинацией. Если, например, комбинация 409 - это текущая комбинация, подлежащая декодированию, то комбинация 405 - это самая последняя декодированная комбинация, которая совместно использует тот же диапазон для Метрики 2, но никакая комбинация, которая совместно использует диапазон для Метрики 1, еще не была декодирована. В таких случаях самая последняя декодированная комбинация, которая совместно использует тот же диапазон для Метрики 1, может считаться не соответствующей тому же фильтру, что и текущая комбинация, подлежащая декодированию. Таким образом, в этом случае первое кодовое слово не будет использоваться для комбинации 409. В качестве альтернативы комбинация, которая совместно использует диапазон для Метрики 1, может быть заменена другой комбинацией, такой как самая последняя декодированная комбинация или другая прежде декодированная комбинация. В таком случае самая последняя декодированная комбинация до комбинации 409 могла бы быть комбинацией 408. Таким образом, если комбинация 408 соответствует тому же фильтру, что и комбинация 409, модуль 349 фильтра может генерировать первое кодовое слово. Аналогичные методики могут использоваться для тех комбинаций, где предыдущая комбинация, совместно использующая общий диапазон для Метрики 1, еще не была декодирована.

[0105] Для первой комбинации в порядке передачи (например, комбинации 401 в примере с фиг. 4А), где ни комбинация, которая совместно использует тот же диапазон для Метрики 1, ни комбинация, которая совместно использует тот же диапазон для Метрики 2, не были декодированы, модуль 349 фильтра может генерировать кодовое слово, указывающее фильтр, который соответствует первой комбинации. Фильтр может быть, например, сигнализирован с использованием третьего кодового слова или может быть сигнализирован с использованием другой методики, в этом случае методики,

описанные в этом раскрытии, могли бы начаться со второй комбинации в порядке передачи или более поздней комбинации.

[0106] В соответствии с другой методикой этого раскрытия, модуль 349 фильтра может использовать серию кодовых слов, чтобы сигнализировать соответствию декодеру.

5 В некоторых вариантах осуществления модуль 349 фильтра может генерировать первое кодовое слово для индикации того, если текущая декодируемая комбинация соответствует тому же фильтру, что и самая последняя декодированная комбинация, которая совместно использует тот же диапазон для первой метрики. Если текущая декодируемая комбинация не соответствует тому же фильтру, что самая последняя

10 декодированная комбинация, которая совместно использует тот же диапазон для первой метрики, то модуль 349 фильтра может генерировать второе кодовое слово, вместо первого кодового слова, которое указывает фильтр, который соответствует текущей декодируемой комбинации. Например, первое кодовое слово может представлять собой один бит (например, 0), тогда как второе кодовое слово - это больше битов (например,

15 первый бит 1, плюс дополнительные биты). В отличие от предыдущей методики, где короткое кодовое слово могло бы генерироваться, если текущая комбинация соответствует тому же фильтру, что и прежде декодированная комбинация, которая совместно использует тот же диапазон для любой из Метрики 1 или Метрики 2, эта методика включает в себя только генерирование короткого кодового слова, если

20 текущая комбинация соответствует тому же фильтру, что и прежде декодированная комбинация, которая совместно использует тот же диапазон для Метрики 1. Таким образом, даже если текущая комбинация соответствует тому же фильтру, что и прежде декодированная комбинация, которая совместно использует тот же диапазон для Метрики 2, модуль 349 фильтра все еще генерирует второе кодовое слово (например,

25 1 + дополнительные биты). Хотя это раскрытие использует Метрику 1 в целях объяснения, те же методики также могут применяться с использованием только Метрики 2.

[0107] В соответствии еще с одной методикой этого раскрытия модуль 349 фильтра может использовать различные серии кодовых слов, чтобы сигнализировать

30 соответствию к декодеру. Например, модуль 349 фильтра может генерировать первое кодовое слово для индикации того, если текущая декодируемая комбинация соответствует тому же фильтру, что и самая последняя декодированная комбинация, независимо от того, какой диапазон, если он был, был для текущей комбинации общим с прежде декодированной комбинацией. Если текущая декодируемая комбинация не

35 соответствует тому же фильтру, что и самая последняя декодированная комбинация, то модуль 349 фильтра может генерировать второе кодовое слово, идентифицирующее фильтр, который соответствует текущей комбинации. В этом конкретном варианте осуществления первое слово может быть относительно коротким по сравнению со вторым кодовым словом. Например, первое кодовое слово может представлять собой

40 один бит (например, 0), тогда как второе кодовое слово - это больше битов (например, первый бит 1, плюс дополнительные биты).

[0108] Снова стоит отметить, что при использовании примера с фиг. 4А и порядка передачи сверху-вниз, слева-направо комбинация 401 была бы самой последней декодированной комбинацией, если комбинация 402 декодируется в текущий момент,

45 комбинация 402 была бы самой последней декодированной комбинацией, если комбинация 403 - текущая комбинация, и так далее. 404 была бы самой последней декодированной комбинацией, если комбинация 405 - это комбинация, которая декодируется в текущий момент. Так, модуль 349 фильтра может генерировать первое

кодовое слово, если комбинация 402 соответствует тому же фильтру, что и комбинация 401, если комбинация 403 соответствует тому же фильтру, что и комбинация 402 и т.д. В противном случае модуль 349 фильтра может генерировать второе кодовое слово, идентифицирующее фильтр, который соответствует текущей комбинации.

5 [0109] В соответствии еще с одной методикой этого раскрытия модуль 349 фильтра может использовать два кодовых слова, чтобы сигнализировать соответствие фильтров комбинациям. Первое кодовое слово, такое как «0», может использоваться для
10 сигнализации того, что текущая комбинация использует тот же фильтр, что и предыдущая комбинация. Второе кодовое слово, такое как «1», может использоваться для сигнализации того, что текущая комбинация имеет другой фильтр, нежели
15 предыдущая комбинация. Второе кодовое слово, однако, не имеет надобности идентифицировать новый фильтр. Вместо этого новый фильтр может быть определен на основе порядка передачи для классов и порядка, при котором передаются
коэффициенты фильтра. При использовании порядка передачи слева-направо, снизу-
20 вверх, описанного выше для фиг. 4В в качестве примера, кодовые слова могли бы передаваться соответственно: 421 (0)=>422 (0)=>423 (1)=>424 (0)=>425 (0)=>426 (0)=>427
(0)=>428 (1)=>429 (0)=>430 (0)=>431 (0)=>432 (1)=>433 (0)=>434 (0)=>435 (0), с числом в
круглых скобках, представляющим кодовое слово для этой комбинации. В этом примере
25 комбинации 421-422 были бы приведены в соответствие первому фильтру, комбинации 423-427 - второму фильтру, комбинации 428-431 - третьему фильтру, а комбинации 432-435 - четвертому фильтру. Коэффициенты для первого фильтра, второго фильтра, третьего фильтра и четвертого фильтра могут соответствовать порядку, при котором
наборы коэффициентов фильтра сигнализируются, где сигнализированный первый
набор коэффициентов фильтра соответствует первому фильтру, сигнализированный
30 второй набор коэффициентов фильтра соответствует второму фильтру и т.д. Определение порядка для передачи наборов коэффициентов фильтра рассматривается ниже более подробно.

[0110] Различные методики, описанные в этом раскрытии для сигнализации
соответствия фильтров комбинациям диапазонов, не являются взаимоисключающими
35 альтернативами, а наоборот, могут использоваться совместно друг с другом. Например, в некоторых вариантах осуществления определенные комбинации могли бы
сигнализироваться с использованием первой методики, тогда как другие комбинации
сигнализируются с использованием второй методики. В качестве одного примера, где
одна из комбинации, которая совместно использует тот же диапазон для Метрики 1,
35 или комбинации, которая совместно использует тот же диапазон для Метрики 2, еще не была декодирована (например, комбинации 402, 403, 404, 405, 409 и 413), модуль 349
фильтра может использовать первую методику. Там, где и комбинация, которая
совместно использует тот же диапазон для Метрики 1, и комбинация, которая совместно
использует тот же диапазон для Метрики 2, были декодированы (например, комбинации
40 406, 407, 408, 410, 411, 412, 414, 415 и 416), могла бы использоваться вторая методика. Дополнительно кодовые слова, использованные для любого из первого, второго и
третьего кодовых слов, описанных выше, могут быть любыми из кодовых слов
фиксированной длины, переменных кодовых слов или контекстно-адаптивных
переменных кодовых слов.

45 [0111] В дополнение к генерированию информации, позволяющей декодеру
реконструировать соответствие фильтров комбинациям диапазонов, модуль 349 фильтра
также генерирует информацию, позволяющую декодеру реконструировать сами
фильтры. Реконструкция фильтров включает в себя реконструкцию коэффициентов

фильтра этих фильтров. Как будет описано более подробно ниже, модуль 349 фильтра может использовать методики разностного кодирования, чтобы сигнализировать коэффициенты фильтра. Чтобы использовать методики разностного кодирования, модуль 349 фильтра определяет порядок, в котором сигнализировать наборы

5 коэффициентов фильтра.

[0112] В качестве части определения порядка модуль 349 фильтра определяет идентификацию комбинации (ID), которая представляет порядковое значение для каждой комбинации диапазонов. При использовании фиг. 4А в качестве примера комбинациям можно было бы назначить ID комбинации, которые представляют

10 порядковые значения в порядке слева-направо, сверху-вниз, в этом случае комбинация 401 получила бы первое порядковое значение, комбинация 402 получила бы второе порядковое значение, а оставшиеся комбинации получили бы порядковые значения в порядке 403=>404=>405=>406=>407=>408=>409 =>410=>411=>412=>413=>414=>415=>416. Модуль 349 фильтра также может назначить ID комбинации, используя порядок

15 сверху-вниз, зигзагообразно, где комбинации получили бы ID комбинации с порядковыми значениями в порядке 401=>402=>403=>404=>408=>407=> 406=>405=>409=>410=>411=>412=>416=>415=>414=>413. Модуль 349 фильтра также может назначить ID комбинации, используя порядок сверху-вниз, слева-направо, где комбинации получили бы ID комбинации с порядковыми значениями в порядке 401=>405=>409=>413 =>402=>406=>410=>414=>403=>407=>411=>415=>404=>408=>412=>416. Модуль 349 фильтра также может назначить ID комбинации, используя порядок слева-направо, зигзагообразно, где комбинации получили бы ID комбинации с порядковыми значениями в порядке 401 =>405=>409=>413=>414=>410=>406=>402=>403=>407=>411=>415=>416=>412=>408=>404. Как можно представить, это только некоторые из множества порядков,

20 которые могли бы использоваться. Кроме того, любой из описанных порядков, может быть и от самого низкого к самому высокому, и от самого высокого к самому низкому.

[0113] После того как модуль 349 фильтра определил соответствие фильтров комбинациям диапазонов, модуль 349 фильтра может идентифицировать группирование комбинаций диапазонов, которые приведены в соответствие одному и тому же фильтру.

30 С использованием фиг. 4А в качестве примера группирования были бы следующими.

Группа Фильтра 1: комбинации 413, 414 и 415

Группа Фильтра 2: комбинации 409, 410

Группа Фильтра 3: комбинации 411 и 412

Группа Фильтра 4: комбинация 416

35 Группа Фильтра 5: комбинации 401 и 405

Группа Фильтра 6: комбинации 402 и 406

Группа Фильтра 7: комбинации 403 и 407

Группа Фильтра 8: комбинации 404 и 408.

[0114] Модуль 349 фильтра затем может назначить каждой группе ID группы, и ID группы может представлять порядковое значение. ID группы может назначаться группам на основе порядковых значений, связанных с комбинациями, которые содержат группу. Например, группа, которая имеет комбинацию с самым низким соответствующим порядковым значением, на основе ID комбинации могла бы получить ID группы с самым низким порядковым значением. Из оставшихся групп оставшаяся

40 группа, группа, которая имеет комбинацию с самым низким соответствующим порядковым значением, могла бы получить ID группы со следующим самым низким порядковым значением. Это процесс может повторяться до тех пор, пока всем группам не будет назначен ID группы. В некоторых вариантах осуществления ID группы может

назначаться на основе комбинаций с самыми высокими соответствующими порядковыми значениями, нежели самыми низкими. В некоторых вариантах осуществления группа, которая имеет комбинацию с самым низким соответствующим порядковым значением, на основе ID комбинации могла бы получить ID группы с самым высоким порядковым значением или наоборот.

[0115] Снова стоит упомянуть, что используя фиг. 4А в качестве примера и допуская, что комбинациям 401-416 присвоены ID комбинации с порядковыми значениями в порядке слева-направо, сверху-вниз, модуль 349 фильтра может назначать ID группы группам фильтров, как показано ниже в Таблице 1.

Таблица 1

Имя группы	Комбинации в группе	Комбинация с самым низким порядковым значением	ID группы
Группа Фильтра 1	413, 414, 415	413	7
Группа Фильтра 2	409, 410	409	5
Группа Фильтра 3	411, 412	411	6
Группа Фильтра 4	416	416	8
Группа Фильтра 5	401, 405	401	1
Группа Фильтра 6	402, 406	402	2
Группа Фильтра 7	403, 407	403	3
Группа Фильтра 8	404, 408	404	4

[0116] В примере с фиг. 4А, показанном в Таблице 1, модуль 349 фильтра назначает Группе Фильтра 5 ID группы с самым низким порядковым значением, поскольку Группа Фильтра 5 включает в себя комбинацию диапазонов с самым низким порядковым значением (то есть комбинацию 401). Модуль 349 фильтра назначает Группе Фильтра 6 ID группы со вторым самым низким порядковым значением, поскольку из оставшихся групп фильтров (то есть всех групп за исключением Группы Фильтра 5) Группа Фильтра 6 включает в себя комбинацию диапазонов со вторым самым низким порядковым значением (то есть комбинацию 402). Модуль 349 фильтра назначает Группе Фильтра 7 ID группы с третьим самым низким порядковым значением, поскольку из оставшихся групп фильтров (то есть всех групп за исключением Группы Фильтра 5 и Группы Фильтра 6) Группа Фильтра 7 включает в себя комбинацию диапазонов с самым низким порядковым значением (то есть комбинацию 403). Модуль 349 фильтра назначает Группе Фильтра 8 ID группы с четвертым самым низким порядковым значением, поскольку из оставшихся групп фильтров (то есть всех групп за исключением Группы Фильтра 5, Группы Фильтра 6 и Группы Фильтра 7) Группа Фильтра 8 включает в себя комбинацию диапазонов с четвертым самым низким порядковым значением (то есть комбинацию 404). Модуль 349 фильтра назначает Группе Фильтра 2 ID группы с пятым самым низким порядковым значением, поскольку из оставшихся групп фильтров (то есть всех групп за исключением Группы Фильтра 5, Группы Фильтра 6, Группы Фильтра 7 и Группы Фильтра 8) Группа Фильтра 2 включает в себя комбинацию диапазонов с самым низким порядковым значением (то есть комбинацию 409). Модуль 349 фильтра назначает Группе Фильтра 3 ID группы с шестым самым низким порядковым значением, поскольку из оставшихся групп фильтров (то есть всех групп за исключением Группы Фильтра 5, Группы Фильтра 6, Группы Фильтра 7, Группы Фильтра 8 и Группы Фильтра 2) Группа Фильтра 3 включает в себя комбинацию диапазонов с самым низким порядковым значением (то есть комбинацию 411). Модуль 349 фильтра назначает Группе Фильтра 1 ID группы с седьмым самым низким порядковым значением, поскольку из оставшихся групп фильтров (то есть всех групп за исключением Группы Фильтра 5, Группы Фильтра 6, Группы Фильтра 7, Группы Фильтра 8, Группы Фильтра

2 и Группы Фильтра 3) Группа Фильтра 1 включает в себя комбинацию диапазонов с самым низким порядковым значением (то есть комбинацию 413). Наконец, модуль 349 фильтра назначает Группе Фильтра 4, последней оставшейся группе фильтров, ID группы с самым высоким порядковым значением (8 в конкретном примере).

5 [0117] На основе ID группы фильтров модуль 349 фильтра определяет порядок, в котором сигнализировать коэффициенты фильтра выбранного фильтра. Снова, при использовании примера с фиг. 4А и Таблицы 1 модуль 349 фильтра сначала
10 сигнализирует коэффициент для Фильтра 5, затем коэффициент для Фильтра 6, после этого коэффициент для Фильтра 7, затем коэффициент для Фильтра 8, после этого коэффициент для Фильтра 2, затем коэффициент для Фильтра 3, после этого коэффициент для Фильтра 1 и, наконец, коэффициент для Фильтра 4. Используя методики разностного кодирования, как описывается в данном раскрытии, модуль 349 фильтра может
15 кодировать коэффициенты для Фильтра 6 как информацию разности относительно коэффициентов фильтра от Фильтра 5, кодировать коэффициенты для Фильтра 7 как информацию разности относительно коэффициентов фильтра от Фильтра 6 и так далее, на основе упорядочивания ID группы.

[0118] Приведение в соответствие двух или более метрик для вводов фильтрам может реализовываться несколькими способами. Например, в некоторых вариантах осуществления каждый ввод может быть иметь уникальный набор фильтров, тогда как
20 в некоторых вариантах осуществления вводы совместно используют общий набор фильтров. Дополнительно в некоторых вариантах осуществления две или более метрик для каждого ввода могли бы использоваться, чтобы идентифицировать конкретный фильтр для каждого ввода. В других вариантах осуществления, однако, две или более метрик для единственного ввода могли бы использоваться, чтобы идентифицировать
25 фильтры для всех вводов. В еще других вариантах осуществления две или более метрик для первого ввода могли бы использоваться, чтобы идентифицировать фильтр для второго, другого ввода.

[0119] В соответствии с этим вариантом осуществления модуль 349 фильтра может реализовывать методики кодирования в отношении информации фильтра, что может
30 сократить количество данных, необходимых, чтобы кодировать и переносить информацию фильтра от кодера 350 к другому устройству. Снова отметим, что для каждого кадра или вырезки модуль 349 фильтра может определять или выбирать один или более наборов коэффициентов фильтра, подлежащих применению к пикселям разных CU для этого кадра или вырезки. Модуль 349 фильтра применяет коэффициенты
35 фильтра с целью отфильтровать видеоблоки реконструированных видеокадров, хранимых в памяти 334, которые могут использоваться для кодирования с предсказанием, совместимого с внутриконтурной фильтрацией. Модуль 349 фильтра может кодировать коэффициенты фильтра как информацию фильтра, которая пересылается к элементу 346 энтропийного кодирования для включения в кодированный
40 битовый поток.

[0120] Дополнительно методики этого раскрытия могут использовать тот факт, что некоторые из коэффициентов фильтра, определенных или выбранных FSU 353, могут быть очень похожими на другие коэффициенты фильтра, применяемые в отношении пикселей CU другого кадра или вырезки. Один и тот же тип фильтра может применяться
45 для разных кадров и вырезок (например, одинаковая поддержка фильтра), но фильтры могут разными с точки зрения значений коэффициентов фильтра, связанных с различными индексами поддержки фильтра. Соответственно, с целью уменьшить количество данных, необходимых для переноса таких коэффициентов фильтра, модуль

349 фильтра может подвергаться кодированию с предсказанием один или более коэффициентов фильтра, подлежащих использованию для фильтрации на основе коэффициентов фильтра другого СУ, потенциально используя сходства между коэффициентами фильтра. В некоторых случаях, однако, может быть более желательным вариантом кодировать коэффициенты фильтра напрямую, например, без использования какого-либо предсказания. Различные методики, такие как методики, которые задействуют использование метрики активности, чтобы определить, когда кодировать коэффициенты фильтра, используя методики кодирования с предсказанием, или когда кодировать коэффициенты фильтра напрямую, без какого-либо кодирования с предсказанием, могут использоваться для эффективного сообщения коэффициентов фильтра декодеру. Дополнительно симметрия также может налагаться так, что поднабор коэффициентов (например, 5, -2, 10), известная декодеру, может использоваться, чтобы определить полный набор коэффициентов (например, 5, -2, 10, 10, -2, 5). Симметрия может налагаться как для сценария прямого кодирования, так и для сценария кодирования с предсказанием.

[0121] Как было описано выше, видеокодер 350 представляет собой пример видеокодера, выполненного с возможностью определять первую метрику для группы пикселей в пределах блока пикселей, определять вторую метрику для группы пикселей, определять фильтр на основе первой метрики и второй метрики и генерировать отфильтрованное изображение путем применения фильтра к группе пикселей. Видеокодер 350 также представляет собой пример видеокодера, выполненного с возможностью определять первую метрику для блока пикселей, при этом первая метрика определяется на основе сравнения поднабора пикселей в блоке с другими пикселями в блоке; определять вторую метрику для блока пикселей; определять фильтр на основе первой метрики и второй метрики; и генерировать отфильтрованное изображение путем применения фильтра к блоку пикселей.

[0122] Как было описано выше, видеокодер 350 также представляет собой пример видеокодера, выполненного с возможностью определять соответствие комбинаций диапазонов фильтрам, при этом комбинация диапазонов содержит диапазон для первой метрики и диапазон для второй метрики, при этом каждая комбинация диапазонов имеет уникальную идентификацию комбинации диапазонов (ID), при этом каждая уникальная комбинация диапазонов ID соответствует порядковому значению для комбинации диапазонов; назначать уникальный ID группы группам комбинаций диапазонов на основе порядковых значений для комбинаций диапазонов, при этом каждый уникальный ID группы соответствует порядковому значению для группы; и кодировать группы коэффициентов фильтра, соответствующих фильтрам, на основе уникальных ID группы. Видеокодер 350 может кодировать наборы коэффициентов фильтра путем сигнализации наборов коэффициентов фильтра в кодированном битовом потоке в порядке, который выбирается на основе порядковых значений уникальных ID группы. Видеокодер 350 может сигнализировать наборы коэффициентов фильтра с использованием методик разностного кодирования.

[0123] Как было описано выше, видеокодер 350 также представляет собой пример видеокодера, выполненного с возможностью определять соответствие комбинаций диапазонов фильтрам, при этом комбинация диапазонов содержит диапазон значений для первой метрики и диапазон значений для второй метрики; генерировать первое кодовое слово, если текущая комбинация диапазонов приведена в соответствие тому же фильтру, что и предыдущая комбинация диапазонов, которая содержит тот же диапазон значений для первой метрики; генерировать второе кодовое слово, если

текущая комбинация диапазонов приведена в соответствие тому же фильтру, что и предыдущая комбинация диапазонов, которая содержит тот же диапазон значений для второй метрики; и генерировать третье кодовое слово, если текущая комбинация диапазонов приведена в соответствие другому фильтру, нежели предыдущая комбинация диапазонов, которая содержит тот же диапазон значений для первой метрики и предыдущая комбинация диапазонов, которая содержит тот же диапазон значений для второй метрики. Вideoкодер 350 также представляет собой пример видеокодера, выполненного с возможностью определять соответствие комбинаций диапазонов фильтрам, при этом комбинация диапазонов содержит диапазон для первой метрики и диапазон для второй метрики; генерировать первое кодовое слово, если текущая комбинация диапазонов приведена в соответствие тому же фильтру, что и предыдущая комбинация диапазонов; и генерировать второе кодовое слово, если текущая комбинация диапазонов приведена в соответствие другому фильтру, нежели предыдущая комбинация диапазонов, при этом второе кодовое слово идентифицирует фильтр, приведенный в соответствие текущей комбинации диапазонов.

[0124] Фиг. 5 представляет собой блок-схему, иллюстрирующую пример видеодекодера 560, который декодирует видеопоследовательность, которая кодирована способом, описанным здесь. Приятая видеопоследовательность может содержать кодированный набор кадров изображения, набор вырезок кадра, обычно кодированную группу изображений (GOP) или широкое многообразие типов серий видеоблоков, которые включают в себя кодированные видеоблоки и синтаксис, чтобы определить, как декодировать такие видеоблоки.

[0125] Видеодекодер 560 включает в себя модуль 552 энтропийного декодирования, который осуществляет функцию обратного декодирования в отношении кодирования, выполненного элементом 346 энтропийного кодирования с фиг. 3. В частности, модуль 552 энтропийного декодирования может осуществлять декодирование CAVLC и CABAC или любой другой тип энтропийного декодирования, используемый видеокодером 350. Энтропийно декодированные видеоблоки в одномерном преобразованном в последовательную форму формате могут подвергнуться обратному сканированию, чтобы преобразовать один или более одномерных векторов коэффициентов обратно в формат двухмерного блока. Количество и размер векторов, как и порядок сканирования, определенный для видеоблоков, могут определять, как реконструируется двухмерный блок. Энтропийно декодированный синтаксис предсказания может быть отправлен от модуля 552 энтропийного декодирования к модулю 554 предсказания, и информация энтропийно декодированного фильтра может быть отправлена от модуля 552 энтропийного декодирования к модулю 559 фильтра.

[0126] Видеодекодер 560 также включает в себя модуль 554 предсказания, модуль 556 обратного квантования, модуль 558 обратного преобразования, запоминающее устройство и сумматор 564. В дополнение к этому видеодекодер 560 также включает в себя фильтр 557 удаления блочности, который фильтрует вывод сумматора 564. Совместимый с этим раскрытием, модуль 559 фильтра может принимать энтропийно декодированную информацию фильтра, которая включает в себя один или более фильтров, подлежащих применению к одному или более вводам. Хотя это и не показано на фиг. 5, фильтр 557 удаления блочности также может принимать энтропийно декодированную информацию фильтра, которая включает в себя один или более фильтров, подлежащих применению.

[0127] Фильтры, применяемые модулем 559 фильтра, могут определяться наборами коэффициентов фильтра. Модуль 559 фильтра может быть выполнен с возможностью

генерировать наборы коэффициентов фильтра на основе информации фильтра, принятой от модуля 552 энтропийного декодирования. Информация фильтра может включать в себя синтаксис описания фильтра, который идентифицирует максимальное количество фильтров в наборе фильтров и/или форму фильтров в наборе фильтров, например.

5 Синтаксис описания фильтра может быть включен в заголовок серии видеоблоков, например, заголовок LCU, заголовок кадра, заголовок вырезки, заголовок GOP, заголовок последовательности и т.п. В других примерах синтаксис описания фильтра может быть включен в нижний колонтитул или другую структуру данных. На основе синтаксиса описания фильтра модуль 559 фильтра может реконструировать набор
10 фильтров, использованный в кодере.

[0128] Информация фильтра также может включать в себя дополнительный синтаксис сигнализирования, который сигнализирует декодеру способ кодирования, использованный для любого заданного набора коэффициентов. В некоторых вариантах осуществления информация фильтра может, например, также включать в себя диапазоны
15 для двух или более метрик, для которых должна использоваться любой заданный набор коэффициентов. После декодирования фильтров модуль 559 фильтра может фильтровать значения пикселей декодированных видеоблоков на основе одного или более наборов коэффициентов фильтра и синтаксиса сигнализирования, который включает в себя диапазоны, для которых должны использоваться различные наборы коэффициентов
20 фильтра.

[0129] Модуль 559 фильтра может принимать в битовом потоке один или более синтаксических компонентов, указывающих набор фильтров для каждого кадра или вырезки, как и соответствие фильтров двум или более метрикам. Например, если кодер использует соответствие диапазонов для метрик фильтрам, что показано на фиг. 4А,
25 кодер будет или сигнализировать это соответствие, или передавать данные, чтобы позволить модулю 559 фильтра реконструировать это соответствие. Вне зависимости от того, ясно ли сообщается это соответствие, модуль 559 фильтра может поддерживать то же соответствие фильтров комбинациям диапазонов, что и использовалось кодером.

[0130] Как было упомянуто выше, модуль 559 фильтра генерирует соответствие на
30 основе информации фильтра, сигнализированной в битовом потоке. На основе этого соответствия модуль 559 фильтра может определять группы и назначать ID группы таким же образом, как описывалось выше в отношении модуля 349 фильтра. Используя эти ID группы, модуль 559 фильтра может связывать принятые коэффициенты фильтра с...

35 [0131] Для каждого CU в пределах кадра или вырезки модуль 559 фильтра может вычислять одну или более метрик, связанных с декодированными пикселями CU для множественных вводов (то есть PI, EI, pRI и RI), чтобы определить, какой фильтр (какие фильтры) из набора (наборов) применять к каждому вводу. В качестве альтернативы модуль 559 фильтра может вычислять одну или более метрик для единственного ввода,
40 такого как pRI или RI. Модуль 559 фильтра определяет, какой фильтр применять, на основе метрик, определенных для конкретного пикселя или группы пикселей. При использовании значения Лапласиана с измененной суммой и направления как примеров для Метрики 1 и Метрики 2 и при использовании соответствий, показанных на фиг. 4А, в качестве примера, если модуль 559 фильтра определяет, что пиксель или группа
45 пикселей имеет значение Лапласиана с измененной суммой в диапазоне 1-2 и направление, соответствующее Диапазону 2-3, модуль 559 фильтра может применять Фильтр 2 к этому пикселю или группе пикселей. Если модуль 559 фильтра определяет, что пиксель или группа пикселей имеет значение Лапласиана с измененной суммой в

Диапазоне 1-4 и направление, соответствующе Диапазону 2-2, модуль 559 фильтра может применять Фильтр 6 к этому пикселю или группе пикселей и так далее. Фильтр может, в общем, допускать любой тип формы или распределения поддержки фильтра. Поддержка фильтра относится к форме фильтра касательно заданного пикселя, подвергаемого фильтрации, и коэффициенты фильтра могут определять взвешивание, применяемое к значениям соседних пикселей в соответствии с поддержкой фильтра. В соответствии с методиками этого раскрытия данные синтаксиса могут быть включены в битовый поток, чтобы сигнализировать декодеру, как были кодированы фильтры (например, как были кодированы коэффициенты фильтра), как и диапазоны метрики активности, для которых должны использоваться различные фильтры.

[0132] Для каждого CU в пределах кадра или вырезки модуль 559 фильтра может вычислять одну или более метрик, связанных с декодированными пикселями CU для множественных вводов (то есть PI, EI, pRI и RI), чтобы определить, какой фильтр (какие фильтры) из набора (наборов) применять к каждому вводу. В качестве альтернативы модуль 559 фильтра может вычислять одну или более метрик для единственного ввода, такого как pRI или RI. В качестве альтернативы модуль 559 фильтра может вычислять одну или более метрик для единственного ввода, такого как pRI или RI. Модуль 559 фильтра определяет, какой фильтр применять, на основе метрик, определенных для конкретного пикселя или группы пикселей. При использовании значения Лапласиана с измененной суммой и направления как примеров для Метрики 1 и Метрики 2 и при использовании соответствий, показанных на фиг. 4А, в качестве примера, если модуль 559 фильтра определяет, что пиксель или группа пикселей имеет значение Лапласиана с измененной суммой в диапазоне 1-2 и направление, соответствующе Диапазону 2-3, модуль 559 фильтра может применять Фильтр 2 к этому пикселю или группе пикселей. Если модуль 559 фильтра определяет, что пиксель или группа пикселей имеет значение Лапласиана с измененной суммой в Диапазоне 1-4 и направление, соответствующе Диапазону 2-2, модуль 559 фильтра может применять Фильтр 6 к этому пикселю или группе пикселей и так далее. Фильтр может, в общем, допускать любой тип формы или распределения поддержки фильтра. Поддержка фильтра относится к форме фильтра касательно заданного пикселя, подвергаемого фильтрации, и коэффициенты фильтра могут определять взвешивание, применяемое к значениям соседних пикселей в соответствии с поддержкой фильтра. В соответствии с методиками этого раскрытия данные синтаксиса могут быть включены в битовый поток, чтобы сигнализировать декодеру, как были кодированы фильтры (например, как были кодированы коэффициенты фильтра), а также диапазоны метрики активности, для которых должны использоваться различные фильтры.

[0133] Модуль 554 предсказания принимает синтаксис предсказания (такой как векторы движения) от модуля 552 энтропийного кодирования. Используя синтаксис предсказания, модуль 554 предсказания генерирует блоки предсказания, которые были использованы, чтобы кодировать видеоблоки. Модуль 556 обратного квантования осуществляет обратное квантование, а модуль 558 обратного преобразования осуществляет обратное преобразование, чтобы изменить коэффициенты остаточных видеоблоков обратно в область пикселя. Сумматор 564 комбинирует каждый блок предсказания с соответствующим остаточным блоком, выведенным модулем 558 обратного преобразования, чтобы реконструировать видеоблок.

[0134] Модуль 559 фильтра генерирует коэффициенты фильтра, подлежащие применению для каждого ввода CU, и затем применяет такие коэффициенты фильтра, чтобы отфильтровать реконструированные видеоблоки этого CU. Фильтрация, к

примеру, может содержать дополнительную деблокирующую фильтрацию, которая сглаживает края и/или устраняет артефакты, связанные с видеоблоками, фильтрацию с понижением уровня шума, чтобы уменьшить шум квантования, или любой другой тип фильтрации, который может улучшить качество кодирования. Отфильтрованные видеоблоки аккумулируются в памяти 562, чтобы реконструировать декодированные кадры (или другие декодируемые элементы) видеoinформации. Декодированные элементы могут быть выведены от видеodeкодера 560 для представления пользователю, но также могут храниться для использования при последующем декодировании с предсказанием.

[0135] В области кодирования видео принято применять фильтрацию в кодере и декодере с целью улучшить качество декодированного видеосигнала. Фильтрация может применяться посредством пост-фильтра, в этом случае отфильтрованный кадр не используется для предсказания будущих кадров. В качестве альтернативы фильтрация может применяться «внутриконтурно», в этом случае отфильтрованный кадр может использоваться для предсказания будущих кадров. Желаемый фильтр может быть создан путем минимизации ошибки между оригинальным сигналом и декодированным отфильтрованным сигналом. Обычно такая фильтрация основывалась на применении одного или более фильтров к реконструированному изображению. Например, деблокирующий фильтр может применяться к реконструированному изображению до того, как это изображение сохраняется в запоминающем устройстве, или деблокирующий фильтр и один дополнительный фильтр могли бы применяться к реконструированному изображению до того, как это изображение сохраняется в запоминающем устройстве.

[0136] Подобно квантованию коэффициентов преобразования, коэффициенты фильтра $h(k, l)$, где $k = -K, \dots, K$, и $l = -L, \dots, L$, также могут быть квантованы. K и L могут представлять собой целые значения. Коэффициенты фильтра $h(k, l)$ могут быть квантованы как:

$$f(k, l) = \text{round}(\text{normFact} \cdot h(k, l)),$$

где normFact - фактор нормализации и round - операция округления, выполняемая, чтобы достичь квантования до желаемой битовой глубины. Квантование коэффициентов фильтра может осуществляться модулем 349 фильтра с фиг. 3 во время кодирования, а деквантование или обратное квантование может осуществляться в отношении декодированных коэффициентов фильтра модулем 559 фильтра с фиг. 5. Фильтр $h(k, l)$ предназначен для того, чтобы в общем представлять любой фильтр. Например, фильтр $h(k, l)$ может применяться к любому из множественных вводов. В некоторых случаях множественные вводы, связанные с видеоблоком, будут использовать разные фильтры, в этом случае множественные фильтры, подобные $h(k, l)$, могут быть квантованы и деквантованы, как описывается выше.

[0137] Квантованные коэффициенты фильтра кодируются и отправляются от устройства-источника, связанного с кодером 350, к устройству-адресату, связанному с декодером 560, как часть кодированного битового потока. В примере, представленном выше, значение normFact обычно равно 2^n , хотя другие значения также могли бы быть использованы. Более большие значения normFact приводят к более точному квантованию, так что квантованные коэффициенты фильтра $f(k, l)$ обеспечивают более высокую эффективность. Однако более большие значения normFact могут создать коэффициенты $f(k, l)$, которые требуют большего количества битов для сигнализирования декодеру.

[0138] В декодере 560 декодированные коэффициенты фильтра $f(k, l)$ могут применяться к подходящему вводу. Например, если декодированные коэффициенты

фильтра подлежат использованию в отношении RI, коэффициенты фильтра могут применяться к пост-деблокированному реконструированному изображению RI(i, j), где i=0,..., M и j=0,..., N следующим образом:

$$\tilde{RI}(i, j) = \sum_{k=-K}^K \sum_{l=-L}^L f(k, l) RI(i+k, j+l) / \sum_{k=-K}^K \sum_{l=-L}^L f(k, l)$$

Переменные M, N, K и L могут представлять собой целые числа. K и L могут определять блок пикселей, который захватывает два измерения от -K к K и от -L к L. Фильтры, применяемые к другим вводам, могут применяться аналогичным образом.

[0139] Методики этого раскрытия могут повышать эффективность пост-фильтра или внутриконтурного фильтра, а также могут сокращать количество битов, необходимых, чтобы сигнализировать коэффициенты фильтра f(k, l). В некоторых случаях некоторое количество различных пост-фильтров или внутрициклических фильтров сигнализируется декодеру для каждой серии видеоблоков, например, для каждого кадра, вырезки, части кадра, группы кадров (GOP) и т.п. Для каждого фильтра дополнительная информация включена в битовый поток, чтобы идентифицировать CU, макроблоки и/или пиксели, для которых должен применяться заданный фильтр.

[0140] Кадры могут быть идентифицированы по номеру кадра и/или типу кадра (например, I-кадры, P-кадры или B-кадры). I-кадры относятся к внутренним кадрам, которые подверглись внутреннему предсказанию. P-кадры относятся к предсказанным кадрам, которые содержат видеоблоки, предсказанные на основе списка данных (например, один предшествующий кадр). B-кадры относятся к двунаправленным предсказанным кадрам, которые предсказаны на основе двух списков данных (например, предшествующий и следующий кадр). Макроблоки могут быть идентифицированы по типам макроблока и/или значениям параметра диапазона квантования (QP), используемым для реконструкции макроблока.

[0141] Коэффициенты фильтра f(k, l) для любого ввода могут быть кодированы с использованием предсказания от коэффициентов, сигнализированных для предыдущих CU. Для каждого ввода CU m (например, каждый кадр, вырезка или GOP) кодер может кодировать и передавать набор M-фильтров:

g_i^m , где i=0,..., M-1.

Для каждого фильтра битовый поток также может кодироваться, чтобы идентифицировать комбинацию диапазонов для двух или более метрик, для которых должен использоваться фильтр.

[0142] Коэффициенты фильтра могут быть предсказаны с использованием реконструированных коэффициентов фильтра, использованных в предыдущем CU. Предыдущие коэффициенты фильтра могут быть представлены как:

f_i^n , где i=0,..., N-1,

В этом случае количество CU n может использоваться, чтобы идентифицировать один или более фильтров, использованных для предсказания текущих фильтров, и количество n может быть отправлено к декодеру как часть кодированного битового потока. В дополнение к этому информация может кодироваться и передаваться к декодеру, чтобы идентифицировать комбинации диапазонов для двух или более метрик, для которых используется кодирование с предсказанием.

[0143] Амплитуда коэффициентов фильтра g(k, l) зависит от значений k и l. Обычно коэффициент с самой большой амплитудой - это коэффициент g(0,0). Другие коэффициенты, которые, как ожидается, имеют большие амплитуды, являются коэффициентами, для которых значение k или l равно 0. Это явление может

использоваться, чтобы дополнительно сократить количество битов, необходимых для сигнализирования коэффициентов. Индексные значения k и l могут определять места расположения в пределах известной поддержки фильтра.

[0144] Коэффициенты:

$$g_i^m(k, l), i=0, \dots, M-1$$

для каждого кадра m могут кодироваться с использованием параметризованных кодов переменной длины, таких как код Голomba (Golomb) или экспоненциальный код Голomba (exp-Golomb), определенных в соответствии с параметром p . Путем изменения значения параметра p , которые определяет параметризованные коды переменной длины, эти коды могут использоваться, чтобы эффективно представить широкий диапазон распределения источников. Распределение коэффициентов $g(k, l)$ (то есть вероятность того, что они будут иметь большие или маленькие значения) зависит от значений k и l . Следовательно, чтобы увеличить эффективность кодирования, для каждого кадра m значение параметра p передается для каждой пары (k, l) . Параметр p может использоваться для параметризованного кодирования переменной длины, когда коэффициенты кодирования:

$$g_i^m(k, l), \text{ где } k=-K, \dots, K, l=-L, \dots, L.$$

[0145] Как было описано выше, видеodeкодер 560 представляет собой пример видеodeкодера, выполненного с возможностью определять первую метрику для группы пикселей в пределах блока пикселей, определять вторую метрику для группы пикселей, определять фильтр на основе первой метрики и второй метрики и генерировать отфильтрованное изображение путем применения фильтра к группе пикселей.

Видеodeкодер 560 также представляет собой пример видеodeкодера, выполненного с возможностью определять первую метрику для блока пикселей, при этом первая метрика определяется на основе сравнения поднабора пикселей в блоке с другими пикселями в блоке; определять вторую метрику для блока пикселей; определять фильтр на основе первой метрики и второй метрики; и генерировать отфильтрованное изображение путем применения фильтра к блоку пикселей.

[0146] Как было описано выше, видеodeкодер 560 также представляет собой пример видеodeкодера, выполненного с возможностью определять соответствие комбинаций диапазонов фильтрам, при этом комбинация диапазонов содержит диапазон для первой метрики и диапазон для второй метрики, при этом каждая комбинация диапазонов имеет уникальную идентификацию комбинации диапазонов (ID), при этом каждая уникальная комбинация диапазонов ID соответствует порядковому значению для комбинации диапазонов; назначать уникальный ID группы группам комбинаций диапазонов на основе порядковых значений для комбинаций диапазонов, при этом каждый уникальный ID группы соответствует порядковому значению для группы; и кодировать наборы коэффициентов фильтра, соответствующих фильтрам, на основе уникальных ID группы. Видеodeкодер 560 может кодировать наборы коэффициентов фильтра путем генерирования наборов коэффициентов фильтра на основе информации, принятой в кодированном битовом потоке. Видеodeкодер 560 может генерировать наборы коэффициентов фильтра с использованием методик разностного кодирования.

[0147] Видеodeкодер 560 также представляет собой пример видеodeкодера, выполненного с возможностью приводить в соответствие первую комбинацию диапазонов первому фильтру, при этом первая комбинация диапазонов содержит первый диапазон значений для первой метрики и первый диапазон значений для второй метрики; приводить в соответствие вторую комбинацию диапазонов второму фильтру,

при этом вторая комбинация диапазонов содержит второй диапазон значений для первой метрики и второй диапазон значений для второй метрики; приводить в соответствие текущую комбинацию диапазонов фильтру, при этом текущая комбинация диапазонов содержит первый диапазон значений для первой метрики и второй диапазон значений для второй метрики. Приведение в соответствие текущей комбинации диапазонов фильтру может включать в себя приведение в соответствие текущей комбинации диапазонов первому фильтру в ответ на прием первого кодового слова, при этом первое кодовое слово показывает, что текущая комбинация диапазонов приведена в соответствие тому же фильтру, что и первая комбинация диапазонов; приведение в соответствие текущей комбинации диапазонов второму фильтру в ответ на прием второго кодового слова, при этом второе кодовое слово показывает, что текущая комбинация диапазонов приведена в соответствие тому же фильтру, что и вторая комбинация диапазонов; и приведение в соответствие текущей комбинации диапазонов третьему фильтру в ответ на прием третьего кодового слова, при этом третье кодовое слово показывает этот третий фильтр. Видеодекодер 560 также представляет собой пример видеодекодера, выполненного с возможностью генерировать соответствие комбинаций диапазонов фильтрам, при этом комбинация диапазонов содержит диапазон для первой метрики и диапазон для второй метрики; приводить в соответствие текущую комбинацию диапазонов тому же фильтру, что и для предыдущей комбинации диапазонов, в ответ на прием первого кодового слова, сигнализирующего, что текущая комбинация диапазонов приведена в соответствие тому же фильтру, что и предыдущая комбинация диапазонов; и приводить в соответствие текущую комбинацию диапазонов фильтру, идентифицированному вторым кодовым словом, в ответ на прием второго кодового слова, сигнализирующего, что текущая комбинация диапазонов приведена в соответствие другому фильтру, нежели предыдущая комбинация диапазонов.

[0148] Как было представлено выше, несколько разные типов метрик могут использоваться вместе с методиками мультиметрической фильтрации, которая описывается в этом раскрытии. Некоторые из этих метрик - это метрики активности, которые определяют количество активности, связанное с одним или более блоками пикселей в пределах видеоданных. Метрики активности могут содержать метрики дисперсии, показывающие дисперсию пикселя в пределах набора пикселей. Как будет описано, некоторые из этих метрик активности - связанные с направлением. Например, метрика горизонтальной активности определяет количество активности вдоль горизонтальной оси, метрика вертикальной активности определяет количество активности вдоль вертикальной оси, метрика диагональной активности определяет количество активности вдоль диагональной оси и т.д.

[0149] Некоторые метрики активности - не связанные с направлением. Например, значение Лапласиана с измененной суммой является метрикой активности, основанной на двухмерном окне пикселей, которые окружают текущий пиксель или текущую группу пикселей. Для текущего пикселя (i,j) значение Лапласиана с измененной суммой может вычисляться следующим образом:

$$\text{var}(i, j) = \sum_{k=-K}^K \sum_{l=-L}^L |2R(i+k, j+l) - R(i+k-1, j+l) - R(i+k+1, j+l) + 2R(i+k, j+l-1) - R(i+k, j+l-1) - R(i+k, j+l+1)| \quad (1)$$

где k представляет значение сложения пиксельных значений от -K до K и l представляют значение сложения от -L до L для двухмерного окна, которое проходит от -K до K и от -L до L, при этом i и j представляют координаты пикселя данных пикселя,

$RI(i, j)$ представляет значение данного пикселя в координатах i и j , а $var(i, j)$ - это метрика активности (то есть значение Лапласиана с измененной суммой).

[0150] Методики этого раскрытия также могут применяться с использованием метрик, связанных с направлением, для горизонтальной активности, вертикальной активности и диагональной активности. Уравнения 2 и 3 демонстрируют примеры того, как горизонтальная активность и вертикальная активность могут вычисляться для текущего пикселя (x, y) путем сравнения значения пикселя (Rec), такого как интенсивность, текущего пикселя с пиксельным значением соседних пикселей.

$$Hor_act(x, y) = R(2 * Rec[x][y] - Rec[x+1][y] - Rec[x-1][y]) \quad (2)$$

$$Ver_act(x, y) = R(2 * Rec[x][y] - Rec[x][y+1] - Rec[x][y-1]) \quad (3)$$

Как показано в уравнении 2, при определении горизонтальной активности текущий пиксель (x, y) может сравниваться с левым соседним $(x-1, y)$ и правым соседним $(x+1, y)$. Как показано в уравнении 3, при определении вертикальной активности текущий пиксель (x, y) может сравниваться с верхним соседним $(x, y+1)$ и нижним соседним $(x, y-1)$.

[0151] Уравнения 4 и 5 демонстрируют примеры того, как вычисляется диагональная активность для текущего пикселя (x, y) , путем сравнения пиксельного значения (Rec) текущего пикселя с пиксельным значением соседних пикселей.

$$45deg_act(x, y) = R(2 * Rec[x][y] - Rec[x+1][y+1] - Rec[x-1][y-1]) \quad (4)$$

$$135deg_act(x, y) = R(2 * Rec[x][y] - Rec[x-1][y+1] - Rec[x+1][y-1]) \quad (5)$$

Как показано в уравнении 4, диагональная активность может вычисляться, например, в направлении 45 градусов путем сравнения текущего пикселя (x, y) с соседним верхним правым $(x+1, y+1)$ и соседним нижним левым $(x-1, y-1)$. Как показано в уравнении 5, диагональная активность может вычисляться, например, в направлении 135 градусов путем сравнения текущего пикселя (x, y) с соседним верхним левым $(x-1, y+1)$ и соседним нижним правым $(x+1, y-1)$.

[0152] Представленные выше уравнения 2-5 иллюстрируют, как горизонтальная активность, вертикальная активность и диагональная активность могут определяться на попиксельной основе, но в некоторых вариантах осуществления горизонтальная активность, вертикальная активность и диагональная активность могут определяться на погрупповой основе, где группа пикселей - блок пикселей 2×2 , 4×4 или $M \times N$. В таком варианте осуществления горизонтальная активность, например, может определяться путем сравнения пиксельных значений из текущей группы со значениями пикселей из левой группы и правой группы, аналогично тому, как описывается в уравнении 2; и вертикальная активность может определяться путем сравнения текущей группы пикселей с верхней группой и нижней группой, аналогично тому, как описывается в уравнении 3. Подобным образом, 45-градусная диагональная активность может определяться путем сравнения текущей группы пикселей с верхней правой соседней группой и левой нижней соседней группой, аналогично тому, как описывается в уравнении 4, а 135-градусная диагональная активность может определяться путем сравнения текущей группы пикселей с верхней левой соседней группой и правой нижней соседней группой, аналогично тому, как описывается в уравнении 5.

[0153] В некоторых вариантах осуществления горизонтальная активность, вертикальная активность, 45-градусная диагональная активность и 135-градусная диагональная активность могут определяться путем сравнения текущего пикселя или группы пикселей с соседними пикселями или группами пикселей только в одном направлении. Например, вместо определения горизонтальной активности на основе сравнения текущего пикселя с левым соседним и правым соседним, горизонтальная

активность может определяться на основе только левого соседа или только правого соседа. Дополнительно, в некоторых вариантах осуществления, горизонтальная активность, вертикальная активность, 45-градусная диагональная активность и 135-градусная диагональная активность могут определяться с использованием средних или взвешенных средних из областей соседних пикселей вместо отдельных соседних пикселей или отдельных групп пикселей.

[0154] Значения, полученные из уравнений 2-5, могут быть разделены на конечное число диапазонов, как то 2, 4, 8 или любое другое конечное число, и каждому диапазону может назначаться идентификация диапазона. Со ссылкой снова на фиг. 4А, например, Диапазон 1-1, Диапазон 1-2, Диапазон 2-1 и т.д. - это все примеры идентификаций диапазона. В качестве одного примера значения горизонтальной активности могут быть разделены на четыре диапазона, и диапазонам могут назначаться ID Диапазона 1-1, Диапазона 1-2, Диапазона 1-3 и Диапазона 1-4. Горизонтальные пороговые значения (то есть $ThH_1, \dots, ThH_P$) могут определять, где начинаются и заканчиваются диапазоны. Таблица 2, представленная ниже, демонстрирует общий случай того, как горизонтальные ID могли бы назначаться Р диапазонам.

Таблица 2	
Индекс метрики активности	
Состояние Hor_act_B	Горизонтальный ID
$Hor_act_B \leq ThH_1$	Диапазон 2-1
$ThH_1 < Hor_act_B \leq ThH_2$	Диапазон 2-2
...	...
$ThH_{P-1} < Hor_act_B$	Диапазон 2-P

При использовании примера из Таблицы 2, если текущий пиксель имеет значение горизонтальной активности больше ThH_1 , но меньше ThH_2 , то текущий пиксель находится в диапазоне 2-2 для метрики 2. Текущие пиксели могут назначаться вертикальным диапазонам с вертикальными ID, 45-градусным диагональным диапазонам - с 45-градусными диагональными ID и 135-градусным диагональным диапазонам - со 135-градусными диагональными ID, подобно тому, как описывается выше в Таблице 2 для горизонтальных диапазонов и горизонтальных ID.

[0155] Любая из горизонтальной активности, вертикальной активности, 45-градусной диагональной активности и 135-градусной диагональной активности может использоваться как метрика в соответствии с методиками фильтрации мультиметрического фильтра, описываемыми в этом раскрытии. Например, возвращаясь к фиг. 4А, Метрика 1 могла бы быть измерением для вертикальной активности, а Метрика 2 могла бы быть измерением для горизонтальной активности. В таком примере элемент фильтра, такой как модуль 349 фильтра с фиг. 4А или модуль 559 фильтра с фиг. 5, может определять фильтр для пикселя или группы пикселей на основе горизонтальной активности пикселя или группы пикселей и вертикальной активности пикселя или группы пикселей. Если, например, текущий пиксель имеет метрику горизонтальной активности, которая попадает в диапазон 2-3, и метрику вертикальной активности, которая попадает в диапазон 1-3, то элемент фильтра фильтрует пиксель, используя Фильтр 4. Подобным образом, комбинации 45-градусной диагональной активности и 135-градусной диагональной активности, 45-градусной диагональной активности и горизонтальной активности, 45-градусной диагональной активности и вертикальной активности, 135-градусной диагональной активности и горизонтальной активности или 135-градусной диагональной активности и вертикальной активности

также могут использоваться блоком фильтра для выбора фильтра для пикселя или группы пикселей. В некоторых вариантах осуществления три или все четыре из горизонтальной активности, вертикальной активности, 45-градусной диагональной активности и 135-градусной диагональной активности могут использоваться элементом

[0156] В вариантах осуществления, описанных выше, горизонтальная активность, вертикальная активность, 45-градусная диагональная активность и 135-градусная диагональная активность - все могут использоваться как метрики, как Метрика 1 и/или Метрика 2 на фиг. 4А, к примеру. В некоторых вариантах осуществления, однако горизонтальная активность, вертикальная активность, 45-градусная диагональная активность и 135-градусная диагональная активность сами по себе могут и не быть метриками, но вместо этого могут использоваться как промежуточные определения для определения общей метрики направления. Метрика направления в общем описывает, в каком направлении (отсутствие направления, горизонтальное, вертикальное, 45-градусное диагональное или 135-градусное диагональное) пиксели больше всего изменяются.

[0157] В одном примере, с использованием только горизонтальной активности и вертикальной активности, как описывается в уравнениях 2 и 3, направление для пикселя могло бы определяться на основе следующих условий:

Направление 1=горизонтальное, если $\text{Hor_activity} > k1 * \text{Ver_activity}$

Направление 2=вертикальное, если $\text{Ver_activity} > k2 * \text{Hor_activity}$

Направление 0=отсутствие направления, иначе.

Константы, $k1$ и $k2$, могут выбираться так, что направление только считается направлением 1 или направлением 2, если горизонтальная активность значительно больше, чем вертикальная активность, или вертикальная активность значительно больше, чем горизонтальная активность. Если горизонтальная активность и вертикальная активность равны или приблизительно равны, то направление - это направление 0. Направление 1 в общем указывает, что пиксельного значения больше изменяются в горизонтальном направлении, чем в вертикальном направлении, а направление 2 указывает, что пиксельного значения больше изменяются в вертикальном направлении, чем в горизонтальном направлении. Направление 0 указывает, что изменения пиксельных значений в горизонтальном направлении приблизительно равны изменениям пиксельных значений в вертикальном направлении.

[0158] Определенная метрика направления (например, направление 0, направление 1, направление 2) может использоваться как метрика для методик мультиметрической фильтрации, описанных в этом раскрытии. При использовании примера с фиг. 4А снова Метрика 1 могла бы быть метрикой дисперсии, как то значение Лапласиана с измененной суммой, тогда как Метрика 2 могла бы быть определением направления, как описано выше. Как описывается со ссылкой на фиг. 4А, каждое из направления 1, направления 2 и направления 0 может быть связано с диапазоном Метрики 2, хотя направление 1, направление 2 и направление 0 представляют конечные определения вместо спектра значений.

[0159] В дополнение к использованию только горизонтальной активности и вертикальной активности, как описано выше, методики этого раскрытия также включают в себя использование 45-градусной диагональной активности и 135-градусной диагональной активности, как описывается в уравнениях 4 и 5, чтобы определить направления на основе следующих условий:

Направление =1, если $45\text{deg_activity} > k1 * 135\text{deg_activity}$

Направление =2, если $135deg_activity > k2 * 45deg_activity$

Направление =0 - иначе.

Определения направления на основе 45-градусной диагональной активности и 135-градусной диагональной активности могут использоваться как метрика с другой метрикой, как то значение Лапласиана с измененной суммой, как описано выше.

[0160] Дополнительно метрика направления может также определяться на основе следующих условий:

Направление =1, если $45deg_activity > k1 * 135deg_activity$, $k2 * Hor_activity$ И $k3 * Ver_activity$

Направление =2, если $135deg_activity > k4 * 45deg_activity$, $k5 * Hor_activity$ И $k6 * Ver_activity$

Направление =3, если $Hor_activity > k7 * Ver_activity$, $k8 * 135deg_activity$ И $k9 * 45deg_activity$

Направление =4, если $Ver_activity > k10 * Hor_activity$, $k11 * 135deg_activity$ И $k12 * 45deg_activity$

Направление =0, иначе.

Как описано выше, $k1-k12$ - константы, выбранные для определения того, насколько больше одна из горизонтальной активности, вертикальной активности, 45-градусной диагональной активности и 135-градусной диагональной активности нуждается в сравнении с другими для того, чтобы было выбрано определенное направление. Определения направления на основе горизонтальной активности, вертикальной активности, 45-градусной диагональной активности и 135-градусной диагональной активности могут использоваться как метрика с другой метрикой, как то значение Лапласиана с измененной суммой, как описывается выше.

[0161] Другая метрика, которая может использоваться с методиками этого раскрытия, включает в себя метрику края. Метрика края в общем определяет количество активности, которое могло бы показывать наличие края в блоке пикселей. Край может появиться, например, в блоке пикселей, если этот блок пикселей содержит границу объекта в пределах изображения. Один пример обнаружения края включает в себя использование четырех пикселей, соседних с текущим пикселем (например, левый, правый, верхний, нижний) или использование восьми пикселей, соседних с текущим пикселем (левый, правый, верхний, нижний, верхний правый, верхний левый, нижний правый, нижний левый). Дополнительно определение типа края может включать в себя использование двух соседних пикселей, таких как верхний и нижний, левый и правый, верхний левый и нижний правый или верхний правый и нижний левый.

[0162] Псевдокод, представленный ниже, демонстрирует примеры того, как информация края может вычисляться для текущего пикселя (x, y) путем сравнения пиксельного значения (Rec), такого как интенсивность, текущего пикселя с пиксельными значениями тех соседних пикселей (то есть 4/8 пикселей).

[0163] Переменная `EdgeType` (ТипКрая) иницируется на 0. Каждый раз, когда утверждение истинное, переменная `EdgeType` или увеличивается на 1 (как показано в псевдокоде `EdgeType ++`), или уменьшается на 1 (как показано в псевдокоде `EdgeType --`). `Rec[x][y]` относится к значению пикселя, такому как интенсивность пикселя, пикселя, расположенного на (x, y). Первое группирование утверждений «если» («if») - для сравнения текущего пикселя с верхним, нижним, левым и правыми соседними. Второе группирование утверждений «если» - для сравнения текущего пикселя с верхним левым, верхним правым, нижним левым и нижним правым соседями. Методики этого раскрытия могут реализовываться с использованием или группы, или обеих групп.

```

EdgeType =0;
if (Rec[x][y]>Rec[x-1][y]) EdgeType ++;
if (Rec[x][y]<Rec[x-1][y]) EdgeType --;
if (Rec[x][y]>Rec[x+1][y]) EdgeType ++;
5 if (Rec[x][y]<Rec[x+1][y]) EdgeType --;
if (Rec[x][y]>Rec[x][y-1]) EdgeType ++;
if (Rec[x][y]<Rec[x][y-1]) EdgeType --;
if (Rec[x][y]>Rec[x][y+1]) EdgeType ++;
if (Rec[x][y]<Rec[x][y+1]) EdgeType --;
10 if (Rec[x][y]>Rec[x-1][y-1]) EdgeType ++;
if (Rec[x][y]<Rec[x-1][y-1]) EdgeType --;
if (Rec[x][y]>Rec[x+1][y-1]) EdgeType ++;
if (Rec[x][y]<Rec[x+1][y-1]) EdgeType --;
if (Rec[x][y]>Rec[x-1][y+1]) EdgeType ++;
15 if (Rec[x][y]<Rec[x-1][y+1]) EdgeType --;
if (Rec[x][y]>Rec[x+1][y+1]) EdgeType ++;
if (Rec[x][y]<Rec[x+1][y+1]) EdgeType --;

```

[0164] Если текущий пиксель - локальный максимум, значение пикселя этого пикселя будет больше, чем у всех его соседей, и будет иметь тип края 4 при использовании
 20 четырех соседей или тип края 8 при использовании восьми соседей. Если текущий пиксель - локальный минимум, значение пикселя этого пикселя будет меньше, чем у всех его соседей, и будет иметь тип края -4 при использовании четырех соседей или тип края -8 при использовании восьми соседей. Так, использование примерных методик, описанных выше, для определения типа края между -4 и 4 или -8 и 8 может
 25 использоваться и при определении фильтра. Значения, определенные для типа края (то есть значения -4 и 4 или значения -8 и 8), могут приводиться в соответствие диапазонам метрики, такой как Метрика 1 или Метрика 2 с фиг. 4А. В некоторых вариантах осуществления абсолютные значения определения типа края могут приводиться в соответствие диапазонам так, что тип края -3 и 3, например, приводился бы в
 30 соответствие одному и тому же фильтру.

[0165] Вычисления различных метрик, описанных в этом раскрытии, направлены только на то, чтобы быть примерами, но не исчерпывающими. Например, метрики могут определяться на основе окон или строк пикселей, которые включают в себя больше соседних пикселей, чем описывается в этом раскрытии.

35 [0166] Дополнительно в некоторых вариантах осуществления метрики, описанные в этом раскрытии, могут вычисляться с использованием подвыборки пикселей в конкретной строке или окне. Например, чтобы вычислить метрику активности блока для блока пикселей 4x4, метрики для активности и направления могут вычисляться следующим образом:

```

40 Метрика Направления
Ver_act(i, j)=abs(X(i, j)<<1-X(i, j-1)-X(i, j+1))
Hor_act(i, j)=abs(X(i, j)<<1-X(i-1, j)-X(i+1, j))
HB=Σi=0,2 Σj=0,2 Hor_act(i, j)
VB=Σi=0,2 Σj=0,2 Ver_act(i, j)
45 Направление =0, 1(HB>k1*VB), 2(VB>k2*HB)

Метрика Активности
LB=HB+VB

```

5 классов (0, 1, 2, 3, 4)

Метрика

Комбинация Активности и Направления (например, 15 или 16 комбинаций, как было объяснено выше в примере с фиг. 4В).

5 [0167] $\text{Hor_act}(i, j)$ в общем относится к горизонтальной активности текущего пикселя (i, j) , и $\text{Vert_act}(i, j)$ в общем относится к вертикальной активности текущего пикселя (i, j) . $X(i, j)$ в общем относится к значению пикселя (i, j) . H_B в общем относится к горизонтальной активности блока 4×4 , который в этом примере определяется на основе суммы горизонтальной активности для пикселей $(0,0)$, $(0,2)$, $(2,0)$ и $(2,2)$. V_B относится к вертикальной активности блока 4×4 , который в этом примере определяется на основе суммы вертикальной активности для пикселей $(0,0)$, $(0,2)$, $(2,0)$ и $(2,2)$. « $\ll 1$ » представляет собой операцию умножения на два. Как объясняется выше, на основе значений H_B и V_B может быть определено направление. С использованием примера, представленного выше, если значение H_B больше, чем k раз, значения V_B , то направление может быть определено как направление 1 (то есть горизонтальное), что может соответствовать более горизонтальной активности, нежели вертикальной активности. Если значение V_B больше, чем k раз, значения H_B , то направление может быть определено как направление 2 (то есть вертикальное), что может соответствовать более вертикальной активности, нежели горизонтальной активности. В противном случае направление может быть определено как направление 0 (то есть отсутствие направления), что означает, что ни горизонтальная, ни вертикальная активность не является доминантной. Метки для различных направлений и соотношений, используемых, чтобы определить направления, составляющие только один пример, как и другие метки и соотношения, также могут использоваться.

[0168] Активность (L_B) для блока 4×4 может определяться как сумма горизонтальной и вертикальной активности. Значение L_B может классифицироваться в диапазон, как описывается выше. Этот конкретный пример демонстрирует пять диапазонов, хотя большее или меньшее количество диапазонов может подобным образом использоваться. Как описано выше, фильтр может быть выбран на основе двухмерного соответствия активности и направления фильтрам, как описывается со ссылкой на фиг. 4А и фиг. 4В, или активность и направление могут комбинироваться в единую метрику, и эта единая метрика может использоваться для выбора фильтра.

35 [0169] Фиг. 6А представляет блок пикселей 4×4 . При использовании методик подвыборки, описанных выше, только четыре из шестнадцати пикселей используются. Четыре пикселя - пиксель $(0,0)$, который обозначается как пиксель 601, пиксель $(2,0)$, который обозначается как пиксель 602, пиксель $(0,2)$, который обозначается как пиксель 603, и пиксель $(2,2)$, который обозначается как пиксель 604. Горизонтальная активность пикселя 601 (есть $\text{hor_act}(0,0)$), например, определяется на основе левого соседнего пикселя и правого соседнего пикселя. Правый соседний пиксель обозначается как пиксель 605. Левый соседний пиксель располагается в другом блоке, нежели блок 4×4 , и не показан на фиг. 6А. Вертикальная активность пикселя 602 (то есть $\text{vert_act}(2,0)$), например, определяется на основе верхнего соседнего пикселя и нижнего соседнего пикселя. Нижний соседний пиксель обозначается как пиксель 606, и верхний соседний пиксель располагается в другом блоке, нежели блок 4×4 , и не показан на фиг. 6А.

[0170] В общем, используя те же методики, что и описаны выше, метрика активности блока может также вычисляться с использованием разных поднаборов пикселей

следующим образом:

Метрика Направления

$Ver_act(i, j) = \text{abs}(X(i, j) - X(i, j-1) - X(i, j+1))$

$Hor_act(i, j) = \text{abs}(X(i, j) - X(i-1, j) - X(i+1, j))$

5 $H_B = \sum_{i=1,2} \sum_{j=1,2} H(i, j)$

$V_B = \sum_{i=1,2} \sum_{j=1,2} V(i, j)$

Направление = 0, 1 ($H > k_1 * V$), 2 ($V > k_2 * H$)

Метрика Активности

10 $L_B = H_B + V_B$

5 классов (0, 1, 2, 3, 4)

Метрика

Комбинация Активности и Направления (например, 15 или 16 комбинаций, как было объяснено выше в примере с фиг. 4В)

15 [0171] Этот другой поднабор пикселей для вычисления H_B и V_B включает в себя пиксели (1,1), (2,1), (1,2) и (2,2), показанные на фиг. 6В как пиксели 611, 612, 613 и 614 соответственно. Как можно увидеть с фиг. 6В, все из верхних соседних пикселей, нижних соседних пикселей, правых соседних пикселей и левых соседних пикселей для пикселей 611, 612, 613 и 614 располагаются в пределах блока 4x4. В примере с фиг. 6В пиксели 20 611, 612, 613 и 614 все располагаются внутри блока, в противоположность расположению на границе блока. В других вариантах осуществления дополнительные разные поднаборы пикселей могут быть выбраны. Например, поднаборы могут быть выбраны так, что соседние пиксели сверху и снизу для пикселей поднабора - в пределах блока 4x4, но некоторые соседние пиксели слева и справа - в соседних блоках. Поднаборы 25 также могут выбираться так, что соседние пиксели слева и справа для пикселей поднабора - в пределах блока 4x4, но некоторые соседние пиксели сверху и снизу - в соседних блоках.

[0172] В общем, используя те же методики, что и описаны выше, метрика активности блока может также вычисляться с использованием поднабора из восьми пикселей 30 следующим образом:

Метрика Направления

$Ver_act(i, j) = \text{abs}(X(i, j) - X(i, j-1) - X(i, j+1))$

$Hor_act(i, j) = \text{abs}(X(i, j) - X(i-1, j) - X(i+1, j))$

$H_B = \sum_{i=0,1,2,3} \sum_{j=1,2} H(i, j)$

35 $V_B = \sum_{i=0,1,2,3} \sum_{j=1,2} V(i, j)$

Направление = 0, 1 ($H > k_1 * V$), 2 ($V > k_2 * H$)

Метрика Активности

$L_B = H_B + V_B$

40 5 классов (0, 1, 2, 3, 4)

Метрика

Комбинация Активности и Направления (например, 15 или 16 комбинаций, как было объяснено выше в примере с фиг. 4В)

45 [0173] Этот другой поднабор пикселей для вычисления H_B и V_B включает в себя пиксели (0,1), (1,1), (2,1), (3,1), (0,2), (1,2), (2,2) и (3,2), показанные на фиг. 6С как пиксели 621, 622, 623 и 624, 625, 626, 627 и 628 соответственно. Как можно увидеть с фиг. 6С, все из верхних соседних пикселей и нижних соседних пикселей 621, 622, 623 и 624, 625, 626, 627 и 628 располагаются в пределах блока 4x4, хотя пиксели 621 и 625 имеют левые

соседние пиксели в левом соседнем блоке каждый, а пиксели 624 и 628 имеют правые соседние пиксели в правом соседнем блоке каждый. Этот конкретный выбор пикселей может уменьшить сложность кодера и/или декодера путем избегания потребности в линейном буфере для хранения пиксельных значений соседних блоков сверху и/или снизу. В силу порядка растрового сканирования слева-направо, сверху-вниз линейные буферы для пиксельных значений соседних блоков сверху и снизу часто должны хранить пиксельного значения для всей верхней или нижней строки, которая в случае 1080P видео, например, могла бы составлять 1920 пикселей. Линейные буферы для соседних блоков слева и справа, однако, часто должны хранить только значения для одного LCU или пары LCU, что могло бы быть только 64 или 128 пикселей, к примеру. Так, линейные буферы для пиксельных значений для соседних блоков сверху и снизу могут нуждаться в том, чтобы быть значительно больше, чем линейные буферы, используемые для пиксельных значений соседних блоков слева и справа. Выбор пикселей, показанный на фиг. 6С, может позволить избежать использования линейных буферов для пиксельных значений верхнего и нижнего соседнего блока, таким образом снижая сложность кодирования.

[0174] Примеры с фиг. 6А - фиг. 6С - только представленные методики этого раскрытия. Предполагается, что эти методики могут быть расширены для блоков, отличных от только 4x4, и что разные поднаборы пикселей могут быть выбраны.

[0175] При вычислении метрики активности блока, вместо оригинальных пикселей могут использоваться квантованные пиксели (то есть $X(i, j) \gg N$), чтобы снизить сложность операций, таких как операции добавления. Дополнительно вычисления могут быть на основе абсолютной разности вместо основанных на значении Лапласиана. Например, при вычислении $\text{Hor_act}(i, j)$ или $\text{Ver_act}(i, j)$ абсолютные разности могут использоваться вместо значений Лапласиана следующим образом:

Метрика Направления

$\text{Ver_act}(i, j) = \text{abs}(X(i, j) - X(i, j-1))$

$\text{Hor_act}(i, j) = \text{abs}(X(i, j) - X(i-1, j))$

$H_B = \sum_{i=0,1,2} \sum_{j=0,1,2} H(i, j)$

$V_B = \sum_{i=0,1,2} \sum_{j=0,1,2} V(i, j)$

Направление = 0, 1 ($H > 2V$), 2 ($V > 2H$)

Метрика Активности

$L_B = H_B + V_B$

5 классов (0, 1, 2, 3, 4)

Метрика

Активность + Направление (например, 15 или 16 комбинаций, как было объяснено выше в примере с фиг. 4В)

[0176] Это раскрытие описало методики подвыборки со ссылкой на ограниченную группу конкретных метрик. Предполагается, что, однако, эти методики подвыборки в общем применимы к другим метрикам, таким как другие метрики, рассмотренные в этом раскрытии, которые могут использоваться в целях определения фильтра. Дополнительно, хотя методики подвыборки этого раскрытия были описаны со ссылкой на блоки пикселей 4x4, методики также могут быть применимы к блокам других размеров.

[0177] Фиг. 7 представляет собой блок-схему последовательности операций, иллюстрирующую методику кодирования видео, совместимую с данным раскрытием. Методики, описанные на фиг. 7, могут реализовываться элементом фильтра видеокодера

или видеodeкодера, как то модуль 349 фильтра видеodeкодера 350 или модуль 559 фильтра видеodeкодера 560. Элемент фильтра определяет первую метрику для группы пикселей в пределах блока пикселей (710). Первая метрика может, к примеру, быть метрикой активности, такой как значение Лапласиана с измененной суммой, или первая метрика может быть метрикой направления. Первая метрика может определяться, например, на основе сравнения набора пикселей в блоке или на основе поднабора пикселей в блоке с остальными пикселями в блоке. Элемент фильтра дополнительно определяет вторую метрику для блока (720). Вторая метрика может, к примеру, быть метрикой направления, которая определяется на основе сравнения измерения горизонтальной активности с измерением вертикальной активности. На основе первой метрики и второй метрики модуль фильтра определяет фильтр (730). Модуль фильтра генерирует отфильтрованное изображение путем применения фильтра к блоку (740). Как рассмотрено выше, в некоторых вариантах осуществления блок может быть блоком пикселей 2x2, 4x4 или MxN, используемым для определения первой метрики или второй метрики. В некоторых вариантах осуществления первая метрика может быть метрикой горизонтальной активности, тогда как вторая метрика - метрика вертикальной активности, или первая метрика может быть метрикой края, тогда как вторая метрика - метрика направления.

[0178] Фиг. 8А представляет собой блок-схему последовательности операций, иллюстрирующие методики кодирования видео в соответствии с данным раскрытием. Методики, описанные на фиг. 8А, могут реализовываться модулем фильтра видеodeкодера, как то модуль 559 фильтра видеodeкодера 560. Модуль 559 фильтра приводит в соответствие первую комбинацию диапазона первому фильтру (810А). Первая комбинация диапазона - комбинация первого диапазона значений для первой метрики и первого диапазона значений для второй метрики. Первая метрика может, например, быть значением Лапласиана с измененной суммой, а вторая метрика может быть метрикой направления, хотя другие метрики также могут использоваться. Модуль 559 фильтра приводит в соответствие вторую комбинацию диапазона второму фильтру (820А). Вторая комбинация диапазона - комбинация второго диапазона значений для первой метрики и второго диапазона значений для второй метрики. Модуль 559 фильтра затем устанавливает соответствие текущей комбинации диапазонов фильтру на основе принятого кодового слова. Текущая комбинация диапазонов включает в себя первый диапазон значений первой метрики и второй диапазон значений второй метрики. Если кодовое слово - это первое кодовое слово (830А, да), модуль 559 фильтра устанавливает соответствие текущей комбинации диапазонов первому фильтру (840А). Первое кодовое слово указывает, что текущая комбинация диапазонов приведена в соответствие с тем же фильтром, что и первая комбинация диапазонов. Если кодовое слово - это второе кодовое слово (830А, нет), модуль 559 фильтра устанавливает соответствие текущей комбинации диапазонов второму фильтру (860А). Второе кодовое слово указывает, что текущая комбинация диапазонов приведена в соответствие с тем же фильтром, что и вторая комбинация диапазонов. Если кодовое слово - это ни первое кодовое слово, ни второе кодовое слово (850А, нет), то модуль 559 фильтра приводит текущую комбинацию диапазонов в соответствие третьему фильтру (870А). Если в ответ на прием третьего кодового слова, при этом третье кодовое слово идентифицирует тот третий фильтр. В примере с фиг. 8А первое кодовое слово и второе кодовое слово - каждое из них - могут включать в себя меньше битов, чем третье кодовое слово.

[0179] Фиг. 8В представляет собой блок-схему последовательности операций, иллюстрирующие методики кодирования видео в соответствии с данным раскрытием.

Методики, описанные на фиг. 8В, могут реализовываться модулем фильтра видеodeкодера, как то модуль 559 фильтра видеodeкодера 560. Модуль 559 фильтра генерирует соответствие комбинаций диапазонов фильтрам (810В). Каждая комбинация диапазонов, например, может включать в себя диапазон для первой метрики и диапазон для второй метрики. В ответ на прием первого кодового слова, которое сигнализирует, что текущая комбинация диапазонов приводится в соответствие тому же фильтру, что и предыдущая комбинация диапазонов (820В, да), модуль 559 фильтра устанавливает соответствие текущей комбинации диапазонов тому же фильтру, что и для предыдущей комбинации диапазонов (830В). В ответ на прием второго кодового слова, которое сообщает, что текущая комбинация диапазонов приводится в соответствие другому фильтру, нежели предыдущая комбинация диапазонов (820В, нет), модуль 559 фильтра устанавливает соответствие текущей комбинации диапазонов новому фильтру (840В). Как описано выше, текущая комбинация диапазонов может определяться на основе известного порядка передачи. В некоторых примерах новый фильтр может быть идентифицирован на основе второго кодового слова, тогда как в других примерах новый фильтр может определяться на основе порядка, в котором сигнализируются коэффициенты фильтра.

[0180] Фиг. 9А представляет собой блок-схему последовательности операций, иллюстрирующие методики кодирования видео в соответствии с данным раскрытием. Методики, описанные на фиг. 9А, могут реализовываться блоком фильтра видеodeкодера, как то модуль 349 фильтра видеodeкодера 350. Модуль 349 фильтра определяет соответствие комбинаций диапазонов фильтрам (910А). Каждая комбинация диапазонов включает в себя диапазон значений для первой метрики и диапазон значений для второй метрики. Для текущей комбинации диапазонов, если текущая комбинация диапазонов приведена в соответствие тому же фильтру, что и предыдущая комбинация диапазонов, которая содержит тот же диапазон значений для первой метрики (920А, да), то модуль 349 фильтра генерирует первое кодовое слово (930А). Если текущая комбинация диапазонов приведена в соответствие тому же фильтру, что и предыдущая комбинация диапазонов, которая содержит тот же диапазон значений для второй метрики (940А, да), то модуль 349 фильтра генерирует второе кодовое слово (950А). Если текущая комбинация диапазонов не приведена в соответствие ни предыдущей комбинации диапазонов, которая содержит тот же диапазон значений для первой метрики, ни предыдущей комбинации диапазонов, которая содержит тот же диапазон значений для второй метрики (950А, нет), то модуль 349 фильтра генерирует третье кодовое слово (960А). Третье кодовое слово может идентифицировать фильтр, приведенный в соответствие текущей комбинации диапазонов.

[0181] Фиг. 9В представляет собой блок-схему последовательности операций, иллюстрирующие методики кодирования видео в соответствии с данным раскрытием. Методики, описанные на фиг. 9В, могут реализовываться модулем фильтра видеodeкодера, как то модуль 349 фильтра видеodeкодера 350. Модуль 349 фильтра определяет соответствие комбинаций диапазонов фильтрам (910В). Каждая комбинация диапазонов может, например, включать в себя диапазон для первой метрики и диапазон для второй метрики. Когда текущая комбинация диапазонов, подвергаемая кодированию, имеет тот же фильтр, что и прежде кодированная комбинация диапазонов (920В, да), модуль 349 фильтра может генерировать первое кодовое слово, чтобы сигнализировать, что текущая комбинация диапазонов приведена в соответствие тому же фильтру, что и предыдущая комбинация диапазонов (930В). Когда текущая комбинация диапазонов, подвергаемая кодированию, не имеет того же фильтра, что и прежде кодированная

комбинация диапазонов (920В, нет), модуль 349 фильтра может генерировать второе кодовое слово (940В). Второе кодовое слово может идентифицировать фильтр, приведенный в соответствие текущей комбинации диапазонов. Как описано выше, текущая комбинация диапазонов может определяться на основе известного порядка передачи. В примере с фиг. 9В первое кодовое слово может включать в себя меньше битов, чем второе кодовое слово.

[0182] В примерах с фиг. 8А и фиг. 8В и фиг. 9А и фиг. 9В термины «первое кодовое слово», «второе кодовое слово» и «третье кодовое слово» используются, чтобы обозначить разницу между разными кодовыми словами и не подразумевается последовательного упорядочивания кодовых слов.

[0183] Фиг. 10 представляет собой блок-схему последовательности операций, иллюстрирующую методики кодирования видео в соответствии с данным раскрытием. Методики, описанные на фиг. 10, могут реализовываться модулем фильтра видеокодера, как то модуль 349 фильтра видеокодера 350, или модулем фильтра видеodeкодера, как то модуль 559 фильтра. Модуль фильтра определяет соответствие комбинаций диапазонов фильтрам (1010). Комбинации диапазонов включают в себя диапазон для первой метрики и диапазон для второй метрики. Модуль фильтра определяет уникальную идентификацию комбинации диапазонов (ID) для каждой комбинации диапазонов (1020). Уникальные ID комбинации диапазонов соответствуют порядковым значениям. Модуль фильтра назначает первый уникальный ID группы первой группе комбинаций диапазонов на основе порядкового значения ID комбинации диапазонов по меньшей мере одной комбинации диапазонов в первой группе комбинаций диапазонов (1030). Группы комбинаций диапазонов включают в себя комбинации диапазонов, приведенные в соответствие одному и тому же фильтру, уникальные ID группы соответствуют набору порядковых значений. Модуль фильтра кодирует первый набор коэффициентов фильтра, соответствующий одному и тому же фильтру, на основе порядкового значения первого уникального ID фильтра (1040). В случае видеокодера кодирование первого набора коэффициентов фильтра может включать в себя, например, сигнализирование коэффициентов фильтра в кодированном битовом потоке с использованием методик разностного кодирования. В случае видеodeкодера кодирование первого набора коэффициентов фильтра может включать в себя реконструкцию коэффициентов фильтра на основе информации, принятой в кодированном битовом потоке.

[0184] Фиг. 11 представляет собой блок-схему последовательности операций, иллюстрирующую методики кодирования видео в соответствии с данным раскрытием. Методики, описанные на фиг. 11, могут реализовываться блоком фильтра видеокодера, как то модуль 349 фильтра видеокодера 350, или модулем фильтра видеodeкодера, как то модуль 559 фильтра. Модуль фильтра определяет соответствие комбинаций диапазонов фильтрам (1110). Комбинации диапазонов включают в себя диапазон для первой метрики и диапазон для второй метрики. Каждая комбинация диапазонов может иметь уникальную идентификацию комбинации диапазонов (ID), и каждый уникальный ID комбинации диапазонов может соответствовать порядковому значению для этой комбинации диапазонов. Модуль фильтра может назначать уникальный ID группы каждой группе комбинаций диапазонов (1120). Модуль фильтра может назначать уникальные ID группы, например, на основе порядковых значений комбинаций диапазонов. Группа комбинаций диапазонов может включать в себя комбинации диапазонов, приведенные в соответствие общему фильтру, и уникальные ID группы могут соответствовать набору порядковых значений. Модуль фильтра может кодировать

наборы коэффициентов фильтра для фильтров на основе уникальных ID группы (1140).

[0185] В примере с фиг. 11 модуль фильтра может назначать уникальные ID группы, назначая, например, уникальный ID группы, соответствующий самому низкому порядковому значению из уникальных ID группы, группе комбинаций диапазонов, которая содержит комбинацию диапазонов с ID комбинации диапазонов, соответствующим самому низкому порядковому значению из ID комбинации диапазонов. В другом примере модуль фильтра может назначать уникальный ID группы, соответствующий самому высокому порядковому значению из уникальных ID группы, группе комбинаций диапазонов, которая содержит комбинацию диапазонов с ID комбинации диапазонов, соответствующим самому высокому порядковому значению из ID комбинации диапазонов.

[0186] В случаях, когда модуль фильтра является частью видеodeкодера, модуль фильтра может кодировать наборы коэффициентов фильтра путем генерирования наборов коэффициентов фильтра на основе информации, принятой в кодированном битовом потоке. Модуль фильтра может, например, генерировать наборы коэффициентов фильтра, используя методики разностного кодирования. В случаях, когда модуль фильтра является частью видеокодера, модуль фильтра может кодировать наборы коэффициентов фильтра путем сигнализирования наборов коэффициентов фильтра в кодированном битовом потоке в порядке, выбранном на основе порядковых значений уникальных ID группы. Модуль фильтра может, к примеру, сигнализировать наборы коэффициентов фильтра, используя методики разностного кодирования.

[0187] Вышеизложенное раскрытие было упрощено до некоторой степени с целью передать детали. Например, раскрытие в общем описывает наборы фильтров, которые сигнализируются на покaдровой основе или на основе «вырезка за вырезкой», но наборы фильтров также могут сигнализироваться на основе «последовательность за последовательностью», на основе «группа изображений за группой изображений», на основе «группа вырезок за группой вырезок», на основе «CU за CU», на основе «LCU за LCU» или другой подобной основе. В общем, фильтры могут сигнализироваться для любого группирования одного или более CU. Дополнительно, в варианте осуществления, может быть множество фильтров на каждый ввод на каждый CU, множество коэффициентов на каждый фильтр и множество разных уровней дисперсии с каждым из фильтров, определяемых для разного диапазона дисперсии. Например, в некоторых случаях может быть шестнадцать или больше фильтров, определенных для каждого ввода CU, и шестнадцать разных диапазонов дисперсии, соответствующих каждому фильтру. Дополнительно, когда это раскрытие описывает передачу информации фильтра, не должно допускаться, что вся информация фильтра передается на одном и том же уровне кодирования. К примеру, в некоторых вариантах осуществления некоторая информация фильтра, такая как синтаксис описания фильтра, может сигнализироваться на покaдровой основе или на основе вырезка за вырезкой, тогда как другая информация фильтра, такая как коэффициенты фильтра, сигнализируется на основе LCU-за-LCU. Синтаксис и другие уровни иерархии кодирования, как то уровень последовательности, GOP-уровень или другие уровни, могут также определяться для переноса некоторой или всей подобной информации фильтра.

[0188] Каждый из фильтров для каждого ввода может включать в себя много коэффициентов. В одном примере фильтры содержат двухмерные фильтры с 81 разными коэффициентами, определенными для поддержки фильтра, который охватывает два измерения. Однако количество коэффициентов фильтра, которые сигнализируются для каждого фильтра, может быть меньше 81 в некоторых случаях. Симметрия

коэффициентов, к примеру, может устанавливаться так, что коэффициенты фильтра в одном измерении или квадранте могут соответствовать обратным или симметричным значениям относительно коэффициентов в других измерениях или квадрантах.

Симметрия коэффициентов может допускать выражение 81 разных коэффициентов меньшим количеством коэффициентов, в этом случае кодер и декодер могут допускать, что обратные или зеркальные значения коэффициентов определяют другие коэффициенты. Например, коэффициенты (5, -2, 10, 10, -2, 5) могут кодироваться и сигнализироваться как поднабор коэффициентов (5, -2, 10). В этом случае декодер может знать, что эти три коэффициента определяют большой симметричный набор коэффициентов (5, -2, 10, 10, -2, 5).

[0189] Методики этого раскрытия могут реализовываться в широком многообразии устройств и аппаратов, включая беспроводную радиотелефонную гарнитуру и интегральную схему (IC) или набор IC (то есть набор чипов). Любые компоненты, модули или элементы были описаны с целью выделить функциональные аспекты и не требуют обязательной реализации различными элементами аппаратного обеспечения.

[0190] Соответственно, методики, описанные здесь, могут реализовываться в аппаратном обеспечении, программном обеспечении, программно-аппаратном обеспечении или любой их комбинации. При применении в аппаратном обеспечении любые признаки, описанные как модули, элементы или компоненты, могут быть выполнены вместе в интегральном логическом устройстве или по отдельности как отдельные имеющие возможность взаимодействовать логические устройства. При применении в программном обеспечении методики могут быть реализованы по меньшей мере частично машиночитаемым носителем, содержащим команды, которые при выполнении в процессоре выполняют один или более способов, описанных выше.

Машиночитаемый носитель может содержать машиночитаемый носитель данных и может составлять часть компьютерного программного продукта, который может включать в себя упаковочные материалы. Машиночитаемый носитель данных может содержать оперативное запоминающее устройство (RAM), такое как синхронное динамическое оперативное запоминающее устройство (SDRAM), постоянное запоминающее устройство (ROM), энергонезависимое оперативное запоминающее устройство (NVRAM), электрически стираемое программируемое постоянное запоминающее устройство (EEPROM), FLASH-память, магнитные и оптические носители данных и т.п. Методики дополнительно или в качестве альтернативы могут быть реализованы, по меньшей мере частично, машиночитаемым средством связи, которое переносит или сообщает код в виде команд или структур данных и доступ к которому, чтение и/или выполнение которого может осуществляться с компьютера.

[0191] Код может выполняться одним или более процессорами, такими как один или более цифровых процессоров сигналов (DSP), универсальных микропроцессоров, специализированных интегральных схем (ASIC), программируемых пользователем вентильных матриц (FPGA) или других эквивалентов интегральных или дискретных логических схем. Соответственно, термин «процессор», как используется здесь, может относиться к любой из вышеупомянутых структур или любой другой структуре, подходящей для реализации методик, описанных здесь. Кроме того, в некоторых аспектах функциональность, описанная здесь, может быть предусмотрена в специализированных модулях программного обеспечения или модулях аппаратного обеспечения, сконфигурированных для кодирования и декодирования, или встроена в комбинированный видеокодек. Также методики могли бы в полной мере быть реализованы в одной или более схемах или логических элементах.

Различные аспекты раскрытия были описаны. Эти и другие аспекты находятся в пределах объема следующей формулы изобретения.

Формула изобретения

- 5 1. Способ декодирования данных видео, содержащий:
 определение направления, по меньшей мере, одного пикселя из блока пикселей,
 определение метрики для упомянутого, по меньшей мере, одного пикселя из
 упомянутого блока пикселей, при этом метрика определяется на основе сравнения
 пиксельных значений для поднабора пикселей в этом блоке с упомянутым, по меньшей
 10 мере, одним пикселем, при этом блок пикселей представляет собой блок пикселей $M \times N$,
 и при этом по меньшей мере одно из M и N составляет 2 или больше;
 определение, на основе упомянутой метрики и упомянутого направления, фильтра
 из набора фильтров, сигнализированного в данных видео, для упомянутого, по меньшей
 мере, одного пикселя из упомянутого блока пикселей; и
 15 генерирование отфильтрованного изображения путем применения фильтра к
 упомянутому, по меньшей мере, одному пикселю из упомянутого блока пикселей.
2. Способ по п. 1, в котором направление содержит одно из горизонтального
 направления, вертикального направления, направления 45 градусов и направления 135
 градусов.
- 20 3. Способ по п. 1, в котором определение направления содержит сравнение измерения
 горизонтальной активности с измерением вертикальной активности, и причем
 горизонтальная активность определяется на основе сравнения пиксельного значения,
 по меньшей мере, одного пикселя в упомянутом поднаборе с пиксельным значением
 левого соседнего пикселя и пиксельным значением правого соседнего пикселя, при
 25 этом левый соседний пиксель и правый соседний пиксель оба находятся в упомянутом
 блоке пикселей.
4. Способ по п. 3, в котором вертикальная активность определяется на основе
 сравнения пиксельного значения, по меньшей мере, одного пикселя в упомянутом
 поднаборе с пиксельным значением верхнего соседнего пикселя и пиксельным значением
 30 нижнего соседнего пикселя, при этом верхний соседний пиксель и нижний соседний
 пиксель оба находятся в упомянутом блоке пикселей.
5. Способ по п. 1, в котором поднабор пикселей содержит пиксели не на границе
 блока $M \times N$.
6. Способ по п. 1, в котором блок пикселей представляет собой блок пикселей 4×4 ,
 35 включающий в себя шестнадцать пикселей, и при этом поднабор пикселей включает в
 себя четыре пикселя.
7. Способ по п. 1, в котором определение метрики содержит:
 сравнение, по меньшей мере, одного пикселя с первым соседним пикселем из
 упомянутого поднабора пикселей,
 40 в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного
 пикселя больше пиксельного значения первого соседнего пикселя, увеличение
 переменной метрики края; и
 в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного
 пикселя меньше пиксельного значения первого соседнего пикселя, уменьшение
 45 переменной метрики края.
8. Способ по п. 7, в котором определение метрики содержит:
 сравнение, по меньшей мере, одного пикселя со вторым соседним пикселем из
 упомянутого поднабора пикселей,

в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя больше пиксельного значения второго соседнего пикселя, увеличение переменной метрики края; и

5 в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя меньше пиксельного значения второго соседнего пикселя, уменьшение переменной метрики края.

9. Способ по п. 8, в котором первый соседний пиксель и второй соседний пиксель содержат одно из:

10 левого соседнего пикселя и правого соседнего пикселя,
верхнего соседнего пикселя и нижнего соседнего пикселя,
нижнего левого соседнего пикселя и верхнего правого соседнего пикселя, и
верхнего левого соседнего пикселя и нижнего правого соседнего пикселя.

10. Способ по п. 1, в котором способ выполняется устройством кодирования видео, содержащим видеокодер, выполненный с возможностью выполнения декодирования
15 видео как части процесса кодирования видео.

11. Устройство декодирования данных видео, содержащее:

модуль фильтра, выполненный с возможностью:

определять направление, по меньшей мере, одного пикселя из блока пикселей,
определять метрику для упомянутого, по меньшей мере, одного пикселя из

20 упомянутого блока пикселей, при этом метрика определяется на основе сравнения пиксельных значений поднабора пикселей в упомянутом блоке с упомянутым, по меньшей мере, одним пикселем, и при этом блок пикселей представляет собой блок пикселей $M \times N$, и при этом, по меньшей мере, одно из M и N составляет 2 или больше;

определять фильтр из набора фильтров, сигнализированного в данных видео, для
25 упомянутого, по меньшей мере, одного пикселя из упомянутого блока пикселей на основе упомянутой метрики и упомянутого направления; и

генерировать отфильтрованное изображение путем применения фильтра к упомянутому, по меньшей мере, одному пикселю упомянутого блока пикселей;

30 запоминающее устройство, выполненное с возможностью хранить отфильтрованный результат модуля фильтра.

12. Устройство по п. 11, в котором направление содержит одно из горизонтального направления, вертикального направления, направления 45 градусов и направления 135 градусов.

13. Устройство по п. 11, в котором модуль фильтра определяет направление
35 посредством сравнения измерения горизонтальной активности с измерением вертикальной активности, и причем горизонтальная активность определяется на основе сравнения пиксельного значения, по меньшей мере, одного пикселя в упомянутом поднаборе с пиксельным значением левого соседнего пикселя и пиксельным значением правого соседнего пикселя, при этом левый соседний пиксель и правый соседний пиксель
40 оба находятся в блоке пикселей.

14. Устройство по п. 13, в котором вертикальная активность определяется на основе сравнения пиксельного значения, по меньшей мере, одного пикселя в упомянутом поднаборе с пиксельным значением верхнего соседнего пикселя и пиксельным значением нижнего соседнего пикселя, при этом верхний соседний пиксель и нижний соседний
45 пиксель оба находятся в упомянутом блоке пикселей.

15. Устройство по п. 11, в котором поднабор пикселей содержит пиксели не на границе блока $M \times N$.

16. Устройство по п. 11, в котором блок пикселей представляет собой блок пикселей

4×4, включающий в себя шестнадцать пикселей, и при этом поднабор пикселей включает в себя четыре пикселя.

17. Устройство по п. 11, в котором модуль фильтра дополнительно выполнен с возможностью определять метрику посредством:

5 сравнения, по меньшей мере, одного пикселя с первым соседним пикселем из упомянутого поднабора пикселей,

в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя больше пиксельного значения первого соседнего пикселя, увеличения переменной метрики края; и

10 в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя меньше пиксельного значения первого соседнего пикселя, уменьшения переменной метрики края.

18. Устройство по п. 17, в котором модуль фильтра дополнительно выполнен с возможностью определять метрику посредством:

15 сравнения, по меньшей мере, одного пикселя со вторым соседним пикселем из поднабора пикселей,

в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя больше пиксельного значения второго соседнего пикселя, увеличения переменной метрики края; и

20 в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя меньше пиксельного значения второго соседнего пикселя, уменьшения переменной метрики края.

19. Устройство по п. 18, в котором

первый соседний пиксель и второй соседний пиксель содержат одно из:

25 левого соседнего пикселя и правого соседнего пикселя,

верхнего соседнего пикселя и нижнего соседнего пикселя,

нижнего левого соседнего пикселя и верхнего правого соседнего пикселя, и

верхнего левого соседнего пикселя и нижнего правого соседнего пикселя.

20. Устройство по п. 11, в котором упомянутое устройство содержит видеокодер, выполненный с возможностью выполнения декодирования видео как части процесса кодирования видео.

21. Аппарат для декодирования данных видео, содержащий:

средство для определения направления, по меньшей мере, одного пикселя из блока пикселей,

35 средство для определения метрики для упомянутого, по меньшей мере, одного пикселя из упомянутого блока пикселей, при этом метрика определяется на основе сравнения пиксельных значений для поднабора пикселей в этом блоке с упомянутым по меньшей мере одним пикселем, и при этом блок пикселей представляет собой блок пикселей M×N, и при этом по меньшей мере одно из M и N составляет 2 или больше;

40 средство для определения фильтра из набора фильтров, сигнализированного в данных видео, для упомянутого, по меньшей мере, одного пикселя из блока пикселей на основе упомянутой метрики и упомянутого направления; и

средство для генерирования отфильтрованного изображения путем применения фильтра к упомянутому, по меньшей мере, одному пикселю из упомянутого блока пикселей.

22. Аппарат по п. 21, в котором направление содержит одно из горизонтального направления, вертикального направления, направления 45 градусов и направления 135 градусов.

23. Аппарат по п. 21, в котором средство для определения метрики содержит средство для сравнения измерения горизонтальной активности с измерением вертикальной активности, и причем средство для сравнения измерения горизонтальной активности с измерением вертикальной активности содержит средство для сравнения пиксельного значения, по меньшей мере, одного пикселя в поднаборе с пиксельным значением левого соседнего пикселя и пиксельным значением правого соседнего пикселя, при этом левый соседний пиксель и правый соседний пиксель оба находятся в блоке пикселей.

24. Аппарат по п. 23, в котором средство для сравнения измерения горизонтальной активности с измерением вертикальной активности дополнительно содержит средство для сравнения пиксельного значения, по меньшей мере, одного пикселя в поднаборе с пиксельным значением верхнего соседнего пикселя и пиксельным значением нижнего соседнего пикселя, при этом верхний соседний пиксель и нижний соседний пиксель оба находятся в блоке пикселей.

25. Аппарат по п. 21, в котором поднабор пикселей содержит пиксели не на границе блока $M \times N$.

26. Аппарат по п. 21, в котором блок пикселей представляет собой блок пикселей 4×4 , включающий в себя шестнадцать пикселей, и при этом поднабор пикселей включает в себя четыре пикселя.

27. Аппарат по п. 21, в котором средство для определения метрики содержит: средство для сравнения, по меньшей мере, одного пикселя с первым соседним пикселем из упомянутого поднабора пикселей,

средство для увеличения переменной метрики края в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя больше пиксельного значения первого соседнего пикселя; и

средство для уменьшения переменной метрики края в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя меньше пиксельного значения первого соседнего пикселя.

28. Аппарат по п. 27, в котором средство для определения метрики дополнительно содержит

средство для сравнения, по меньшей мере, одного пикселя со вторым соседним пикселем из упомянутого поднабора пикселей,

средство для увеличения переменной метрики края в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя больше пиксельного значения второго соседнего пикселя; и

средство для уменьшения переменной метрики края в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя меньше пиксельного значения второго соседнего пикселя.

29. Аппарат по п. 28, в котором первый соседний пиксель и второй соседний пиксель содержат одно из:

левого соседнего пикселя и правого соседнего пикселя,

верхнего соседнего пикселя и нижнего соседнего пикселя,

нижнего левого соседнего пикселя и верхнего правого соседнего пикселя, и

верхнего левого соседнего пикселя и нижнего правого соседнего пикселя.

30. Аппарат по п. 21, в котором аппарат содержит видеокодер, выполненный с возможностью выполнения декодирования видео как части процесса кодирования видео.

31. Машиночитаемый носитель данных, в котором хранятся команды, которые при выполнении заставляют один или более процессоров для декодирования данных видео:

определять направление, по меньшей мере, одного пикселя из блока пикселей, определять метрику для упомянутого, по меньшей мере, одного пикселя из упомянутого блока пикселей, при этом метрика определяется на основе сравнения пиксельных значений для упомянутого поднабора пикселей в упомянутом блоке с упомянутым, по меньшей мере, одним пикселем, и при этом блок пикселей представляет собой блок пикселей $M \times N$, и при этом по меньшей мере одно из M и N составляет 2 или больше;

определять фильтр из набора фильтров, сигнализированного в данных видео, для упомянутого, по меньшей мере, одного пикселя из упомянутого блока пикселей на основе упомянутой метрики и упомянутого направления; и

генерировать отфильтрованное изображение путем применения фильтра к упомянутому, по меньшей мере, одному пикселю из упомянутого блока пикселей.

32. Машиночитаемый носитель данных по п. 31, в котором направление содержит одно из горизонтального направления, вертикального направления, направления 45 градусов и направления 135 градусов.

33. Машиночитаемый носитель данных по п. 31, в котором упомянутые один или более процессоров определяют направление посредством сравнения измерения горизонтальной активности с измерением вертикальной активности, и определяют горизонтальную активность на основе сравнения пиксельного значения, по меньшей мере, одного пикселя в упомянутом поднаборе с пиксельным значением левого соседнего пикселя и пиксельным значением правого соседнего пикселя, при этом левый соседний пиксель и правый соседний пиксель оба находятся в упомянутом блоке пикселей.

34. Машиночитаемый носитель данных по п. 33, в котором упомянутые один или более процессоров определяют вертикальную активность на основе сравнения пиксельного значения, по меньшей мере, одного пикселя в упомянутом поднаборе с пиксельным значением верхнего соседнего пикселя и пиксельным значением нижнего соседнего пикселя, при этом верхний соседний пиксель и нижний соседний пиксель оба находятся в упомянутом блоке пикселей.

35. Машиночитаемый носитель данных по п. 31, в котором поднабор пикселей содержит пиксели не на границе блока $M \times N$.

36. Машиночитаемый носитель данных по п. 31, в котором блок пикселей представляет собой блок пикселей 4×4 , включающий в себя шестнадцать пикселей, и при этом поднабор пикселей включает в себя четыре пикселя.

37. Машиночитаемый носитель данных по п. 31, в котором хранятся дополнительные команды, которые при выполнении заставляют упомянутые один или более процессоров определять метрику посредством:

сравнения, по меньшей мере, одного пикселя с первым соседним пикселем из упомянутого поднабора пикселей,

в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя больше пиксельного значения первого соседнего пикселя, увеличения переменной метрики края; и

в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя меньше пиксельного значения первого соседнего пикселя, уменьшения переменной метрики края.

38. Машиночитаемый носитель данных по п. 37, в котором хранятся дополнительные команды, которые при выполнении заставляют упомянутые один или более процессоров определять метрику посредством:

сравнения, по меньшей мере, одного пикселя со вторым соседним пикселем из

упомянутого поднабора пикселей,

в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя больше пиксельного значения второго соседнего пикселя, увеличения переменной метрики края; и

5 в ответ на то, что пиксельное значение упомянутого, по меньшей мере, одного пикселя меньше пиксельного значения второго соседнего пикселя, уменьшения переменной метрики края.

39. Машиночитаемый носитель данных по п. 38, в котором первый соседний пиксель и второй соседний пиксель содержат одно из:

10 левого соседнего пикселя и правого соседнего пикселя,
верхнего соседнего пикселя и нижнего соседнего пикселя,
нижнего левого соседнего пикселя и верхнего правого соседнего пикселя, и
верхнего левого соседнего пикселя и нижнего правого соседнего пикселя.

40. Машиночитаемый носитель данных по п. 31, в котором один или более
15 процессоров осуществляют декодирование видео как часть процесса кодирования видео.

20

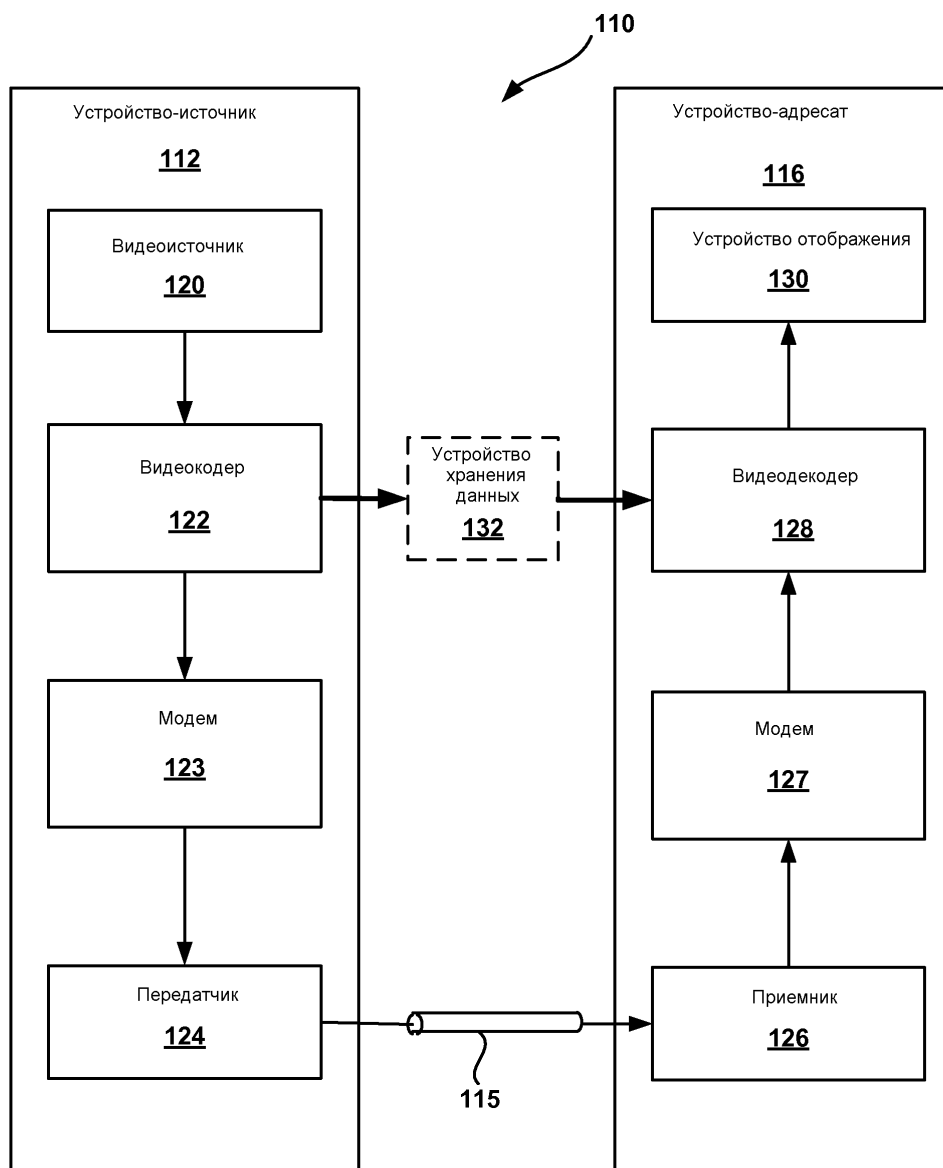
25

30

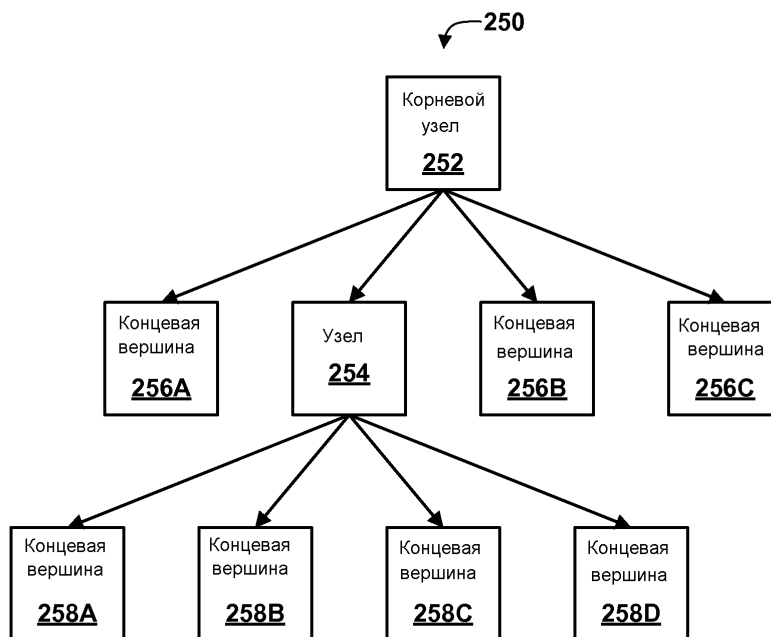
35

40

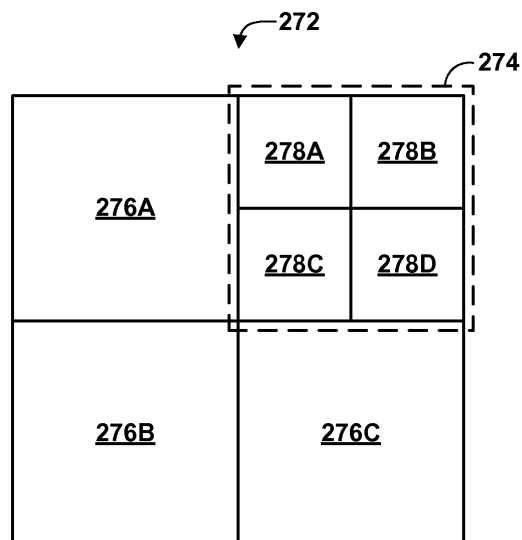
45



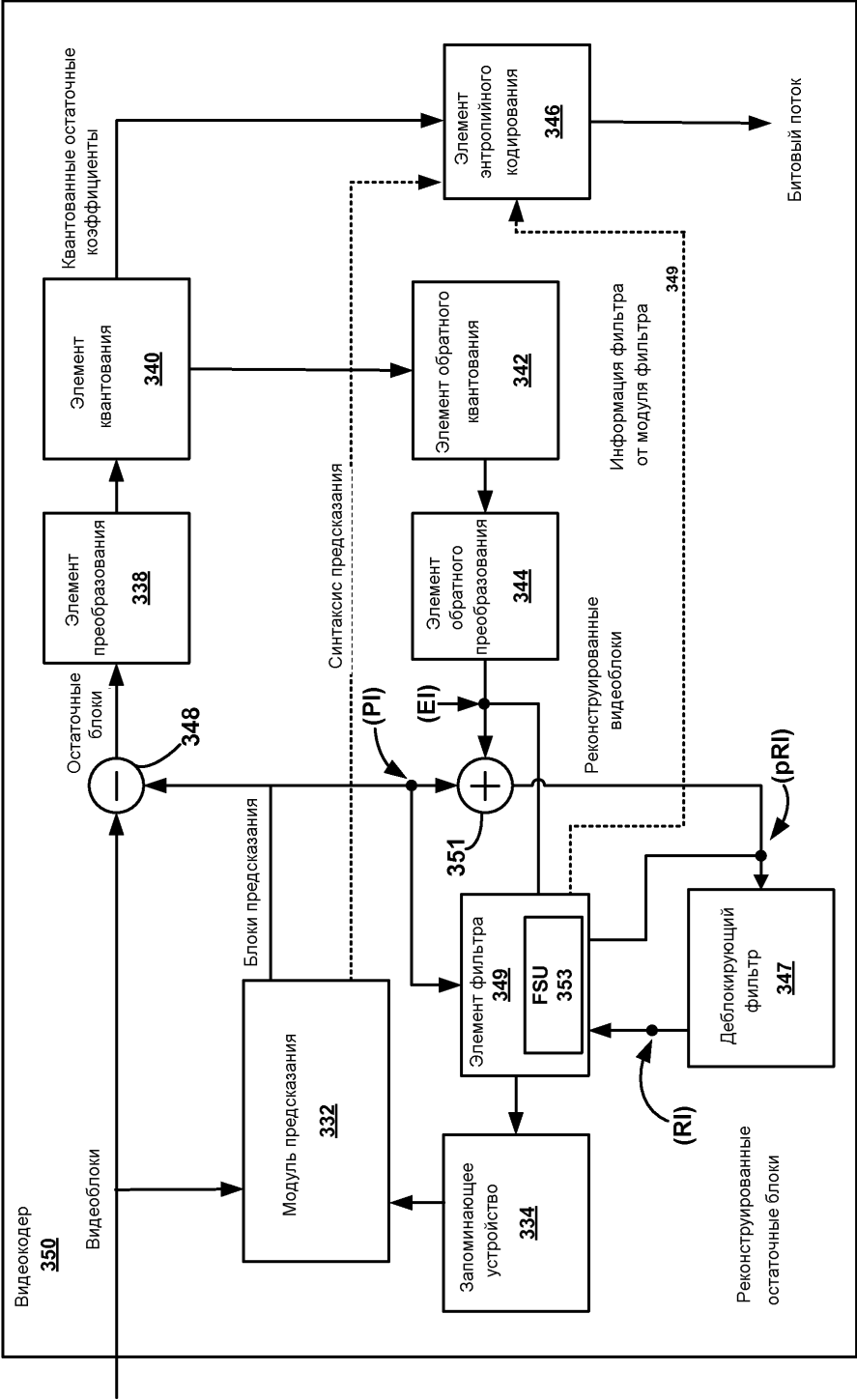
ФИГ.1



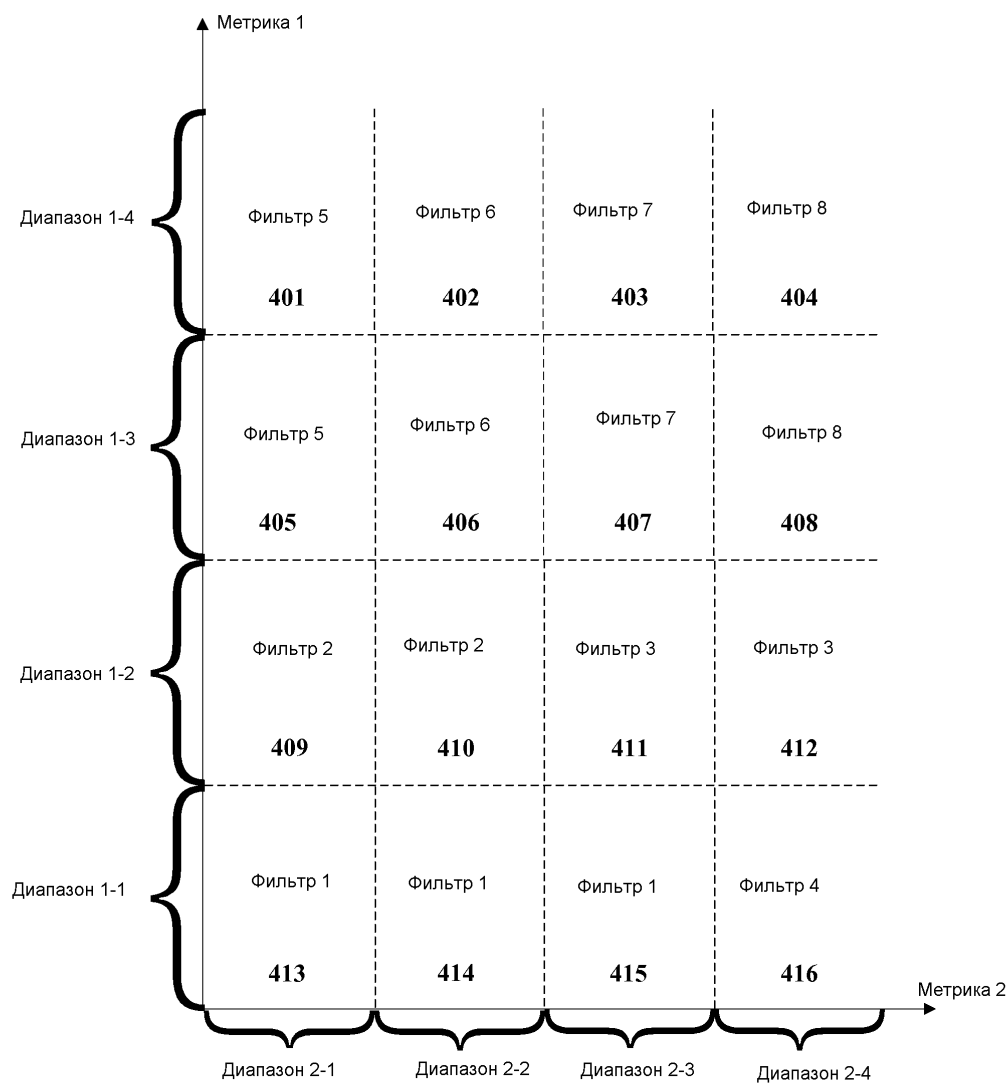
ФИГ.2А



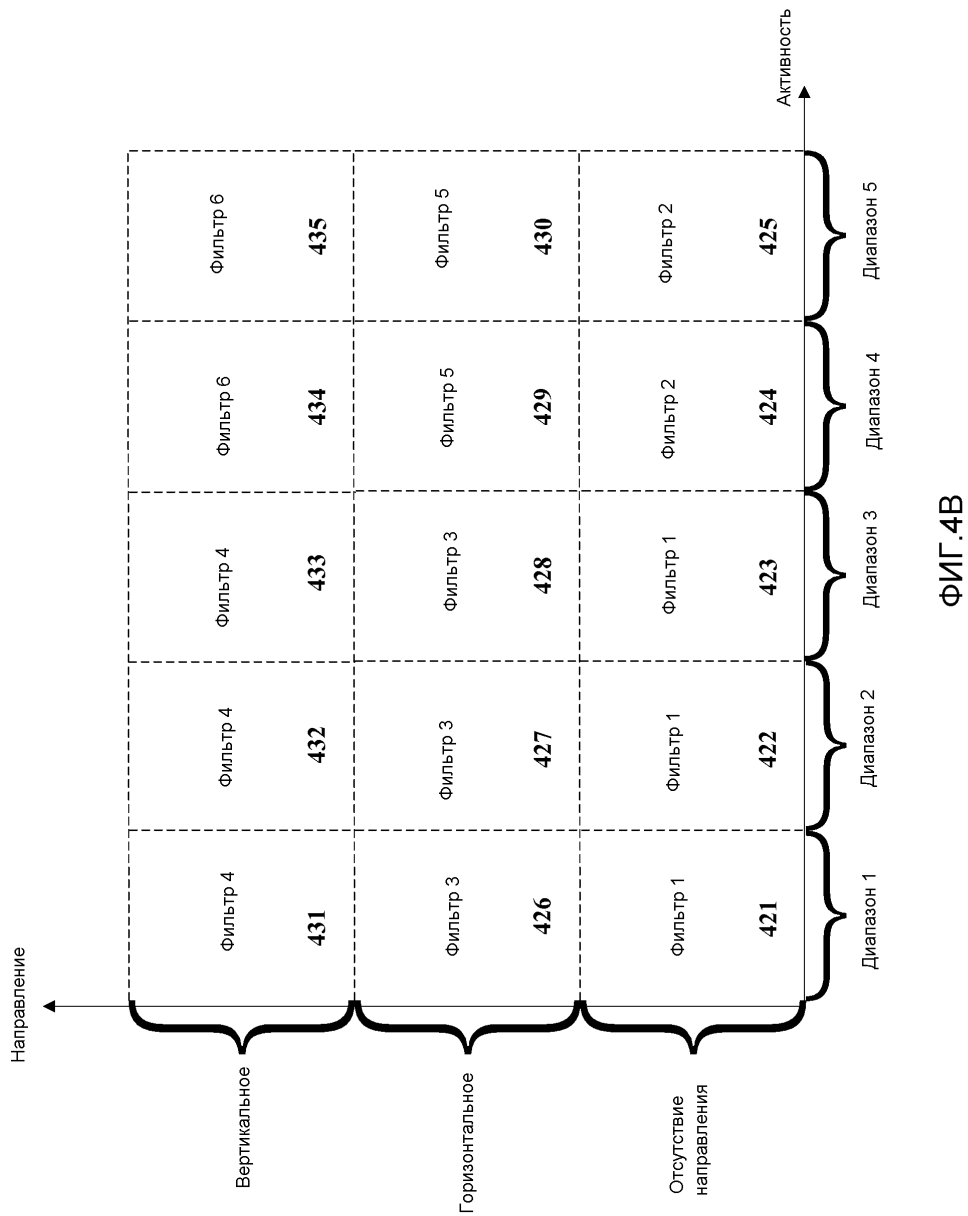
ФИГ.2В

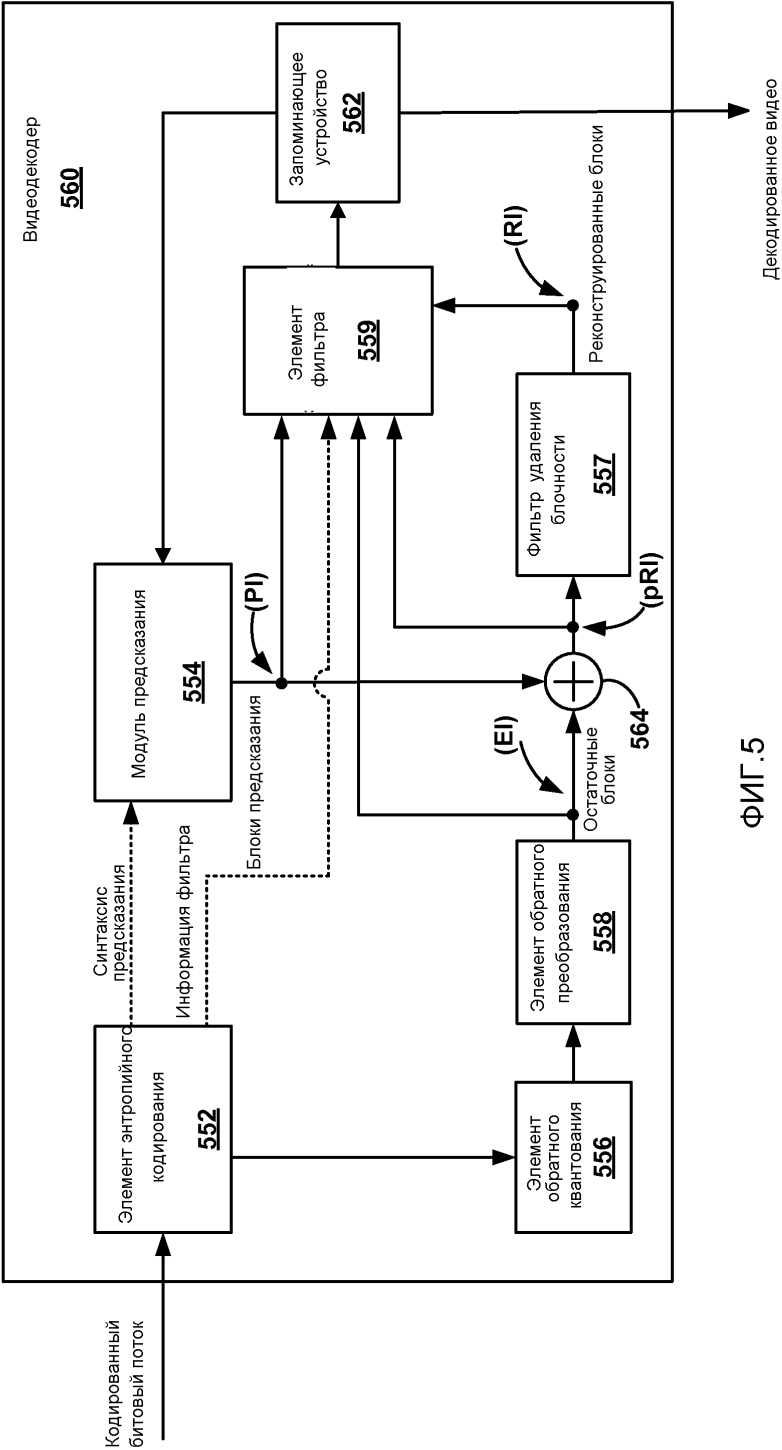


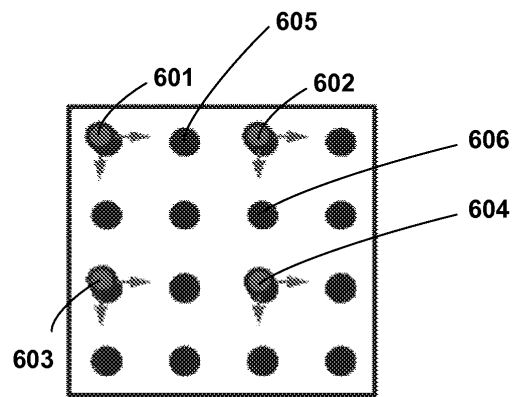
ФИГ.3



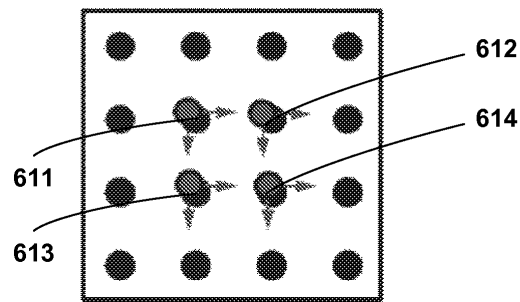
ФИГ.4А



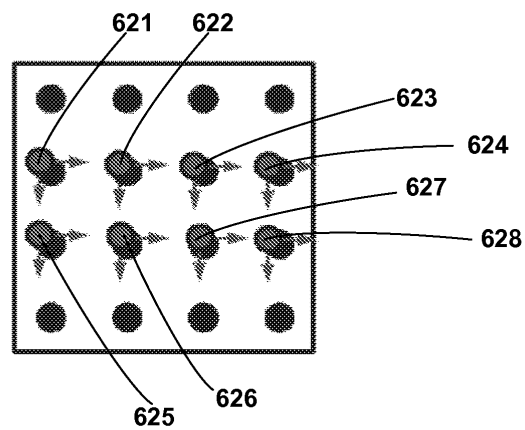




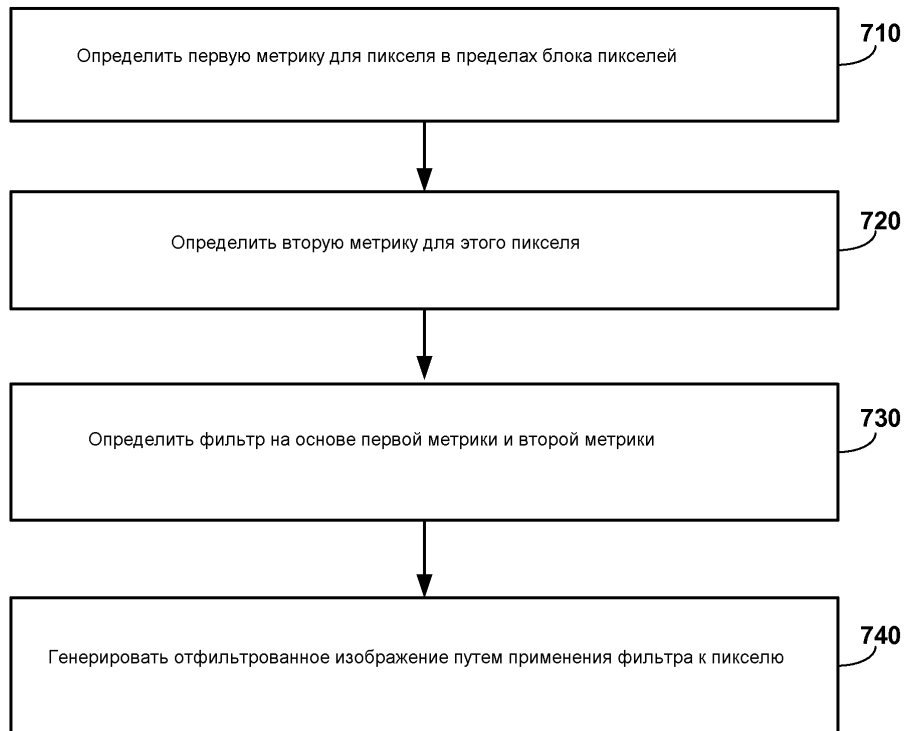
ФИГ.6А



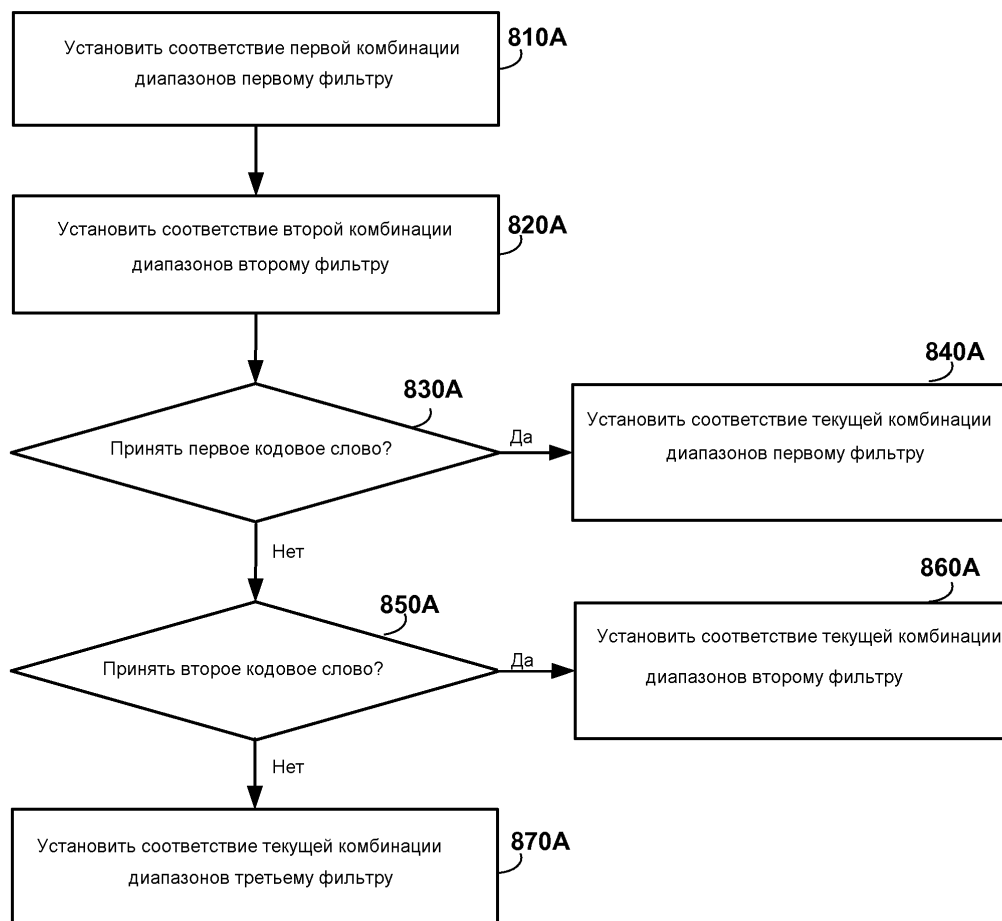
ФИГ.6В



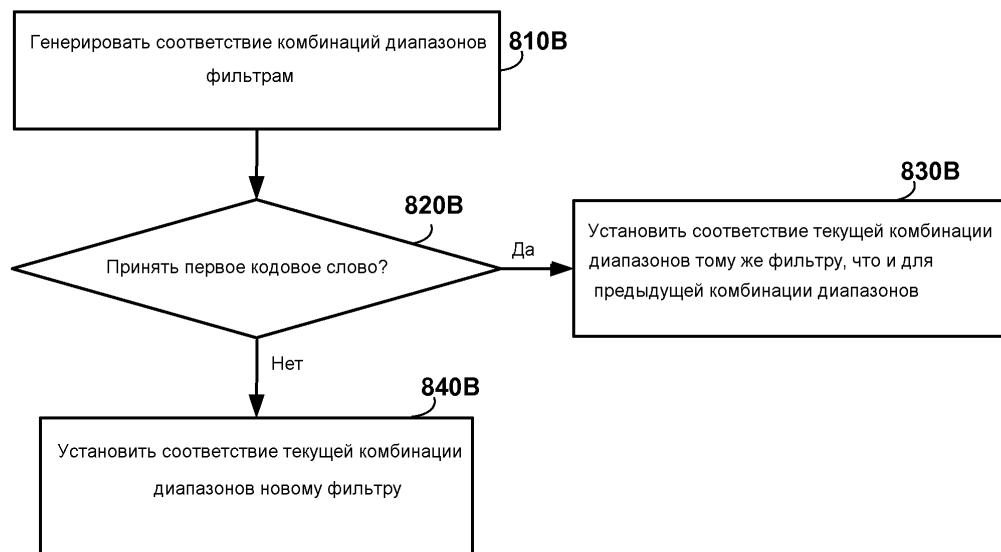
ФИГ.6С



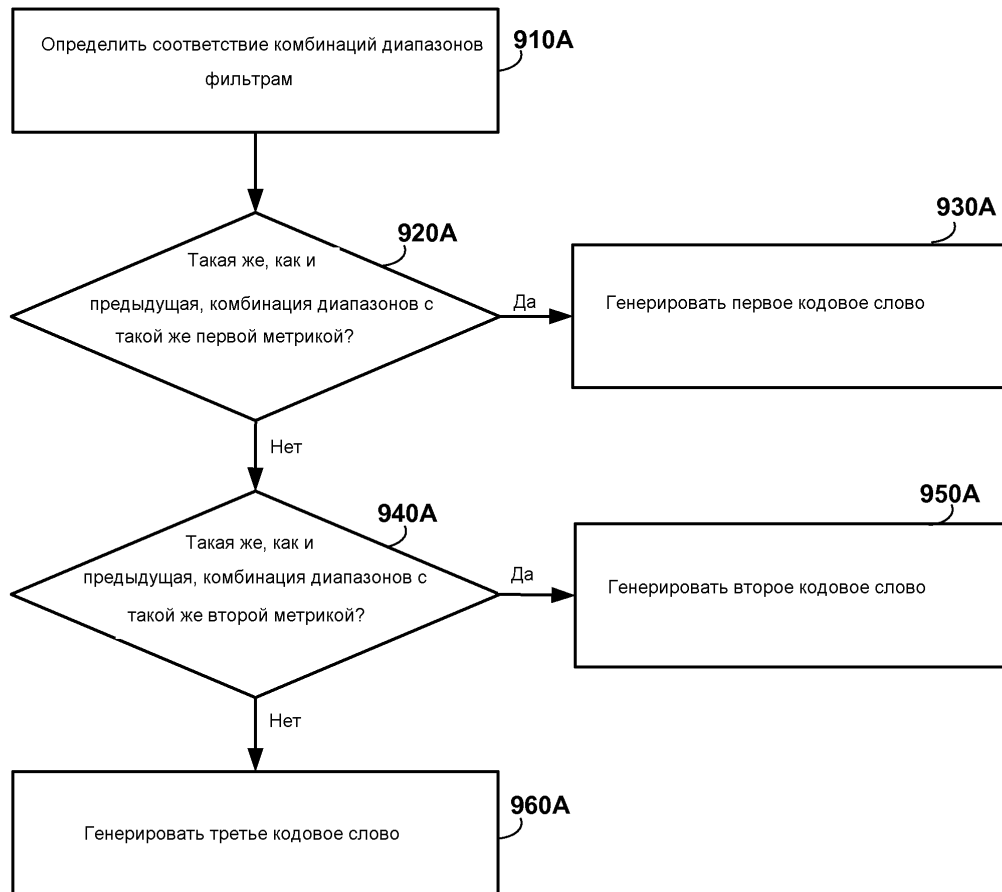
ФИГ.7



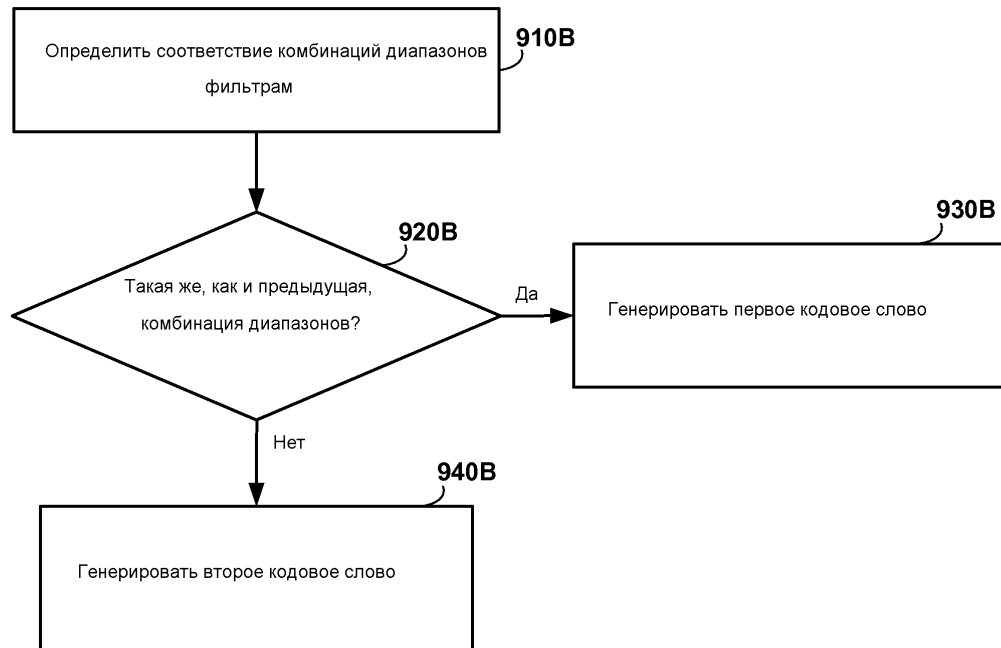
ФИГ.8А



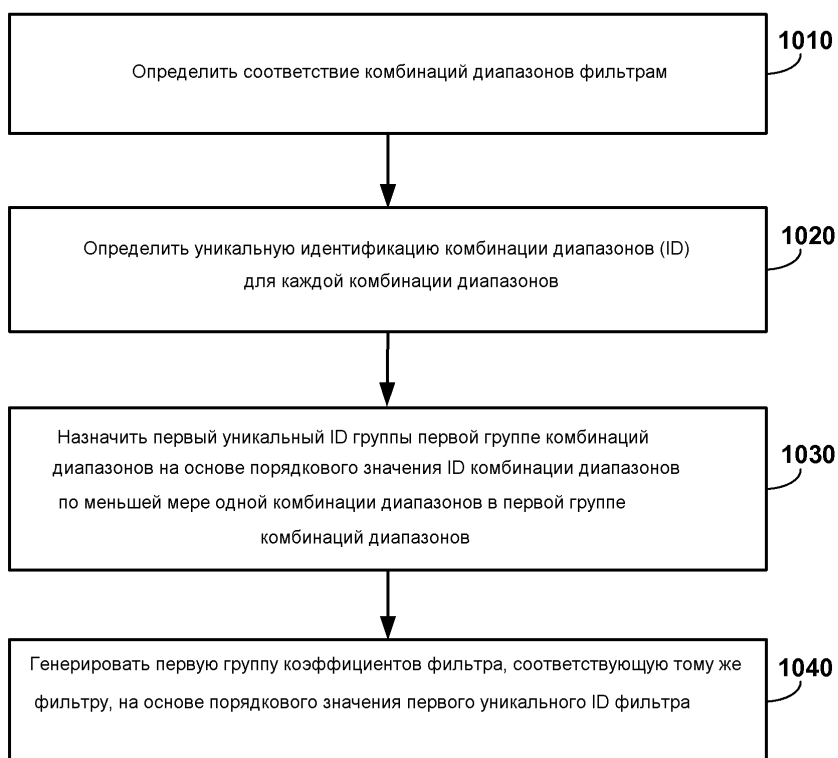
ФИГ.8В



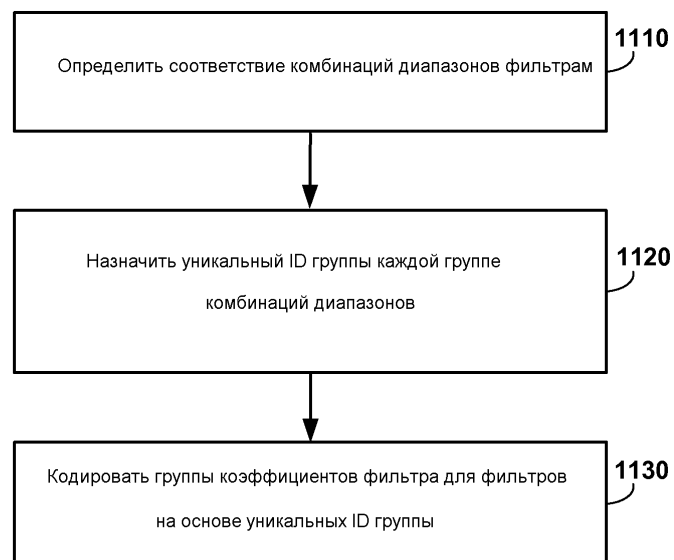
ФИГ.9А



ФИГ.9В



ФИГ.10



ФИГ.11