



(51) МПК
H04N 19/436 (2014.01)
H04N 19/593 (2014.01)
H04N 19/11 (2014.01)

**ФЕДЕРАЛЬНАЯ СЛУЖБА
 ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ**

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК

H04N 19/436 (2006.01); H04N 19/593 (2006.01); H04N 19/11 (2006.01)

(21)(22) Заявка: 2017110397, 30.09.2014

(24) Дата начала отсчета срока действия патента:
30.09.2014

Дата регистрации:
06.02.2019

Приоритет(ы):

(22) Дата подачи заявки: 30.09.2014

(43) Дата публикации заявки: 01.10.2018 Бюл. № 28

(45) Опубликовано: 06.02.2019 Бюл. № 4

(85) Дата начала рассмотрения заявки РСТ на национальной фазе: 29.03.2017

(86) Заявка РСТ:
CN 2014/087885 (30.09.2014)

(87) Публикация заявки РСТ:
WO 2016/049839 (07.04.2016)

Адрес для переписки:
129090, Москва, ул. Б.Спасская, 25, строение 3,
ООО "Юридическая фирма Городиский и
Партнеры"

(72) Автор(ы):

**ЛИ Бинь (CN),
СЮЙ Цзичжэн (CN)**

(73) Патентообладатель(и):

**МАЙКРОСОФТ ТЕКНОЛОДЖИ
ЛАЙСЕНСИНГ, ЭлЭлСи (US)**

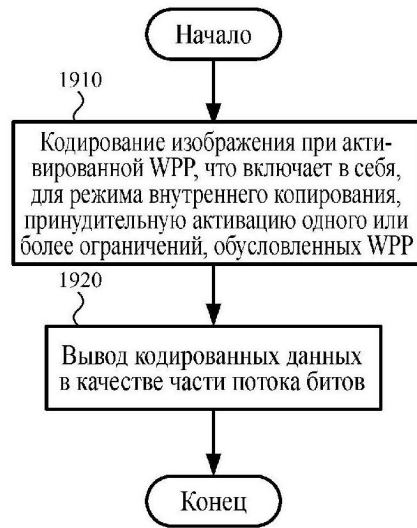
(56) Список документов, цитированных в отчете о поиске: P. LAI et al. "Non-RCE4: Major color table (palette) sharing", JCTVC-P0153 (version 3), опубли. 11.01.2014 на 6 страницах. US 2013/0272370 A1, 17.10.2013. С. PANG et al. "Non-RCE3: Intra Motion Compensation with 2-D MVs", JCTVC-N0256 (version 4), опубли. 02.08.2013 на 12 страницах. DO-KYOUNG KWON et al. "Fast intra block copy (IntraBC) search for (см. прод.)

(54) ПРАВИЛА ДЛЯ РЕЖИМОВ ВНУТРИКАДРОВОГО ПРОГНОЗИРОВАНИЯ, КОГДА АКТИВИРУЕТСЯ ВОЛНОВАЯ ПАРАЛЛЕЛЬНАЯ ОБРАБОТКА ДАННЫХ

(57) Реферат:

Изобретение относится к вычислительной технике. Технический результат заключается в ускорении кодирования при использовании режима внутреннего копирования для текущего изображения. Способ кодирования изображения, используемый в вычислительной системе, в котором кодируют изображение при активированной волновой параллельной обработке (WPP), при этом кодирование формирует закодированные данные, что включает в себя, для прогнозирования с

внутренним блочным копированием, этап, на котором принудительно активируют одно или более ограничений, совместимых с WPP, причем одно или более ограничений включает в себя ограничение, что значение горизонтального сдвига от опорной области до текущей области меньше или равно значению вертикального сдвига от текущей области до опорной области; и выводят закодированные данные в качестве части потока битов. 6 н. и 10 з.п. ф-лы, 22 ил.



ФИГ. 19

(56) (продолжение):

HEVC screen content coding", опубл. 28.07.2014 на 4 страницах. RU 2420021 С2, 27.05.2011.



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(51) Int. Cl.
H04N 19/436 (2014.01)
H04N 19/593 (2014.01)
H04N 19/11 (2014.01)

(12) **ABSTRACT OF INVENTION**

(52) CPC
H04N 19/436 (2006.01); H04N 19/593 (2006.01); H04N 19/11 (2006.01)

(21)(22) Application: **2017110397, 30.09.2014**

(24) Effective date for property rights:
30.09.2014

Registration date:
06.02.2019

Priority:

(22) Date of filing: **30.09.2014**

(43) Application published: **01.10.2018** Bull. № 28

(45) Date of publication: **06.02.2019** Bull. № 4

(85) Commencement of national phase: **29.03.2017**

(86) PCT application:
CN 2014/087885 (30.09.2014)

(87) PCT publication:
WO 2016/049839 (07.04.2016)

Mail address:
**129090, Moskva, ul. B.Spaskaya, 25, stroenie 3,
OOO "Yuridicheskaya firma Gorodisskij i
Partnery"**

(72) Inventor(s):
**LI Bin (CN),
SYUJ Tszichzhen (CN)**

(73) Proprietor(s):
**MAJKROSOFT TEKNOLODZHI
LAJSENSING, EIEISI (US)**

(54) **RULES FOR INTRA-PICTURE PREDICTION MODES WHEN WAVEFRONT PARALLEL PROCESSING IS ENABLED**

(57) Abstract:

FIELD: computer equipment.

SUBSTANCE: invention relates to the computer equipment. Image encoding method used in a computer system in which an image is encoded with wavefront parallel processing (WPP) enabled, wherein the encoding generates encoded data, which includes, for prediction with internal blocking, a step at which one or more constraints compatible with WPP are forcibly activated, wherein one or more constraints includes a

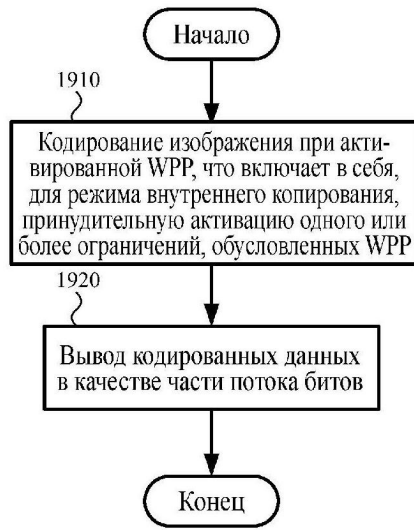
constraint that the value of horizontal shift from the reference area to the current area is less than or equal to the value of vertical shift from the current area to the reference area; and the encoded data are output as part of a bit stream.

EFFECT: encoding acceleration when using internal copy mode for the current image.

16 cl, 22 dwg

RU 2 679 201 C2

RU 2 679 201 C2



ФИГ. 19

Уровень техники

[001] Специалисты используют сжатие (также называемое "исходной кодировкой" или "исходным кодированием"), чтобы уменьшать скорость передачи битов цифрового видео. Сжатие снижает затраты на хранение и передачу видеoinформации посредством преобразования информации в форму с более низкой скоростью передачи битов.

Распаковка (также называемая "декодированием") восстанавливает версию исходной информации из сжатой формы. "Кодек" представляет собой систему кодера/декодера.

[002] За последние 25 лет приняты различные стандарты видеокодеков, включающие в себя стандарты ITU-T H.261, H.262 (MPEG-2 или ISO/IEC 13818-2), H.263 и H.264 (MPEG-4 AVC или ISO/IEC 14496-10), стандарты MPEG-1 (ISO/IEC 11172-2) и MPEG-4 Visual (ISO/IEC 14496-2) и стандарт SMPTE 421M (VC-1). Позднее, утвержден H.265/HEVC-стандарт (ITU-T H.265 или ISO/IEC 23008-2). В настоящее время разрабатываются расширения для H.265/HEVC-стандарта (например, для масштабируемого кодирования/декодирования видео, для кодирования/декодирования видео с более высокой точностью воспроизведения с точки зрения выборочной битовой глубины или частоты дискретизации сигнала цветности, для контента захвата экрана или для многовидового кодирования/декодирования).

Стандарт видеокодеков типично задает опции для синтаксиса закодированного потока видеобитов, детализируя параметры в потоке битов, когда конкретные признаки используются при кодировании и декодировании. Во многих случаях, стандарт видеокодеков также предоставляет подробную информацию касательно операций декодирования, которые должен выполнять декодер, чтобы достигать надлежащих результатов декодирования. Помимо стандартов кодеков, различные собственные форматы кодеков задают другие опции для синтаксиса закодированного потока видеобитов и соответствующих операций декодирования.

[003] Волновая параллельная обработка (WPP) представляет собой инструментальное средство, доступное для кодирования и декодирования в H.265/HEVC-стандарте. Когда WPP активируется, часть изображения разделяется на строки специальных секций, называемых "единицами дерева кодирования (CTU)". В ходе кодирования или декодирования, первая строка CTU может обрабатываться "CTU за CTU", слева направо. Обработка (кодирование или декодирование) вторых строк CTU не должна ожидать завершения обработки для первой строки CTU. Вместо этого, обработка второй строки может начинаться после того, как обработка завершается для нескольких CTU первой строки, которые предоставляют информацию, используемую при обработке начальной CTU второй строки. Аналогично, обработка третьей строки CTU может начинаться после того, как обработка завершается для нескольких CTU второй строки. WPP упрощает параллельную обработку различных строк CTU. Различные подпроцессы или ядра обработки могут выполнять обработку для различных строк CTU на основе временной задержки со сдвигом.

[004] Внутреннее блочное копирование (BC) представляет собой разрабатываемый режим прогнозирования для расширений H.265/HEVC. Для режима прогнозирования с внутренним BC, значения выборок текущего блока изображения прогнозируются с использованием ранее восстановленных значений выборок в том же изображении. Блочный вектор (BV) указывает сдвиг от текущего блока до опорного блока изображения, которое включает в себя ранее восстановленные значения выборок, используемые для прогнозирования. BV передается в служебных сигналах в потоке битов. Прогнозирование с внутренним BC представляет собой форму внутрикадрового прогнозирования, причем прогнозирование с внутренним BC для блока изображения не использует значения выборок, за исключением значений выборок в том же

изображении. Режим внутреннего строкового копирования (SC) и режим внутреннего линейного копирования (LC) являются другими примерами режимов внутрикадрового прогнозирования, которые, аналогично режиму внутреннего ВС, используют значение смещения для того, чтобы указывать сдвиг до позиции в ранее восстановленных значениях выборок, используемых для прогнозирования. Режим палитрового прогнозирования, который является другим примером режима внутрикадрового прогнозирования, прогнозирует палитру, используемую для того, чтобы представлять цвета в такой секции, как единица кодирования (CU). Как указано в настоящее время для расширений H.265/HEVC-стандарта, различные режимы внутрикадрового прогнозирования не используются эффективно, когда WPP активируется.

Сущность изобретения

[005] В общих словах, подробное описание представляет новшества в правилах, принудительно активированных для режимов внутрикадрового прогнозирования, когда волновая параллельная обработка (WPP) активируется. Синтаксический элемент в потоке битов может указывать то, активируется или нет WPP для видеопоследовательности, набора изображений или изображения. Новшества упрощают использование режимов внутрикадрового прогнозирования, к примеру, режима палитрового прогнозирования, режима внутреннего блочного копирования, режима внутреннего линейного копирования и режима внутреннего строкового копирования, посредством кодера или декодера, когда WPP активируется.

[006] Согласно одному аспекту новшеств, описанных в данном документе, кодер кодирует изображение при активированной WPP. Кодирование формирует закодированные данные. Для режима палитрового кодирования кодер прогнозирует палитру для начальной единицы в текущей WPP-строке изображения с использованием предыдущих палитровых данных из предыдущей единицы в предыдущей WPP-строке изображения. Кодер выводит закодированные данные в качестве части потока битов.

[007] Соответствующий декодер принимает закодированные данные в качестве части потока битов. Декодер декодирует закодированные данные при активированной WPP. Декодирование восстанавливает изображение. Для режима палитрового декодирования, декодер прогнозирует палитру для начальной единицы в текущей WPP-строке изображения с использованием предыдущих палитровых данных из предыдущей единицы в предыдущей WPP-строке изображения.

[008] Согласно другому аспекту новшеств, описанных в данном документе, кодер кодирует изображение при активированной WPP. Кодирование формирует закодированные данные. Для режима внутреннего копирования (например, режима внутреннего блочного копирования, режима внутреннего строкового копирования, режима внутреннего линейного копирования), кодер принудительно активирует одно или более ограничений, обусловленных WPP. Кодер выводит закодированные данные в качестве части потока битов.

[009] Соответствующий декодер принимает закодированные данные в качестве части потока битов. Для режима внутреннего копирования (например, режима внутреннего блочного копирования, режима внутреннего строкового копирования, режима внутреннего линейного копирования), закодированные данные удовлетворяют одному или более ограничений, обусловленных WPP. Декодер декодирует закодированные данные при активированной WPP. Декодирование восстанавливает изображение.

[010] Новшества могут реализовываться в качестве части способа, в качестве части вычислительной системы, сконфигурированной с возможностью осуществлять способ, или в качестве части материальных машиночитаемых носителей, хранящих

машиноисполняемые инструкции для вынуждения вычислительной системы осуществлять способ. Различные новшества могут использоваться в комбинации или отдельно. Эта сущность изобретения предоставлена для того, чтобы представлять в упрощенной форме выбор концепций, которые дополнительно описаны ниже в
5 подробном описании. Эта сущность не предназначена для того, чтобы идентифицировать ключевые признаки или важнейшие признаки заявленной сущности изобретения, как и не предназначена для использования таким образом, чтобы ограничивать объем заявляемой сущности изобретения. Вышеприведенные и другие цели, признаки и преимущества изобретения должны становиться более очевидными из нижеприведенного
10 подробного описания, которое приводится со ссылками на прилагаемые чертежи.

Краткое описание чертежей

[011] Фиг. 1 является схемой примерной вычислительной системы, в которой могут реализовываться некоторые описанные варианты осуществления.

[012] Фиг. 2a и 2b являются схемами примерных сетевых окружений, в которых могут
15 реализовываться некоторые описанные варианты осуществления.

[013] Фиг. 3 является схемой примерной системы кодера, в сочетании с которой могут реализовываться некоторые описанные варианты осуществления.

[014] Фиг. 4 является схемой примерной системы декодера, в сочетании с которой могут реализовываться некоторые описанные варианты осуществления.

[015] Фиг. 5a и 5b являются схемами, иллюстрирующими примерный видеокодер, в
20 сочетании с которым могут реализовываться некоторые описанные варианты осуществления.

[016] Фиг. 6 является схемой, иллюстрирующей примерный видеodeкодер, в сочетании с которым могут реализовываться некоторые описанные варианты осуществления.

[017] Фиг. 7 является схемой, иллюстрирующей синхронизацию WPP, а фиг. 8 является схемой, иллюстрирующей восстановленный контент, который может использоваться для прогнозирования, когда WPP активируется.

[018] Фиг. 9 и 10 являются схемами, иллюстрирующими аспекты палитрового прогнозирования, согласно некоторым описанным вариантам осуществления.

[019] Фиг. 11 и 12 являются блок-схемами последовательности операций способа, иллюстрирующими кодирование и декодирование, соответственно, с палитровым прогнозированием, когда WPP активируется, согласно некоторым описанным вариантам
30 осуществления.

[020] Фиг. 13 и 14 являются схемами, иллюстрирующими аспекты примера прогнозирования внутреннего блочного копирования для текущего блока изображения.

[021] Фиг. 15 и 16 являются схемами, иллюстрирующими аспекты примеров прогнозирования с внутренним линейным копированием и прогнозирования с внутренним строковым копированием, соответственно.

[022] Фиг. 17 является схемой, иллюстрирующей примерный порядок Z-сканирования
40 для единиц изображений.

[023] Фиг. 18 является схемой, иллюстрирующей пример ограничений на местоположение опорной области для режима внутреннего копирования, когда WPP активируется, согласно некоторым описанным вариантам осуществления.

[024] Фиг. 19 и 20 являются блок-схемами последовательности операций способа, иллюстрирующими кодирование и декодирование, соответственно, с правилами, принудительно активированными для режима внутреннего копирования, когда WPP активируется, согласно некоторым описанным вариантам осуществления.

Подробное описание изобретения

[025] Подробное описание представляет новшества в правилах, принудительно активированных для режимов внутрикадрового прогнозирования, когда волновая параллельная обработка (WPP) активируется. Например, некоторые новшества связаны с прогнозированием палитр для режима палитрового кодирования/декодирования, когда WPP активируется. Другие новшества связаны с ограничениями, принудительно активированными в ходе режима внутреннего копирования (такого как режим внутреннего блочного копирования, режим внутреннего линейного копирования или режим внутреннего строкового копирования), когда WPP активируется. Новшества упрощают использование режимов внутрикадрового прогнозирования посредством кодера или декодера, когда WPP активируется.

[026] Хотя операции, описанные в данном документе, находятся в местах, описанных как выполняемые посредством видеокодера или видеодекодера, во многих случаях операции могут выполняться посредством другого типа инструментального средства обработки мультимедиа (например, кодера изображений или декодера изображений).

[027] Некоторые новшества, описанные в данном документе, проиллюстрированы со ссылкой на термины, конкретные для расширений H.265/HEVC-стандарта. Например, следует обратиться к предварительной версии JCTVC-R1005 расширений для кодирования/декодирования экранного контента для H.265/HEVC-стандарта: "High Efficiency Video Coding (HEVC) Screen Content Coding: Draft 1", JCTVC-R1005 v2, август 2014 года. Новшества, описанные в данном документе, также могут реализовываться для других стандартов или форматов.

[028] Многие новшества, описанные в данном документе, могут повышать производительность искажения в зависимости от скорости передачи при кодировании некоторого "искусственно созданного" видеоконтента, такого как контент захвата экрана. В общем, видео захвата экрана (также называемое "видео экранного контента") представляет собой видео, которое содержит подготовленный посредством рендеринга текст, компьютерную графику, сформированный с помощью анимации контент или другие аналогичные типы контента, захваченного при рендеринге на компьютерном дисплее, в отличие от только захваченного камерой видеоконтента. Контент захвата экрана типично включает в себя повторяющиеся структуры (например, графику, текстовые символы). Контент захвата экрана обычно закодирован в формате (например, YUV 4:4:4 или RGB 4:4:4) с высоким разрешением дискретизации сигнала цветности, хотя он также может быть закодирован в формате с более низким разрешением дискретизации сигнала цветности (например, YUV 4:2:0). Общие сценарии для кодирования/декодирования контента захвата экрана включают в себя конференц-связь с использованием удаленного рабочего стола и кодирование/декодирование графических наложений на естественное видео или другое видео "со смешанным контентом". Некоторые из новшеств, описанных в данном документе, сконфигурированы с возможностью кодирования видео экранного контента или другого искусственно созданного видео. Эти новшества также могут использоваться для естественного видео, но могут не быть настолько эффективными.

[029] Если обобщать, возможны различные альтернативы примерам, описанным в данном документе.

Например, некоторые способы, описанные в данном документе, могут быть изменены посредством изменения упорядочения описанных этапов способа, посредством разбиения, повторения или опускания некоторых этапов способа и т.д. Различные аспекты раскрытой технологии могут использоваться в комбинации или отдельно. Различные варианты осуществления используют одно или более описанных новшеств.

Некоторые новшества, описанные в данном документе, разрешают одну или более проблем, отмеченных в разделе "Уровень техники". Типично, данная технология/инструментальное средство не разрешает все такие проблемы.

1. Примерные вычислительные системы

5 [030] Фиг. 1 иллюстрирует обобщенный пример подходящей вычислительной системы (100), в которой могут реализовываться несколько из описанных новшеств.

Вычислительная система (100) не имеет намерение накладывать какие-либо ограничения в отношении объема использования или функциональности, поскольку новшества могут реализовываться во множестве вычислительных систем общего назначения или
10 специального назначения.

[031] Со ссылкой на фиг. 1, вычислительная система (100) включает в себя один или более процессоров (110, 115) и запоминающее устройство (120, 125). Процессоры (110, 115) выполняют машиноисполняемые инструкции. Процессор может представлять собой центральный процессор (CPU) общего назначения, процессор в
15 специализированной интегральной схеме (ASIC) либо любой другой тип процессора.

В многопроцессорной системе, несколько процессоров выполняют машиноисполняемые инструкции, с тем чтобы повышать вычислительную мощность. Например, фиг. 1 показывает центральный процессор (110), а также графический процессор или сопроцессор (115). Материальное запоминающее устройство (120, 125) может
20 представлять собой энергозависимое запоминающее устройство (например, регистры, кэш, RAM), энергонезависимое запоминающее устройство (например, ROM, EEPROM, флэш-память и т.д.) или некоторую комбинацию вышеозначенного, доступную посредством процессора(ов). Запоминающее устройство (120, 125) сохраняет программное обеспечение (180), реализующее одно или более новшеств для правил,
25 принудительно активированных для режимов внутрикадрового прогнозирования, когда WPP активируется, в форме машиноисполняемых инструкций, подходящих для выполнения посредством процессора(ов).

[032] Вычислительная система может иметь дополнительные признаки. Например, вычислительная система (100) включает в себя устройство (140) хранения, устройства
30 (150) ввода, одно или более устройств (160) вывода и одно или более соединений (170) связи. Механизм межкомпонентного соединения (не показан), такой как шина, контроллер или сеть, соединяет между собой компоненты вычислительной системы (100). Типично, программное обеспечение операционной системы (не показано) предоставляет операционное окружение для другого программного обеспечения,
35 выполняемого в вычислительной системе (100), и координирует действия компонентов вычислительной системы (100).

[033] Материальное устройство (140) данных хранения может быть съемным или стационарным и включает в себя в себя магнитные диски, магнитные ленты или кассеты, CD-ROM, DVD или любой другой носитель, который может быть использован, чтобы
40 сохранять информацию, и к которому можно осуществлять доступ в пределах вычислительной системы (100). Устройство (140) хранения данных хранит инструкции для программного обеспечения (180), реализующего одно или более новшеств для правил, принудительно активированных для режимов внутрикадрового прогнозирования, когда WPP активируется.

45 [034] Устройство (150) ввода может представлять собой устройство сенсорного ввода, такое как клавиатура, мышь, перо или шаровой манипулятор, устройство голосового ввода, устройство сканирования или другое устройство, которое предоставляет ввод в вычислительную систему (100). Для видео, устройство (150) ввода

может представлять собой камеру, видеокарту, карту ТВ-тюнера, модуль захвата экрана или аналогичное устройство, которое принимает видеоввод в аналоговой или цифровой форме, либо CD-ROM или CD-RW, который считывает видеоввод в вычислительную систему (100). Устройство (160) вывода может представлять собой дисплей, принтер, динамик, устройство записи CD-RW или другое устройство, которое предоставляет вывод из вычислительной системы (100).

[035] Соединения (170) связи обеспечивают связь по среде связи с другим вычислительным объектом. Среда связи передает такую информацию, как машиноисполняемые инструкции, аудио- или видеоввод или вывод либо другие данные, в модулированном сигнале данных. Модулированный сигнал данных представляет собой сигнал, который имеет одну или более характеристик, заданных или измененных таким образом, чтобы кодировать информацию в сигнале. В качестве примера, а не ограничения, среды связи могут использовать электрическую, оптическую, RF- или другую несущую.

[036] Новшества могут быть описаны в общем контексте машиночитаемых носителей.

Машиночитаемые носители представляют собой любые доступные материальные носители, к которым можно осуществлять доступ в вычислительном окружении. В качестве примера, а не ограничения, в вычислительной системе (100) машиночитаемые носители включают в себя запоминающее устройство (120, 125), устройство (140) хранения данных и комбинации любых вышеозначенных элементов.

[037] Новшества могут быть описаны в общем контексте машиноисполняемых инструкций, таких как машиноисполняемые инструкции, включенные в программные модули, выполняемые в вычислительной системе на целевом реальном или виртуальном процессоре. В общем, программные модули включают в себя процедуры, программы, библиотеки, объекты, классы, компоненты, структуры данных и т.д., которые выполняют конкретные задачи или реализуют конкретные абстрактные типы данных. Функциональность программных модулей может быть комбинирована или разделена между программными модулями, как требуется в различных вариантах осуществления. Машиноисполняемые инструкции для программных модулей могут выполняться в локальной или распределенной вычислительной системе.

[038] Термины "система" и "устройство" используются взаимозаменяемо в данном документе. Если контекст явно не указывает иное, ни один из терминов не подразумевает ограничения на тип вычислительной системы или вычислительного устройства. В общем, вычислительная система или вычислительное устройство могут быть локальными или распределенными и могут включать в себя любую комбинацию аппаратных средств специального назначения и/или аппаратных средств общего назначения с программным обеспечением, реализующим функциональность, описанную в данном документе.

[039] Раскрытые способы также могут реализовываться с использованием специализированных вычислительных аппаратных средств, сконфигурированных с возможностью осуществлять любой из раскрытых способов. Например, раскрытые способы могут реализовываться посредством интегральной схемы (например, ASIC (такой как процессор цифровых сигналов (DSP) на ASIC, графический процессор (GPU) или программируемое логическое устройство (PLD), такое как программируемая пользователем вентиляционная матрица (FPGA)) специально сконструированной или сконфигурированной с возможностью реализовывать любой из раскрытых способов.

[040] Для целей представления, подробное описание использует такие термины, как "определять" и "использовать", для того чтобы описывать компьютерные операции в вычислительной системе. Эти термины являются высокоуровневыми абстракциями для

операций, выполняемых компьютером, и не должны путаться с действиями, выполняемыми человеком. Фактические компьютерные операции, соответствующие этим терминам, различаются в зависимости от реализации.

II. Примерные сетевые окружения

5 [041] Фиг. 2а и 2b показывают примерные сетевые окружения (201, 202), которые включают в себя видеокодеры (220) и декодеры (270). Кодеры (220) и декодеры (270) соединены по сети (250) с использованием надлежащего протокола связи. Сеть (250) может включать в себя Интернет или другую компьютерную сеть.

10 [042] В сетевом окружении (201), показанном на фиг. 2а, каждое инструментальное средство (210) для обмена данными в реальном времени (RTC) включает в себя как кодер (220), так и декодер (270) для двунаправленной связи. Данный кодер (220) может формировать вывод, совместимый с варьированием или расширением H.265/HEVC-стандарта, стандартом SMPTE 421M, стандартом ISO-IEC 14496-10 (также известным как H.264 или AVC), другим стандартом или собственным форматом, причем
15 соответствующий декодер (270) принимает закодированные данные из кодера (220). Двунаправленная связь может быть частью видеоконференции, видеотелефонного вызова либо другого сценария двусторонней или многосторонней связи. Хотя сетевое окружение (201) на фиг. 2а включает в себя два инструментальных средства (210) для обмена данными в реальном времени, сетевое окружение (201) вместо этого может
20 включать в себя три или более инструментальных средства (210) для обмена данными в реальном времени, которые участвуют в многосторонней связи.

[043] Инструментальное средство (210) для обмена данными в реальном времени управляет кодированием посредством кодера (220). Фиг. 3 показывает примерную систему (300) кодера, которая может быть включена в инструментальное средство (210)
25 для обмена данными в реальном времени. Альтернативно, инструментальное средство (210) для обмена данными в реальном времени использует другую систему кодера. Инструментальное средство (210) для обмена данными в реальном времени также управляет декодированием посредством декодера (270). Фиг. 4 показывает примерную систему (400) декодера, которая может быть включена в инструментальное средство
30 (210) для обмена данными в реальном времени. Альтернативно, инструментальное средство (210) для обмена данными в реальном времени использует другую систему декодера.

[044] В сетевом окружении (202), показанном на фиг. 2b, инструментальное средство (212) кодирования включает в себя кодер (220), который кодирует видео для доставки
35 в несколько инструментальных средств (214) воспроизведения, которые включают в себя декодеры (270). Однонаправленная связь может предоставляться для системы видеонаблюдения, системы мониторинга на основе веб-камер, представления конференц-связи с использованием удаленного рабочего стола или другого сценария, в котором видео закодировано и отправляется из одного местоположения в одно или более других
40 местоположений. Хотя сетевое окружение (202) на фиг. 2b включает в себя два инструментальных средства (214) воспроизведения, сетевое окружение (202) может включать в себя большее или меньшее число инструментальных средств (214) воспроизведения. В общем, инструментальное средство (214) воспроизведения обменивается данными с инструментальным средством (212) кодирования, чтобы
45 определять поток видео для приема посредством инструментального средства (214) воспроизведения. Инструментальное средство (214) воспроизведения принимает поток, буферизует принимаемые закодированные данные в течение надлежащего периода и начинает декодирование и воспроизведение.

[045] Фиг. 3 показывает примерную систему (300) кодера, которая может быть включена в инструментальное средство (212) кодирования. Альтернативно, инструментальное средство (212) кодирования использует другую систему кодера. Инструментальное средство (212) кодирования также может включать в себя серверную логику контроллера для управления соединениями с одним или более инструментальными средствами (214) воспроизведения. Фиг. 4 показывает примерную систему (400) декодера, которая может быть включена в инструментальное средство (214) воспроизведения. Альтернативно, инструментальное средство (214) воспроизведения использует другую систему декодера. Инструментальное средство (214) воспроизведения также может включать в себя клиентскую логику контроллера для управления соединениями с инструментальным средством (212) кодирования.

III. Примерные системы кодера

[046] Фиг. 3 является блок-схемой примерной системы (300) кодера, в сочетании с которой могут реализовываться некоторые описанные варианты осуществления. Система (300) кодера может представлять собой инструментальное средство кодирования общего назначения, допускающее работу в любом из нескольких режимов кодирования, таких как режим кодирования с низкой задержкой для обмена данными в реальном времени, режим транскодирования и режим кодирования с более высокой задержкой для формирования мультимедиа для воспроизведения из файла или потока, либо она может представлять собой инструментальное средство кодирования специального назначения, адаптированное для одного такого режима кодирования. Система (300) кодера может быть сконфигурирована с возможностью кодирования конкретного типа контента (например, контента захвата экрана). Система (300) кодера может реализовываться в качестве части модуля операционной системы, в качестве части библиотеки приложений, в качестве части автономного приложения или с использованием аппаратных средств специального назначения. В целом, система (300) кодера принимает последовательность исходных видеоизображений (311) из видеоисточника (310) и формирует закодированные данные в качестве вывода в канал (390). Закодированные данные, выводимые в канал, могут включать в себя контент, закодированный с использованием правил, принудительно активированных для режимов внутрикадрового прогнозирования, когда WPP активируется.

[047] Видеоисточник (310) может представлять собой камеру, карту тюнера, носители хранения данных, модуль захвата экрана или другой цифровой видеоисточник. Видеоисточник (310) формирует последовательность видеоизображений на частоте кадров, например, 30 кадров в секунду. При использовании в данном документе термин "изображение", в общем, означает исходные, закодированные или восстановленные данные изображений. Для видео с построчным сканированием, изображение представляет собой видеокادر с построчным сканированием. Для чересстрочного видео, в примерных вариантах осуществления, чересстрочная развертка чересстрочного видеокадра может устраняться до кодирования. Альтернативно, два комплементарных чересстрочных видеополя закодированы вместе в качестве одного видеокадра или закодированы в качестве двух отдельно кодируемых полей. Кроме указания видеокадра с построчным сканированием или видеокадра с чересстрочным сканированием термин "изображение" может указывать одно неспаренное видеополе, комплементарную пару видеополей, плоскость видеообъектов, которая представляет видеообъект в данное время, или интересующую область в большем изображении. Плоскость или область видеообъектов может быть частью большего изображения, которое включает в себя несколько объектов или областей сцены.

[048] Поступающее исходное изображение (311) сохраняется во временной области (320) хранения запоминающего устройства исходных изображений, которая включает в себя несколько областей (321, 322,..., 32n) хранения буфера изображений. Буфер (321, 322 и т.д.) изображений хранит одно исходное изображение в области (320) хранения исходных изображений. После того, как одно или более исходных изображений (311) сохранены в буферах (321, 322 и т.д.) изображений, модуль (330) выбора изображений выбирает отдельное исходное изображение из области (320) хранения исходных изображений. Порядок, в котором изображения выбираются посредством модуля (330) выбора изображений для ввода в кодер (340), может отличаться от порядка, в котором изображения формируются посредством видеоисточника (310), например, кодирование некоторых изображений может задерживаться, чтобы обеспечивать возможность некоторым последующим изображениям быть закодированными сначала и за счет этого упрощать временно обратное прогнозирование. До кодера (340), система (300) кодера может включать в себя препроцессор (не показан), который выполняет предварительную обработку (например, фильтрацию) выбранного изображения (331) перед кодированием. Предварительная обработка может включать в себя преобразование цветового пространства в первичные (например, сигнала яркости) и вторичные (например, разностей сигнала цветности к красному цвету и к синему цвету) компоненты и обработку повторной дискретизации (например, чтобы уменьшать пространственное разрешение компонентов сигнала цветности) для кодирования. Перед кодированием, видео может преобразовываться в такое цветовое пространство, как YUV, в котором значения выборок компонента сигнала (Y) яркости представляют значения яркости или интенсивности, а значения выборок компонентов сигнала цветности (U, V) представляют значения цветовой разности. Точные определения значений цветовой разности (и операции преобразования в/из цветового YUV-пространства относительно другого цветового пространства, такого как RGB) зависят от реализации. В общем, при использовании в данном документе термин "YUV" указывает любое цветовое пространство с компонентом сигнала яркости (или яркости) и одним или более компонентов сигнала цветности (или цветности), включающих в себя Y'UV, YIQ, Y'IQ и YDbDr, а также варьирования, к примеру, YCbCr и YCoCg. Значения выборок сигнала цветности могут субдискретизироваться до более низкой частоты дискретизации сигнала цветности (например, для формата YUV 4:2:0), или значения выборок сигнала цветности могут иметь разрешение, идентичное разрешению значений выборок сигнала яркости (например, для формата YUV 4:4:4). Или видео может быть закодировано в другом формате (например, в формате RGB 4:4:4, в формате GBR 4:4:4 или в формате BGR 4:4:4). В частности, видео экранного контента может быть закодировано в формате RGB 4:4:4, в формате GBR 4:4:4 или в формате BGR 4:4:4.

[049] Кодер (340) кодирует выбранное изображение (331), чтобы формировать закодированное изображение (341), а также формирует сигналы (342) операции управления запоминающим устройством (MMCO) или информацию (RPS) наборов опорных изображений. RPS представляет собой набор изображений, который может использоваться для ссылки в компенсации движения для текущего изображения или любого последующего изображения. Если текущее изображение не является первым изображением, которое закодировано, при выполнении своего процесса кодирования, кодер (340) может использовать одно или несколько ранее закодированных/ декодированных изображений (369), которые сохранены во временной области (360) хранения запоминающего устройства декодированных изображений. Такие сохраненные

декодированные изображения (369) используются в качестве опорных изображений для межкадрового прогнозирования контента текущего исходного изображения (331). ММСО/RPS-информация (342) указывает декодеру то, какие восстановленные изображения могут использоваться в качестве опорных изображений и, следовательно, должны сохраняться в области хранения изображений.

[050] Обычно кодер (340) включает в себя несколько модулей кодирования, которые выполняют задачи кодирования, такие как сегментация на мозаичные фрагменты, оценка и прогнозирование на основе внутрикадрового прогнозирования, оценка и компенсация движения, преобразования частоты, квантование и энтропийное кодирование. Точные операции, выполняемые посредством кодера (340), могут варьироваться в зависимости от формата сжатия. Формат выходных закодированных данных может представлять собой варьирование или расширение H.265/HEVC-формата, формат Windows Media Video, VC-1-формат, MPEG-х-формат (например, MPEG-1, MPEG-2 или MPEG-4), H.26х-формат (например, H.261, H.262, H.263, H.264) или другой формат.

[051] Кодер (340) может сегментировать изображение на несколько мозаичных фрагментов идентичного размера или различных размеров. Например, кодер (340) разбивает изображение вдоль строк мозаичных фрагментов и столбцов мозаичных фрагментов, которые, с помощью границ изображения, задают горизонтальные и вертикальные границы мозаичных фрагментов в изображении, причем каждый мозаичный фрагмент представляет собой прямоугольную область. Мозаичные фрагменты зачастую используются для того, чтобы предоставлять возможности для параллельной обработки. Изображение также может быть организовано в качестве одного или нескольких слайсов, причем слайс может представлять собой все изображение или секцию изображения. Слайс может декодироваться независимо от других слайсов в изображении, что повышает устойчивость к ошибкам. Контент слайса или мозаичного фрагмента дополнительно сегментируется на блоки или другие наборы значений выборок для целей кодирования и декодирования. Строки некоторых блоков (например, строки единиц дерева кодирования слайса согласно H.265/HEVC-стандарту) могут быть закодированы параллельно с использованием WPP, как подробнее пояснено ниже.

[052] Для синтаксиса согласно H.265/HEVC-стандарту, кодер разбивает контент изображения (или слайса, или мозаичного фрагмента) на единицы дерева кодирования. Единица дерева кодирования (CTU) включает в себя значения выборок сигнала яркости, организованные в качестве блока дерева кодирования (CTB) сигналов яркости, и соответствующие значения выборок сигнала цветности, организованные в качестве двух CTB сигналов цветности. Размер CTU (и ее CTB) выбирается посредством кодера. CTB сигналов яркости может содержать, например, значения выборок сигнала яркости 64×64, 32×32 или 16×16. CTU включает в себя одну или более единиц кодирования. Единица кодирования (CU) имеет блок кодирования (CB) сигналов яркости и два соответствующих CB сигналов цветности. Например, CTU с CTB сигналов яркости 64×64 и двумя CTB сигналов цветности 64×64 (формат YUV 4:4:4) может разбиваться на четыре CU, причем каждая CU включает в себя CB сигналов яркости 32×32 и два CB сигналов цветности 32×32, и причем каждая CU возможно разбивается дополнительно на меньшие CU. Альтернативно, в качестве другого примера, CTU с CTB сигналов яркости 64×64 и двумя CTB сигналов цветности 32×32 (формат YUV 4:2:0) может разбиваться на четыре CU, причем каждая CU включает в себя CB сигналов яркости 32×32 и два CB сигналов цветности 16×16, и причем каждая CU возможно разбивается

дополнительно на меньшие CU. Наименьший допустимый размер CU (например, 8×8, 16×16) может передаваться в служебных сигналах в потоке битов.

[053] Обычно, CU имеет режим прогнозирования, к примеру, взаимный или внутренний. CU включает в себя одну или более единиц прогнозирования для целей передачи в служебных сигналах информации прогнозирования (такой как подробности режима прогнозирования, значения сдвига и т.д.) и/или обработки прогнозирования. Единица прогнозирования (PU) имеет прогнозный блок (PB) сигналов яркости и два PB сигналов цветности. Согласно H.265/HEVC-стандарту, для внутренне прогнозированной CU, PU имеет размер, идентичный размеру CU, если CU не имеет наименьший размер (например, 8×8). В этом случае, CU может разбиваться на четыре меньших PU (например, каждые 4×4, если наименьший CU-размер составляет 8×8 для внутрикадрового прогнозирования), или PU может иметь наименьший CU-размер, как указано посредством синтаксического элемента для CU. Тем не менее, для симметричных или асимметричных сегментов, используемых при прогнозировании с внутренним BC, большая CU может разбиваться на несколько PU.

[054] CU также имеет одну или более единиц преобразования для целей остаточного кодирования/декодирования, при этом единица преобразования (TU) имеет блок преобразования (TB) сигналов яркости и два TB сигналов цветности. PU во внутренне прогнозированной CU может содержать одну TU (равную по размеру PU) или несколько TU. Кодер определяет то, как сегментировать видео на STU, CU, PU, TU и т.д.

[055] В H.265/HEVC-реализациях, слайс может включать в себя один сегмент слайса (независимый сегмент слайса) или разделяться на несколько сегментов слайса (независимый сегмент слайса и один или более зависимых сегментов слайса). Сегмент слайса является целым числом STU, упорядоченных последовательно в сканировании мозаичных фрагментов, содержащихся в одной единице уровня абстрагирования от сети (NAL). Для независимого сегмента слайса, заголовок сегмента слайса включает в себя значения синтаксических элементов, которые применяются для независимого сегмента слайса. Для зависимого сегмента слайса, усеченный заголовок сегмента слайса включает в себя несколько значений синтаксических элементов, которые применяются для этого зависимого сегмента слайса, и значения других синтаксических элементов для зависимого сегмента слайса логически выводятся из значений для предыдущего независимого сегмента слайса в порядке декодирования.

[056] При использовании в данном документе, термин "блок" может указывать макроблок, единицу остаточных данных, CB, PB или TB либо некоторый другой набор значений выборки, в зависимости от контекста. Термин "единица" может указывать макроблок, STU, CU, PU, TU или некоторый другой набор блоков, либо он может указывать один блок, в зависимости от контекста.

[057] Возвращаясь к фиг. 3, кодер представляет внутренне кодированный блок, линию или строку исходного изображения (331) с точки зрения прогнозирования из других ранее восстановленных значений выборки в изображении (331). Для прогнозирования с внутренним копированием, модуль внутрикадровой оценки оценивает сдвиг от текущего блока, линии или строки до позиции в других ранее восстановленных значениях выборки. Опорный блок, линия или строка значений выборки в изображении используются для того, чтобы формировать прогнозные значения для текущего блока, линии или строки. Например, для прогнозирования с внутренним блочным копированием (BC), модуль внутрикадровой оценки оценивает сдвиг от текущего блока до позиции в ранее восстановленных значениях выборки в изображении. Опорный блок представляет собой блок значений выборки в изображении, которые предоставляют

значения ВС-прогнозирования для текущего блока. Опорный блок может указываться с помощью значения блочного вектора (BV) (определенного в BV-оценке). В качестве другого примера, для прогнозирования с внутренним линейным копированием (LC), модуль внутрикадровой оценки оценивает сдвиг от текущей линии (текущего блока) до позиции в ранее восстановленных значениях выборок в изображении. Опорная линия представляет собой линию значений выборок в изображении, которые предоставляют значения LC-прогнозирования для текущей линии. Опорная линия может указываться с помощью значения смещения, которое указывает сдвиг от текущей линии до опорной линии. В качестве другого примера, для прогнозирования с внутренним строковым копированием (SC), модуль внутрикадровой оценки оценивает сдвиг от текущей строки (текущего блока) до позиции в ранее восстановленных значениях выборок в изображении. Опорная строка представляет собой последовательность значений выборок в изображении, которые используются для того, чтобы формировать значения SC-прогнозирования для текущей строки. Опорная строка может указываться с помощью значения смещения (указывающего сдвиг от текущей строки до опорной строки) и значения длины строки. В зависимости от реализации, кодер может выполнять оценку смещения для блока, линии или строки с использованием входных значений выборок или восстановленных значений выборок (ранее закодированных значений выборок в том же изображении). Когда WPP активируется, модуль внутрикадровой оценки может определять сдвиги (например, для BV-значений при прогнозировании с внутренним ВС либо для значений смещения при прогнозировании с внутренним SC или прогнозировании с внутренним LC) в соответствии с ограничениями на местоположения опорных областей, как пояснено ниже.

[058] Для внутреннего пространственного прогнозирования для блока, модуль внутрикадровой оценки оценивает экстраполяцию соседних восстановленных значений выборок в блок.

[059] Модуль внутрикадровой оценки может выводить информацию прогнозирования (к примеру, BV-значения для прогнозирования с внутренним ВС, значения смещения для прогнозирования с внутренним LC или прогнозирования с внутренним SC либо режим (направление) прогнозирования для внутреннего пространственного прогнозирования), которая энтропийно кодируется. Предиктор внутрикадрового прогнозирования применяет информацию прогнозирования, чтобы определять внутренние прогнозные значения.

[060] Для режима палитрового кодирования, кодер (340) представляет, по меньшей мере, некоторые значения выборок CU или другой единицы с использованием палитры. Палитра представляет цвета, используемые в единице. Например, палитра преобразует значения 0, 1, 2, ..., p индекса в соответствующие цвета. В ходе кодирования единицы, надлежащие значения индекса заменяют значения выборок в позициях в единице. Редкое значение в единице может быть закодировано с использованием значения управляющего кода и литеральных значений вместо использования значения индекса в палитре. Палитра может изменяться между единицами, и информация, указывающая палитры, может передаваться в служебных сигналах в потоке битов.

[061] Кодер (340) представляет межкадрово закодированный прогнозированный блок исходного изображения (331) с точки зрения прогнозирования из опорных изображений. Модуль оценки движения оценивает движение блока относительно одного или более опорных изображений (369). Когда используются несколько опорных изображений, несколько опорных изображений могут исходить из различных временных направлений или одного и того же временного направления. Опорная область прогнозирования с

компенсацией движения представляет собой область значений выборок в опорном изображении(ях), которые используются для того, чтобы формировать значения прогнозирования с компенсацией движения для блока значений выборок текущего изображения. Модуль оценки движения выводит информацию движения, такую
5 информацию вектора движения (MV), которая энтропийно кодирована. Модуль компенсации движения применяет MV к опорным изображениям (369), чтобы определять значения прогнозирования с компенсацией движения для межкадрового прогнозирования.

[062] Кодер может определять разности (если имеются) между прогнозными
10 значениями (внутренними или взаимными) блока и соответствующими исходными значениями. Эти значения остатка прогнозирования дополнительно закодированы с использованием преобразования частоты (если преобразование частоты не пропускается), квантования и энтропийного кодирования. Например, кодер (340) задает значения для параметра квантования (QP) для изображения, мозаичного фрагмента,
15 слайса и/или другой части видео и квантует коэффициенты преобразования соответствующим образом. Энтропийный кодер кодера (340) сжимает значения квантованных коэффициентов преобразования, а также некоторую вспомогательную информацию (например, MV-информацию, BV-информацию, QP-значения, решения по выбору режима, варианты выбора параметров). Типичные технологии энтропийного
20 кодирования включают в себя кодирование экспоненциальным кодом Голомба, кодирование Голомба-Райса, арифметическое кодирование, дифференциальное кодирование, кодирование Хаффмана, кодирование по длинам серий, кодирование переменного-переменной длины (V2V), кодирование переменного-фиксированной длины (V2F), кодирование Лемпеля-Зива (LZ), словарное кодирование, энтропийное
25 кодирование на основе сегментации на интервалы вероятности (PIPE) и комбинации вышеозначенного. Энтропийный кодер может использовать различные технологии кодирования для различных видов информации, может применять несколько технологий в комбинации (например, посредством применения кодирования Голомба-Райса с последующим применением арифметического кодирования) и может выбирать из числа
30 нескольких кодовых таблиц в пределах конкретной технологии кодирования. В некоторых реализациях, преобразование частоты может пропускаться. В этом случае, значения остатка прогнозирования могут квантоваться и энтропийно кодироваться. Когда используется режим палитрового кодирования, энтропийный кодер может кодировать палитровые данные. Кодер (340) может использовать палитровое
35 прогнозирование, как пояснено ниже.

[063] Адаптивный фильтр удаления блочности включен в контур компенсации движения (т.е. "внутриконтурную" фильтрацию) в кодере (340), чтобы сглаживать неоднородности по строкам и/или столбцам границ блоков в декодированном изображении. Другая фильтрация (к примеру, фильтрация с подавлением периодически
40 повторяющихся помех, адаптивная контурная фильтрация (ALF) или фильтрация на основе дискретизированного адаптивного смещения (SAO); не показаны) альтернативно или дополнительно может применяться в качестве операций внутриконтурной фильтрации.

[064] Закодированные данные, сформированные посредством кодера (340), включают
45 в себя синтаксические элементы для различных слоев синтаксиса потока битов. Для синтаксиса согласно H.265/HEVC-стандарту, например, набор параметров изображения (PPS) представляет собой синтаксическую структуру, которая содержит синтаксические элементы, которые могут быть ассоциированы с изображением. PPS может

использоваться для отдельного изображения, или PPS может быть повторно использован для нескольких изображений в последовательности. PPS типично передается в служебных сигналах отдельно от закодированных данных для изображения (например, одной единицы уровня абстрагирования от сети (NAL) для PPS и одной или более других NAL-единиц для закодированных данных для изображения). В закодированных данных для изображения, синтаксический элемент указывает то, какой PPS следует использовать для изображения. Аналогично, для синтаксиса согласно H.265/HEVC-стандарту, набор параметров последовательности (SPS) представляет собой синтаксическую структуру, которая содержит синтаксические элементы, которые могут быть ассоциированы с последовательностью изображений. Поток битов может включать в себя один SPS или несколько SPS. SPS типично передается в служебных сигналах отдельно от других данных для последовательности, и синтаксический элемент в других данных указывает то, какой SPS следует использовать.

[065] Кодированные изображения (341) и MMCO/RPS-информация (342) (или информация, эквивалентная MMCO/RPS-информации (342), поскольку зависимости и структуры упорядочения для изображений уже известны в кодере (340)) обрабатываются посредством эмулятора (350) процесса декодирования. Эмулятор (350) процесса декодирования реализует часть функциональности декодера, например, задачи декодирования, чтобы восстанавливать опорные изображения. Способом, совместимым с MMCO/RPS-информацией (342), эмулятор процессов декодирования (350) определяет то, должно или нет данное кодированное изображение (341) восстанавливаться и сохраняться для использования в качестве опорного изображения при межкадровом прогнозировании последующих изображений, которые должны быть закодированы. Если кодированное изображение (341) должно сохраняться, эмулятор (350) процесса декодирования моделирует процесс декодирования, который должен осуществляться посредством декодера, который принимает кодированное изображение (341) и формирует соответствующее декодированное изображение (351). При этом, когда кодер (340) использует декодированное изображение(я) (369), которые сохранены в области (360) хранения декодированных изображений, эмулятор (350) процесса декодирования также использует декодированное изображение(я) (369) из области (360) хранения в качестве части процесса декодирования.

[066] Временная область (360) хранения запоминающего устройства декодированных изображений включает в себя несколько областей (361, 362, ..., 36n) хранения буфера изображений. Способом, совместимым с MMCO/RPS-информацией (342), эмулятор (350) процесса декодирования управляет контентом области (360) хранения, чтобы идентифицировать любые буферы (361, 362 и т.д.) изображений с изображениями, которые более не требуются посредством кодера (340) для использования в качестве опорных изображений. После моделирования процесса декодирования, эмулятор (350) процесса декодирования сохраняет заново декодированное изображение (351) в буфере (361, 362 и т.д.) изображений, который идентифицирован таким образом.

[067] Кодированные изображения (341) и MMCO/RPS-информация (342) буферизованы во временной области (370) кодированных данных. Кодированные данные, которые агрегированы в области (370) кодированных данных, содержат, в качестве части синтаксиса элементарного кодированного потока видеобитов, закодированные данные для одного или более изображений. Кодированные данные, которые агрегированы в области (370) кодированных данных, также могут включать в себя мультимедийные метаданные, связанные с кодированными видеоданными (например, в качестве одного или более параметров в одном или более сообщениях с дополнительной улучшающей

информацией (SEI) или сообщений с информацией применимости видео (VUI)).

[068] Агрегированные данные (371) из временной области (370) кодированных данных обрабатываются посредством канального кодера (380). Канальный кодер (380) может пакетизировать и/или мультиплексировать агрегированные данные для передачи или хранения в качестве мультимедийного потока (например, согласно формату потока мультимедийных программ или транспортного потока, такому как ITU-T H.222.0 | ISO/IEC 13818-1, или формату транспортного Интернет-протокола реального времени, такому как IETF RFC 3550), причем в этом случае канальный кодер (380) может добавлять синтаксические элементы в качестве части синтаксиса потока передачи мультимедиа. Альтернативно, канальный кодер (380) может организовывать агрегированные данные для хранения в качестве файла (например, согласно формату мультимедийного контейнера, такому как ISO/IEC 14496-12), причем в этом случае канальный кодер (380) может добавлять синтаксические элементы в качестве части синтаксиса файла хранения мультимедиа. Альтернативно, если обобщать, канальный кодер (380) может реализовывать один или более протоколов мультиплексирования или транспортных протоколов мультимедийной системы, причем в этом случае канальный кодер (380) может добавлять синтаксические элементы в качестве части синтаксиса протокола(ов). Канальный кодер (380) предоставляет вывод в канал (390), который представляет устройство хранения, соединение связи или другой канал для вывода. Канальный кодер (380) или канал (390) также может включать в себя другие элементы (не показаны), например, для кодирования с прямой коррекцией ошибок (FEC) и модуляции аналоговых сигналов.

IV. Примерная система декодера

[069] Фиг. 4 является блок-схемой примерной системы (400) декодера, в сочетании с которой могут реализовываться некоторые описанные варианты осуществления. Система (400) декодера может представлять собой инструментальное средство декодирования общего назначения, допускающее работу в любом из нескольких режимов декодирования, таких как режим декодирования с низкой задержкой для обмена данными в реальном времени и режим декодирования с более высокой задержкой для воспроизведения мультимедиа из файла или потока, либо она может представлять собой инструментальное средство декодирования специального назначения, адаптированное для одного такого режима декодирования. Система (400) декодера может реализовываться в качестве части модуля операционной системы, в качестве части библиотеки приложений, в качестве части автономного приложения или с использованием аппаратных средств специального назначения. В целом, система (400) декодера принимает кодированные данные из канала (410) и формирует восстановленные изображения в качестве вывода для назначения (490) вывода. Принимаемые закодированные данные могут включать в себя контент, закодированный с использованием правил, принудительно активированных для режимов внутрикадрового прогнозирования, когда WPP активируется.

[070] Система (400) декодера включает в себя канал (410), который может представлять устройство хранения данных, соединение связи или другой канал для кодированных данных в качестве ввода. Канал (410) формирует кодированные данные, которые канально кодированы. Канальный декодер (420) может обрабатывать кодированные данные. Например, канальный декодер (420) депакетизирует и/или демультимплексирует данные, которые агрегированы для передачи или хранения в качестве мультимедийного потока (например, согласно формату потока мультимедийных программ или транспортного потока, такому как ITU-T H.222.0 | ISO/

IEC 13818-1, или формату транспортного Интернет-протокола реального времени, такому как IETF RFC 3550), причем в этом случае каналный декодер (420) может синтаксически анализировать синтаксические элементы, добавленные в качестве части синтаксиса потока передачи мультимедиа. Альтернативно, каналный декодер (420) разделяет кодированные видеоданные, которые агрегированы для хранения в качестве файла (например, согласно формату мультимедийного контейнера, такому как ISO/IEC 14496-12), причем в этом случае каналный декодер (420) может синтаксически анализировать синтаксические элементы, добавленные в качестве части синтаксиса файла хранения мультимедиа. Альтернативно, если обобщать, каналный декодер (420) может реализовывать один или более протоколов демультимплексирования или транспортных протоколов мультимедийной системы, причем в этом случае каналный декодер (420) может синтаксически анализировать синтаксические элементы, добавленные в качестве части синтаксиса протокола(ов). Канал (410) или каналный декодер (420) также может включать в себя другие элементы (не показаны), например, для FEC-декодирования и демодуляции аналоговых сигналов.

[071] Кодированные данные (421), которые выводятся из каналного декодера (420), сохраняются во временной области (430) кодированных данных до тех пор, пока достаточное количество таких данных не будет принято. Кодированные данные (421) включают в себя кодированные изображения (431) и MMCO/RPS-информацию (432). Кодированные данные (421) в области (430) кодированных данных содержат, в качестве части синтаксиса элементарного кодированного потока видеобитов, кодированные данные для одного или более изображений. Кодированные данные (421) в области (430) кодированных данных также могут включать в себя мультимедийные метаданные, связанные с закодированными видеоданными (например, в качестве одного или более параметров в одном или более SEI-сообщений или VUI-сообщений).

[072] В общем, область (430) кодированных данных временно сохраняет кодированные данные (421) до тех пор, пока такие кодированные данные (421) не будут использоваться посредством декодера (450). В этот момент, кодированные данные для кодированного изображения (431) и MMCO/RPS-информации (432) передаются из области (430) кодированных данных в декодер (450). По мере того, как продолжается декодирование, новые кодированные данные добавляются в область (430) кодированных данных, и самые старые кодированные данные, остающиеся в области (430) кодированных данных, передаются в декодер (450).

[073] Декодер (450) декодирует кодированное изображение (431), чтобы формировать соответствующее декодированное изображение (451). Изображение может быть сегментировано на несколько мозаичных фрагментов идентичного размера или различных размеров. Изображение также может быть организовано в качестве одного или более слайсов. Контент слайса или мозаичного фрагмента может дополнительно сегментироваться на блоки или другие наборы значений выборок. Если изображение было закодировано при активированной WPP (с использованием WPP или иным способом, совместимым с использованием WPP в ходе декодирования), строки некоторых блоков (например, строки STU согласно H.265/HEVC-стандарту) могут декодироваться параллельно с использованием WPP, как подробнее пояснено ниже.

[074] Надлежащим образом, при выполнении своего процесса декодирования, декодер (450) может использовать одно или более ранее декодированных изображений (469) в качестве опорных изображений для межкадрового прогнозирования. Декодер (450) считывает такие ранее декодированные изображения (469) из временной области (460) хранения запоминающего устройства декодированных изображений. Обычно, декодер

(450) включает в себя несколько модулей декодирования, которые выполняют задачи декодирования, такие как энтропийное декодирование, внутрикадровое прогнозирование, межкадровое прогнозирование с компенсацией движения, обратное квантование, обратные преобразования частоты (если не пропущены) и объединение мозаичных фрагментов. Точные операции, выполняемые посредством декодера (450), могут варьироваться в зависимости от формата сжатия.

[075] Например, декодер (450) принимает закодированные данные для сжатого изображения или последовательности изображений и формирует вывод, включающий в себя декодированное изображение (451). В декодере (450), буфер принимает закодированные данные для сжатого изображения и в подходящее время делает принимаемые закодированные данные доступными для энтропийного декодера. Энтропийный декодер энтропийно декодирует энтропийно кодированные квантованные данные, а также энтропийно кодированную вспомогательную информацию, типично применяя инверсию энтропийного кодирования, выполняемого в кодере. Когда используется режим палитрового декодирования, энтропийный декодер может декодировать палитровые данные. Декодер (450) может использовать палитровое прогнозирование, как пояснено ниже.

[076] Модуль компенсации движения применяет информацию движения к одному или более опорных изображений, чтобы формировать значения прогнозирования с компенсацией движения для любых взаимно кодированных блоков восстанавливаемого изображения. Модуль внутрикадрового прогнозирования может пространственно прогнозировать значения выборок текущего блока из соседних ранее восстановленных значений выборок. Альтернативно, для прогнозирования с внутренним BC, прогнозирования с внутренним LC или прогнозирования с внутренним SC, модуль внутрикадрового прогнозирования может прогнозировать значения выборок текущего блока, линии или строки с использованием ранее восстановленных значений выборок опорного блока, линии или строки в изображении, который указывается с помощью значения сдвига. В частности, опорный блок/линия/строка может указываться с помощью BV-значения (для прогнозирования с внутренним BC), значения смещения (для прогнозирования с внутренним LC) или значения смещения и значения длины строки (для прогнозирования с внутренним SC). Когда WPP активируется, сдвиги (например, для BV-значений при прогнозировании с внутренним BC либо для значений смещения при прогнозировании с внутренним SC или прогнозировании с внутренним LC) являются согласованными с ограничениями на местоположения опорных областей, как пояснено ниже.

[077] Декодер (450) также восстанавливает значения остатка прогнозирования. Обратный квантователь обратно квантует энтропийно декодированные данные. Например, декодер (450) задает значения для QP для изображения, мозаичного фрагмента, слайса и/или другой части видео на основе синтаксических элементов в потоке битов и обратно квантует коэффициенты преобразования соответствующим образом. Обратный преобразователь частоты преобразует квантованные данные частотной области в данные пространственной области. В некоторых реализациях, преобразование частоты может пропускаться, причем в этом случае обратное преобразование частоты также пропускается. Если да, значения остатка прогнозирования могут энтропийно декодироваться и обратно квантоваться. Для межкадрового прогнозированного блока, декодер (450) комбинирует восстановленные значения остатка прогнозирования со значениями прогнозирования с компенсацией движения. Декодер (450) аналогично может комбинировать значения остатка

прогнозирования с прогнозными значениями из внутрикадрового прогнозирования.

[078] Для режима палитрового декодирования, декодер (450) использует палитру, которая представляет, по меньшей мере, некоторые значения выборок CU или другой единицы. Палитра преобразует значения индекса в соответствующие цвета. В ходе декодирования, для позиций в единице, значения индекса из палитры заменяются надлежащими значениями выборок. Управляюще кодированное значение в единице может декодироваться с использованием значения управляющего кода и литеральных значений. Палитра может изменяться между единицами, и информация, указывающая палитры, может передаваться в служебных сигналах в потоке битов.

[079] Адаптивный фильтр удаления блочности включен в контур компенсации движения в видеодекодере (450), чтобы сглаживать неоднородности по строкам и/или столбцам границ блоков в декодированном изображении (451). Другая фильтрация (к примеру, фильтрация с подавлением периодически повторяющихся помех, ALF- или SAO-фильтрация; не показаны) альтернативно или дополнительно может применяться в качестве операций внутриконтурной фильтрации.

[080] Временная область (460) хранения запоминающего устройства декодированных изображений включает в себя несколько областей (461, 462, ..., 46n) хранения буфера изображений. Область (460) хранения декодированных изображений является примером буфера декодированных изображений. Декодер (450) использует MMCO/RPS-информацию (432), чтобы идентифицировать буфер (461, 462 и т.д.) изображений, в котором он может сохранять декодированное изображение (451). Декодер (450) сохраняет декодированное изображение (451) в этом буфере изображений.

[081] Модуль (480) упорядочения вывода идентифицирует, когда следующее изображение, которое должно формироваться в порядке вывода, доступно в области (460) хранения декодированных изображений. Когда следующее изображение (481), которое должно формироваться в порядке вывода, доступно в области (460) хранения декодированных изображений, оно считывается посредством модуля (480) упорядочения вывода и выводится в назначение (490) вывода (например, на дисплей). В общем, порядок, в котором изображения выводятся из области (460) хранения декодированных изображений посредством модуля (480) упорядочения вывода, может отличаться от порядка, в котором изображения декодируются посредством декодера (450).

V. Примерные видеокодеры

[082] Фиг. 5а и 5б являются блок-схемой обобщенного видеокодера (500), в сочетании с которым могут реализовываться некоторые описанные варианты осуществления.

Кодер (500) принимает последовательность видеоизображений, включающую в себя текущее изображение в качестве входного видеосигнала (505), и формирует закодированные данные в кодированном потоке (595) видеобитов в качестве вывода.

[083] Кодер (500) является блочным и использует блочный формат, который зависит от реализации. Блоки дополнительно могут подразделяться на различных стадиях, например, на стадиях прогнозирования, преобразования частоты и/или энтропийного кодирования. Например, изображение может разделяться на блоки 64×64, блоки 32×32 или блоки 16×16, которые, в свою очередь, могут разделяться на меньшие блоки значений выборок для кодирования и декодирования. В реализациях кодирования для H.265/HEVC-стандарта, кодер сегментирует изображение на STU (СТВ), CU (СВ), PU (РВ) и TU (ТВ).

[084] Кодер (500) сжимает изображения с использованием внутрикадрового кодирования и/или межкадрового кодирования. Многие компоненты кодера (500) используются как для внутрикадрового кодирования, так и для межкадрового

кодирования. Точные операции, выполняемые посредством этих компонентов, могут варьироваться в зависимости от типа сжимаемой информации.

[085] Модуль (510) мозаичного размещения необязательно сегментирует изображение на несколько мозаичных фрагментов идентичного размера или различных размеров. Например, модуль (510) мозаичного размещения разбивает изображение вдоль строк мозаичных фрагментов и столбцов мозаичных фрагментов, которые, с помощью границ изображения, задают горизонтальные и вертикальные границы мозаичных фрагментов в изображении, причем каждый мозаичный фрагмент представляет собой прямоугольную область. В H.265/HEVC-реализациях, кодер (500) сегментирует изображение на один или более слайсов, при этом каждый слайс включает в себя один или более сегментов слайса. Строки некоторых блоков (например, строки STU слайса согласно H.265/HEVC-стандарту) могут быть закодированы параллельно с использованием WPP, как подробнее пояснено ниже.

[086] Общий контроллер (520) кодирования принимает изображения для входного видеосигнала (505), а также обратную связь (не показана) из различных модулей кодера (500). В целом, общий контроллер (520) кодирования предоставляет управляющие сигналы (не показаны) в другие модули (к примеру, в модуль (510) мозаичного размещения, преобразователь (530)/модуль масштабирования/квантователь, модуль (535) масштабирования/обратный преобразователь, модуль (540) внутрикадровой оценки, модуль (550) оценки движения и переключатель внутреннего/взаимного режима), чтобы задавать и изменять параметры кодирования в ходе кодирования. В частности, общий контроллер (520) кодирования может определять то, следует или нет и как следует использовать палитровое прогнозирование, прогнозирование с внутренним ВС, прогнозирование с внутренним LC и прогнозирование с внутренним SC в ходе кодирования. Общий контроллер (520) кодирования также может оценивать промежуточные результаты в ходе кодирования, например, при выполнении анализа искажения в зависимости от скорости передачи. Общий контроллер (520) кодирования формирует общие управляющие данные (522), которые указывают решения, принимаемые в ходе кодирования, так что соответствующий декодер может принимать согласованные решения. Общие управляющие данные (522) предоставляются в модуль (590) форматирования заголовков/энтропийный кодер.

[087] Если текущее изображение прогнозируется с использованием межкадрового прогнозирования, модуль (550) оценки движения оценивает движение блоков значений выборки текущего изображения входного видеосигнала (505) относительно одного или более опорных изображений. Буфер (570) декодированных изображений буферизует одно или более восстановленных ранее кодированных изображений для использования в качестве опорных изображений. Когда используются несколько опорных изображений, несколько опорных изображений могут исходить из различных временных направлений или одного и того же временного направления. Модуль (550) оценки движения формирует в качестве вспомогательной информации данные (552) движения, такие как MV-данные, значения индекса режима объединения и данные выбора опорных изображений. Данные (552) движения предоставляются в модуль (590) форматирования заголовков/энтропийный кодер, а также в модуль (555) компенсации движения.

[088] Модуль (555) компенсации движения применяет MV к восстановленному опорному изображению(ям) из буфера (570) декодированных изображений. Модуль (555) компенсации движения формирует прогнозирования с компенсацией движения для текущего изображения.

[089] В отдельном тракте в кодере (500), модуль (540) внутрикадровой оценки

определяет то, как выполнять внутрикадровое прогнозирование для блоков значений выборок текущего изображения входного видеосигнала (505). Текущее изображение может быть полностью или частично кодировано с использованием внутрикадрового кодирования. С использованием значений восстановления (538) текущего изображения, для внутреннего пространственного прогнозирования, модуль (540) внутрикадровой оценки определяет то, как пространственно прогнозировать значения выборок текущего блока текущего изображения из соседних ранее восстановленных значений выборок текущего изображения.

[090] Альтернативно, для прогнозирования с внутренним копированием, модуль (540) внутрикадровой оценки оценивает сдвиг от текущего блока, линии или строки до позиции в других ранее восстановленных значениях выборок. Опорный блок, линия или строка значений выборок в изображении используются для того, чтобы формировать прогнозные значения для текущего блока, линии или строки. Например, для прогнозирования с внутренним ВС, модуль (540) внутрикадровой оценки оценивает сдвиг от текущего блока до опорного блока, который может указываться с помощью BV-значения. В качестве другого примера, для прогнозирования с внутренним LC, модуль (540) внутрикадровой оценки оценивает сдвиг от текущей линии (текущего блока) до опорной линии, который может указываться с помощью значения смещения (указывающего сдвиг от текущей линии до опорной линии). В качестве другого примера, для прогнозирования с внутренним SC, модуль внутрикадровой оценки оценивает сдвиг от текущей строки (текущего блока) до опорной строки, который может указываться с помощью значения смещения (указывающего сдвиг от текущей строки до опорной строки) и значения длины строки. Когда WPP активируется, модуль (540) внутрикадровой оценки может определять сдвиги (например, для BV-значений при прогнозировании с внутренним ВС либо для значений смещения при прогнозировании с внутренним SC или прогнозировании с внутренним LC) в соответствии с ограничениями на местоположения опорных областей, как пояснено ниже.

[091] В зависимости от реализации, модуль (540) внутрикадровой оценки может выполнять оценку смещения для текущего блока, линии или строки с использованием входных значений выборок, восстановленных значений выборок до внутриконтурной фильтрации или восстановленных значений выборок после внутриконтурной фильтрации. В общем, посредством использования входных значений выборок или нефильтрованных восстановленных значений выборок для оценки смещения, модуль (540) внутрикадровой оценки может не допускать узкого места при последовательной обработке (которое может возникать в результате фильтрации восстановленных значений выборок опорного блока, линии, строки и т.д. до оценки смещения/прогнозирования с внутренним копированием). С другой стороны, сохранение нефильтрованных восстановленных значений выборок использует дополнительное запоминающее устройство. Кроме того, если внутриконтурная фильтрация применяется до оценки смещения, может быть предусмотрена область влияния, которая перекрывается между процессом фильтрации, который должен применяться после того, как текущий блок/линия/строка декодируется, и областью, используемой для оценки смещения/прогнозирования с внутренним копированием. В таком случае, оценка смещения/прогнозирование с внутренним копированием должна применяться перед этим аспектом операции фильтрации. В некоторых реализациях, кодер может применять некоторые операции внутриконтурной фильтрации до оценки смещения/прогнозирования с внутренним копированием и выполнять дополнительную или альтернативную фильтрацию на последующей стадии обработки.

[092] Модуль (540) внутрикадровой оценки формирует в качестве вспомогательной информации данные (542) внутреннего прогнозирования, такие как информация, указывающая то, внутреннее прогнозирование использует пространственное прогнозирование, прогнозирование с внутренним ВС, прогнозирование с внутренним LC или прогнозирование с внутренним SC, направление режима прогнозирования (для внутреннего пространственного прогнозирования), BV-значения (для прогнозирования с внутренним ВС), значения смещения (для прогнозирования с внутренним LC) или значения смещения и значения длины (для прогнозирования с внутренним SC). Данные (542) внутреннего прогнозирования предоставляются в модуль (590) форматирования заголовков/энтропийный кодер, а также в модуль (545) внутрикадрового прогнозирования.

[093] Согласно данным (542) внутреннего прогнозирования, модуль (545) внутрикадрового прогнозирования пространственно прогнозирует значения выборок текущего блока текущего изображения из соседних ранее восстановленных значений выборок текущего изображения. Альтернативно, для прогнозирования с внутренним копированием, модуль (545) внутрикадрового прогнозирования прогнозирует значения выборок текущего блока, линии, строки или другой секции с использованием ранее восстановленных значений выборок опорного блока, линии, строки или другой секции, которая указывается посредством сдвига (BV-значения, значения смещения и т.д.) для текущего блока, линии, строки и т.д. В некоторых случаях, BV-значение (или другое значение смещения) может быть прогнозированным значением. В других случаях, BV-значение (или другое значение смещения) может отличаться от своего прогнозированного значения, причем в этом случае дифференциал указывает разность между прогнозированным значением и BV-значением (или другим значением смещения). Для режима внутреннего SC, модуль (545) внутрикадрового прогнозирования также использует значение длины строки при прогнозировании значений выборок текущей строки.

[094] Для режима палитрового кодирования, кодер (500) представляет, по меньшей мере, некоторые значения выборок CU или другой единицы с использованием палитры. Палитра представляет цвета, используемые в единице. Например, палитра преобразует значения 0, 1, 2, ..., p индекса в соответствующие цвета, которые могут иметь формат RGB 4:4:4, формат BGR 4:4:4, формат GBR 4:4:4, формат YUV 4:4:4 или другой формат (цветовое пространство, частота дискретизации цветов). Значение индекса может представлять RGB-триплет, BGR-триплет или GBR-триплет для пиксела, при этом пиксел представляет собой набор совместно размещенных значений выборок. Для кодирования единицы, значения индекса заменяют значения выборок пикселов в единице. Редкое значение в единице может быть закодировано с использованием значения управляющего кода и литеральных значений вместо использования значения индекса в палитре. Палитра может изменяться между единицами, и палитровые данные, указывающие палитры, могут передаваться в служебных сигналах в потоке битов.

[095] Переключатель внутреннего/взаимного режима выбирает то, представляет собой прогнозирование (558) для данного блока прогнозирование с компенсацией движения или внутрикадровое прогнозирование.

[096] В некоторых примерных реализациях, остаток не вычисляется для единицы, закодированной в режиме палитрового кодирования или в режиме внутреннего копирования (прогнозирование с внутренним ВС, прогнозирование с внутренним LC или прогнозирование с внутренним SC). Вместо этого, остаточное кодирование пропускается, и прогнозные значения выборок используются в качестве

восстановленных значений выборок.

[097] Когда остаточное кодирование не пропускается, разность (если имеется) между блоком прогнозирования (558) и соответствующей частью исходного текущего изображения входного видеосигнала (505) предоставляет значения остатка (518). В ходе восстановления текущего изображения, когда остаточные значения закодированы/переданы в служебных сигналах, восстановленные остаточные значения комбинируются с прогнозированием (558), чтобы формировать приблизительное или точное восстановление (538) исходного контента из видеосигнала (505). (При сжатии с потерями, некоторая информация теряется из видеосигнала (505)).

[098] В качестве части остаточного кодирования, в преобразователе (530)/модуле масштабирования/квантователе, когда преобразование частоты не пропускается, преобразователь частоты преобразует видеоинформацию пространственной области в данные частотной области (т.е. спектрального преобразования). Для кодирования видео на основе блоков, преобразователь частоты применяет дискретное косинусное преобразование (DCT), его целочисленную аппроксимацию или другой тип прямого блочного преобразования (например, дискретное синусное преобразование или его целочисленную аппроксимацию) к блокам данных остатков прогнозирования (или данных значений выборок, если прогнозирование (558) является нулем), формируя блоки коэффициентов преобразования частоты. Преобразователь (530)/модуль масштабирования/квантователь может применять преобразование с переменными размерами блоков. В этом случае, преобразователь (530)/модуль масштабирования/квантователь может определять то, какие размеры блоков преобразований следует использовать для остаточных значений для текущего блока. Модуль масштабирования/квантователь масштабирует и квантует коэффициенты преобразования. Например, квантователь применяет скалярное квантование с мертвыми зонами к данным частотной области с размером шага квантования, который варьируется на основе каждого изображения, на основе каждого мозаичного фрагмента, на основе каждого слайса, на поблочной основе, на конкретной для частоты основе или на другой основе. Данные (532) квантованных коэффициентов преобразования предоставляются в модуль (590) форматирования заголовков/энтропийный кодер. Если преобразование частоты пропускается, модуль масштабирования/квантователь может масштабировать и квантовать блоки данных остатков прогнозирования (или данных значений выборок, если прогнозирование (558) является нулем), формируя квантованные значения, которые предоставляются в модуль (590) форматирования заголовков/энтропийный кодер.

[099] Чтобы восстанавливать остаточные значения, в модуле (535) масштабирования/обратном преобразователе, модуль масштабирования/обратный квантователь выполняет обратное масштабирование и обратное квантование для квантованных коэффициентов преобразования. Когда стадия преобразования не пропущена, обратный преобразователь частоты выполняет обратное преобразование частоты, формируя блоки восстановленных значений остатка прогнозирования или значений выборок. Если стадия преобразования пропущена, обратное преобразование частоты также пропускается. В этом случае, модуль масштабирования/обратный квантователь может выполнять обратное масштабирование и обратное квантование для блоков данных остатков прогнозирования (или данных значений выборок), формируя восстановленные значения. Когда остаточные значения закодированы/переданы в служебных сигналах, кодер (500) комбинирует восстановленные остаточные значения со значениями прогнозирования (558) (например, значениями прогнозирования с компенсацией движения, значениями внутрикадрового прогнозирования), чтобы формировать

восстановление (538). Когда остаточные значения не закодированы/переданы в служебных сигналах, кодер (500) использует значения прогнозирования (558) в качестве восстановления (538).

5 [0100] Для внутрикадрового прогнозирования, значения восстановления (538) могут быть возвращены в модуль (540) внутрикадровой оценки и модуль (545) внутрикадрового прогнозирования. Значения восстановления (538) могут использоваться для прогнозирования с компенсацией движения последующих изображений. Значения восстановления (538) дополнительно могут фильтроваться. Контроллер (560) фильтрации определяет то, как выполнять фильтрацию для удаления блочности и SAO-фильтрацию
10 для значений восстановления (538), для данного изображения видеосигнала (505). Контроллер (560) фильтрации формирует управляющие данные (562) фильтрации, которые предоставляются в модуль (590) форматирования заголовков/энтропийный кодер и модуль (565) объединения/фильтр.

15 [0101] В модуле (565) объединения/фильтре, кодер (500) объединяет контент из различных мозаичных фрагментов в восстановленную версию изображения. Кодер (500) избирательно выполняет фильтрацию для удаления блочности и SAO-фильтрацию согласно управляющим данным (562) фильтрации и правилам для адаптации фильтра, с тем чтобы адаптивно сглаживать неоднородности на границах в изображениях. Альтернативно или дополнительно, может применяться другая фильтрация (к примеру,
20 фильтрация с подавлением периодически повторяющихся помех или ALF; не показаны). Границы мозаичных фрагментов могут избирательно фильтроваться или вообще не фильтроваться, в зависимости от настроек кодера (500), и кодер (500) может предоставлять синтаксические элементы в кодированном потоке битов, чтобы указывать то, применяется или нет такая фильтрация. Буфер (570) декодированных изображений
25 буферизует восстановленное текущее изображение для использования в последующем прогнозировании с компенсацией движения.

[0102] Модуль (590) форматирования заголовков/энтропийный кодер форматирует и/или энтропийно кодирует общие управляющие данные (522), данные (532) квантованных коэффициентов преобразования, данные (542) внутреннего
30 прогнозирования, данные (552) движения и управляющие данные (562) фильтрации. Для данных (552) движения, модуль (590) форматирования заголовков/энтропийный кодер может выбирать и энтропийно кодировать значения индекса режима объединения, либо может использоваться MV-предиктор по умолчанию. В некоторых случаях, модуль (590) форматирования заголовков/энтропийный кодер также определяет MV-
35 дифференциалы для MV-значений (относительно MV-предикторов для MV-значений), затем энтропийно кодирует MV-дифференциалы, например, с использованием контекстно-адаптивного двоичного арифметического кодирования. Для данных (542) внутреннего прогнозирования, BV-значение (или другое значение смещения) может быть закодировано с использованием прогнозирования. Прогнозирование может
40 использовать предиктор по умолчанию (например, BV-значение или другое значение смещения из одного или более соседних блоков). Когда несколько предикторов являются возможными, индекс предиктора может указывать то, какой из нескольких предикторов следует использовать для прогнозирования BV-значения (или другого значения смещения). Модуль (590) форматирования заголовков/энтропийный кодер может
45 выбирать и энтропийно кодировать значения индекса предиктора (для прогнозирования с внутренним копированием), либо может использоваться предиктор по умолчанию. В некоторых случаях, модуль (590) форматирования заголовков/энтропийный кодер также определяет дифференциалы (относительно предикторов для BV-значений или

других значений смещения), затем энтропийно кодирует дифференциалы, например, с использованием контекстно-адаптивного двоичного арифметического кодирования. Для режима палитрового кодирования, модуль (590) форматирования заголовков/энтропийный кодер может кодировать палитровые данные. В частности, модуль (590) форматирования заголовков/энтропийный кодер может использовать палитровое прогнозирование, как пояснено ниже.

[0103] Модуль (590) форматирования заголовков/энтропийный кодер предоставляет закодированные данные в кодированном потоке (595) видеобитов. Формат кодированного потока (595) видеобитов может представлять собой варьирование или расширение H.265/HEVC-формата, формат Windows Media Video, VC-1-формат, MPEG-x-формат (например, MPEG-1, MPEG-2 или MPEG-4), H.26x-формат (например, H.261, H.262, H.263, H.264) или другой формат.

[0104] В зависимости от реализации и требуемого типа сжатия, модули кодера (500) могут добавляться, опускаться, разбиваться на несколько модулей, комбинироваться с другими модулями и/или заменяться аналогичными модулями. В альтернативных вариантах осуществления, кодеры с различными модулями и/или другими конфигурациями модулей выполняют одну или более описанных технологий. Конкретные варианты осуществления кодеров типично используют вариацию или дополненную версию кодера (500). Взаимосвязи, показанные между модулями в кодере (500), указывают общие потоки информации в кодере; другие взаимосвязи не показаны для простоты.

VI. Примерные видеодекодеры

[0105] Фиг. 6 является блок-схемой обобщенного декодера (600), в сочетании с которым могут реализовываться некоторые описанные варианты осуществления.

Декодер (600) принимает закодированные данные в кодированном потоке (605) видеобитов и формирует вывод, включающий в себя изображения для восстановленного видео (695). Формат кодированного потока (605) видеобитов может представлять собой варьирование или расширение H.265/HEVC-формата, формат Windows Media Video, VC-1-формат, MPEG-x-формат (например, MPEG-1, MPEG-2 или MPEG-4), H.26x-формат (например, H.261, H.262, H.263, H.264) или другой формат.

[0106] Изображение может быть организовано в качестве нескольких мозаичных фрагментов идентичного размера или различных размеров. Изображение также может быть организовано в качестве одного или более слайсов. Контент слайса или мозаичного фрагмента может быть дополнительно организован в качестве блоков или других наборов значений выборок. Декодер (600) является блочным и использует блочный формат, который зависит от реализации. Блоки дополнительно могут подразделяться на различных стадиях. Например, изображение может разделяться на блоки 64×64, блоки 32×32 или блоки 16×16, которые, в свою очередь, могут разделяться на меньшие блоки значений выборок. В реализациях декодирования для H.265/HEVC-стандарта, изображение сегментировано на CTU (CTB), CU (CB), PU (PB) и TU (TB). Если изображение закодировано при активированной WPP (с использованием WPP или способом, совместимым с использованием WPP в ходе декодирования), строки некоторых блоков (например, строки CTU согласно H.265/HEVC-стандарту) могут декодироваться параллельно с использованием WPP, как подробнее пояснено ниже.

[0107] Декодер (600) распаковывает изображения с использованием внутрикадрового декодирования и/или межкадрового декодирования. Многие компоненты декодера (600) используются как для внутрикадрового декодирования, так и для межкадрового декодирования. Точные операции, выполняемые посредством этих компонентов, могут

варьироваться в зависимости от типа распаковываемой информации.

[0108] Буфер принимает закодированные данные в кодированном потоке (605) видеобитов и делает принимаемые закодированные данные доступными для синтаксического анализатора (610)/энтропийного декодера. Синтаксический анализатор (610)/энтропийный декодер энтропийно декодирует энтропийно кодированные данные, типично применяет инверсию энтропийного кодирования, выполняемого в кодере (500) (например, выполняет контекстно-адаптивное двоичное арифметическое декодирование). В качестве результата синтаксического анализа и энтропийного декодирования, синтаксический анализатор (610)/энтропийный декодер формирует общие управляющие данные (622), данные (632) квантованных коэффициентов преобразования, данные (642) внутреннего прогнозирования, данные (652) движения и управляющие данные (662) фильтрации. Для данных (642) внутреннего прогнозирования, если значения индекса предиктора передаются в служебных сигналах, синтаксический анализатор (610)/энтропийный декодер может энтропийно декодировать значения индекса предиктора, например, с использованием контекстно-адаптивного двоичного арифметического декодирования. В некоторых случаях, синтаксический анализатор (610)/энтропийный декодер также энтропийно декодирует дифференциалы для BV-значений или других значений смещения (например, с использованием контекстно-адаптивного двоичного арифметического декодирования), затем комбинирует дифференциалы с соответствующими предикторами, чтобы восстанавливать BV-значения (или другие значения смещения). В других случаях, дифференциал опускается из потока битов, и BV-значение (или другое значение смещения) представляет собой просто предиктор (например, указываемый с помощью значения индекса предиктора). Для режима палитрового декодирования, синтаксический анализатор (610)/энтропийный декодер может декодировать палитровые данные. В частности, синтаксический анализатор (610)/энтропийный декодер может использовать палитровое прогнозирование, как пояснено ниже.

[0109] Общий контроллер (620) декодирования принимает общие управляющие данные (622) и предоставляет управляющие сигналы (не показаны) в другие модули (к примеру, в модуль (635) масштабирования/обратный преобразователь, модуль (645) внутрикадрового прогнозирования, модуль (655) компенсации движения и переключатель внутреннего/взаимного режима), чтобы задавать и изменять параметры декодирования в ходе декодирования.

[0110] Если текущее изображение прогнозируется с использованием межкадрового прогнозирования, модуль (655) компенсации движения принимает данные (652) движения, такие как MV-данные, данные выбора опорных изображений и значения индекса режима объединения. Модуль (655) компенсации движения применяет MV к восстановленному опорному изображению(ям) из буфера (670) декодированных изображений. Модуль (655) компенсации движения формирует прогнозирование с компенсацией движения для взаимно кодированных блоков текущего изображения. Буфер (670) декодированных изображений сохраняет одно или более ранее восстановленных изображений для использования в качестве опорных изображений.

[0111] В отдельном тракте в декодере (600), модуль (645) внутрикадрового прогнозирования принимает данные (642) внутреннего прогнозирования, такие как информация, указывающая то, внутрикадровое прогнозирование использует пространственное прогнозирование, прогнозирование с внутренним BC, прогнозирование с внутренним LC или прогнозирование с внутренним SC, а также направление режима прогнозирования (для внутреннего пространственного

прогнозирования), BV-значения (для прогнозирования с внутренним BC), значения смещения (для прогнозирования с внутренним LC) или значения смещения и значения длины (для прогнозирования с внутренним SC). Для внутреннего пространственного прогнозирования, с использованием значений восстановления (638) текущего изображения, согласно данным режима прогнозирования, модуль (645) внутрикадрового прогнозирования пространственно прогнозирует значения выборок текущего блока текущего изображения из соседних ранее восстановленных значений выборок текущего изображения. Альтернативно, для прогнозирования с внутренним копированием, модуль (645) внутрикадрового прогнозирования прогнозирует значения выборок текущего блока, линии, строки или другой секции с использованием ранее восстановленных значений выборок опорного блока, линии, строки или другой секции, которая указывается посредством сдвига (BV-значения, значения смещения и т.д.) для текущего блока, линии, строки и т.д. В некоторых случаях, BV-значение (или другое значение смещения) может быть прогнозированным значением. В других случаях, BV-значение (или другое значение смещения) может отличаться от своего прогнозированного значения, причем в этом случае BV-значение (или другое значение смещения) восстанавливается с использованием дифференциала и прогнозированного значения. Для режима внутреннего SC, модуль (645) внутрикадрового прогнозирования также использует значение длины строки при прогнозировании значений выборок текущей строки.

[0112] Для режима палитрового декодирования, декодер (600) использует палитру, которая представляет, по меньшей мере, некоторые значения выборок CU или другой единицы. Палитра преобразует значения индекса в соответствующие цвета, используемые в единице. Например, палитра преобразует значения 0, 1, 2,..., p индекса в соответствующие цвета, которые могут иметь формат RGB 4:4:4, формат BGR 4:4:4, формат GBR 4:4:4, формат YUV 4:4:4 или другой формат (цветовое пространство, частота дискретизации цветов). Значение индекса может представлять RGB-триплет, BGR-триплет или GBR-триплет для пиксела. В ходе декодирования, для позиций в единице, значения индекса из палитры заменяются надлежащими значениями выборок. Управляюще кодированное значение в единице может декодироваться с использованием значения управляющего кода и литеральных значений. Палитра может изменяться между единицами, на основе палитровых данных, передаваемых в служебных сигналах в потоке битов.

[0113] Переключатель внутреннего/взаимного режима выбирает значения прогнозирования с компенсацией движения или внутрикадрового прогнозирования для использования в качестве прогнозирования (658) для данного блока. Например, когда придерживаются H.265/HEVC-синтаксиса, переключатель внутреннего/взаимного режима может управляться на основе синтаксического элемента, закодированного для CU изображения, которое может содержать внутренне прогнозированные CU и взаимно прогнозированные CU. Когда остаточные значения закодированы/переданы в служебных сигналах, декодер (600) комбинирует прогнозирование (658) с восстановленными остаточными значениями, чтобы формировать восстановление (638) контента из видеосигнала. Когда остаточные значения не закодированы/переданы в служебных сигналах, декодер (600) использует значения прогнозирования (658) в качестве восстановления (638).

[0114] Чтобы восстанавливать остаток, когда остаточные значения закодированы/переданы в служебных сигналах, модуль (635) масштабирования/обратный преобразователь принимает и обрабатывает данные (632) квантованных коэффициентов

преобразования. В модуле (635) масштабирования/обратном преобразователе, модуль масштабирования/обратный квантователь выполняет обратное масштабирование и обратное квантование для квантованных коэффициентов преобразования. Обратный преобразователь частоты выполняет обратное преобразование частоты, формируя
 5 блоки восстановленных значений остатка прогнозирования или значений выборки. Например, обратный преобразователь частоты применяет обратное блочное преобразование к коэффициентам преобразования частоты, формируя данные значений выборки или данные остатков прогнозирования. Обратное преобразование частоты может представлять собой обратное DCT, его целочисленную аппроксимацию или
 10 другой тип обратного преобразования частоты (например, обратное дискретное синусное преобразование или его целочисленную аппроксимацию). Если преобразование частоты пропущено в ходе кодирования, обратное преобразование частоты также пропускается. В этом случае, модуль масштабирования/обратный квантователь может выполнять обратное масштабирование и обратное квантование для блоков данных
 15 остатков прогнозирования (или данных значений выборки), формируя восстановленные значения.

[0115] Для внутрикадрового прогнозирования, значения восстановления (638) могут быть возвращены в модуль (645) внутрикадрового прогнозирования. Для межкадрового прогнозирования, значения восстановления (638) дополнительно могут фильтроваться.
 20 В модуле (665) объединения/фильтре, декодер (600) объединяет контент из различных мозаичных фрагментов в восстановленную версию изображения. Декодер (600) избирательно выполняет фильтрацию для удаления блочности и SAO-фильтрацию согласно управляющим данным (662) фильтрации и правилам для адаптации фильтра, с тем чтобы адаптивно сглаживать неоднородности на границах в изображениях.
 25 Альтернативно или дополнительно, может применяться другая фильтрация (к примеру, фильтрация с подавлением периодически повторяющихся помех или ALF; не показаны). Границы мозаичных фрагментов могут избирательно фильтроваться или вообще не фильтроваться, в зависимости от настроек декодера (600) или синтаксического элемента в закодированных данных потоков битов. Буфер (670) декодированных изображений
 30 буферизует восстановленное текущее изображение для использования в последующем прогнозировании с компенсацией движения.

[0116] Декодер (600) также может включать в себя постобрабатывающий фильтр. Постобрабатывающий фильтр (608) может включать в себя фильтрацию для удаления блочности, фильтрацию с подавлением периодически повторяющихся помех, адаптивную
 35 фильтрацию Винера, фильтрацию на основе воспроизведения зерна пленки, SAO-фильтрацию или другой вид фильтрации. С учетом того, что "внутриконтурная" фильтрация выполняется для восстановленных значений выборки изображений в контуре компенсации движения и, следовательно, затрагивает значения выборки опорных изображений, постобрабатывающий фильтр (608) применяется к
 40 восстановленным значениям выборки за пределами контура компенсации движения перед выводом для отображения.

[0117] В зависимости от реализации и требуемого типа распаковки, модули декодера (600) могут добавляться, опускаться, разбиваться на несколько модулей, комбинироваться с другими модулями и/или заменяться аналогичными модулями. В
 45 альтернативных вариантах осуществления, декодеры с различными модулями и/или другими конфигурациями модулей выполняют одну или более описанных технологий. Конкретные варианты осуществления декодеров типично используют вариацию или дополненную версию декодера (600). Взаимосвязи, показанные между модулями в

декодере (600), указывают общие потоки информации в декодере; другие взаимосвязи не показаны для простоты.

VII. Правила для режимов внутрикадрового прогнозирования, когда WPP активируется

5 [0118] Этот раздел представляет примеры правил для режимов внутрикадрового прогнозирования, когда волновая параллельная обработка (WPP) активируется. Новшества упрощают использование режимов внутрикадрового прогнозирования, к примеру, режима палитрового прогнозирования, режима внутреннего блочного копирования (BC), режима внутреннего линейного копирования (LC) и режима
10 внутреннего строкового копирования (SC), посредством кодера или декодера, когда WPP активируется.

A. Волновая параллельная обработка - введение

[0119] В общем, WPP представляет собой инструментальное средство кодирования/декодирования, которое упрощает параллельную обработку посредством
15 дифференциальной задержки начала обработки для строк единиц в изображении. Когда WPP активируется, различные строки единиц в изображении могут быть закодированы или декодированы параллельно. В ходе кодирования или декодирования, первая строка единиц может обрабатываться "единица за единицей", слева направо. Обработка (кодирование или декодирование) второй строки единиц не должна ожидать завершения
20 обработки для всей первой строки единиц. Вместо этого, обработка второй строки единиц может начинаться после того, как обработка завершается для нескольких единиц первой строки, которые предоставляют информацию, используемую при обработке начальной единицы второй строки. Аналогично, обработка третьей строки единиц может начинаться после того, как обработка завершается для нескольких единиц второй
25 строки. Таким образом, WPP упрощает параллельную обработку различных строк единиц, т.е. различные подпроцессы или ядра обработки могут выполнять обработку для различных строк единиц на основе временной задержки со сдвигом.

[0120] Например, согласно H.265/HEVC-стандарту, когда WPP активируется, слайс разделяется на строки STU. В ходе кодирования или декодирования, первая строка
30 STU может обрабатываться "STU за STU". Обработка второй строки STU может начинаться после того, как обработка завершается для первых двух STU первой строки, которые предоставляют информацию (например, восстановленные значения выборок, восстановленные MV-значения или BV-значения, информацию контекстной модели), используемую при обработке начальной STU второй строки. Аналогично, обработка
35 третьей строки STU может начинаться после того, как обработка завершается для первых двух STU второй строки.

[0121] Фиг. 7 иллюстрирует синхронизацию (700) WPP для текущего изображения (710) согласно H.265/HEVC-стандарту. Изображение (710) сегментировано на STU, которые организованы в STU-столбцы и STU-строки. Различные STU-строки могут
40 быть закодированы или декодированы параллельно с использованием WPP. Синхронизация WPP отражает зависимости между STU в ходе кодирования или декодирования. В этом примере, данная STU может зависеть от информации (такой как восстановленные значения выборок, восстановленные MV-значения или BV-значения, информация контекстной модели) из (1) соседней STU слева от нее, (2) соседней
45 STU выше и слева от нее, (3) соседней STU выше данной STU и (4) соседней STU выше и справа от данной STU, если такие соседние STU (1)-(4) доступны (например, находятся в пределах изображения, в одном и том же слайсе и мозаичном фрагменте). Каждая соседняя STU, в свою очередь, может зависеть от своих соседних STU (1)-(4), если

доступны. Фиг. 8 иллюстрирует каскадные зависимости для начальной STU пятой STU-строки. Начальная STU пятой STU-строки зависит от первых двух STU четвертой STU-строки, которые совместно зависят от первых трех STU третьей STU-строки и т.д.

[0122] Со ссылкой на фиг. 7, для WPP, первая STU-строка (т.е. STU-строка 0) обрабатывается "STU за STU" для волны 0. Для волны 1, обработка начальной STU в STU-строке 1 может начинаться после того, как кодирование/декодирование второй STU в STU-строке 0 завершается. Аналогично, для волны 2, обработка начальной STU в STU-строке 2 может начинаться после того, как кодирование/декодирование второй STU в STU-строке 1 завершается. Для волны 3, обработка начальной STU в STU-строке 3 может начинаться после того, как кодирование/декодирование второй STU в STU-строке 2 завершается, и для волны 4, обработка начальной STU в STU-строке 4 начинается после того, как кодирование/декодирование второй STU в STU-строке 3 завершается.

[0123] Даже когда STU-строки обрабатываются параллельно, эта обработка с временной задержкой и со сдвигом обеспечивает то, что зависимости между STU удовлетворяются, когда обработка начинается для STU-строки. На фиг. 7, вывод с двумя STU для каждой STU-строки поддерживается в ходе обработки. Для каждой STU-строки, обработка для текущей STU (показана полужирным выделением) выполняется за две STU до обработки для текущей STU в следующей STU-строке. Тем не менее, на практике обработка для данной STU-строки может останавливаться или замедляться, вызывая каскадные задержки в обработке для последующих STU-строк, которые зависят от завершения (прямо или косвенно) обработки для STU в данной STU-строке. Для данной STU последующей STU-строки, зависимости каскадируются из предыдущих STU-строк. В примере (800), показанном на фиг. 8, для начальной STU пятой STU-строки кадра (810), первые две STU четвертой STU-строки обрабатываются. В противном случае, обработка для начальной STU пятой STU-строки не может начинаться. Вообще говоря, третья STU третьей STU-строки обрабатывается, поскольку иначе обработка для второй STU четвертой STU-строки не может начинаться. Аналогично, четвертая STU второй STU-строки обрабатывается, в качестве предварительного условия для обработки для третьей STU третьей STU-строки. В завершение, пятая STU первой STU-строки обрабатывается, в качестве предварительного условия для обработки для четвертой STU второй STU-строки. Таким образом, фиг. 8 показывает восстановленный контент, который гарантированно доступен для прогнозирования начальной STU в волне 4 (пятой STU-строке), когда WPP активируется.

[0124] Напротив, когда WPP не активируется, STU обрабатываются слева направо в STU-строке, "STU-строка за STU-строкой" сверху вниз изображения (или слайса и мозаичного фрагмента, при использовании).

Информация контекстной модели (также называемая "информацией САВАС-состояния" или "информацией энтропийного кодирования"), используемая для данной STU, может учитывать результаты обработки любых предыдущих STU в той же STU-строке и любых предыдущих STU-строках изображения (или слайса/мозаичного фрагмента), которые являются более ранними STU в порядке кодирования/декодирования. Например, обработка второй STU в STU-строке зависит от результатов обработки первой STU в STU-строке и т.д. В качестве другого примера, обработка для начальной STU в STU-строке зависит от результатов обработки для последней STU в предыдущей STU-строке в изображении (или слайсе/мозаичном фрагменте).

[0125] С другой стороны, когда WPP активируется, обработка для начальной STU

в STU-строке не зависит от результатов обработки для последней STU в предыдущей STU-строке в изображении (или слайсе/мозаичном фрагменте). Обработка для начальной STU в STU-строке может начинаться после обработки концов для второй STU в предыдущей STU-строке в изображении (или слайсе/мозаичном фрагменте), даже если информация контекстной модели из последней STU в предыдущей STU-строке недоступна. Обработка третьей STU в предыдущей STU-строке по-прежнему зависит от результатов обработки второй STU в этой STU-строке, идентично тому, когда WPP не активируется.

[0126] Когда WPP активируется, кодер может фактически использовать (или нет) WPP в ходе кодирования. Независимо, кодер принудительно активирует ограничения и правила прогнозирования, которые применяются, когда WPP фактически используется. Аналогично, когда WPP активируется, декодер может фактически использовать (или нет) WPP в ходе декодирования. Синтаксический элемент в потоке битов может указывать то, активируется или нет WPP для видеопоследовательности, набора изображений или изображения. Например, синтаксический элемент может передаваться в служебных сигналах в SPS, PPS или другой синтаксической структуре в потоке битов. В H.265/HEVC-реализациях, например, значение синтаксического элемента флага активации синхронизации энтропийного кодирования, который передается в служебных сигналах в синтаксической PPS-структуре, указывает то, активируется или нет WPP для изображения(й), ассоциированного с синтаксической PPS-структурой. Если флаг активации синхронизации энтропийного кодирования равен 1, WPP активируется для изображения(й). В противном случае, WPP не активируется для изображения(й).

В. Режимы палитрового кодирования/декодирования и палитровое прогнозирование - введение

[0127] В общем, режим палитрового кодирования/декодирования использует палитру, чтобы представлять, по меньшей мере, некоторые значения выборок единицы (например, CU в H.265/HEVC-реализациях или другой единицы). Например, палитра преобразует значения 0, 1, 2, ..., p индекса в соответствующие цвета, которые могут иметь формат RGB 4:4:4, формат BGR 4:4:4, формат GBR 4:4:4, формат YUV 4:4:4 или другой формат (цветовое пространство, частота дискретизации цветов). Значение индекса может представлять RGB-триплет, BGR-триплет или GBR-триплет для пиксела. Фиг. 9 показывает два примера палитр. Палитра для текущей единицы (т.е. "текущая палитра") включает в себя p значений 0, ..., p-1 индекса, которые ассоциированы с RGB-триплетами. Палитра для предыдущей единицы (представленная в "предыдущих палитровых данных") включает в себя q значений 0, ..., q-1 индекса, которые ассоциированы с RGB-триплетами. Значения p и q могут быть идентичными или отличающимися. В ходе кодирования, кодер может заменять значения выборок пиксела с использованием значения индекса, которое может быть дополнительно закодировано, например, с использованием энтропийного кодирования. В ходе декодирования, декодер может восстанавливать значения выборок пиксела из значения индекса с использованием палитры, например, после энтропийного декодирования значения индекса.

[0128] В частности, режим палитрового кодирования/декодирования может быть эффективным, когда единица включает в себя относительно небольшое число разных цветов, что является общей характеристикой для видео экранного контента. Например, CU 64×64 в формате RGB 4:4:4 включает в себя 64×64=4096 пикселей, но может включать в себя гораздо меньшее число цветов (например, 1-20 цветов). Редкий цвет в единице может быть закодирован с использованием значения управляющего кода и литеральных значений (для соответствующих значений выборок) вместо включения редкого цвета

непосредственно в палитру.

[0129] Две единицы могут использовать одну и ту же палитру, но цвета типично изменяются между единицами в изображении. В связи с этим, палитры типично изменяются между единицами в изображении. Для палитр, кодер передает в служебных сигналах палитровые данные, которые могут энтропийно кодироваться в потоке битов. Декодер принимает и синтаксически анализирует палитровые данные, энтропийно декодирует их надлежащим образом и восстанавливает палитры. Кодер и декодер могут использовать палитровое прогнозирование, чтобы уменьшать скорость передачи битов, ассоциированную с передачей служебных сигналов палитровых данных.

[0130] В общем, палитровое прогнозирование может представлять собой любой подход, используемый в ходе кодирования или декодирования для того, чтобы прогнозировать значения палитры для текущей единицы (текущей палитры) с использованием палитровых данных из одной или более палитр для предыдущих единиц (предыдущих палитровых данных). Хотя цвета типично изменяются между единицами в изображении, во многих случаях, по меньшей мере, некоторые цвета в данной единице также используются в смежной единице. Палитровое прогнозирование использует это наблюдение для того, чтобы уменьшать скорость передачи битов для палитровых данных.

[0131] Фиг. 9 показывает упрощенный пример (900) палитрового прогнозирования, в котором палитра для текущей единицы (текущая палитра) прогнозируется с использованием палитры для предыдущей единицы (представленной в предыдущих палитровых данных). Например, предыдущие палитровые данные могут быть из палитры, используемой посредством предыдущей единицы, закодированной или декодированной перед текущей единицей, если предыдущая единица использует режим палитрового кодирования/декодирования. Если предыдущая единица не использует режим палитрового кодирования/декодирования, предыдущие палитровые данные для текущей единицы могут быть "унаследованы" из предыдущей единицы. Иными словами, предыдущие палитровые данные, которые доступны для предыдущей единицы, могут быть повторно использованы в качестве предыдущих палитровых данных, доступных для текущей единицы. (В некоторых примерных реализациях, предыдущие палитровые данные, фактически, являются информацией состояния или состояния для текущей единицы. Данная единица, которая сама не использует режим палитрового кодирования/декодирования, по-прежнему имеет предыдущие палитровые данные, которые могут быть унаследованы посредством следующей единицы). Таким образом, через цепочку наследования, предыдущие палитровые данные для текущей единицы могут включать в себя палитровые данные из палитры последнего ранее обработанной единицы (если имеется), которая использует режим палитрового кодирования/декодирования.

[0132] Для палитрового прогнозирования, кодер может определять, является ли текущая палитра идентичной предыдущим палитровым данным. Если да, кодер может просто указывать то, что предыдущие палитровые данные должны быть повторно использованы в качестве текущей палитры. Иначе (текущая палитра отличается от предыдущих палитровых данных), кодер определяет изменения между предыдущими палитровыми данными и текущей палитрой и передает в служебных сигналах синтаксические элементы, которые указывают изменения. Например, как показано на фиг. 9, для данного значения индекса текущей палитры, кодер передает в служебных сигналах то, представляет собой соответствующий цвет для данного значения индекса цвет в предыдущих палитровых данных ("prev") или новый цвет ("new"). Если соответствующий цвет представляет собой цвет в предыдущих палитровых данных,

кодер передает в служебных сигналах значение индекса (из предыдущих палитровых данных) для цвета в предыдущих палитровых данных, которые могут использоваться для того, чтобы заполнять текущую палитру. На фиг. 9, цвет для значения 1 индекса из предыдущих палитровых данных переназначен значению 0 индекса в текущей палитре, а цвет для значения 0 индекса предыдущих палитровых данных переназначен значению 2 индекса в текущей палитре. Таким образом, цвета могут изменять позиции между палитрами, например, так что наиболее распространенные цвета имеют наименьшие значения индекса, что позволяет повышать эффективность энтропийного кодирования. Если соответствующий цвет для данного значения индекса представляет собой новый цвет, кодер передает в служебных сигналах триплет для нового цвета. На фиг. 9, например, для значения 1 индекса в текущей палитре, кодер передает в служебных сигналах новый триплет (215, 170, 200), который используется для того, чтобы обновлять текущую палитру.

[0133] На основе синтаксических элементов, передаваемых в служебных сигналах посредством кодера, декодер может определять, является или нет текущая палитра идентичной предыдущим палитровым данным, которые доступны в декодере. Если да, декодер может повторно использовать предыдущие палитровые данные в качестве текущей палитры. Иначе (текущая палитра отличается от предыдущих палитровых данных), декодер принимает и синтаксически анализирует синтаксические элементы, которые указывают изменения между предыдущими палитровыми данными и текущей палитрой. Например, как показано на фиг. 9, для данного значения индекса текущей палитры, декодер определяет то, представляет собой соответствующий цвет для данного значения индекса цвет в предыдущих палитровых данных ("prev") или новый цвет ("new"). Если соответствующий цвет представляет собой цвет в предыдущих палитровых данных, декодер принимает и синтаксически анализирует синтаксический элемент, который указывает значение индекса (из предыдущих палитровых данных) для цвета в предыдущих палитровых данных, которые могут использоваться для того, чтобы заполнять текущую палитру. Если соответствующий цвет для данного значения индекса представляет собой новый цвет, декодер принимает и синтаксически анализирует синтаксические элементы, которые указывают новый цвет.

[0134] После составления текущей палитры, кодер и декодер обновляют предыдущие палитровые данные для следующей единицы, чтобы сохранять палитровые данные из текущей палитры. Эти новые "предыдущие палитровые данные" могут использоваться для того, чтобы прогнозировать палитру для следующей единицы.

[0135] Фиг. 9 показывает упрощенный пример (900). На практике, синтаксические элементы и правила, используемые для того, чтобы передавать в служебных сигналах палитровые данные, могут быть более сложными. На предмет дополнительных сведений по палитровому кодированию/декодированию и палитровому прогнозированию в примерных реализациях следует обратиться, например, к разделам 7.3.8.8, 7.4.9.6 и 8.4.1 JCTVC-R1005. Альтернативно, другой подход используется для того, чтобы передавать в служебных сигналах палитровые данные.

С. Палитровое прогнозирование, когда WPP активируется

[0136] В общем, предыдущие палитровые данные для целей палитрового прогнозирования используются для того, чтобы прогнозировать палитру текущей единицы (текущую палитру). Например, предыдущие палитровые данные могут представлять собой фактические палитровые данные из соседней единицы, закодированной или декодированной перед текущей единицей (если соседняя единица использует режим палитрового кодирования/декодирования). Альтернативно, (если

соседняя единица не использует режим палитрового кодирования/декодирования), предыдущие палитровые данные, доступные для соседней единицы, могут быть повторно использованы (или "унаследованы") в качестве предыдущих палитровых данных для текущей единицы.

5 [0137] В некоторых примерных реализациях, когда WPP не активируется, для начальной единицы в текущей строке, предыдущие палитровые данные из последней единицы в предыдущей строке (например, из палитры последней единицы, если последняя единица использует режим палитрового кодирования/декодирования; иначе, предыдущие палитровые данные, доступные для последней единицы). Когда WPP активируется, для
10 начальной единицы в текущей строке, предыдущие палитровые данные из последней единицы в предыдущей строке могут быть недоступны. Тем не менее, сброс палитрового прогнозирования в начале каждой строки единиц (отсутствие предиктора для палитрового прогнозирования для начальной единицы в текущей единице) может оказывать негативное влияние на эффективность кодирования.

15 [0138] Вместо этого, когда WPP активируется, кодер или декодер может использовать предыдущие палитровые данные из одной из первых двух единиц в предыдущей строке (например, одной из первых двух CU в первых двух STU предыдущей строки), которые уже обработаны для начала обработки начальной единицы в текущей строке. Например, для начальной единицы в текущей строке, предыдущие палитровые данные для целей
20 палитрового прогнозирования из второй единицы предыдущей строки (фактические палитровые данные второй единицы, если вторая единица использует режим палитрового кодирования/декодирования, или предыдущие палитровые данные, доступные для второй единицы предыдущей строки, в противном случае). Таким образом, для начальной единицы в текущей строке, если вторая единица не использует режим
25 палитрового кодирования/декодирования, предыдущие палитровые данные потенциально являются палитровыми данными из первой единицы в предыдущей строке (если первая единица использует режим палитрового кодирования/декодирования), или (если ни одна из первых двух единиц в предыдущей строке не использует режим палитрового кодирования/декодирования), предыдущими палитровыми данными из
30 первой единицы предыдущей строки, которая может зависеть от палитры для второй единицы строки перед предыдущей строкой, и т.д. (Альтернативно, если ни одна из первых двух единиц в предыдущей строке не использует режим палитрового кодирования/декодирования, предыдущие палитровые данные могут быть нулем, без предиктора, или использовать предиктор по умолчанию). После начальной единицы в
35 текущей строке, для текущей единицы, предыдущие палитровые данные для целей палитрового прогнозирования являются палитровыми данными из соседней единицы, закодированной или декодированной перед текущей единицей (если соседняя единица использует режим палитрового кодирования/декодирования), или (если соседняя единица не использует режим палитрового кодирования/декодирования) предыдущими
40 палитровыми данными, доступными для соседней единицы, идентично тому, когда WPP не активируется.

[0139] Фиг. 10 показывает зависимости (1000) при палитровом прогнозировании для текущего изображения (1010), обработанного при активированной WPP согласно некоторым примерным реализациям. Изображение (1010) организовано посредством
45 STU-строк и STU-столбцов. В примере по фиг. 10, каждая STU включает в себя одну CU. Если обобщать, данная STU может рекурсивно разбиваться на несколько CU (например, одна CU может разбиваться на четыре CU, каждая из которых дополнительно может разбиваться на меньшие CU). Состояние обработки (например, кодирование

или декодирование текущих STU), в общем, соответствует состоянию обработки, показанному на фиг. 7. Стрелки на фиг. 10 показывают направление палитрового прогнозирования, когда WPP активируется. Для начальной CU в первой STU-строке, предиктор отсутствует. Альтернативно, может использоваться предиктор по умолчанию.

5 Для каждой последующей CU в первой STU-строке, предыдущие палитровые данные находятся от CU слева. Для начальной CU в любой STU-строке после первой STU-строки, предыдущие палитровые данные исходят из второй CU предыдущей строки. Для каждой последующей CU в любой STU-строке после первой STU-строки, предыдущие палитровые данные исходят из предыдущей CU в той же строке. Для любой
10 CU, которая использует режим палитрового кодирования/декодирования, его палитра используется для того, чтобы составлять предыдущие палитровые данные для прогнозирования палитры следующей CU. Если данная CU не использует режим палитрового кодирования/декодирования, предыдущие палитровые данные, которые доступны для данной CU, сохраняются в качестве предыдущих палитровых данных
15 следующей CU.

[0140] Когда STU включает в себя несколько CU для целей палитрового прогнозирования для текущей CU, предыдущая CU может быть более ранней CU в порядке Z-сканирования. Более ранняя CU может находиться в той же STU или другой STU (для первой CU STU). Раздел VII.D поясняет, со ссылкой на фиг. 17, примеры
20 порядка Z-сканирования в некоторых примерных реализациях.

[0141] Кодер и декодер могут использовать запоминающее устройство для того, чтобы сохранять предыдущие палитровые данные, используемые для палитрового прогнозирования. Объем палитровых данных, которые сохранены для предыдущих палитровых данных, зависит от реализации. В общем, предыдущие палитровые данные
25 могут включать в себя количество C цветов в предыдущих палитровых данных и подробности значений выборок (например, цветовые триплеты в формате RGB 4:4:4, формате GBR 4:4:4, формате BGR 4:4:4, формате YUV 4:4:4 или другом формате) для цветов. Кодер и декодер могут сохранять все палитровые данные для предыдущей палитры. Тем не менее, сохранение всех палитровых данных для предыдущей палитры
30 может потреблять существенный объем запоминающего устройства для большой палитры. Альтернативно, чтобы уменьшать потребление запоминающего устройства в некоторых случаях, кодер и декодер ограничивают C посредством порогового количества, причем пороговое количество зависит от реализации. В этом случае, кодер и декодер сохраняют предыдущие палитровые данные для первых C различных цветов
35 в предыдущей палитре, при этом C ограничено посредством порогового количества. Например, пороговое количество равно 16 или 32. Альтернативно, когда WPP активируется, для палитрового прогнозирования для начальной единицы текущей строки, кодер и декодер сохраняют предыдущие палитровые данные, самое большее, для первых C1 различных цветов. После этого, кодер и декодер сохраняют предыдущие
40 палитровые данные, самое большее, для первых C2 различных цветов для палитрового прогнозирования в текущей строке. Например, C1 равно 16 или 32, и C2 равно 64.

[0142] Фиг. 11 показывает обобщенную технологию (1100) для кодирования, которая включает в себя палитровое прогнозирование, когда WPP активируется. Кодер, к примеру, кодер, показанный на фиг. 3 или фиг. 5a и 5b, либо другой кодер может
45 выполнять технологию (1100). Фиг. 12 показывает обобщенную технологию (1200) для декодирования, которая включает в себя палитровое прогнозирование, когда WPP активируется. Декодер, к примеру, декодер, показанный на фиг. 4 или фиг. 6, либо другой декодер может выполнять технологию (1200).

[0143] Со ссылкой на фиг. 11, кодер кодирует (1110) изображение при активированной WPP. Кодирование (1110) формирует закодированные данные. В качестве части кодирования (1110), для режима палитрового кодирования, кодер прогнозирует палитру для начальной единицы в текущей WPP-строке изображения с использованием предыдущих палитровых данных из предыдущей единицы в предыдущей WPP-строке изображения. Кодер выводит (1120) закодированные данные в качестве части потока битов.

[0144] Со ссылкой на фиг. 12, декодер принимает (1210) закодированные данные в качестве части потока битов. Декодер декодирует (1220) закодированные данные при активированной WPP. Декодирование (1220) восстанавливает изображение. В качестве части декодирования (1220), для режима палитрового декодирования, декодер прогнозирует палитру для начальной единицы в текущей WPP-строке изображения с использованием предыдущих палитровых данных из предыдущей единицы в предыдущей WPP-строке изображения.

[0145] В некоторых примерных реализациях, в кодере или декодере, текущая WPP-строка и предыдущая WPP-строка представляют собой строки STU, и начальная единица и предыдущая единица представляют собой CU. Альтернативно, WPP-строки представляют собой строки другого типа единицы, и/или начальные и предыдущие единицы представляют собой другой тип единицы. Синтаксический элемент в потоке битов может указывать то, активируется или нет WPP для видеопоследовательности, набора изображений или изображения. Синтаксический элемент может передаваться в служебных сигналах в синтаксической SPS-структуре, синтаксической PPS-структуре (например, синтаксический элемент флага активации синхронизации энтропийного кодирования в H.265/HEVC-реализациях) или другой синтаксической структуре в потоке битов.

[0146] В общем, в ходе кодирования или декодирования, палитра для начальной единицы в текущей WPP-строке представляет, по меньшей мере, некоторые цвета, используемые в начальной единице в текущей WPP-строке. Предыдущие палитровые данные из предыдущей единицы в предыдущей WPP-строке могут представлять, по меньшей мере, некоторые цвета, используемые в палитре для предыдущей единицы в предыдущей WPP-строке (если предыдущая единица использует режим палитрового кодирования/декодирования). Альтернативно, (если предыдущая единица не использует режим палитрового кодирования/декодирования), предыдущие палитровые данные из предыдущей единицы в предыдущей WPP-строке, возможно, унаследованы посредством предыдущей единицы из более ранней единицы, которая, возможно, использует режим палитрового кодирования/декодирования либо непосредственно наследует предыдущие палитровые данные. Цвета, не представленные в палитре, могут управляюще кодироваться. Предыдущая WPP-строка может представлять собой WPP-строку непосредственно выше текущей WPP-строки. Предыдущая единица в предыдущей WPP-строке может находиться выше начальной единицы в текущей WPP-строке (например, первой единицы в предыдущей строке на фиг. 10), либо она может находиться выше и справа от начальной единицы в текущей WPP-строке (например, второй единицы в предыдущей строке на фиг. 10). После обработки начальной единицы в текущей WPP-строке, для палитры последующей единицы в текущей WPP-строке изображения, кодер или декодер может прогнозировать палитру (последующей единицы) с использованием палитровых данных из палитры для начальной единицы в текущей WPP-строке.

[0147] В ходе кодирования или декодирования, палитровое прогнозирование может включать в себя несколько стадий. Например, в ходе кодирования, кодер может

определять (на основе сравнения текущей палитры с предыдущими палитровыми данными) то, следует или нет повторно использовать предыдущие палитровые данные из предыдущей единицы в предыдущей WPP-строке в качестве палитры для начальной единицы в текущей WPP-строке (текущей палитре). В ходе декодирования, декодер может определять (на основе информации, передаваемой в служебных сигналах посредством кодера) то, следует или нет повторно использовать предыдущие палитровые данные из предыдущей единицы в предыдущей WPP-строке в качестве палитры для начальной единицы в текущей WPP-строке (текущей палитре). В качестве другого примера, в ходе кодирования, когда существуют изменения палитры, кодер может выбирать один или более цветов из предыдущих палитровых данных из предыдущей единицы в предыдущей WPP-строке для того, чтобы включать в палитру для начальной единицы в текущей WPP-строке (текущей палитре). Кодер передает в служебных сигналах синтаксические элементы, указывающие выбранные цвета. Затем в ходе декодирования, декодер может принимать и синтаксически анализировать синтаксические элементы и, на основе синтаксических элементов, выбирать один или более цветов из предыдущих палитровых данных из предыдущей единицы в предыдущей WPP-строке для того, чтобы включать в палитру для начальной единицы в текущей WPP-строке (текущей палитре). Альтернативно, кодер и декодер могут использовать другой подход для того, чтобы передавать в служебных сигналах палитровые данные.

[0148] Кодер и декодер могут сохранять все предыдущие палитровые данные из предыдущей единицы в предыдущей WPP-строке. Например, кодер и декодер могут сохранять, для каждого из C цветов в предыдущих палитровых данных из предыдущей единицы в предыдущей WPP-строке, одно или более значений цветовых компонентов. Альтернативно, чтобы уменьшать потребление запоминающего устройства в некоторых случаях, кодер и декодер могут ограничивать C посредством порогового количества для предыдущих палитровых данных, так что палитровые данные для первых C цветов, ограниченные посредством порогового количества, сохраняются для палитрового прогнозирования.

D. Прогнозирование с внутренним копированием - введение

[0149] В общем, режим внутреннего копирования использует прогнозирование с внутренним копированием, в котором текущий блок, линия, строка или другая секция в изображении прогнозируются с использованием ранее восстановленных значений выборок в опорном блоке, линии, строке или другой секции в том же изображении. Например, режим внутреннего копирования может использовать прогнозирование с внутренним блочным копированием (BC), прогнозирование с внутренним линейным копированием (LC) или прогнозирование с внутренним строковым копированием (SC).

[0150] В общем, режим внутреннего BC использует прогнозирование с внутренним BC, в котором значения выборок текущего блока изображения прогнозируются с использованием значений выборок в том же изображении. Значение блочного вектора (BV) указывает сдвиг от текущего блока до блока изображения ("опорного блока"), который включает в себя значения выборок, используемые для прогнозирования. Опорный блок предоставляет прогнозируемые значения для текущего блока. Значения выборок, используемые для прогнозирования, являются ранее восстановленными значениями выборок, которые в силу этого доступны в кодере в ходе кодирования и в декодере в ходе декодирования. BV-значение передается в служебных сигналах в потоке битов, и декодер может использовать BV-значение для того, чтобы определять опорный блок изображения, который следует использовать для прогнозирования.

[0151] Фиг. 13 показывает пример (1300) прогнозирования с внутренним BC для

текущего блока (1330) текущего изображения (1310). Текущий блок может представлять собой блок кодирования (CB) единицы кодирования (CU), прогнозный блок (PB) единицы прогнозирования (PU), блок преобразования (TB) единицы преобразования (TU) или другой блок. Размер текущего блока может составлять 64×64, 32×32, 16×16, 8×8 или
 5 некоторый другой размер. Блоки могут быть симметрично или асимметрично сегментированы на меньшие блоки для целей прогнозирования с внутренним ВС. Если обобщать, размер текущего блока составляет $m \times n$, где каждое из m и n является целым числом, и где m и n могут быть равны друг другу или могут иметь различные значения. Таким образом, текущий блок может быть квадратным или прямоугольным.

10 Альтернативно, текущий блок может иметь некоторую другую форму.

[0152] BV (1340) указывает сдвиг (или смещение) от текущего блока (1330) до опорного блока (1350) изображения, которое включает в себя значения выборок, используемые для прогнозирования. Опорный блок (1350) может быть идентичным
 15 текущему блоку (1330), или он может представлять собой аппроксимацию текущего блока (1330). Предположим, что левая верхняя позиция текущего блока находится в позиции (x_0, y_0) в текущем изображении, и предположим, что левая верхняя позиция опорного блока находится в позиции (x_1, y_1) в текущем изображении. BV указывает сдвиг $(x_1 - x_0, y_1 - y_0)$. Например, если левая верхняя позиция текущего блока находится в позиции $(256, 128)$, и левая верхняя позиция опорного блока находится в позиции $(126,$
 20 $104)$, BV-значение составляет $(-130, -24)$. В этом примере, отрицательный горизонтальный сдвиг указывает позицию слева от текущего блока, а отрицательный вертикальный сдвиг указывает позицию выше текущего блока.

[0153] Прогнозирование с внутренним ВС может повышать эффективность кодирования посредством использования избыточности (к примеру, повторяющихся
 25 шаблонов в изображении) с использованием ВС-операций. Тем не менее, нахождение совпадающего опорного блока для текущего блока может быть вычислительно сложным и длительным, с учетом числа возможных вариантов опорных блоков, которые может оценивать кодер. Фиг. 14 показывает пример (1400), иллюстрирующий некоторые возможные варианты опорных блоков для текущего блока (1430) текущего изображения
 30 (1410), когда WPP не активируется. Четыре BV (1441, 1442, 1443, 1444) указывают сдвиги для четырех возможных вариантов опорных блоков. Когда WPP не активируется, возможные варианты опорных блоков могут находиться в любом месте в восстановленном контенте текущего изображения (1410). (Блоки, в общем, кодируются слева направо, затем сверху вниз). Возможный вариант опорного блока может
 35 перекрываться с другими возможными вариантами опорных блоков, как показано для возможных вариантов опорных блоков, указываемых посредством BV (1443, 1444). В некоторых примерных реализациях, опорный блок ограничивается нахождением в пределах того же слайса и мозаичного фрагмента, что и текущий блок. Такое прогнозирование с внутренним ВС не использует значения выборок в других слайсах
 40 или мозаичных фрагментах. Местоположение опорного блока может подвергаться одному или более другим ограничениям, например, ограничению, когда WPP активируется, как описано ниже.

[0154] Блок с режимом прогнозирования для прогнозирования с внутренним ВС может представлять собой CB, PB или другой блок.

45 Когда блок представляет собой CB, BV для блока может передаваться в служебных сигналах на уровне CU (и другие CB в CU используют идентичный BV или его масштабированную версию). Альтернативно, когда блок представляет собой PB, BV для блока может передаваться в служебных сигналах на уровне PU (и другие PB в PU

используют идентичный BV или его масштабированную версию). Если обобщать, BV для блока прогнозирования с внутренним BC передается в служебных сигналах на надлежащем уровне синтаксиса для блока.

[0155] В общем, режим внутреннего LC использует прогнозирование с внутренним LC, в котором значения выборок текущей линии текущего блока изображения прогнозируются с использованием значений выборок в том же изображении. Значение смещения указывает сдвиг от текущей линии до линии изображения ("опорной линии"), которая включает в себя значения выборок, используемые для прогнозирования. Значение смещения передается в служебных сигналах в потоке битов, и декодер может использовать значение смещения для того, чтобы определять опорную линию, которую следует использовать для прогнозирования.

[0156] Фиг. 15 иллюстрирует пример (1500) прогнозирования с внутренним LC для линий текущего блока (1530) текущего изображения. Текущий блок может представлять собой CB CU, PB PU, TB TU или другой блок. Размер текущего блока может составлять 64×64, 32×32, 16×16, 8×8 или некоторый другой размер. Если обобщать, размер текущего блока составляет $m \times n$, где каждое из m и n является целым числом, и где m и n могут быть равны друг другу или могут иметь различные значения. Таким образом, текущий блок может быть квадратным или прямоугольным. Для прогнозирования с внутренним LC, блок разбивается на горизонтальные линии или вертикальные линии. Горизонтальные линии имеют высоту в 1 выборку и имеют ширину текущего блока. Вертикальные линии имеют ширину в 1 выборку и имеют высоту текущего блока.

[0157] На фиг. 15, первое смещение (1551) указывает сдвиг от первой линии (1541) текущего блока (1530) до опорной линии (1561), которая включает в себя значения выборок, используемые для прогнозирования первой линии (1541). Опорная линия может быть идентичной данной линии, или она может представлять собой аппроксимацию данной линии. Второе смещение (1552) указывает сдвиг от второй линии (1542) текущего блока (1530) до опорной линии (1562), которая включает в себя значения выборок, используемые для прогнозирования второй линии (1542). Значение (1551, 1552) смещения является аналогичным BV-значению в том, что оно указывает сдвиг в текущем изображении. Хотя фиг. 15 показывает горизонтальные линии, текущий блок (1530) вместо этого может разбиваться на вертикальные линии для прогнозирования с внутренним LC. Синтаксический элемент в расчете на блок, в расчете на единицу или в расчете на изображение может указывать то, используются горизонтальные или вертикальные линии для прогнозирования с внутренним LC.

[0158] Прогнозирование с внутренним LC может повышать эффективность кодирования посредством использования избыточности (к примеру, повторяющихся шаблонов в изображении) с использованием LC-операций. Когда WPP не активируется, возможные варианты опорных линий могут находиться в любом месте в восстановленном контенте текущего изображения. Возможный вариант опорной линии может перекрываться с другими возможными вариантами опорных линий. В некоторых примерных реализациях, опорная линия ограничивается нахождением в пределах того же слайса и мозаичного фрагмента, что и текущая линия. Местоположение опорной линии может подвергаться одному или более другим ограничениям, например, ограничению, когда WPP активируется, как описано ниже.

[0159] В общем, режим внутреннего SC использует прогнозирование с внутренним SC, в котором значения выборок текущей строки текущего блока изображения прогнозируются с использованием значений выборок в том же изображении. Значение смещения указывает сдвиг от текущей строки до строки изображения ("опорной строки"),

которая включает в себя значения выборок, используемые для прогнозирования. Значение длины строки указывает длину строки с точки зрения значений выборок. Значение смещения и значение длины строки передаются в служебных сигналах в потоке битов, и декодер может использовать значение смещения и значение длины строки для того, чтобы определять опорную строку, которую следует использовать для прогнозирования.

[0160] Фиг. 16 показывает пример (1600) прогнозирования с внутренним SC для строк текущего блока (1630) текущего изображения. Текущий блок может представлять собой CB CU, PB PU, TB TU или другой блок. Размер текущего блока может составлять 64×64, 32×32, 16×16, 8×8 или некоторый другой размер. Если обобщать, размер текущего блока составляет $m \times n$, где каждое из m и n является целым числом, и где m и n могут быть равны друг другу или могут иметь различные значения. Таким образом, текущий блок может быть квадратным или прямоугольным. Для прогнозирования с внутренним SC, блок разбивается на одну или более строк. На фиг. 16, текущий блок (1630) разбивается на три строки. Строки блока могут сканироваться горизонтально, т.е. слева направо в данной строке, затем следующая строка и т.д. Альтернативно, строки блока могут сканироваться вертикально, т.е. сверху вниз в данном столбце, затем следующий столбец и т.д. Строки на фиг. 16 сканируются горизонтально.

[0161] На фиг. 16, первое смещение (1651) указывает сдвиг от первой строки (1641) текущего блока (1630) до опорной строки (1661), которая включает в себя значения выборок, используемые для прогнозирования первой строки (1641). Длина каждой первой строки (1641) и соответствующей опорной строки (1661) составляет 6 значений выборок. Опорная строка может быть идентичной данной строке, или она может представлять собой аппроксимацию данной строки. Второе смещение (1652) указывает сдвиг от второй строки (1642) текущего блока (1630) до опорной строки (1662), которая включает в себя значения выборок, используемые для прогнозирования второй строки (1642). Длина каждой второй строки (1642) и соответствующей опорной строки (1662) составляет 14 значений выборок. Смещение не показано для третьей строки текущего блока (1630), которая имеет длину в 44 выборки. Аналогично BV-значению, значение (1651, 1652) смещения указывает сдвиг в текущем изображении. Хотя фиг. 16 показывает горизонтальное сканирование, текущий блок (1630) вместо этого может разбиваться на строки, которые сканируются вертикально для прогнозирования с внутренним SC. Синтаксический элемент в расчете на блок, в расчете на единицу или в расчете на изображение может указывать то, используется горизонтальный или вертикальный порядок сканирования для прогнозирования с внутренним SC.

[0162] Прогнозирование с внутренним SC может повышать эффективность кодирования посредством использования избыточности (к примеру, повторяющихся шаблонов в изображении) с использованием SC-операций. Прогнозирование с внутренним SC является более гибким по сравнению с прогнозированием с внутренним BC или прогнозированием с внутренним LC (обеспечивает возможность сегментации на произвольные строки, а не только на фиксированные сегменты), но передает в служебных сигналах больший объем информации (значения длины строки в дополнение к значениям смещения). Когда WPP не активируется, возможные варианты опорных строк могут находиться в любом месте в восстановленном контенте текущего изображения. Возможный вариант опорной строки может перекрываться с другими возможными вариантами опорных строк. В некоторых примерных реализациях, опорная строка ограничивается нахождением в пределах того же слайса и мозаичного фрагмента, что и текущая строка. Местоположение опорной строки может подвергаться одному

или более другим ограничениям, например, ограничению, когда WPP активируется, как описано ниже.

[0163] Операции прогнозирования с внутренним копированием для режима внутреннего СВ, режима внутреннего LC или режима внутреннего SC могут применяться на уровне СВ (когда BV-значение или другое значение смещения передается в служебных сигналах в расчете на СВ или его часть) или РВ (когда BV-значение или другое смещение передаются в служебных сигналах в расчете на РВ или его часть). В этом случае, опорная область ограничивается таким образом, чтобы не перекрывать текущую область или блок, включающий в себя текущую область. Альтернативно, операции прогнозирования с внутренним копированием могут применяться для меньших секций в РВ или СВ, даже когда BV-значение или другое значение смещения передается в служебных сигналах для РВ или СВ (или его части). Например, для первой секции блока, опорная область включает в себя позиции за пределами блока. Тем не менее, для второй секции блока, опорная область, используемая в операциях прогнозирования с внутренним копированием, может включать в себя позиции в ранее восстановленной первой секции того же блока. Таким образом, BV-значение или значение смещения может ссылаться на позиции в том же РВ или СВ. Предоставление возможности применения операций прогнозирования с внутренним копированием для секций в РВ или СВ упрощает использование BV-значений или других смещений с относительно небольшими абсолютными величинами.

[0164] Когда опорная область для текущей области в текущей единице может включать в себя позицию в той же единице, кодер может проверять достоверность BV-значения или другого значения смещения с учетом порядков Z-сканирования текущей области и опорной области (которые находятся в одном и том же слайсе и одном и том же мозаичном фрагменте). Например, кодер проверяет то, что порядок Z-сканирования блока, содержащего правую нижнюю позицию $(x_0 + offset_x + m - 1, y_0 + offset_y + n - 1)$ опорной области, меньше порядка Z-сканирования блока, содержащего левую верхнюю позицию (x_0, y_0) текущей области, где $offset$ указывает значение смещения, и текущая область и опорная область имеют размеры $m \times n$. Если да, блок, который содержит правую нижнюю позицию опорной области, ранее восстановлен (а следовательно, также и остальная часть опорной области). Кодер также может проверять то, что значение смещения удовлетворяет, по меньшей мере, одному из условий $offset_x + m \leq 0$ и $offset_y + n \leq 0$, что обеспечивает то, что опорная область не перекрывает текущую область.

[0165] В общем, порядок Z-сканирования придерживается последовательно указанного упорядочения единиц, которые сегментируют изображение. Фиг. 17 показывает примерный порядок (1700) Z-сканирования для текущей области (1730) и единицы, которые могут включать в себя правую нижнюю позицию опорной области. Текущая область (1730) может представлять собой СВ, РВ либо другой блок, линию или прямоугольник, который включает в себя строку. Порядки Z-сканирования, в общем, назначаются единицам последовательно слева направо в строке, повторяющейся в последовательных строках сверху вниз. Когда единица разбивается, порядки Z-сканирования назначаются в разбитой единице, рекурсивно. Для реализаций кодирования/декодирования для H.265/HEVC-стандарта, порядок Z-сканирования продолжается от STU к STU вдоль шаблона растрового сканирования STU (слева направо в STU-строке, повторяясь в последовательных STU-строках сверху вниз). Если STU разбивается, порядок Z-сканирования придерживается шаблона растрового сканирования для CU дерева квадрантов в разбитой STU. Так же, если CU разбивается

(например, на несколько CU или на несколько PU), порядок Z-сканирования придерживается шаблона растрового сканирования для блоков в разбитой CU.

[0166] В некоторых примерных реализациях, BV-значение или другое значение смещения передается в служебных сигналах для CU, PU или другой единицы и применяется для всех блоков единицы. В зависимости от цветового пространства и частоты дискретизации цветов, BV-значение или другое значение смещения может использоваться для всех блоков без масштабирования, либо оно может масштабироваться для блоков в различных цветовых компонентах. Альтернативно, различные BV-значения или другие значения смещения могут передаваться в служебных сигналах для различных блоков единицы. В некоторых примерных реализациях, один и тот же режим прогнозирования (например, режим внутреннего BC) применяется для всех блоков единицы. Альтернативно, различные блоки могут иметь различные режимы прогнозирования.

Е. Ограничения для прогнозирования с внутренним копированием, когда WPP активируется

[0167] В общем, прогнозирование с внутренним BC, прогнозирование с внутренним LC и прогнозирование с внутренним SC используют ранее восстановленные значения выборок в изображении, чтобы прогнозировать значения выборок текущего блока, линии или строки одного и того же изображения. В качестве правила, область изображения, которое включает в себя ранее восстановленные значения выборок, когда WPP активируется, отличается от области изображения, которое включает в себя ранее восстановленные значения выборок, когда WPP не активируется. Для прогнозирования с внутренним копированием, некоторые ограничения на местоположение опорной области принудительно активированы независимо от того, активируется или нет WPP. Одно или более других ограничений на местоположение опорной области применяются, когда WPP активируется. Ограничения на местоположение опорной области могут принимать форму пределов для разрешенных BV-значений для прогнозирования с внутренним BC или пределов для разрешенных значений смещения для прогнозирования с внутренним LC или прогнозирования с внутренним SC.

[0168] Фиг. 18 показывает пример (1800) ограничений на местоположение опорной области для текущей области (1830) изображения (1810), когда WPP активируется, согласно некоторым примерным реализациям. Изображение (1810) организовано посредством STU-строк и STU-столбцов.

[0169] Текущая область (1830) кодируется или декодируется в режиме внутреннего копирования. Текущая область (1830) является частью текущей STU. Для текущей STU, зависимости каскадируются из предыдущих STU-строк. Когда WPP активируется, для начала обработки текущей STU, обрабатывается STU слева от нее в той же строке, и обрабатываются STU в предыдущей строке вплоть до правой верхней STU. Аналогично, для любых из этих уже обработанных STU, обрабатываются STU слева в той же строке и STU в предыдущей строке вплоть до правой верхней STU. Как показано на фиг. 18, эти ранее обработанные STU предоставляют восстановленный контент, гарантированно доступный для прогнозирования с внутренним копированием, когда WPP активируется.

[0170] Фиг. 18 показывает некоторые возможные варианты опорных областей для текущей области (1830) текущего изображения (1810). Четыре значения (1841, 1842, 1843, 1844) смещения указывают сдвиги для четырех возможных вариантов опорных областей. Возможный вариант опорной области может перекрываться с другими возможными вариантами опорных областей. Возможные варианты опорных областей могут находиться в любом месте в восстановленном контенте текущего изображения

(1810). Когда WPP не активируется, восстановленный контент текущего изображения (1810), в общем, включает в себя STU слева от текущей STU и все STU в предыдущих STU-строках. Тем не менее, когда WPP активируется, меньший объем восстановленного контента доступен, как показано на фиг. 18. На фиг. 18, три из значений (1841, 1842, 1843) смещения указывают допустимые опорные области. Эти допустимые опорные области включают в себя только значения выборок в восстановленном контенте, гарантированно доступные для прогнозирования с внутренним копированием, когда WPP активируется. Одно из значений (1844) смещения указывает недопустимую опорную область, которая включает в себя, по меньшей мере, некоторые значения выборок за пределами восстановленного контента, гарантированно доступные для прогнозирования с внутренним копированием, когда WPP активируется.

1. Кодирование и декодирование с ограничениями на режим внутреннего копирования, когда WPP активируется

[0171] Фиг. 19 показывает обобщенную технологию (1900) для кодирования с правилами, принудительно активированными для режима внутреннего копирования, когда WPP активируется. Кодер, к примеру, кодер, показанный на фиг. 3 или фиг. 5a и 5b, либо другой кодер может выполнять технологию (1900). Фиг. 20 показывает обобщенную технологию (2000) для декодирования с правилами, принудительно активированными для режима внутреннего копирования, когда WPP активируется. Декодер, к примеру, декодер, показанный на фиг. 4 или фиг. 6, либо другой декодер может выполнять технологию (2000).

[0172] Со ссылкой на фиг. 19, кодер кодирует (1910) изображение при активированной WPP. Кодирование (1910) формирует закодированные данные. В качестве части кодирования (1910), для режима внутреннего копирования (например, режима с использованием прогнозирования с внутренним ВС, прогнозирования с внутренним LC или прогнозирования с внутренним SC), кодер принудительно активирует одно или более ограничений, обусловленных WPP. Кодер выводит (1920) закодированные данные в качестве части потока битов.

[0173] Со ссылкой на фиг. 20, декодер принимает (2010) закодированные данные в качестве части потока битов. Закодированные данные удовлетворяют, для режима внутреннего копирования (например, режима с использованием прогнозирования с внутренним ВС, прогнозирования с внутренним LC или прогнозирования с внутренним SC), одному или более ограничений, обусловленных WPP. Декодер декодирует (2020) закодированные данные при активированной WPP. Декодирование (2020) восстанавливает изображение.

[0174] Синтаксический элемент в потоке битов может указывать то, активируется или нет WPP для видеопоследовательности, набора изображений или изображения. Синтаксический элемент может передаваться в служебных сигналах в синтаксической SPS-структуре, синтаксической PPS-структуре (например, синтаксический элемент флага активации синхронизации энтропийного кодирования в H.265/HEVC-реализациях) или другой синтаксической структуре в потоке битов.

[0175] Режим внутреннего копирования может представлять собой режим внутреннего ВС. В этом случае, для текущего блока в изображении, значение смещения указывает сдвиг для опорного блока в изображении. Опорный блок включает в себя ранее восстановленные значения выборок. Альтернативно, режим внутреннего копирования может представлять собой режим внутреннего LC. В этом случае, для текущей линии в блоке изображения, значение смещения указывает сдвиг для опорной линии в изображении. Опорная линия включает в себя ранее восстановленные значения выборок,

которые сканируются в направлении линейного сканирования, которое может быть горизонтальным или вертикальным. Альтернативно, режим внутреннего копирования может представлять собой режим внутреннего SC. В этом случае, для текущей строки в блоке изображения, значение смещения указывает сдвиг для опорной строки в изображении. Опорная строка включает в себя ранее восстановленные значения выборок, которые сканируются в порядке строкового сканирования, который может быть горизонтальным или вертикальным. Для текущей строки, значение длины указывает длину для каждой текущей строки и опорной строки. Альтернативно, режим внутреннего копирования может представлять собой некоторой другой режим, который использует значение смещения для того, чтобы находить опорный блок, линию, строку или другую секцию в изображении, для текущего блока, линии, строки или другой секции в том же изображении, при этом опорный блок, линия, строка и т.д. включает в себя ранее восстановленные значения выборок.

[0176] Ограничение(я), обусловленное WPP, может включать в себя такое ограничение, что значение горизонтального сдвига от опорной области до текущей области меньше или равно значению вертикального сдвига от текущей области до опорной области. Например, значение горизонтального сдвига измеряет разность от WPP-столбца, который включает в себя правый край опорной области, до WPP-столбца, который включает в себя левый край текущей области, и значение вертикального сдвига измеряет разность от WPP-строки, которая включает в себя верхний край текущей области, до WPP-строки, которая включает в себя нижний край опорной области. В некоторых примерных реализациях, каждый из WPP-столбцов представляет собой столбец STU, и каждая из WPP-строк представляет собой строку STU. Альтернативно, ограничение (я), обусловленное WPP, включает в себя одно или более других и/или дополнительных ограничений.

[0177] Если режим внутреннего копирования представляет собой режим внутреннего ВС, текущая область представляет собой текущий блок, а опорная область представляет собой опорный блок. Если режим внутреннего копирования представляет собой режим внутреннего LC, текущая область представляет собой текущую линию, а опорная область представляет собой опорную линию. Если режим внутреннего копирования представляет собой режим внутреннего SC, текущая область представляет собой прямоугольник, который включает в себя текущую строку, а опорная область представляет собой прямоугольник, который включает в себя опорную строку. Кодер или декодер может идентифицировать прямоугольник, который включает в себя текущую строку, посредством определения прямоугольника, который включает в себя начальную позицию текущей строки, конечную позицию текущей строки и любые позиции между начальной позицией и конечной позицией текущей строки в порядке строкового сканирования (например, горизонтальный, вертикальный). Кодер или декодер может идентифицировать прямоугольник, который включает в себя опорную строку, с использованием значения смещения, применяемого к прямоугольнику, который включает в себя текущую строку.

[0178] Кодер также может принудительно активировать одно или более других ограничений, которые не обусловлены WPP. Например, кодер может проверять то, что левая верхняя позиция текущей области и левая верхняя позиция опорной области находятся в одном и том же слайсе, если применимо, и в одном и том же мозаичном фрагменте, если применимо. В качестве другого примера, кодер может проверять то, что левая верхняя позиция текущей области и правая нижняя позиция опорной области находятся в одном и том же слайсе, если применимо, и в одном и том же мозаичном

фрагменте, если применимо. В качестве еще одного другого примера, кодер может проверять то, что удовлетворяется одно из следующих трех условий: (а) STU-строка, которая включает в себя нижний край опорной области, находится выше STU-строки, которая включает в себя верхний край текущей области; (b) если STU-строка, которая включает в себя нижний край опорной области, равна STU-строке, которая включает в себя верхний край текущей области, то STU-столбец, который включает в себя правый край опорной области, находится слева от STU-столбца, который включает в себя левый край текущей области; и (с) если STU-строка, которая включает в себя нижний край опорной области, равна STU-строке, которая включает в себя верхний край текущей области, и если STU-столбец, который включает в себя правый край опорной области, равен STU-столбцу, который включает в себя левый край текущей области, то правая нижняя позиция опорной области находится раньше в порядке Z-сканирования, чем левая верхняя позиция текущей области.

[0179] Следующие разделы детализируют примеры ограничений, которые кодер может принудительно активировать для прогнозирования с внутренним BC, прогнозирования с внутренним LC и прогнозирования с внутренним SC, соответственно, когда WPP активируется, согласно некоторым примерным реализациям.

2. Примерные ограничения на BV-значения для прогнозирования с внутренним BC, когда WPP активируется

[0180] Этот раздел детализирует примеры ограничений, которые кодер может принудительно активировать для прогнозирования с внутренним BC, когда WPP активируется. Для текущего блока, ограничения верифицируют то, что возможный вариант опорного блока, указываемый посредством BV-значения, включает в себя восстановленные значения выборок, которые должны быть доступны, когда текущий блок закодирован или декодирован, даже когда WPP активируется.

[0181] Определения

Текущий блок начинается с позиции (x_0, y_0) относительно левой верхней позиции текущего изображения. Ширина и высота текущего блока представляет собой w_{block} и h_{block} , соответственно. Текущий блок является частью текущей CU. STU-размер составляет S. Текущая CU начинается с (x_{CU}, y_{CU}) относительно левой верхней позиции изображения. Блочный вектор представляет собой $(BV_x,)$.

[0182] Кодер верифицирует то, что удовлетворяются все следующие ограничения.

[0183] Первое ограничение. Кодер верифицирует то, что позиция (x_0, y_0) и позиция $(x_0 + BV_x, y_0 + BV_y)$ находятся в одном и том же слайсе и в одном и том же мозаичном фрагменте. Иными словами, кодер верифицирует то, что левая верхняя позиция текущего блока и левая верхняя позиция опорного блока находятся в одном и том же слайсе и в одном и том же мозаичном фрагменте. Если две позиции находятся в различных слайсах или различных мозаичных фрагментах, первое ограничение не удовлетворяется.

[0184] Второе ограничение. Кодер верифицирует то, что позиция (x_0, y_0) и позиция $(x_0 + BV_x + w_{block} - 1, y_0 + BV_y + h_{block} - 1)$ находятся в одном и том же слайсе и одном и том же мозаичном фрагменте. Иными словами, кодер верифицирует то, что левая верхняя позиция текущего блока и правая нижняя позиция опорного блока находятся в одном и том же слайсе и в одном и том же мозаичном фрагменте. Если две позиции находятся в различных слайсах или различных мозаичных фрагментах, второе ограничение не удовлетворяется.

[0185] Для первого и второго ограничений, если несколько слайсов не используются, две позиции, которые проверяются, обязательно находятся в одном и том же слайсе, и

первое и второе ограничения для слайсов не должны проверяться. Аналогично, если несколько мозаичных фрагментов не используются, две позиции, которые проверяются, обязательно находятся в том же мозаичном фрагменте, и первое и второе ограничения для мозаичных фрагментов не должны проверяться. Все позиции текущего блока находятся в одном слайсе и одном мозаичном фрагменте. Если первое и второе ограничения удовлетворяются, все позиции опорного блока также находятся в этом слайсе и мозаичном фрагменте. Кодер проверяет первое и второе ограничения независимо от того, активируется или нет WPP.

[0186] Третье ограничение. Для третьего ограничения, кодер верифицирует то, что удовлетворяется одно из следующих трех условий. Кодер проверяет третье ограничение независимо от того, активируется или нет WPP.

[0187] Первое условие третьего ограничения. Кодер проверяет то, удовлетворяется или нет $(y_0 + BV_y + h_{block} - 1)/S < y_0/S$. Иными словами, кодер вычисляет STU-строку, которая включает в себя нижний край опорного блока: $(y_0 + BV_y + h_{block} - 1)/S$. Кодер также вычисляет STU-строку, которая включает в себя верхний край текущего блока: y_0/S . Кодер затем проверяет то, находится или нет STU-строка, которая включает в себя нижний край опорного блока, выше STU-строки, которая включает в себя верхний край текущего блока. Если да, опорный блок обязательно включает в себя ранее восстановленные значения выборок, по меньшей мере, когда WPP не активируется.

[0188] Второе условие третьего ограничения. Когда $(y_0 + BV_y + h_{block} - 1)/S = y_0/S$, кодер проверяет то, удовлетворяется или нет $(x_0 + BV_x + w_{block} - 1)/S < x_0/S$. Иными словами, если STU-строка, которая включает в себя нижний край опорного блока, равна STU-строке, которая включает в себя верхний край текущего блока (та же STU-строка), кодер вычисляет (а) STU-столбец, который включает в себя правый край опорного блока $((x_0 + BV_x + w_{block} - 1)/S)$, и (б) STU-столбец, который включает в себя левый край текущего блока (x_0/S) . Кодер затем проверяет то, находится или нет STU-столбец, который включает в себя правый край опорного блока, слева от STU-столбца, который включает в себя левый край текущего блока. Если да, опорный блок обязательно включает в себя ранее восстановленные значения выборок.

[0189] Третье условие третьего ограничения. Когда $(y_0 + BV_y + h_{block} - 1)/S = y_0/S$ и $(x_0 + BV_x + w_{block} - 1)/S = x_0/S$, кодер проверяет то, меньше или нет порядок Z-сканирования позиции $(x_0 + BV_x + w_{block} - 1, y_0 + uBV + h_{block} - 1)$ порядка Z-сканирования позиции (x_0, y_0) . Иными словами, если STU-строка, которая включает в себя нижний край опорного блока, равна STU-строке, которая включает в себя верхний край текущего блока (та же STU-строка), и STU-столбец, который включает в себя правый край опорного блока, равен STU-столбцу, который включает в себя левый край текущего блока (тот же STU-столбец), то кодер проверяет то, находится или нет правая нижняя позиция опорного блока раньше в порядке Z-сканирования, чем левая верхняя позиция текущего блока. Третье условие применяется, если прогнозирование из текущей CU разрешается. Если прогнозирование из текущей CU не разрешается, (x_0, y_0) должен быть (x_{CU}, y_{CU}) .

[0190] Четвертое ограничение. Кодер проверяет четвертое ограничение, когда WPP активируется. Для четвертого ограничения, кодер верифицирует то, что $(x_0 + BV_x + w_{block} - 1)/S - x_0/S \leq y_0/S - (y_0 + BV_y + h_{block} - 1)/S$. Иными словами, кодер вычисляет разность между STU-столбцом, который включает в себя правый край опорного блока, и STU-столбцом, который включает в себя левый край текущего блока: $(x_0 + BV_x + w_{block} - 1)/S - x_0/S$. Кодер также вычисляет разность между STU-строкой, которая

включает в себя верхний край текущего блока, и STU-строкой, которая включает в себя нижний край опорного блока: $y_0/S - (v_0 + BV_y + h_{block} - 1)/S$. Кодер верифицирует то, что первая разность (между STU-столбцами) меньше или равна второй разности (между STU-строками). Как показано с помощью неровной линии STU от текущей STU вверх и вправо на фиг. 8 или фиг. 18, это верифицирует то, что опорный блок является частью восстановленного контента, который гарантированно доступен для прогнозирования, когда WPP активируется.

3. *Примерные ограничения на значения смещения для прогнозирования с внутренним LC, когда WPP активируется*

[0191] Этот раздел детализирует примеры ограничений, которые кодер может принудительно активировать для прогнозирования с внутренним LC, когда WPP активируется. Для текущей линии, ограничения верифицируют то, что возможный вариант опорной линии, указываемый посредством значения смещения, включает в себя восстановленные значения выборок, которые должны быть доступны, когда текущая линия закодирована или декодирована, даже когда WPP активируется.

[0192] *Определения*

Текущий блок начинается с позиции (x_0, y_0) относительно левой верхней позиции текущего изображения. Ширина и высота текущего блока представляет собой w_{block} и h_{block} , соответственно. Текущий блок является частью текущей CU. STU-размер составляет S . Значение смещения для текущей линии составляет $(offset_x, offset_y)$. L линий текущего блока уже обработаны.

[0193] Кодер задает начальную и конечную позиции текущей линии текущего блока.

Позиция $(x_{curr_line_start}, y_{curr_line_start})$ представляет собой начальную позицию текущей линии, и $(x_{curr_line_end}, y_{curr_line_end})$ представляет собой конечную позицию текущей линии. Когда используется горизонтальное сканирование (также называемое "строковым режимом"), каждая линия представляет собой строку текущего блока:

$(x_{curr_line_start}, y_{curr_line_start}) = (x_0, y_0 + L)$ и $(x_{curr_line_end}, y_{curr_line_end}) = (x_0 + w_{block} - 1, y_0 + L)$ Когда используется вертикальное сканирование (также называемое "столбцовым режимом"), каждая линия представляет собой столбец текущего блока:

$(x_{curr_line_start}, y_{curr_line_start}) = (x_0 + L, y_0)$ и $(x_{curr_line_end}, y_{curr_line_end}) = (x_0 + L, y_0 + h_{block} - 1)$

[0194] Кодер также задает начальную и конечную позиции возможного варианта опорной линии, который прогнозирует текущую линию. Позиция $(x_{ref_line_start}, y_{ref_line_start})$ представляет собой начальную позицию опорной линии. Позиция $(x_{ref_line_end}, y_{ref_line_end})$ представляет собой конечную позицию опорной линии. Опорная линия может представлять собой строку (когда используется горизонтальное сканирование) или столбец (когда используется вертикальное сканирование). Независимо от того, используется горизонтальное сканирование (строковый режим) или вертикальное сканирование (столбцовый режим):

$(x_{ref_line_start}, y_{ref_line_start}) = (x_{curr_line_start}, y_{curr_line_start}) + (offset_x, offset_y)$ и $(x_{ref_line_end}, y_{ref_line_end}) = (x_{curr_line_end}, y_{curr_line_end}) + (offset_x, offset_y)$.

[0195] Кодер верифицирует то, что удовлетворяются все следующие ограничения.

Для некоторых из этих ограничений, рассматривается левая верхняя позиция (x_0, y_0) текущего блока. Для таких ограничений, альтернативно, начальная позиция

$(x_{curr_line_start}, y_{curr_line_start})$ текущей линии может проверяться вместо левой верхней позиции (x_0, y_0) текущего блока.

[0196] Первое ограничение. Кодер верифицирует то, что позиция (x_0, y_0) и позиция $(x_{ref_line_start}, y_{ref_line_start})$ находятся в одном и том же слайсе и в одном и том же мозаичном фрагменте. Иными словами, кодер верифицирует то, что левая верхняя позиция текущего блока и начальная позиция опорной линии находятся в одном и том же слайсе и в одном и том же мозаичном фрагменте. Если две позиции находятся в различных слайсах или различных мозаичных фрагментах, первое ограничение не удовлетворяется.

[0197] Второе ограничение. Кодер верифицирует то, что позиция (x_0, y_0) и позиция $(x_{ref_line_end}, y_{ref_line_end})$ находятся в одном и том же слайсе и одном и том же мозаичном фрагменте. Иными словами, кодер верифицирует то, что левая верхняя позиция текущего блока и конечная позиция опорной линии находятся в одном и том же слайсе и в одном и том же мозаичном фрагменте. Если две позиции находятся в различных слайсах или различных мозаичных фрагментах, второе ограничение не удовлетворяется.

[0198] Для первого и второго ограничений, если несколько слайсов не используются, две позиции, которые проверяются, обязательно находятся в одном и том же слайсе, и первое и второе ограничения для слайсов не должны проверяться. Аналогично, если несколько мозаичных фрагментов не используются, две позиции, которые проверяются, обязательно находятся в том же мозаичном фрагменте, и первое и второе ограничения для мозаичных фрагментов не должны проверяться. Все позиции текущей линии находятся в одном слайсе и одном мозаичном фрагменте. Если первое и второе ограничения удовлетворяются, все позиции опорной линии также находятся в этом слайсе и мозаичном фрагменте. Кодер проверяет первое и второе ограничения независимо от того, активируется или нет WPP.

[0199] Третье ограничение. Для третьего ограничения, кодер верифицирует то, что удовлетворяется одно из следующих трех условий. Кодер проверяет третье ограничение независимо от того, активируется или нет WPP.

[0200] Первое условие третьего ограничения. Кодер проверяет то, удовлетворяется или нет $(y_{ref_line_end}/S < y_0/S$. Иными словами, кодер вычисляет STU-строку, которая включает в себя нижний край или позицию опорной линии: $y_{ref_line_end}/S$. Кодер также вычисляет STU-строку, которая включает в себя верхний край текущего блока: y_0/S . Кодер затем проверяет то, находится или нет STU-строка, которая включает в себя нижний край или позицию опорной линии, выше STU-строки, которая включает в себя верхний край текущего блока. Если да, опорная линия обязательно включает в себя ранее восстановленные значения выборок, по меньшей мере, когда WPP не активируется.

[0201] Второе условие третьего ограничения. Когда $y_{ref_line_end}/S = y_0/S$, кодер проверяет то, удовлетворяется или нет $x_{ref_line_end}/S < x_0/S$. Иными словами, если STU-строка, которая включает в себя нижний край или позицию опорной линии, равна STU-строке, которая включает в себя верхний край текущего блока (та же STU-строка), кодер вычисляет (a) STU-столбец, который включает в себя правый край или позицию опорной линии $(x_{ref_line_end}/S)$, и (b) STU-столбец, который включает в себя левый край текущего блока (x_0/S) . Кодер затем проверяет то, находится или нет STU-столбец, который включает в себя правый край или позицию опорной линии, слева от STU-столбца, который включает в себя левый край текущего блока. Если да, опорная линия обязательно включает в себя ранее восстановленные значения выборок.

[0202] Третье условие третьего ограничения. Когда $y_{ref_line_end}/S = y_0/S$ and $x_{ref_line_end}/S = x_0/S$, кодер проверяет то, меньше или нет порядок Z-сканирования позиции $(x_{ref_line_end}, y_{ref_line_end})$

порядка Z-сканирования позиции $(x_{curr_line_start}, y_{curr_line_start})$. Иными словами, если STU-строка, которая включает в себя нижний край или позицию опорной линии, равна STU-строке, которая включает в себя верхний край текущего блока (та же STU-строка), и STU-столбец, который включает в себя правый край или позицию опорной линии, равен STU-столбцу, который включает в себя левый край текущего блока (тот же STU-столбец), то кодер проверяет то, находится или нет конечная позиция опорной линии раньше в порядке Z-сканирования, чем начальная позиция текущей линии. Третье условие применяется, если прогнозирование из текущей CU разрешается. Если прогнозирование из текущей CU не разрешается, $(x_{curr_line_start}, y_{curr_line_start})$ должен быть (x_0, y_0) .

[0203] Четвертое ограничение. Кодер проверяет четвертое ограничение, когда WPP активируется. Для четвертого ограничения, кодер верифицирует то, что $x_{ref_line_end}/S - x_0/S \leq y_0/S - y_{ref_line_end}/S$. Иными словами, кодер вычисляет разность между STU-столбцом, который включает в себя правый край или позицию опорной линии, и STU-столбцом, который включает в себя левый край текущего блока: $x_{ref_line_end}/S - x_0/S$. Кодер также вычисляет разность между STU-строкой, которая включает в себя верхний край текущего блока, и STU-строкой, которая включает в себя нижний край или позицию опорной линии: $y_0/S - y_{ref_line_end}/S$. Кодер верифицирует то, что первая разность (между STU-столбцами) меньше или равна второй разности (между STU-строками).

4. Примерные ограничения на значения смещения для прогнозирования с внутренним SC, когда WPP активируется

[0204] Этот раздел детализирует примеры ограничений, которые кодер может принудительно активировать для прогнозирования с внутренним SC, когда WPP активируется. Для текущей строки, ограничения верифицируют то, что возможный вариант опорной строки, указываемый посредством значения смещения и значения длины, включает в себя восстановленные значения выборок, которые должны быть доступны, когда текущая строка закодирована или декодирована, даже когда WPP активируется.

[0205] Определения

Текущий блок начинается с позиции (x_0, y_0) относительно левой верхней позиции текущего изображения. Ширина и высота текущего блока представляет собой w_{block} и h_{block} , соответственно. Текущий блок является частью текущей CU. STU-размер составляет S. Значение смещения для текущей строки представляет собой $(offset_x, offset_y)$, и значение длины строки для текущей строки представляет собой $length_{string}$. К пикселей текущего блока уже обработаны с использованием прогнозирования с внутренним SC.

[0206] Кодер задает начальную и конечную позиции текущей строки текущего блока. Позиция $(x_{curr_string_start}, y_{curr_string_start})$ представляет собой начальную позицию текущей строки, и $(x_{curr_string_end}, y_{curr_string_end})$ представляет собой конечную позицию текущей строки. Кодер также задает ограничительный прямоугольник, который содержит начальную позицию текущей строки, конечную позицию текущей строки и любые позиции (в порядке строкового сканирования) между начальной и конечной позициями текущей строки. Левая верхняя позиция ограничительного прямоугольника представляет собой $(x_{curr_rect_TL}, y_{curr_rect_TL})$, правая нижняя позиция ограничительного прямоугольника представляет собой $(x_{curr_rect_BR}, y_{curr_rect_BR})$. Кодер задает опорный прямоугольник (включающий в себя опорную строку) в качестве ограничительного прямоугольника,

сдвинутого на значение смещения. Левая верхняя позиция опорного прямоугольника представляет собой $(x_{ref_rect_TL}, y_{ref_rect_TL})$. Правая нижняя позиция ограничительного прямоугольника представляет собой $(x_{ref_rect_BR}, y_{ref_rect_BR})$.

5 [0207] Когда используется горизонтальное строковое сканирование (слева направо в строке текущего блока, от верхней строки до нижней строки), начальная позиция является следующей: $(x_{curr_string_start}, y_{curr_string_start}) = (x_0 + K \% w_{block}, y_0 + K / w_{block})$. Конечная позиция является следующей: $(x_{curr_string_end}, y_{curr_string_end}) = (x_0 + (K + length_{string} - 1) \% w_{block}, y_0 + (K + length_{string} - 1) / w_{block})$. Когда начальная позиция и конечная позиция текущей строки находятся в той же строке текущего блока, $(x_{curr_rect_TL}, y_{curr_rect_TL}) = (x_{curr_string_start}, y_{curr_string_start})$, и $(x_{curr_rect_BR}, y_{curr_rect_BR}) = (x_{curr_string_end}, y_{curr_string_end})$. В противном случае, левая верхняя позиция ограничительного прямоугольника представляет собой $(x_{curr_rect_TL}, y_{curr_rect_TL}) = (x_0, y_0 + K / w_{block})$, и правая нижняя позиция ограничительного

10

15

прямоугольника представляет собой $(x_{curr_rect_BR}, y_{curr_rect_BR}) = (x_0 + w_{block} - 1, y_0 + (K + length_{string} - 1) / w_{block})$.

[0208] Когда используется вертикальное сканирование (сверху вниз в столбце текущего блока, от левого столбца к правому столбцу), начальная позиция является следующей: $(x_{curr_string_start}, y_{curr_string_start}) = (x_0 + K / h_{block}, y_0 + K \% h_{block})$. Конечная позиция является следующей: $(x_{curr_string_end}, y_{curr_string_end}) = (x_0 + (K + length_{string} - 1) / h_{block}, y_0 + (K + length_{string} - 1) \% h_{block})$. Когда начальная позиция и конечная позиция текущей строки находятся в том же столбце текущего блока, $(x_{curr_rect_TL}, y_{curr_rect_TL}) = (x_{curr_string_start}, y_{curr_string_start})$, и $(x_{curr_rect_BR}, y_{curr_rect_BR}) = (x_{curr_string_end}, y_{curr_string_end})$. В противном случае, левая

20

25

верхняя позиция ограничительного прямоугольника представляет собой $(x_{curr_rect_TL}, y_{curr_rect_TL}) = (x_0 + K / h_{block}, y_0)$, и правая нижняя позиция ограничительного прямоугольника представляет собой $(x_{curr_rect_BR}, y_{curr_rect_BR}) = (x_0 + (K + length_{string} - 1) / h_{block}, y_0 + h_{block} - 1)$.

30 [0209] Независимо от того, используется горизонтальное сканирование или вертикальное сканирование, левая верхняя позиция опорного прямоугольника представляет собой $(x_{ref_rect_TL}, y_{ref_rect_TL}) = (x_{curr_rect_TL}, y_{curr_rect_TL}) + (offset_x, offset_y)$, и правая нижняя позиция опорного прямоугольника представляет собой $(x_{ref_rect_BR}, y_{ref_rect_BR}) = (x_{curr_rect_BR}, y_{curr_rect_BR}) + (offset_x, offset_y)$. В завершение, начальная позиция опорной строки представляет собой $(x_{ref_string_start}, y_{ref_string_start}) = (x_{curr_string_start}, y_{curr_string_start}) + (offset_x, offset_y)$.

35

[0210] Кодер верифицирует то, что удовлетворяются все следующие ограничения. Для некоторых из этих ограничений, рассматривается левая верхняя позиция (x_0, y_0) текущего блока. Для таких ограничений, альтернативно, начальная позиция текущей строки или левая верхняя позиция ограничительного прямоугольника могут проверяться вместо левой верхней позиции (x_0, y_0) текущего блока.

40

[0211] Первое ограничение. Кодер верифицирует то, что позиция (x_0, y_0) и позиция $(x_{ref_rect_TL}, y_{ref_rect_TL})$ находятся в одном и том же слайсе и в одном и том же мозаичном фрагменте. Иными словами, кодер верифицирует то, что левая верхняя позиция текущего блока и левая верхняя позиция опорного прямоугольника находятся в одном и том же слайсе и в одном и том же мозаичном фрагменте. Если две позиции находятся в различных слайсах или различных мозаичных фрагментах, первое ограничение не

45

удовлетворяется.

[0212] Второе ограничение. Кодер верифицирует то, что позиция (x_0, y_0) и позиция $(x_{ref_rect_BR}, y_{ref_rect_BR})$ находятся в одном и том же слайсе и одном и том же мозаичном фрагменте. Иными словами, кодер верифицирует то, что левая верхняя позиция текущего блока и правая нижняя позиция опорного прямоугольника находятся в одном и том же слайсе и в одном и том же мозаичном фрагменте. Если две позиции находятся в различных слайсах или различных мозаичных фрагментах, второе ограничение не удовлетворяется.

[0213] Для первого и второго ограничений, если несколько слайсов не используются, две позиции, которые проверяются, обязательно находятся в одном и том же слайсе, и первое и второе ограничения для слайсов не должны проверяться. Аналогично, если несколько мозаичных фрагментов не используются, две позиции, которые проверяются, обязательно находятся в том же мозаичном фрагменте, и первое и второе ограничения для мозаичных фрагментов не должны проверяться. Все позиции текущей строки находятся в одном слайсе и одном мозаичном фрагменте. Если первое и второе ограничения удовлетворяются, все позиции опорного прямоугольника (и, следовательно, опорная строка) также находятся в этом слайсе и мозаичном фрагменте. Кодер проверяет первое и второе ограничения независимо от того, активируется или нет WPP.

[0214] Третье ограничение. Для третьего ограничения, кодер верифицирует то, что одно или более следующих условий удовлетворяются. Кодер проверяет третье ограничение независимо от того, активируется или нет WPP.

[0215] Первое условие третьего ограничения. Кодер проверяет то, удовлетворяется или нет $y_{ref_rect_BR}/S < y_0/S$. Иными словами, кодер вычисляет STU-строку, которая включает в себя нижний край опорного прямоугольника: $y_{ref_rect_BR}/S$. Кодер также вычисляет STU-строку, которая включает в себя верхний край текущего блока: y_0/S . Кодер затем проверяет то, находится или нет STU-строка, которая включает в себя нижний край опорного прямоугольника, выше STU-строки, которая включает в себя верхний край текущего блока. Если да, опорный прямоугольник обязательно включает в себя ранее восстановленные значения выборок, по меньшей мере, когда WPP не активируется.

[0216] Второе условие третьего ограничения. Когда $y_{ref_rect_BR}/S == y_0/S$, кодер проверяет то, удовлетворяется или нет $x_{ref_rect_BR}/S < x_0/S$. Иными словами, если STU-строка, которая включает в себя нижний край опорного прямоугольника, равна STU-строке, которая включает в себя верхний край текущего блока (та же STU-строка), кодер вычисляет (а) STU-столбец, который включает в себя правый край опорного прямоугольника $(x_{ref_rect_BR}/S)$, и (б) STU-столбец, который включает в себя левый край текущего блока (x_0/S) . Кодер затем проверяет то, находится или нет STU-столбец, который включает в себя правый край опорного прямоугольника, слева от STU-столбца, который включает в себя левый край текущего блока. Если да, опорный прямоугольник обязательно включает в себя ранее восстановленные значения выборок.

[0217] Третье условие третьего ограничения. Когда $y_{ref_rect_BR}/S == y_0/S$ и $x_{ref_rect_BR}/S == x_0/S$, кодер проверяет то, меньше или нет порядок Z-сканирования позиции $(x_{ref_rect_BR}, y_{ref_rect_BR})$ порядка Z-сканирования позиции (x_0, y_0) . Иными словами, если STU-строка, которая включает в себя нижний край опорного прямоугольника, равна STU-строке, которая включает в себя верхний край текущего блока (та же STU-строка), и STU-столбец, который включает в себя правый край опорного прямоугольника, равен

СТУ-столбцу, который включает в себя левый край текущего блока (тот же СТУ-столбец), то кодер проверяет то, находится или нет правая нижняя позиция опорного прямоугольника раньше в порядке Z-сканирования, чем левая верхняя позиция текущего блока.

5 [0218] Четвертое условие третьего ограничения. Если прогнозирование из текущей СУ разрешается, когда $y_{ref_string_start} == y_0$, кодер проверяет то, что $x_{ref_string_start} < x_0$. Иными словами, если прогнозирование из текущей СУ разрешается, третье ограничение может удовлетворяться, если текущая строка и опорная строка начинаются в одной и той же строке, и опорная строка начинается слева от текущей строки.

10 [0219] Четвертое ограничение. Кодер проверяет четвертое ограничение, когда WPP активируется. Для четвертого ограничения, кодер верифицирует то, что $x_{ref_rect_BR}/S - x_0/S \leq y_0/S - y_{ref_rect_BR}/S$. Иными словами, кодер вычисляет разность между СТУ-столбцом, который включает в себя правый край или позицию опорного
15 прямоугольника, и СТУ-столбцом, который включает в себя левый край текущего блока: $x_{ref_rect_BR}/S - x_0/S$. Кодер также вычисляет разность между СТУ-строкой, которая включает в себя верхний край текущего блока, и СТУ-строкой, которая включает в себя нижний край или позицию опорного прямоугольника: $y_0/S - y_{ref_rect_BR}/S$. Кодер верифицирует то, что первая разность (между СТУ-столбцами) меньше или равна второй
20 разности (между СТУ-строками).

[0220] Пятое ограничение. Для пятого ограничения, кодер верифицирует то, что $K + length_{string} \leq w_{block} * h_{block}$. Иными словами, кодер проверяет то, что текущий блок включает в себя достаточно позиций для текущей строки, с учетом количества позиций, которые уже обработаны.

25 F. Альтернативы и варьирования

[0221] Во многих примерах, описанных в данном документе, прогнозирование с внутренним копированием и компенсация движения реализуются в отдельных компонентах или процессах, и оценка смещения и оценка движения реализуются в отдельных компонентах или процессах. Альтернативно, прогнозирование с внутренним
30 копированием может реализовываться как частный случай компенсации движения, и оценка смещения может реализовываться как частный случай оценки движения, для которого текущее изображение используется в качестве опорного изображения. В таких реализациях, значение смещения может передаваться в служебных сигналах в качестве MV-значения, но использоваться для прогнозирования с внутренним копированием (в
35 текущем изображении), а не для межкадрового прогнозирования. При использовании в данном документе, термин "прогнозирование с внутренним копированием" указывает прогнозирование в текущем изображении независимо от того, предоставляется это прогнозирование с использованием модуля внутрикадрового прогнозирования, модуля компенсации движения или некоторого другого модуля. Аналогично, BV-значение или
40 другое значение смещения может быть представлено с использованием MV-значения или с использованием различного типа параметра или синтаксического элемента, и оценка смещения может предоставляться с использованием модуля внутрикадровой оценки, модуля оценки движения или некоторого другого модуля.

[0222] В свете множества возможных вариантов осуществления, к которым могут
45 быть применены принципы изобретения, следует признавать, что проиллюстрированные варианты осуществления являются предпочтительными примерами изобретения и не должны восприниматься как ограничивающие объем изобретения. Вместо этого, объем изобретения задается посредством прилагаемой формулы изобретения. Следовательно,

в качестве изобретения заявлено все, что находится в пределах объема и сущности этой формулы.

(57) Формула изобретения

- 5 1. Способ кодирования изображения, используемый в вычислительной системе, содержащий этапы, на которых:
- кодируют изображение при активированной волновой параллельной обработке (WPP), при этом кодирование формирует закодированные данные, что включает в себя, для прогнозирования с внутренним блочным копированием, этап, на котором
 - 10 принудительно активируют одно или более ограничений, совместимых с WPP, причем одно или более ограничений включает в себя ограничение, что значение горизонтального сдвига от опорной области до текущей области меньше или равно значению вертикального сдвига от текущей области до опорной области; и
 - выводят закодированные данные в качестве части потока битов.
- 15 2. Способ по п. 1, в котором:
- значение горизонтального сдвига измеряет разность от столбца единицы дерева кодирования, STU, который включает в себя правый край опорной области, до STU-столбца, который включает в себя левый край текущей области; и
 - значение вертикального сдвига измеряет разность от STU-строки, которая включает
 - 20 в себя верхний край текущей области, до STU-строки, которая включает в себя нижний край опорной области.
3. Способ по п. 1, в котором текущая область является текущим блоком в единице дерева кодирования, STU, и при этом кодирование включает в себя:
- вычисление значения горизонтального сдвига как $(x_0 + BV_x + w_{\text{block}} - 1) / S - x_0 / S$, где x_0 -
 - 25 горизонтальная позиция текущего блока, STU, BV_x - горизонтальный компонент вектора для прогнозирования с внутренним блочным копированием, w_{block} - ширина текущего блока и S указывает размер STU; и
 - вычисление значения вертикального сдвига как $y_0 / S - (y_0 + BV_y + h_{\text{block}} - 1) / S$, где y_0 -
 - 30 вертикальная позиция текущего блока, BV_y - вертикальный компонент вектора для прогнозирования с внутренним блочным копированием и h_{block} - высота текущего блока.
4. Способ по п. 1, в котором, для прогнозирования с внутренним блочным копированием, другие ограничения включают в себя:
- 35 (1) левая верхняя позиция текущей области и левая верхняя позиция опорной области находятся в одном и том же слайсе, если применимо, и в одном и том же мозаичном фрагменте, если применимо;
- (2) левая верхняя позиция текущей области и правая нижняя позиция опорной области находятся в одном и том же слайсе, если применимо, и в одном и том же мозаичном
- 40 фрагменте, если применимо; и
- (3) одно из следующих трех условий удовлетворяется:
- (а) строка единицы дерева кодирования (STU), которая включает в себя нижний край опорной области, находится выше STU-строки, которая включает в себя верхний край текущей области;
 - 45 (б) если STU-строка, которая включает в себя нижний край опорной области, равна STU-строке, которая включает в себя верхний край текущей области, то STU-столбец, который включает в себя правый край опорной области, находится слева от STU-столбца, который включает в себя левый край текущей области; и

(с) если STU-строка, которая включает в себя нижний край опорной области, равна STU-строке, которая включает в себя верхний край текущей области, и если STU-столбец, который включает в себя правый край опорной области, равен STU-столбцу, который включает в себя левый край текущей области, то правая нижняя позиция опорной области находится раньше в порядке Z-сканирования, чем левая верхняя позиция текущей области.

5. Способ по пп. 1-4, в котором текущая область является текущим блоком и опорная область является опорным блоком.

6. Способ по п. 1, в котором для текущего блока в изображении значение смещения указывает сдвиг для опорного блока в изображении, причем опорный блок включает в себя ранее восстановленные значения выборов.

7. Способ декодирования изображения, используемый в вычислительной системе, содержащий этапы, на которых:

- принимают закодированные данные в качестве части потока битов, при этом закодированные данные удовлетворяют, для прогнозирования с внутренним блочным копированием, одному или нескольким ограничениям, совместимым с волновой параллельной обработкой (WPP), причем одно или более ограничений включает в себя ограничение, что значение горизонтального сдвига от опорной области до текущей области меньше или равно значению вертикального сдвига от текущей области до опорной области; и

- декодируют закодированные данные при активированной WPP, при этом декодирование восстанавливает изображение.

8. Способ по п. 7, в котором:

- значение горизонтального сдвига измеряет разность от столбца единицы дерева кодирования, STU, который включает в себя правый край опорной области, до STU-столбца, который включает в себя левый край текущей области; и

- значение вертикального сдвига измеряет разность от STU-строки, которая включает в себя верхний край текущей области, до STU-строки, которая включает в себя нижний край опорной области.

9. Способ по п. 7, в котором текущая область является текущим блоком в единице дерева кодирования, STU, и при этом кодирование включает в себя:

вычисление значения горизонтального сдвига как $(x_0 + BV_x + w_{\text{block}} - 1) / S - x_0 / S$, где x_0 - горизонтальная позиция текущего блока, STU, BV_x - горизонтальный компонент вектора для прогнозирования с внутренним блочным копированием, w_{block} - ширина текущего блока и S указывает размер STU; и

вычисление значения вертикального сдвига как $y_0 / S - (y_0 + BV_y + h_{\text{block}} - 1) / S$, где y_0 - вертикальная позиция текущего блока, BV_y - вертикальный компонент вектора для прогнозирования с внутренним блочным копированием и h_{block} - высота текущего блока.

10. Способ по п. 7, в котором, для прогнозирования с внутренним блочным копированием, другие ограничения включают в себя:

(1) левая верхняя позиция текущей области и левая верхняя позиция опорной области находятся в одном и том же слайсе, если применимо, и в одном и том же мозаичном фрагменте, если применимо;

(2) левая верхняя позиция текущей области и правая нижняя позиция опорной области находятся в одном и том же слайсе, если применимо, и в одном и том же мозаичном фрагменте, если применимо; и

(3) одно из следующих трех условий удовлетворяется:

(а) строка единицы дерева кодирования (СТУ), которая включает в себя нижний край опорной области, находится выше СТУ-строки, которая включает в себя верхний край текущей области;

5 (b) если СТУ-строка, которая включает в себя нижний край опорной области, равна СТУ-строке, которая включает в себя верхний край текущей области, то СТУ-столбец, который включает в себя правый край опорной области, находится слева от СТУ-столбца, который включает в себя левый край текущей области; и

10 (с) если СТУ-строка, которая включает в себя нижний край опорной области, равна СТУ-строке, которая включает в себя верхний край текущей области, и если СТУ-столбец, который включает в себя правый край опорной области, равен СТУ-столбцу, который включает в себя левый край текущей области, то правая нижняя позиция опорной области находится раньше в порядке Z-сканирования, чем левая верхняя позиция текущей области.

15 11. Способ по п. 7, в котором:

текущая область является текущим блоком, а опорная область является опорным блоком.

12. Способ по п. 11, в котором для текущего блока в изображении значение смещения указывает сдвиг для опорного блока в изображении, причем опорный блок включает
20 в себя ранее восстановленные значения выборки.

13. Один или более машиночитаемых носителей, хранящих машиноисполняемые инструкции для вынуждения вычислительной системы, запрограммированной ими, осуществлять способ по п. 1.

14. Один или более машиночитаемых носителей, хранящих машиноисполняемые
25 инструкции для вынуждения вычислительной системы, запрограммированной ими, осуществлять способ по п. 7.

15. Вычислительная система, сконфигурированная с возможностью осуществлять способ по любому из пп. 1-6.

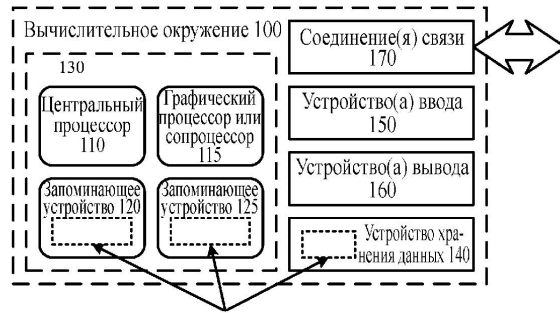
16. Вычислительная система, сконфигурированная с возможностью осуществлять
30 способ по любому из пп. 7-12.

35

40

45

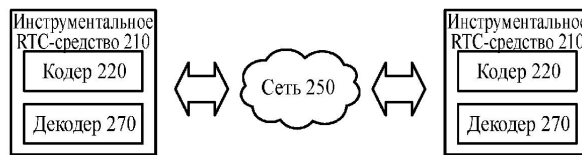
ФИГ. 1



Программное обеспечение 180, реализующее одну или более инноваций для правил для режимов внутрикадрового прогнозирования, когда волновая параллельная обработка (WPP) активируется

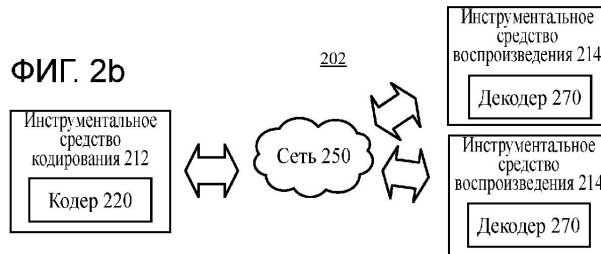
ФИГ. 2а

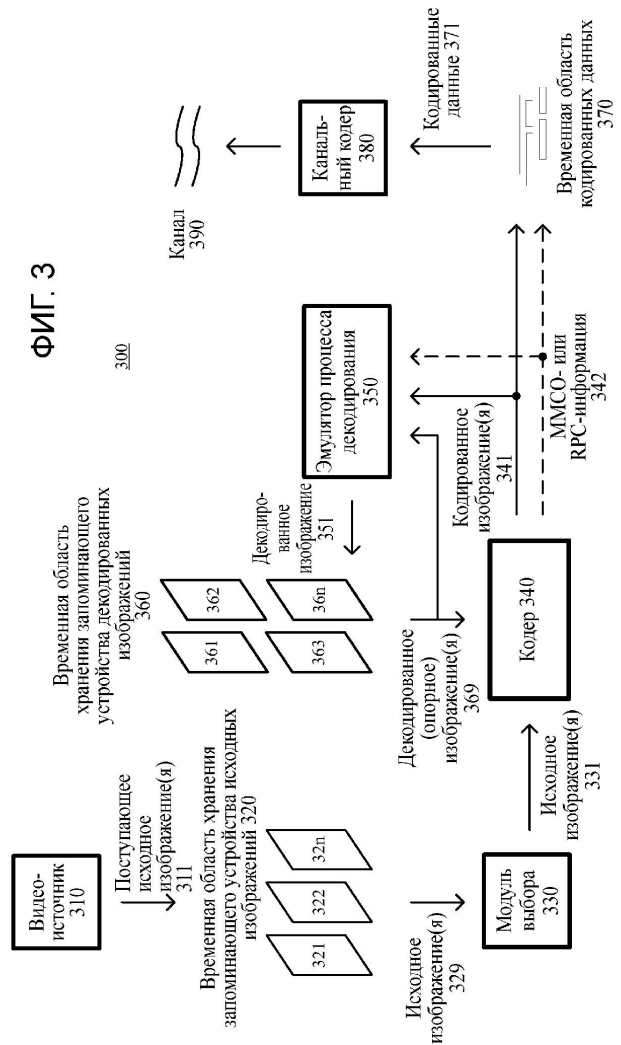
201



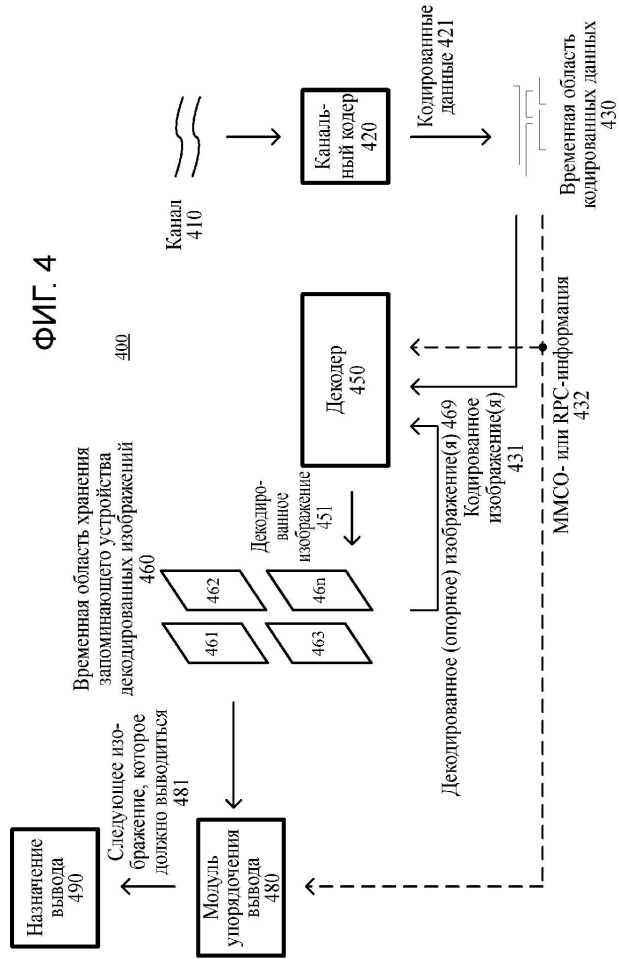
ФИГ. 2б

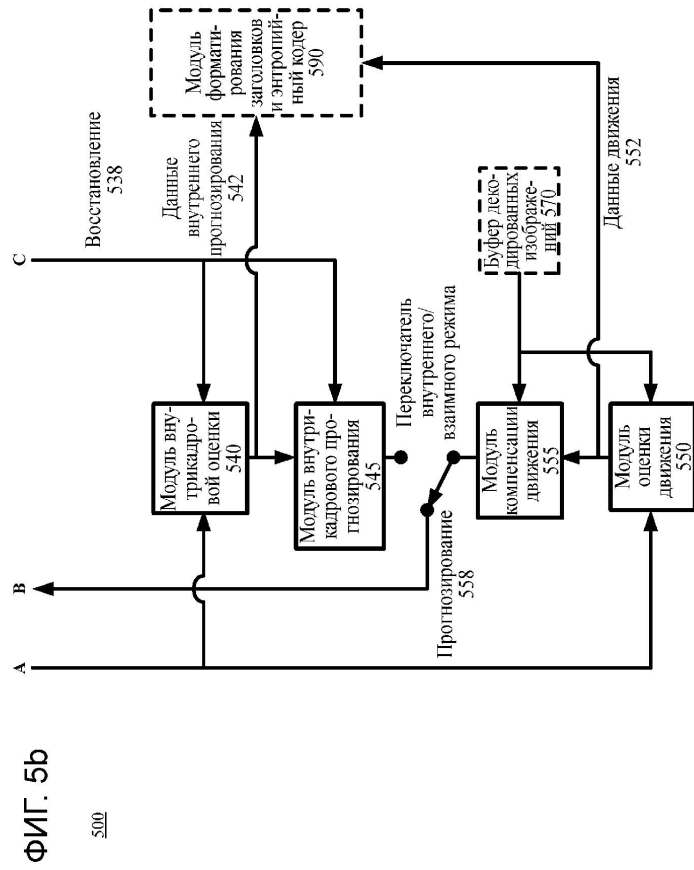
202



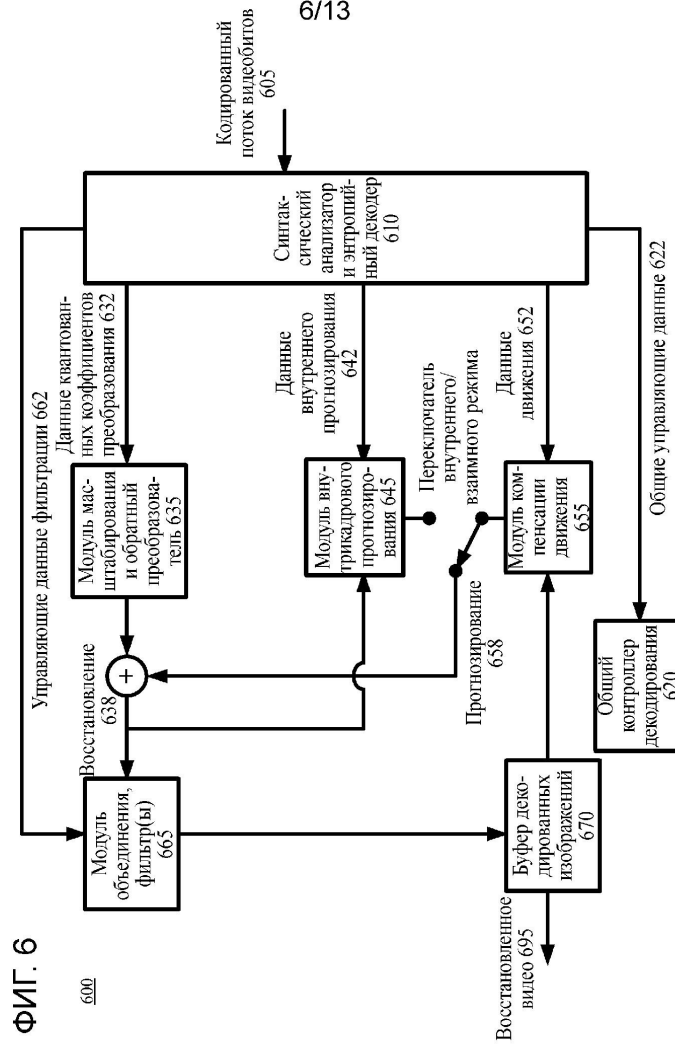


ФИГ. 4





6/13

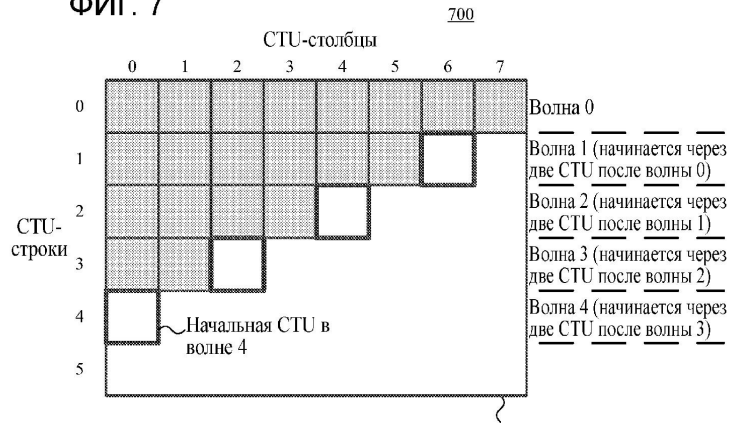


ФИГ. 6

600

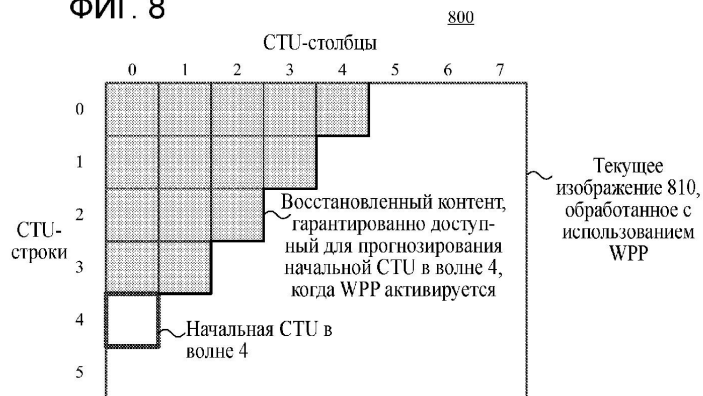
7/13

ФИГ. 7



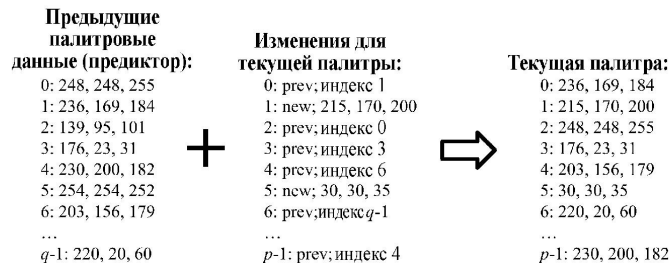
Текущее изображение 710, обработанное с использованием волновой параллельной обработки (WPP)

ФИГ. 8



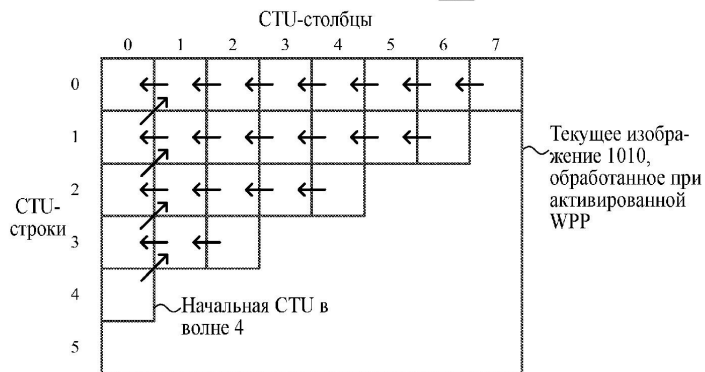
ФИГ. 9

900



ФИГ. 10

1000

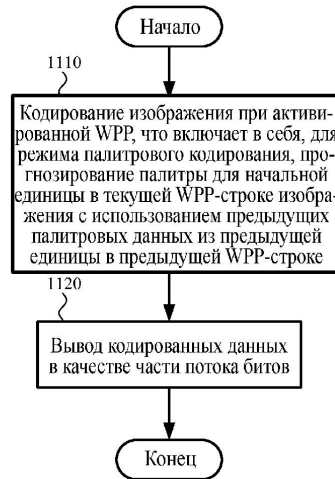


Направление палитрового прогнозирования, когда WPP активизируется:

- Для начальной STU в первой STU-строке - отсутствует
- Для начальной STU в последующих STU-строках - вправо вверх, согласно стрелкам
- Иначе - влево, согласно стрелкам

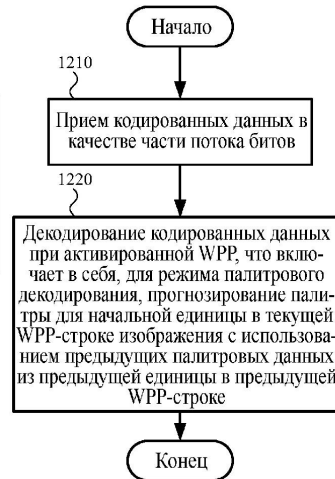
ФИГ. 11

1100



ФИГ. 12

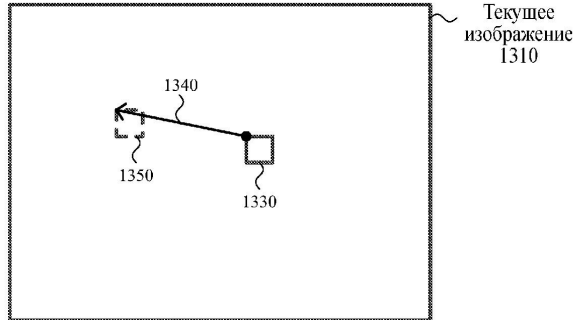
1200



10/13

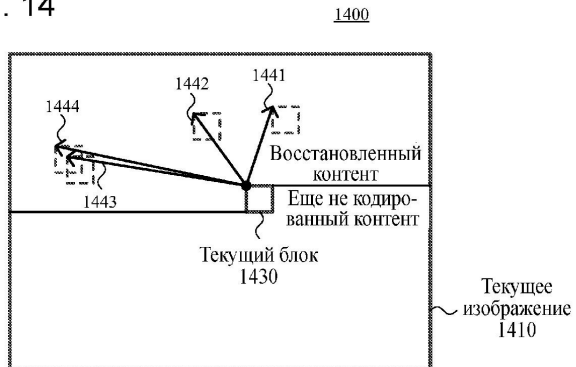
ФИГ. 13

Режим внутреннего блочного копирования



Блочный вектор (1340) для текущего блока (1330) текущего изображения (1310), указывающего сдвиг до опорного блока (1350) в текущем изображении (1310)

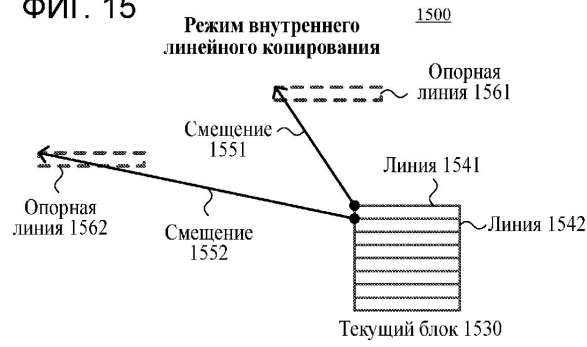
ФИГ. 14



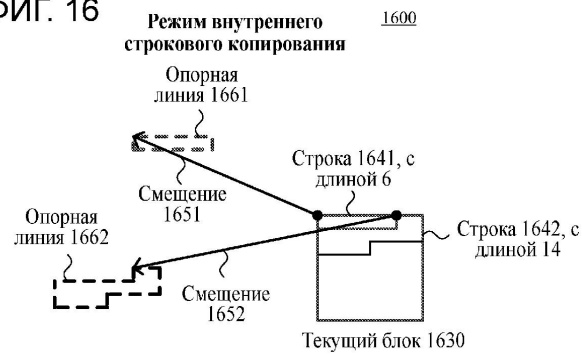
Блочный вектор (1441, 1442, 1443, 1444), указывающие сдвиги до возможных вариантов опорных блоков для текущего блока (1430) текущего изображения (1410)

11/13

ФИГ. 15



ФИГ. 16



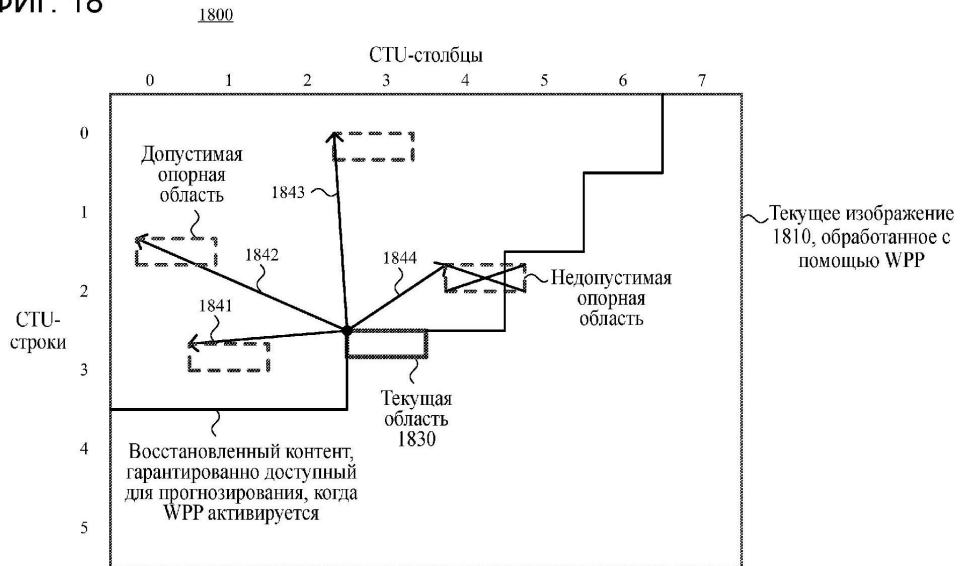
12/13

ФИГ. 17

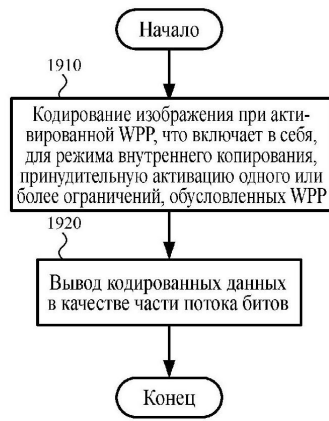
1700
Порядок Z-сканирования для текущей единицы и единиц, которые могут включать в себя правую нижнюю позицию опорной области



ФИГ. 18



ФИГ. 19 1900



ФИГ. 20 2000

