

(19)



**Евразийское
патентное
ведомство**

(11) **035170**(13) **B1**

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ЕВРАЗИЙСКОМУ ПАТЕНТУ

(45) Дата публикации и выдачи патента
2020.05.08

(21) Номер заявки
201791616

(22) Дата подачи заявки
2016.01.25

(51) Int. Cl. **H04N 19/176** (2014.01)
H04N 19/196 (2014.01)
H04N 19/593 (2014.01)

(54) СИСТЕМА И СПОСОБ КОДИРОВАНИЯ И ДЕКОДИРОВАНИЯ ВИДЕОДАНЫХ В РЕЖИМЕ ПАЛИТРЫ

(31) **62/109,568; 15/004,508**

(32) **2015.01.29; 2016.01.22**

(33) **US**

(43) **2017.11.30**

(86) **PCT/US2016/014760**

(87) **WO 2016/123033 2016.08.04**

(71)(73) Заявитель и патентовладелец:
КВЭЛКОММ ИНКОРПОРЕЙТЕД
(US)

(72) Изобретатель:
Пу Вэй, Карчевич Марта, Джоши
Раджан Лаксман, Цзоу Фэн, Серегин
Вадим (US)

(74) Представитель:
Медведев В.Н. (RU)

(56) **US-A1-2015016501**

"HEVC Screen Content Coding Test Model 2 (SCM 2)", 109. MPEG MEETING; 7-7-2014 - 11-7-2014; SAPPORO; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11), no. N14706, 20 October 2014 (2014-10-20), XP030021442, the whole document

PU W. ET AL.: "Non CE1: Palette Mode Semantics, Codeword, and Encoder Fixes", 20. JCT-VC MEETING; 10-2-2105 - 18-2-2015; GENEVA; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG. 16); URL: [HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/ no.JCTVC-T0063](http://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/no.JCTVC-T0063), 31 January 2015 (2015-01-31), XP030117186, the whole document

PU W. ET AL.: "Non CE1: Palette Mode Semantics, Codeword, and Encoder Fixes", 20. JCT-VC MEETING; 10-2-2105 - 18-2-2015; GENEVA; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG. 16); URL: [HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/ no.JCTVC-T0063-V7](http://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/no.JCTVC-T0063-V7), 12 February 2015 (2015-02-12), XP030117187, the whole document

(57) Осуществляют прием множества элементов синтаксиса, которые указывают палитру, которая использовалась для кодирования блока видеоданных, причем множество элементов синтаксиса включает в себя первый элемент синтаксиса, который указывает количество значений палитры для палитры, которые явно сигнализируются в битовом потоке кодированного видеосигнала, причем первый элемент синтаксиса кодируется с использованием одного или более кодов Голомба таким образом, что длина кодированного первого элемента синтаксиса меньше или равна заранее определенному максимальному количеству битов, декодирование множества элементов синтаксиса, включающее в себя декодирование первого элемента синтаксиса с использованием одного или более кодов Голомба, реконструкцию палитры на основании декодированного множества элементов синтаксиса и декодирование блока видеоданных с использованием реконструированной палитры.

B1**035170****035170 B1**

По данной заявке испрашивается приоритет предварительной патентной заявки США № 62/109568, поданной 29 января 2015 г., содержание которой в полном объеме включено сюда посредством ссылки.

Область техники, к которой относится изобретение

Это изобретение относится к кодированию и декодированию видеосигнала.

Уровень техники

Возможности цифрового видео можно включать в самые разнообразные устройства, в том числе цифровые телевизоры, цифровые системы прямого вещания, системы беспроводного вещания, карманные персональные компьютеры (PDA), портативные или настольные компьютеры, планшетные компьютеры, устройства чтения электронных книг, цифровые камеры, цифровые записывающие устройства, цифровые медиапроигрыватели, устройства для видеоигр, игровые консоли, сотовые или спутниковые радиотелефоны, так называемые "смартфоны", устройства телеконференцсвязи, устройства потоковой передачи видео и пр. Устройства цифрового видео реализуют методы сжатия видеосигнала, например, описанные в стандартах, установленных MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, часть 10, улучшенное кодирование видеоданных (AVC), разрабатываемом в настоящее время стандарте высокопроизводительного кодирования видеоданных (HEVC), и расширениях таких стандартов. Видеоустройства могут передавать, принимать, кодировать, декодировать и/или сохранять информацию цифрового видео более эффективно за счет реализации таких методов сжатия видеосигнала.

Методы сжатия видеосигнала осуществляют пространственное (внутри изображения) предсказание и/или временное (между изображениями) предсказание для уменьшения или удаления избыточности, присущей видеопоследовательностям. Для блочного кодирования видеоданных слайс видео (например, кадр видео или участок кадра видео) может делиться на видеоблоки. Видеоблоки во внутренне-кодированном (I) слайсе изображения кодируются с использованием пространственного предсказания относительно опорных выборок в соседних блоках одного и того же изображения. Видеоблоки во внешне-кодированном (P или B) слайсе изображения могут использовать пространственное предсказание относительно опорных выборок в соседних блоках одного и того же изображения или временное предсказание относительно опорных выборок в других опорных изображениях. Изображения могут именоваться кадрами, и опорные изображения могут именоваться опорными кадрами.

Пространственное или временное предсказание дает предсказательный блок для блока, подлежащего кодированию. Остаточные данные представляют пиксельные разности между исходным блоком, подлежащим кодированию, и предсказательным блоком. Внешне-кодируемый блок кодируется согласно вектору движения, который указывает блок опорных выборок, образующий предсказательный блок, и остаточные данные указывают разность между кодированным блоком и предсказательным блоком. Внутренне-кодируемый блок кодируется согласно режиму внутреннего кодирования и остаточным данным. Для дополнительного сжатия остаточные данные можно преобразовывать из пиксельной области в область преобразования, получая остаточные коэффициенты, которые затем можно квантовать. Квантованные коэффициенты, первоначально расположенные в двумерной матрице, можно сканировать для создания одномерного вектора коэффициентов и для достижения еще большего сжатия можно применять энтропийное кодирование.

Сущность изобретения

Это изобретение относится к методам кодирования и декодирования видеосигнала. В частности, это изобретение описывает методы кодирования и декодирования видеоданных в режиме кодирования на основе палитры. В режиме кодирования на основе палитры пиксельные значения для блока видеоданных могут кодироваться относительно палитры значений цвета, связанных с блоком видеоданных. Палитра значений цвета может определяться видеокодером и может содержать значения цвета, наиболее характерные для конкретного блока. Видеокодер может назначать индекс в палитре значений цвета каждому пикселю в блоке видеоданных и сигнализировать такой индекс видеодекодеру в битовом потоке кодированного видеосигнала. Затем видеодекодер может использовать индекс в палитре для определения, какое значение цвета использовать для конкретного пикселя в блоке.

Помимо сигнализации индексов в палитре, видеокодер также может передавать саму палитру в битовом потоке кодированного видеосигнала. Методы передачи палитры могут включать в себя явную сигнализацию значений палитры, а также прогнозирование записей палитры для текущего блока из записей палитры для одного или более ранее кодированных блоков. Это изобретение описывает методы кодирования палитры, включающие в себя методы кодирования элементов синтаксиса, связанных с кодированием палитры и/или предсказанием палитры.

В одном примере изобретения способ декодирования видеоданных содержит прием блока видеоданных в битовом потоке кодированного видеосигнала, причем блок видеоданных кодируется с использованием режима кодирования на основе палитры, прием множества элементов синтаксиса, которые указывают палитру, которая использовалась для кодирования блока видеоданных, причем множество элементов синтаксиса включает в себя первый элемент синтаксиса, который указывает количество значений палитры для палитры, которые явно сигнализируются в битовом потоке кодированного видеосигнала, причем первый элемент синтаксиса кодируется с использованием одного или более кодов Голomba таким образом, что длина кодированного первого элемента синтаксиса меньше или равна заранее определен-

нящий инструкции, которые при выполнении предписывают одному или более процессорам устройства, выполненного с возможностью декодирования видеоданных, принимать блок видеоданных в битовом потоке кодированного видеосигнала, причем блок видеоданных кодируется с использованием режима кодирования на основе палитры принимать множество элементов синтаксиса, которые указывают палитру, которая использовалась для кодирования блока видеоданных, причем множество элементов синтаксиса включает в себя первый элемент синтаксиса, который указывает количество значений палитры для палитры, которые явно сигнализируются в битовом потоке кодированного видеосигнала, причем первый элемент синтаксиса кодируется с использованием одного или более кодов Голомба таким образом, что длина кодированного первого элемента синтаксиса меньше или равна заранее определенному максимальному количеству битов, декодировать множество элементов синтаксиса, в том числе, декодировать первый элемент синтаксиса с использованием одного или более кодов Голомба, реконструировать палитру на основании декодированного множества элементов синтаксиса и декодировать блок видеоданных с использованием реконструированной палитры.

В другом примере это изобретение описывает компьютерно-считываемый носитель данных, хранящий инструкции, которые при выполнении предписывают одному или более процессорам устройства, выполненного с возможностью кодирования видеоданных, кодировать блок видеоданных с использованием режима кодирования на основе палитры и палитры, генерировать множество элементов синтаксиса, которые указывают палитру, которая использовалась для кодирования блока видеоданных, причем множество элементов синтаксиса включает в себя первый элемент синтаксиса, который указывает количество значений палитры для палитры, которые явно сигнализируются в битовом потоке кодированного видеосигнала, кодировать первый элемент синтаксиса с использованием одного или более кодов Голомба таким образом, что длина кодированного первого элемента синтаксиса меньше или равна заранее определенному максимальному количеству битов, и включать множество элементов синтаксиса в битовый поток кодированного видеосигнала.

Детали одного или более примеров изложены в прилагаемых чертежах и нижеследующем описании. Другие признаки, задачи и преимущества явствуют из описания и чертежей, а также из формулы изобретения.

Краткое описание чертежей

Фиг. 1 - блок-схема, демонстрирующая пример системы кодирования видеоданных, которая может использовать методы, описанные в этом изобретении.

Фиг. 2 - блок-схема, демонстрирующая пример видеокодера, который может реализовать методы, описанные в этом изобретении.

Фиг. 3 - блок-схема, демонстрирующая пример видеодекодера, который может реализовать методы, описанные в этом изобретении.

Фиг. 4 - блок-схема, демонстрирующая пример блока кодирования на основе палитры видеокодера, показанного на фиг. 2.

Фиг. 5 - принципиальная схема, демонстрирующая пример метода предсказания палитры согласно методам изобретения.

Фиг. 6 - принципиальная схема, демонстрирующая пример метода кодирования двоичного вектора предсказания согласно методам изобретения.

Фиг. 7 - блок-схема, демонстрирующая пример блока декодирования на основе палитры видеокодера, показанного на фиг. 3.

Фиг. 8 - блок-схема операций, демонстрирующая пример способа кодирования видеосигнала согласно методам изобретения.

Фиг. 9 - блок-схема операций, демонстрирующая пример способа декодирования видеосигнала согласно методам изобретения.

Подробное описание

Это изобретение относится к области кодирования видеоданных и, в частности к прогнозированию или кодированию блока видеоданных в режиме кодирования на основе палитры. При традиционном кодировании видеоданных предполагается, что изображения являются тонально непрерывными и пространственно гладкими. На основании этих предположений разработаны различные инструменты, например блочное преобразование, фильтрация и т.д., и такие инструменты демонстрируют высокие показатели для видеозаписей естественного содержания. Однако в таких применениях, как удаленный рабочий стол, совместная работа и беспроводной дисплей, компьютерно-генерируемое содержимое экрана (например, текст или компьютерная графика) может быть преобладающим содержанием, подлежащим сжатию. Такого рода содержанию свойственно иметь дискретный тон и характерные резкие линии и высококонтрастные границы объектов. Предположение тональной непрерывности и гладкости уже неприемлемо к содержанию экрана и поэтому традиционные методы кодирования видеоданных не позволяют эффективно сжимать видеоданные, включающие в себя содержимое экрана.

Это изобретение описывает кодирование на основе палитры, особенно пригодное для кодирования содержания экрана. Например, исходя из того, что конкретная область видеоданных имеет сравнительно малое количество цветов, видеокодер (например, видеокодер или видеодекодер) может формировать так

называемую "палитру" для представления видеоданных конкретной области. Палитру можно выражать в виде таблицы цветов или пиксельных значений, представляющих видеоданные конкретной области (например, данного блока). Например, палитра может включать в себя наиболее преобладающие пиксельные значения в данном блоке. В ряде случаев, наиболее преобладающие пиксельные значения могут включать в себя одно или более пиксельных значений, наиболее часто возникающих в блоке. Дополнительно, в ряде случаев видеокодер может применять пороговое значение для определения, подлежит ли пиксельное значение включению как одно из наиболее преобладающих пиксельных значений в блоке. Согласно различным аспектам кодирования на основе палитры видеокодер может кодировать значения индекса, указывающие одно или более из пиксельных значений текущего блока, вместо кодирования фактических пиксельных значений или их остатков для текущего блока видеоданных. В контексте кодирования на основе палитры, значения индекса указывают соответствующие записи в палитре, которые используются для представления отдельных пиксельных значений текущего блока.

Например, видеокодер может кодировать блок видеоданных путем определения палитры для блока (например, явно кодируя палитру, прогнозируя палитру или совершая комбинированное действие), определяя положение записи в палитре для представления одного или более из пиксельных значений, и кодируя блок значениями индекса, которые указывают запись в палитре, используемую для представления пиксельных значений блока. В некоторых примерах, видеокодер может сигнализировать палитру и/или значения индекса в кодированном битовом потоке. В свою очередь, видеодекодер может получать, из кодированного битового потока, палитру для блока, а также значения индекса для отдельных пикселей блока. Видеодекодер может соотносить значения индекса пикселей с записями палитры для реконструкции различных пиксельных значений блока.

В соответствии с различными рассмотренными ниже примерами это изобретение описывает методы повышения эффективности кодирования при кодировании блоков видеоданных в режиме кодирования на основе палитры. Примеры этого изобретения включают в себя методы кодирования видеоданных с использованием режима кодирования на основе палитры и методы кодирования элементов синтаксиса, связанных с режимом кодирования на основе палитры. В некоторых примерах методы этого изобретения относятся к кодированию элементов синтаксиса, используемых видеодекодером для определения и/или реконструкции палитры для блока видеоданных.

В некоторых примерах методы кодирования на основе палитры этого изобретения можно конфигурировать для использования с одним или более стандартами кодирования видеоданных. Некоторые иллюстративные стандарты кодирования видеоданных включают в себя ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 или ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual и ITU-T H.264 (также известный как ISO/IEC MPEG-4 AVC), включая расширения "кодирование масштабируемых видеоданных" (SVC) и "кодирование многовидовых видеоданных" (MVC). В другом примере методы кодирования на основе палитры можно конфигурировать для использования с высокопроизводительным кодированием видеоданных (HEVC). HEVC представляет собой новый стандарт кодирования видеоданных, разработанный Joint Collaboration Team on Video Coding (JCT-VC) ITU-T Video Coding Experts Group (VCEG) и ISO/IEC Motion Picture Experts Group (MPEG).

Недавно разработка HEVC была закончена Joint Collaboration Team on Video Coding (JCT-VC) ITU-T Video Coding Experts Group (VCEG) и ISO/IEC Motion Picture Experts Group (MPEG). Последняя спецификация HEVC, именуемая в дальнейшем HEVC версия 1 или HEVC1, описана в "ITU-T H.265 (V1)", опубликованном 24 марта 2015 г. доступна по адресу <http://www.itu.int/ITU-T/recommendations/rec.aspx?rec=11885&lang=en>. Стандарт HEVC также описан в документе ITU-T H.265, SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS, Infrastructure of Audiovisual Services-Coding of Moving Video, "High Efficiency Video Coding", апрель 2013 г. Недавняя спецификация расширений диапазона, в дальнейшем именуемая RExt, описана в "ITU-T H.265 (V2)", опубликованном 24 марта 2015 г. доступна по адресу <http://www.itu.int/ITU-T/recommendations/rec.aspx?rec=12296&lang=en>.

Для обеспечения более эффективного кодирования генерируемого содержания экрана JCT-VC разрабатывает расширение стандарта HEVC, именуемое стандартом кодирования содержания экрана (SCC) HEVC. Недавний рабочий проект стандарта SCC HEVC, именуемый "HEVC SCC Draft 2" или "WD2", описан в документе JCTVC-S1005, R. Joshi and J. Xu, "HEVC screen content coding draft text 2", Joint Collaborative Team on Video Coding (JCT-VC) ITU-T SG 16 WP 3 и ISO/IEC JTC 1/SC 29/WG 11, 19th Meeting: Strasbourg, FR, 17-24 October 2014.

На фиг. 1 показана блок-схема, демонстрирующая пример системы 10 кодирования видеоданных, которая может использовать методы этого изобретения для кодирования видеоданных на основе палитры. Используемый здесь термин "видеокодер", в целом, относится к видеокодерам и видеодекодерам. В этом изобретении термины "кодирование видеоданных" или "кодирование" могут означать, в целом, кодирование видеосигнала или декодирование видеосигнала. Видеокодер 20 и видеодекодер 30 системы 10 кодирования видеоданных представляют примеры устройств, которые могут быть выполнены с возможностью осуществления методов кодирования на основе палитры видеоданных в соответствии с различными примерами, описанными в этом изобретении. Например, видеокодер 20 и видеодекодер 30 могут быть выполнены с возможностью выборочно кодировать различные блоки видеоданных, например CU

или PU при кодировании HEVC, с использованием либо кодирования на основе палитры, либо кодирования без помощи палитры. Режимы кодирования без помощи палитры могут относиться к различным режимам временного кодирования с внешним предсказанием или режимам пространственного кодирования с внутренним предсказанием, например различным режимам кодирования, заданным стандартом HEVC. Однако следует понимать, что методы этого изобретения можно использовать с любыми методами и/или стандартами кодирования видеоданных, которые используют режим кодирования на основе палитры.

Как показано на фиг. 1, система 10 кодирования видеоданных включает в себя исходное устройство 12 и конечное устройство 14. Исходное устройство 12 генерирует данные кодированного видеосигнала. Соответственно, исходное устройство 12 может именоваться устройством кодирования видеосигнала или устройством кодирования видеосигнала. Конечное устройство 14 может декодировать данные кодированного видеосигнала, генерируемые исходным устройством 12. Соответственно, конечное устройство 14 может именоваться устройством декодирования видеосигнала или устройством декодирования видеосигнала. Исходное устройство 12 и конечное устройство 14 могут быть примерами устройств кодирования видеоданных или устройств кодирования видеоданных.

Исходное устройство 12 и конечное устройство 14 могут содержать самые разнообразные устройств, в том числе, настольные компьютеры, мобильные вычислительные устройства, компьютеры типа ноутбук (например, портативный компьютер), планшетные компьютеры, телевизионные приставки, телефонные трубки, например, так называемые "смартфоны", телевизоры, камеры, устройства отображения, цифровые медиапроигрыватели, игровые консоли, автомобильные компьютеры и т.п.

Конечное устройство 14 может принимать данные кодированного видеосигнала от исходного устройства 12 по каналу 16. Канал 16 может содержать одну или более сред или одно или более устройств, способных переносить данные кодированного видеосигнала с исходного устройства 12 на конечное устройство 14. В одном примере канал 16 может содержать одну или более сред передачи данных, которые позволяют исходному устройству 12 передавать данные кодированного видеосигнала непосредственно на конечное устройство 14 в реальном времени. В этом примере, исходное устройство 12 может модулировать данные кодированного видеосигнала согласно стандарту связи, например протоколу беспроводной связи, и может передавать модулированные видеоданные на конечное устройство 14. Одна или более сред передачи данных могут включать в себя беспроводные и/или проводные среды передачи данных, например, радиочастотный (RF) спектр или одну или более физических линий передачи. Одна или более сред передачи данных могут образовывать часть пакетной сети, например, локальной сети, глобальной сети или глобальной сети (например, интернета). Одна или более сред передачи данных могут включать в себя маршрутизаторы, коммутаторы, базовые станции или другое оборудование, которые позволяют осуществлять связь между исходным устройством 12 и конечным устройством 14.

В другом примере канал 16 может включать в себя носитель данных, где хранятся данные кодированного видеосигнала, генерируемые исходным устройством 12. В этом примере конечное устройство 14 может осуществлять доступ к носителю данных посредством доступа к диску или доступа к карте. Носитель данных может включать в себя различные локально адресуемые носители данных, например, диски Blu-ray, DVD, CD-ROM, флеш-память или другие пригодные цифровые носители данных для хранения данных кодированного видеосигнала.

В дополнительном примере канал 16 может включать в себя файловый сервер или другое промежуточное запоминающее устройство, где хранятся данные кодированного видеосигнала, генерируемые исходным устройством 12. В этом примере конечное устройство 14 может осуществлять доступ к данным кодированного видеосигнала, хранящимся на файловом сервере или другом промежуточном запоминающем устройстве, посредством потоковой передачи или загрузки. Файловый сервер может представлять собой разновидность сервера, позволяющую хранить данные кодированного видеосигнала и передавать данные кодированного видеосигнала на конечное устройство 14. Примеры файловых серверов включают в себя веб-серверы (например, для веб-сайта), серверы протокола передачи файла (FTP), устройства сетевого хранения данных (NAS) и локальные приводы дисков.

Конечное устройство 14 может осуществлять доступ к данным кодированного видеосигнала через стандартное соединение для передачи данных, например интернет-соединение. Примеры соединений для передачи данных могут включать в себя беспроводные каналы (например, соединения Wi-Fi), проводные соединения (например, DSL, кабельный модем и т.д.) или их комбинации, пригодные для осуществления доступа к данным кодированного видеосигнала, хранящимся на файловом сервере. Передача данных кодированного видеосигнала с файлового сервера может представлять собой потоковую передачу, загрузочную передачу или их комбинацию.

Методы этого изобретения для кодирования видеоданных на основе палитры не ограничиваются беспроводными применениями или установками. Методы можно применять к кодированию видеоданных для поддержки различных мультимедийных применений, например эфирного телевизионного вещания, кабельных телевизионных передач, спутниковых телевизионных передач, передач потокового видео, например, через интернет, кодированию видеоданных для хранения на носителе данных, декодированию видеоданных, хранящихся на носителе данных, или других применений. В некоторых примерах

система 10 кодирования видеоданных может быть выполнена с возможностью поддержки односторонней или двусторонней передачи видео для поддержки таких применений, как потоковая передача видео, воспроизведение видео, вещание видео и/или видеотелефония.

Система 10 кодирования видеоданных, представленная на фиг. 1, является лишь одним примером. Методы этого изобретения можно применять к случаям использования кодирования видеоданных (например, кодированию видеосигнала или декодированию видеосигнала), которые не обязательно включают в себя какую-либо передачу данных между устройствами кодирования и декодирования. В других примерах данные извлекаются из локальной памяти, передаются в потоке по сети и т.п. Устройство кодирования видеосигнала может кодировать и сохранять данные в памяти и/или устройство декодирования видеосигнала может извлекать и декодировать данные из памяти. Во многих примерах кодирование и декодирование осуществляется устройствами, которые не осуществляют связь друг с другом, но просто кодируют данные в память и/или извлекают и декодируют данные из памяти.

В примере, показанном на фиг. 1, исходное устройство 12 включает в себя источник 18 видеосигнала, видеокодер 20 и выходной интерфейс 22. В некоторых примерах выходной интерфейс 22 может включать в себя модулятор/демодулятор (модем) и/или передатчик. Источник 18 видеосигнала может включать в себя устройство захвата видео, например видеокамеру, архив видеозаписей, содержащий ранее захваченные видеоданные, интерфейс подачи видео для приема видеоданных от поставщика видеоконтента и/или систему компьютерной графики для генерации видеоданных или комбинацию таких источников видеоданных.

Видеокодер 20 может кодировать видеоданные из источника 18 видеосигнала. В некоторых примерах исходное устройство 12 непосредственно передает данные кодированного видеосигнала на конечное устройство 14 через выходной интерфейс 22. В других примерах данные кодированного видеосигнала также могут сохраняться на носителе данных или файловом сервере для дальнейшего доступа со стороны конечного устройства 14 для декодирования и/или воспроизведения.

В примере, показанном на фиг. 1, конечное устройство 14 включает в себя входной интерфейс 28, видеodeкодер 30 и устройство 32 отображения. В некоторых примерах входной интерфейс 28 включает в себя приемник и/или модем. Входной интерфейс 28 может принимать данные кодированного видеосигнала по каналу 16. Устройство 32 отображения может быть объединено с конечным устройством 14 или может быть внешним по отношению к нему. В общем случае устройство 32 отображения отображает декодированные видеоданные. Устройство 32 отображения может содержать различные устройства отображения, например, жидкокристаллический дисплей (LCD), плазменный дисплей, дисплей на органических светодиодах (OLED) или устройство отображения другого типа.

Это изобретение, в целом, может относиться к видеокодеру 20, "сигнализирующему" или "передающему" некоторую информацию на другое устройство, например видеodeкодер 30. Термин "сигнализация" или "передача", в целом, может относиться к передаче элементов синтаксиса и/или других данных, используемых для декодирования сжатых видеоданных. Такая передача может происходить в реальном или почти реальном времени. Попеременно такая передача может происходить в течение промежутка времени, например может происходить при сохранении элементов синтаксиса на компьютерно-считываемый носитель данных в кодированном битовом потоке во время кодирования, которые затем могут извлекаться устройством декодирования в любое время после сохранения на этот носитель. Таким образом, хотя можно говорить, что видеodeкодер 30 "принимает" некоторую информацию, прием информации не обязательно происходит в реальном или почти реальном времени, и она может извлекаться из носителя некоторое время спустя после сохранения.

Каждый из видеокодера 20 и видеodeкодера 30 может быть реализован в виде любой из различных пригодных схем, например одного или более микропроцессоров, цифровых сигнальных процессоров (DSP), специализированных интегральных схем (ASIC), вентильных матриц, программируемых пользователем (FPGA), дискретной логики, оборудования или любых их комбинаций. Если методы реализуются частично программными средствами, устройство может сохранять инструкции для программного обеспечения на пригодном, нетранзиторном компьютерно-считываемом носителе данных и может выполнять инструкции аппаратными средствами с использованием одного или более процессоров для осуществления методов этого изобретения. Любое из вышеупомянутого (включая аппаратное обеспечение, программное обеспечение, комбинацию аппаратного обеспечения и программного обеспечения и т.д.) можно рассматривать как один или более процессоров. Каждый из видеокодера 20 и видеodeкодера 30 может быть включен в один или более кодеров или декодеров, любой из которых может быть встроен как часть объединенного кодера/декодера (кодека) в соответствующее устройство.

В некоторых примерах видеокодер 20 и видеodeкодер 30 действуют согласно стандарту сжатия видеосигнала, например вышеупомянутому стандарту HEVC. На основе базового стандарта HEVC ведутся работы по созданию расширений "кодирование масштабируемых видеоданных", "кодирование многовидео-видеоданных" и "3D кодирование" для HEVC. Кроме того, для расширения стандарта HEVC могут быть обеспечены режимы кодирования на основе палитры, например, описанные в этом изобретении. В некоторых примерах описанные в этом изобретении методы кодирования на основе палитры можно применять к кодерам и декодерам, выполненным с возможностью работать согласно другим стандартам

кодирования видеоданных. Соответственно применение режима кодирования на основе палитры для кодирования единиц кодирования (CU) или единиц предсказания (PU) в кодеке HEVC описано в порядке примера.

В HEVC и других стандартах кодирования видеоданных видеопоследовательность обычно включает в себя последовательность изображений. Изображения также могут именоваться "кадрами". Изображение может включать в себя три матрицы выборок, обозначенных S_L , S_{Cb} и S_{Cr} . S_L представляет собой двухмерную матрицу (например, блок) выборок яркости. S_{Cb} представляет собой двухмерную матрицу выборок цветности Cb . S_{Cr} представляет собой двухмерную матрицу выборок цветности Cr . выборки цветности также могут именоваться здесь выборками "цветности". В других случаях изображение может быть монохромным и может включать в себя только матрицу выборок яркости.

Для генерации кодированного представления изображения в HEVC видеокодер 20 может генерировать набор единиц дерева кодирования (CTU). Каждая из CTU может представлять собой блок выборок яркости дерева кодирования, два соответствующих блока выборок цветности - дерево кодирования и структуры синтаксиса, используемые для кодирования выборок блоков дерева кодирования. Блок дерева кодирования может представлять собой блок $N \times N$ выборок. CTU также могут именоваться "блоком дерева" или "наибольшей единицей кодирования" (LCU). CTU HEVC могут быть, в целом, аналогичны макроблокам других стандартов, например H.264/AVC. Однако CTU не обязана ограничиваться конкретным размером и может включать в себя одну или более единиц кодирования (CU). Слайс может включать в себя целое число последовательно упорядоченных CTU в растровом сканировании. Кодированный слайс может содержать заголовок слайса и данные слайса. Заголовок слайса для слайса может быть структурой синтаксиса, которая включает в себя элементы синтаксиса, которые обеспечивают информацию о слайсе. Данные слайса могут включать в себя кодированные CTU слайса.

В этом изобретении может использоваться термин "единица видео" или "блок видео" или "блок" для обозначения одного или более блоков выборок и структур синтаксиса, используемых для кодирования выборок одного или более блоков выборок. Возможные типы единиц или блоков видео могут включать в себя CTU, CU, PU, единицы преобразования (TU), макроблоки, разделы макроблоков и т.д. В некоторых контекстах вместо PU можно рассматривать макроблоки или разделы макроблоков.

Для генерации кодированной CTU видеокодер 20 может рекурсивно осуществлять разбиение квадрата дерева на блоках дерева кодирования CTU для разделения блоков дерева кодирования на блоки кодирования, отсюда название "единицы дерева кодирования". Блок кодирования является блоком $N \times N$ выборок. CU может представлять собой блок кодирования выборок яркости и два соответствующих блока кодирования выборок цветности изображения, которые имеют матрицу выборок яркости, матрицу выборок Cb и матрицу выборок Cr , и структуры синтаксиса, используемые для кодирования выборок блоков кодирования. Видеокодер 20 может разбивать блок кодирования CU на один или более блоков предсказания. Блок предсказания может представлять собой прямоугольный (например, квадратный или отличный от квадратного) блок выборок, к которым применяется одно и то же предсказание. Единица предсказания (PU) CU может представлять собой блок предсказания выборок яркости, два соответствующих блока предсказания выборок цветности изображения и структуры синтаксиса, используемые для предсказания выборок блока предсказания. Видеокодер 20 может генерировать предсказательные блоки яркости, Cb и Cr для блоков предсказания яркости, Cb и Cr каждой PU CU.

Видеокодер 20 может использовать внутреннее предсказание или внешнее предсказание для генерации предсказательных блоков для PU. Если видеокодер 20 использует внутреннее предсказание для генерации предсказательных блоков PU, видеокодер 20 может генерировать предсказательные блоки PU на основании декодированных выборок изображения, связанного с PU.

Если видеокодер 20 использует внешнее предсказание для генерации предсказательных блоков PU, видеокодер 20 может генерировать предсказательные блоки PU на основании декодированных выборок одного или более изображений, отличных от изображения, связанного с PU. Для генерации предсказательных блоков PU видеокодер 20 может использовать одинарное предсказание или двойное предсказание. Когда видеокодер 20 использует одинарное предсказание для генерации предсказательных блоков для PU, PU может иметь единственный вектор движения (MV). Когда видеокодер 20 использует двойное предсказание для генерации предсказательных блоков для PU, PU может иметь два MV.

После того, как видеокодер 20 генерирует предсказательные блоки (например, предсказательные блоки яркости, Cb и Cr) для одной или более PU CU, видеокодер 20 может генерировать остаточные блоки для CU. Каждая выборка в остаточном блоке CU может указывать разность между выборкой в предсказательном блоке PU CU и соответствующей выборкой в блоке кодирования CU. Например, видеокодер 20 может генерировать остаточный блок яркости для CU. Каждая выборка в остаточном блоке яркости CU указывает разность между выборкой яркости в одном из предсказательных блоков яркости CU и соответствующей выборкой в исходном блоке кодирования яркости CU. Кроме того, видеокодер 20 может генерировать остаточный блок Cb для CU. Каждая выборка в остаточном блоке Cb CU может указывать разность между выборкой Cb в одном из предсказательных блоков Cb CU и соответствующей выборкой в исходном блоке кодирования Cb CU. Видеокодер 20 также может генерировать остаточный блок Cr для CU. Каждая выборка в остаточном блоке Cr CU может указывать разность между выборкой

Сг в одном из предсказательных блоков Сг CU и соответствующей выборкой в исходном блоке кодирования Сг CU.

Кроме того, видеокодер 20 может использовать разбиение квадрадерева для разложения остаточных блоков (например, остаточных блоков яркости, Сb и Сr) CU на один или более блоков преобразования (например, блоков преобразования яркости, Сb и Сr). Блок преобразования может представлять собой прямоугольный блок выборок, к которым применяется одно и то же преобразование. Единица преобразования (TU) CU может представлять собой блок преобразования выборок яркости, два соответствующих блока преобразования выборок цветности и структуры синтаксиса, используемые для преобразования выборок блока преобразования. Таким образом, каждая TU CU может быть связана с блоком преобразования яркости, блоком преобразования Сb и блоком преобразования Сг. Блок преобразования яркости, связанный с TU, может представлять собой подблок остаточного блока яркости CU. Блок преобразования Сb может представлять собой подблок остаточного блока Сb CU. Блок преобразования Сг может представлять собой подблок остаточного блока Сг CU.

Видеокодер 20 может применять одно или более преобразований к блоку преобразования для генерации блока коэффициентов для TU. Блок коэффициентов может представлять собой двухмерную матрицу коэффициентов преобразования. Коэффициент преобразования может быть скалярной величиной. Например, видеокодер 20 может применять одно или более преобразований к блоку преобразования яркости TU для генерации блока коэффициентов яркости для TU. Видеокодер 20 может применять одно или более преобразований к блоку преобразования Сb TU для генерации блока коэффициентов Сb для TU. Видеокодер 20 может применять одно или более преобразований к блоку преобразования Сг TU для генерации блока коэффициентов Сг для TU.

После генерации блока коэффициентов (например, блока коэффициентов яркости, блока коэффициентов Сb или блока коэффициентов Сг) видеокодер 20 может квантовать блок коэффициентов. Квантование в целом означает процесс, в котором коэффициенты преобразования квантуются для уменьшения, по возможности, объема данных, используемых для представления коэффициентов преобразования, обеспечивая дополнительное сжатие. После того, как видеокодер 20 квантует блок коэффициентов, видеокодер 20 может энтропийно кодировать элементы синтаксиса, указывающие квантованные коэффициенты преобразования. Например, видеокодер 20 может осуществлять контекстно-адаптивное двоичное арифметическое кодирование (CABAC) на элементах синтаксиса, указывающих квантованные коэффициенты преобразования. Видеокодер 20 может выводить энтропийно кодированные элементы синтаксиса в битовом потоке. Битовый поток также может включать в себя элементы синтаксиса, не подвергнутые энтропийному кодированию.

Видеокодер 20 может выводить битовый поток, который включает в себя энтропийно кодированные элементы синтаксиса. Битовый поток может включать в себя последовательность битов, которая образует представление кодированных изображений и соответствующих данных. Битовый поток может содержать последовательность единиц уровня сетевой абстракции (NAL). Каждая из единиц NAL включает в себя заголовок единицы NAL и инкапсулирует полезную нагрузку первичной байтовой последовательности (RBSP). Заголовок единицы NAL может включать в себя элемент синтаксиса, который указывает код типа единицы NAL. Код типа единицы NAL, указанный в заголовке единицы NAL единицы NAL, указывает тип единицы NAL. RBSP может быть структурой синтаксиса, содержащей целое число байтов, инкапсулированных в единице NAL. В ряде случаев RBSP включает в себя нулевые биты.

Различные типы единиц NAL могут инкапсулировать различные типы RBSP. Например, первый тип единицы NAL может инкапсулировать RBSP для набора параметров изображения (PPS), второй тип единицы NAL может инкапсулировать RBSP для кодированного слайса, третий тип единицы NAL может инкапсулировать RBSP для информации дополнительного улучшения (SEI) и т.д. Единицы NAL, инкапсулирующие RBSP для данных кодирования видео (в отличие от RBSP для наборов параметров и сообщений SEI) может именоваться единицами NAL уровня кодирования видеоданных (VCL).

Видеодекодер 30 может принимать битовый поток, генерируемый видеокодером 20. Кроме того, видеодекодер 30 может получать элементы синтаксиса из битового потока. Например, видеодекодер 30 может разбирать битовый поток для декодирования элементов синтаксиса из битового потока. Видеодекодер 30 может реконструировать изображения видеоданных, по меньшей мере, частично на основании элементов синтаксиса, полученных (например, декодированных) из битового потока. Процесс для реконструкции видеоданных, в целом, может быть обратным процессу, осуществляемому видеокодером 20. Например, видеодекодер 30 может использовать MV PU для определения блоков выборок внешнего предсказания (например, блоков внешнего предсказания) для PU текущей CU. Кроме того, видеодекодер 30 может обратно квантовать блоки коэффициентов преобразования, связанные с TU текущей CU. Видеодекодер 30 может осуществлять обратные преобразования на блоках коэффициентов преобразования для реконструкции блоков преобразования, связанных с TU текущей CU. Видеодекодер 30 может реконструировать блоки кодирования текущей CU путем добавления выборок блоков предсказательных выборок для PU текущей CU к соответствующим выборкам блоков преобразования TU текущей CU. Реконструируя блоки кодирования для каждой CU изображения, видеодекодер 30 может реконструировать изображение.

В некоторых примерах видеокодер 20 и видеодекодер 30 могут быть выполнены с возможностью осуществления кодирования на основе палитры. Например, в кодировании на основе палитры, вместо осуществления вышеописанных методов кодирования с внутренним предсказанием или с внешним предсказанием, видеокодер 20 и видеодекодер 30 могут кодировать так называемую палитру как таблицу цветов или пиксельные значения, представляющие видеоданные конкретной области (например, данного блока). Таким образом, вместо кодирования фактических пиксельных значений или их остатков для текущего блока видеоданных видеокодер может кодировать значения индекса для одного или более из пиксельных значений текущего блока, где значения индекса указывают записи в палитре, которые используются для представления пиксельных значений текущего блока (например, индекс может отображаться в набор значений Y, Cg и Cb или в набор значений R, G и B).

Например, видеокодер 20 может кодировать блок видеоданных путем определения палитры для блока, определяя положение записи в палитре, имеющей значение, представляющее значение одного или более отдельных пикселей блока, и кодируя блок значениями индекса, которые указывают запись в палитре, используемую для представления одного или более отдельных пиксельных значений блока. Дополнительно видеокодер 20 может сигнализировать значения индекса в кодированном битовом потоке. В свою очередь, устройство декодирования видеосигнала (например, видеодекодер 30) может получать из кодированного битового потока палитру для блока, а также значения индекса, используемые для определения различных отдельных пикселей блока с использованием палитры. Видеодекодер 30 может согласовывать значения индекса отдельных пикселей с записями палитры для реконструкции пиксельных значений блока. В случаях, когда пиксельное значение отдельного пикселя недостаточно близко к любому из пиксельных значений, представленных соответствующей палитрой для блока, видеодекодер 30 может идентифицировать такой отдельный пиксель как выходной пиксель, в целях кодирования на основе палитры. Пиксельное значение выходного пикселя может кодироваться явно, а не индексом палитры.

В другом примере видеокодер 20 может кодировать блок видеоданных согласно следующим операциям. Видеокодер 20 может определять остаточные значения предсказания для отдельных пикселей блока, определять палитру для блока и определять положение записи (например, значения индекса) в палитре, имеющей значение, представляющее значение одного или более из остаточных значений предсказания отдельных пикселей. Дополнительно видеокодер 20 может кодировать блок значениями индекса, которые указывают запись в палитре, используемую для представления соответствующего остаточного значения предсказания для каждого отдельного пикселя блока. Видеодекодер 30 может получать из кодированного битового потока, сигназируемого исходным устройством 12, палитру для блока, а также значения индекса для остаточных значений предсказания, соответствующих отдельным пикселям блока. Как описано, значения индекса могут соответствовать записям в палитре, связанной с текущим блоком. В свою очередь, видеодекодер 30 может соотносить значения индекса остаточных значений предсказания с записями палитры для реконструкции остаточных значений предсказания блока. Остаточные значения предсказания могут добавляться к значениям предсказания (например, полученным с использованием внутреннего или внешнего предсказания) для реконструкции пиксельных значений блока.

Как более подробно описано ниже, основная идея кодирования на основе палитры состоит в том, что для данного блока видеоданных, подлежащего кодированию, видеокодер 20 может выводить палитру, которая включает в себя наиболее преобладающие пиксельные значения в текущем блоке. Например, палитра может относиться к количеству пиксельных значений, которые определяются или предполагаются преобладающими и/или представляющими для текущей CU. Видеокодер 20 может сначала передавать размер и элементы палитры на видеодекодер 30. Дополнительно видеокодер 20 может кодировать пиксельные значения в данном блоке согласно некоторому порядку сканирования. Для каждого пикселя, включенного в данный блок, видеокодер 20 может сигнализировать значение индекса, которое отображает пиксельное значение в соответствующую запись в палитре. Если пиксельное значение недостаточно близко к значению любой из записей палитры (например, по сравнению с некоторым заранее определенным порогом), то такой пиксель определяется как "выходной пиксель". В соответствии с кодированием на основе палитры видеокодер 20 может кодировать и сигнализировать значение индекса, которое зарезервировано для выходного пикселя, т.е. указывать, что это выходной пиксель, а не пиксель, для которого существует запись в палитре. В некоторых примерах видеокодер 20 также может кодировать и сигнализировать пиксельное значение или остаточное значение (или их квантованные версии) для выходного пикселя, включенного в данный блок.

Приняв битовый поток кодированного видеосигнала, сигназируемый видеокодером 20, видеодекодер 30 может сначала определять палитру на основании информации, принятой от видеокодера 20. Затем видеодекодер 30 может отображать принятые значения индекса, связанные с положениями пикселей в данном блоке, в записи палитры для реконструкции пиксельных значений данного блока. В ряде случаев видеодекодер 30 может определять, что пиксель блока, кодированного на основе палитры, является выходным пикселем, например, путем определения, что пиксель кодируется на основе палитры значением индекса, зарезервированным для выходных пикселей. В случаях, когда видеодекодер 30 идентифицирует выходной пиксель в блоке, кодированном на основе палитры, видеодекодер 30 может принимать пиксельное значение или остаточное значение (или их квантованные версии) для выходного пиксе-

ля, включенного в данный блок. Видеодекодер 30 может реконструировать блок, кодированный на основе палитры, отображая отдельные пиксельные значения в соответствующие записи палитры и используя пиксельное значение или остаточное значение (или их квантованные версии) для реконструкции любых выходных пикселей, включенных в блок, кодированный на основе палитры.

Видеокодер 20 и/или видеодекодер 30 может быть выполнен с возможностью действовать согласно методам, описанным в этом изобретении, как будет более подробно описано ниже. В общем случае видеокодер 20 и/или видеодекодер 30 может быть выполнен с возможностью кодирования и декодирования видеоданных с использованием одного или более режимов кодирования палитры, причем режимы кодирования палитры не включают в себя режим совместного использования палитры. Методы этого изобретения включают в себя устройство кодирования видеоданных, например видеокодер 20, выполненный с возможностью определения первого бина первого элемента синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре. Видеокодер 20 может быть дополнительно выполнен с возможностью кодирования битового потока. Битовый поток может включать в себя первый элемент синтаксиса. Битовый поток также может не включать в себя второй элемент синтаксиса, который указывает режим совместного использования палитры. В некоторых примерах, определение первого бина первого элемента синтаксиса содержит определение первого бина первого элемента синтаксиса с использованием контекстно-адаптивного двоичного арифметического кодирования. В других примерах определение первого бина первого элемента синтаксиса содержит определение первого бина первого элемента синтаксиса с использованием одного или более контекстов. В некоторых примерах использования одного или более контекстов один или более контекстов могут базироваться на по меньшей мере одном из предсказанного количества записей кодирования палитры и размера блока.

Кроме того, это изобретение описывает видеокодер 20, выполненный с возможностью определения, что текущий пиксель является первым пикселем в линии в порядке сканирования. Видеокодер 20 может дополнительно определять наличие соседнего пикселя, расположенного над текущим пикселем. В случае определения, что текущий пиксель является первым пикселем в линии в порядке сканирования, и определения наличия соседнего пикселя, расположенного над текущим пикселем, видеокодер 20 может быть дополнительно выполнен с возможностью обходить кодирование первого элемента синтаксиса в битовом потоке, причем первый элемент синтаксиса указывает тип серии, и кодировать оставшуюся часть битового потока.

Кроме того, методы этого изобретения включают в себя видеокодер 20, выполненный с возможностью определения первого элемента синтаксиса, который указывает максимально допустимый размер палитры и имеет минимальное значение нуль. Видеокодер 20 также может быть выполнен с возможностью кодирования битового потока, который включает в себя первый элемент синтаксиса. В некоторых примерах битовый поток дополнительно включает в себя второй элемент синтаксиса, который указывает максимальный размер палитры предсказателя и имеет минимальное значение нуль. В некоторых примерах первый элемент синтаксиса имеет максимальное значение 4096 и второй элемент синтаксиса имеет максимальное значение 8192. В других примерах первый элемент синтаксиса имеет максимальное значение 4095 и второй элемент синтаксиса имеет максимальное значение 4095. В других примерах первый элемент синтаксиса имеет максимальное значение 4095 и второй элемент синтаксиса имеет максимальное значение 8191. В прочих примерах первый элемент синтаксиса имеет максимальное значение, равное количеству пикселей в наибольшей единице кодирования, и второй элемент синтаксиса имеет максимальное значение, равное положительной постоянной, например 2, умноженной на максимальное значение первого элемента синтаксиса. В других примерах битовый поток включает в себя другой элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре. В некоторых примерах этого этот элемент синтаксиса представляется одним из кода Голомба-Райса, экспоненциального кода Голомба, усеченного кода Райса или унарного кода. В других примерах этого этот элемент синтаксиса представляется одним из усеченного кода Голомба-Райса, усеченного экспоненциального кода Голомба, усеченного кода Райса, усеченного унарного кода или кода, который также используется для кодирования третьего элемента синтаксиса, включенного в кодированный битовый поток, который указывает, копируется ли индекс палитры из индекса палитры в строке над текущим пикселем или явно кодируется в кодированном битовом потоке. В некоторых примерах этот элемент синтаксиса представляется режимом усеченного Райса. В некоторых примерах элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре, имеет максимальное значение, равное количеству пикселей в текущем блоке видеоданных.

Кроме того, это изобретение описывает устройство кодирования видеоданных, например видеокодер 30, выполненный с возможностью приема кодированного битового потока. Кодированный битовый поток не включает в себя первый элемент синтаксиса, который указывает режим совместного использования палитры. Кроме того, кодированный битовый поток включает в себя второй элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре. Видеодекодер 30 может быть дополнительно выполнен с возможностью декодирования первого бина второго элемента синтаксиса. В некоторых примерах декодирование первого бина второго элемента синтаксиса содержит декодирование первого бина второго элемента синтаксиса с использованием контекстно-

адаптивного двоичного арифметического кодирования элемент. В других примерах декодирование первого бина второго элемента синтаксиса содержит декодирование первого бина второго элемента синтаксиса с использованием одного или более контекстов. В некоторых примерах использования одного или более контекстов один или более контекстов могут базироваться по меньшей мере на одном из предсказанного количества записей кодирования палитры и размера блока.

Кроме того, методы этого изобретения включают в себя видеodeкодер 30, выполненный с возможностью приема кодированного битового потока. Кодированный битовый поток может включать в себя первый элемент синтаксиса, который указывает тип серии. Видеodeкодер 30 может быть дополнительно выполнен с возможностью определения, что текущий пиксель является первым пикселем в линии в порядке сканирования. Видеodeкодер 30 может дополнительно определять наличие соседнего пикселя, расположенного над текущим пикселем. В случае определения, что текущий пиксель является первым пикселем в линии в порядке сканирования, и определения наличия соседнего пикселя, расположенного над текущим пикселем, видеodeкодер 30 может обходить декодирование первого элемента синтаксиса.

Кроме того, методы этого изобретения включают в себя видеodeкодер 30, выполненный с возможностью приема кодированного битового потока, который включает в себя первый элемент синтаксиса, который указывает максимально допустимый размер палитры и имеет минимальное значение нуль. Видеodeкодер 30 может быть дополнительно выполнен с возможностью декодирования кодированного битового потока. В некоторых примерах кодированный битовый поток дополнительно включает в себя второй элемент синтаксиса, который указывает максимальный размер палитры предсказателя и имеет минимальное значение нуль. В некоторых примерах первый элемент синтаксиса имеет максимальное значение 4096 и второй элемент синтаксиса имеет максимальное значение 8192. В других примерах первый элемент синтаксиса имеет максимальное значение 4095 и второй элемент синтаксиса имеет максимальное значение 4095. В других примерах первый элемент синтаксиса имеет максимальное значение 4095 и второй элемент синтаксиса имеет максимальное значение 8191. В прочих примерах первый элемент синтаксиса имеет максимальное значение, равное количеству пикселей в наибольшей единице кодирования, и второй элемент синтаксиса имеет максимальное значение, равное положительной постоянной, например 2, умноженной на максимальное значение первого элемента синтаксиса. В других примерах кодированный битовый поток включает в себя другой элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре. В некоторых примерах элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре, представляется одним из кода Голомба-Райса, экспоненциального кода Голомба, усеченного кода Райса или унарного кода. В других примерах элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре, представляется одним из усеченного кода Голомба-Райса, усеченного экспоненциального кода Голомба, усеченного кода Райса, усеченного унарного кода или того же кода, который используется для кодирования элемента синтаксиса, который указывает, копируется ли индекс палитры из индекса палитры в строке над текущим пикселем или явно кодируется в кодированном битовом потоке. В некоторых примерах элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре представляется режимом усеченного Райса. В некоторых примерах элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре, имеет максимальное значение, равное количеству пикселей в текущем блоке видеоданных.

В другом примере изобретения видеodeкодер 30 может быть выполнен с возможностью принимать блок видеоданных в битовом потоке кодированного видеосигнала, причем блок видеоданных кодируется с использованием режима кодирования на основе палитры, принимать множество элементов синтаксиса, которые указывают палитру, которая использовалась для кодирования блока видеоданных, причем множество элементов синтаксиса включает в себя первый элемент синтаксиса, который указывает количество значений палитры для палитры, которые явно сигнализируются в битовом потоке кодированного видеосигнала, декодировать множество элементов синтаксиса, в том числе, декодировать первый элемент синтаксиса с использованием одного или более кодов Голомба, реконструировать палитру на основании декодированного множества элементов синтаксиса, и декодировать блок видеоданных с использованием реконструированной палитры.

В другом примере изобретения видеodeкодер 20 может быть выполнен с возможностью кодировать блок видеоданных с использованием режима кодирования на основе палитры и палитры, генерировать множество элементов синтаксиса, которые указывают палитру, которая использовалась для кодирования блока видеоданных, причем множество элементов синтаксиса включает в себя первый элемент синтаксиса, который указывает количество значений палитры для палитры, которые явно сигнализируются в битовом потоке кодированного видеосигнала, кодировать первый элемент синтаксиса с использованием одного или более кодов Голомба, и включения множества элементов синтаксиса в битовый поток кодированного видеосигнала.

На фиг. 2 показана блок-схема, демонстрирующая пример видеodeкодера 20, который может реализовать различные методы этого изобретения. Фиг. 2 обеспечена в целях объяснения и не подлежит рассмотрению в плане ограничения методов, представленных и описанных, в целом, в этом изобретении. В целях объяснения это изобретение описывает видеodeкодер 20 в контексте кодирования HEVC. Однако

методы этого изобретения можно применять к другим стандартам или способам кодирования.

В примере, показанном на фиг. 2, видеокодер 20 включает в себя память 98 видеоданных, блок 100 обработки предсказания, блок 102 генерации остатка, блок 104 обработки преобразования, блок 106 квантования, блок 108 обратного квантования, блок 110 обработки обратного преобразования, блок 112 реконструкции, блок 114 фильтрации, буфер 116 декодированных изображений и блок 118 энтропийного кодирования. Блок 100 обработки предсказания включает в себя блок 120 обработки внешнего предсказания и блок 126 обработки внутреннего предсказания. Блок 120 обработки внешнего предсказания включает в себя блок оценивания движения и блок компенсации движения (не показан). Видеокодер 20 также включает в себя блок 122 кодирования на основе палитры, выполненный с возможностью осуществления различных аспектов методов кодирования на основе палитры, описанных в этом изобретении. В других примерах видеокодер 20 может включать в себя больше, меньше структурных компонентов или другие структурные компоненты.

В памяти 98 видеоданных могут храниться видеоданные, подлежащие кодированию компонентами видеокодера 20. Видеоданные, хранящиеся в памяти 98 видеоданных, можно получать, например, из источника 18 видеосигнала, показанного на фиг. 1. Буфер 116 декодированных изображений может представлять собой память опорных изображений, где хранятся опорные видеоданные для использования при кодировании видеоданных видеокодером 20, например в режимах внутреннего или внешнего кодирования. Память 98 видеоданных и буфер 116 декодированных изображений могут быть образованы любым из различных запоминающих устройств, например динамической оперативной памятью (DRAM), включающей в себя синхронную DRAM (SDRAM), магниторезистивную RAM (MRAM), резистивную RAM (RRAM) или другие типы запоминающих устройств. Память 98 видеоданных и буфер 116 декодированных изображений может обеспечиваться одним и тем же запоминающим устройством или отдельными запоминающим устройствами. В различных примерах память 98 видеоданных может располагаться на одном кристалле с другими компонентами видеокодера 20 или вне кристалла относительно этих компонентов.

Видеокодер 20 может принимать видеоданные. Видеокодер 20 может кодировать каждая CTU в слайсе изображения видеоданных. Каждая из CTU может быть связана с блоками яркости дерева кодирования (CTB) одинакового размера и соответствующими CTB изображения. Как часть кодирования CTU блок 100 обработки предсказания может осуществлять разбиение квадрадерева для разделения CTB CTU на блоки все меньшего размера. Меньший блок может быть блоками кодирования CU. Например, блок 100 обработки предсказания может разбивать CTB, связанный с CTU, на четыре подблока одинакового размера, разбивая один или более из подблоков на четыре подподблока одинакового размера и т.д.

Видеокодер 20 может кодировать CU CTU для генерации кодированных представлений CU (например, кодированных CU). Как часть кодирования CU блок 100 обработки предсказания может разбивать блоки кодирования, связанные с CU среди одной или более PU из CU. Таким образом, каждая PU может быть связана с блоком предсказания яркости и соответствующими блоками предсказания цветности. Видеокодер 20 и видеодекодер 30 могут поддерживать PU, имеющие различные размеры. Как указано выше, размер CU может быть связан с размером блока кодирования яркости CU и размер PU может быть связан с размером блока предсказания яркости PU. Исходя из того, что размер конкретного CU равен $2N \times 2N$, видеокодер 20 и видеодекодер 30 могут поддерживать размеры PU $2N \times 2N$ или $N \times N$ для внутреннего предсказания и симметричные размеры PU $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$ или аналогичные для внешнего предсказания. Видеокодер 20 и видеодекодер 30 также могут поддерживать асимметричное разбиение для размеров PU $2N \times nU$, $2N \times nD$, $nL \times 2N$ и $nR \times 2N$ для внешнего предсказания.

Блок 120 обработки внешнего предсказания может генерировать предсказательные данные для PU, осуществляя внешнее предсказание на каждой PU CU. Предсказательные данные для PU могут включать в себя один или более блоков предсказательных выборок PU и информацию движения для PU. Блок 120 обработки внешнего предсказания может осуществлять разные операции для PU CU в зависимости от того, является ли PU I-слайсом, P-слайсом или B-слайсом. В I-слайсе все PU внутренне предсказываются. Поэтому, если PU находится в I-слайсе, блок 120 обработки внешнего предсказания не осуществляет внешнее предсказание на PU. Таким образом, для блоков, кодированных в I-режиме, предсказательный блок формируется с использованием пространственного предсказания из ранее кодированных соседних блоков в одном и том же кадре.

Если PU находится в P-слайсе, блок оценивания движения блока 120 обработки внешнего предсказания может искать опорные изображения в списке опорных изображений (например, "RefPicList0") для опорной области для PU. Опорная область для PU может быть областью, в опорном изображении, которое содержит блоки выборок, которые наиболее точно соответствуют блокам выборок PU. Блок оценивания движения может генерировать опорный индекс, который указывает позицию в RefPicList0 опорного изображения, содержащего опорную область для PU. Кроме того, блок оценивания движения может генерировать MV, который указывает пространственное перемещение между блоком кодирования PU и опорным положением, связанным с опорной областью. Например, MV может быть двухмерным вектором, который обеспечивает смещение от координат в текущем декодированном изображении к координатам в опорном изображении. Блок оценивания движения может выводить опорный индекс и вектор

движения (MV) в качестве информации движения PU. Блок 120 компенсации движения блока обработки внешнего предсказания может генерировать блоки предсказательных выборок PU на основании фактических или интерполированных выборок в опорном положении, указанном MV PU.

Если PU находится в В-слайсе, блок оценивания движения может осуществлять одинарное предсказание или двойное предсказание для PU. Для осуществления одинарного предсказания для PU, блок оценивания движения может искать опорные изображения RefPicList0 или второй список опорных изображений ("RefPicList1") для опорной области для PU. Блок оценивания движения может выводить, в качестве информации движения PU, опорный индекс, который указывает позицию в RefPicList0 или RefPicList1 опорного изображения, который содержит опорную область, MV, который указывает пространственное перемещение между блоком выборок PU и опорным положением, связанным с опорной областью, и один или более указателей направления предсказания, которые указывают, находится ли опорное изображение в RefPicList0 или RefPicList1. Блок 120 компенсации движения блока обработки внешнего предсказания может генерировать блоки предсказательных выборок PU, по меньшей мере, частично на основании фактических (т.е. целочисленной точности) или интерполированных (т.е. дробной точности) выборок в опорной области, указанной вектором движения PU.

Для осуществления двунаправленного внешнего предсказания для PU, блок оценивания движения может искать опорные изображения в RefPicList0 для опорной области для PU и также может искать опорные изображения в RefPicList1 для другой опорной области для PU. Блок оценивания движения может генерировать индексы опорного изображения, которые указывают позиции в RefPicList0 и RefPicList1 опорных изображений, которые содержат опорные области. Кроме того, блок оценивания движения может генерировать MV, которые указывают пространственные перемещения между опорным положением, связанным с опорными областями, и блоком выборок PU. Информация движения PU может включать в себя опорные индексы и MV PU. Блок компенсации движения может генерировать блоки предсказательных выборок PU, по меньшей мере, частично на основании фактических или интерполированных выборок в опорной области, указанной вектором движения PU.

В соответствии с различными примерами этого изобретения видеокодер 20 может быть выполнен с возможностью осуществления кодирования на основе палитры. В отношении инфраструктуры HEVC, в порядке примера, методы кодирования на основе палитры могут быть сконфигурированы для использования в качестве режима CU. В других примерах методы кодирования на основе палитры могут быть сконфигурированы для использования в качестве режима PU в инфраструктуре HEVC. Соответственно все описанные здесь процессы (в этом изобретении) в контексте режима CU могут, дополнительно или альтернативно, применяться к режиму PU. Однако эти примеры на основе HEVC не следует рассматривать как ограничение описанных здесь методов кодирования на основе палитры, что позволяет применять методы работы независимо или как часть других существующих или еще не разработанных систем/стандартов. В этих случаях единицей для кодирования палитры могут быть квадратные блоки, прямоугольные блоки или даже области непрямоугольной формы.

Например, блок 122 кодирования на основе палитры, может осуществлять кодирование на основе палитры, при выборе режима кодирования на основе палитры, например, для CU или PU. Например, блок 122 кодирования на основе палитры может быть выполнен с возможностью генерации палитры, имеющей записи, указывающие пиксельные значения, выбора пиксельных значений в палитре для представления пиксельных значений, по меньшей мере, некоторых позиций блока видеоданных, и сигнализации информации, связывающей, по меньшей мере, некоторые из позиций блока видеоданных с записями в палитре, соответствующими, соответственно, выбранным пиксельным значениям. Хотя различные функции описаны как осуществляемые блоком 122 кодирования на основе палитры, некоторые или все из таких функций могут осуществляться другими блоками обработки или комбинацией разных блоков обработки.

Блок 122 кодирования на основе палитры может быть выполнен с возможностью генерации любого из различных описанных здесь элементов синтаксиса, связанных с кодированием на основе палитры. Соответственно видеокодер 20 может быть выполнен с возможностью кодирования блоков видеоданных с использованием режимов кодирования на основе палитры, описанных в этом изобретении. Видеокодер 20 может выборочно кодировать блок видеоданных с использованием режима кодирования палитры или кодировать блок видеоданных с использованием другого режима, например, такого HEVC с внешним предсказанием или режима кодирования с внутренним предсказанием. Блок видеоданных может представлять собой, например, CU или PU, генерируемую согласно процессу кодирования HEVC. Видеокодер 20 может кодировать некоторые блоки в режимах временного предсказания с внешним предсказанием или пространственного кодирования с внутренним предсказанием и декодировать другие блоки в режиме кодирования на основе палитры.

Блок 126 обработки внутреннего предсказания может генерировать предсказательные данные для PU, осуществляя внутреннее предсказание на PU. Предсказательные данные для PU могут включать в себя блоки предсказательных выборок для PU и различные элементы синтаксиса. Блок 126 обработки внутреннего предсказания может осуществлять внутреннее предсказание на PU в I-слайсах, P-слайсах и В-слайсах.

Для осуществления внутреннего предсказания на PU блок 126 обработки внутреннего предсказания может использовать множественные режимы внутреннего предсказания для генерации множественных наборов предсказательных данных для PU. При использовании некоторых режимов внутреннего предсказания для генерации набора предсказательных данных для PU блок 126 обработки внутреннего предсказания может расширять значения выборок от блоков выборок соседних PU по предсказательным блокам PU в направлениях, связанных с режимами внутреннего предсказания. Соседние PU могут располагаться выше, выше и справа, выше и слева или слева PU, предполагая порядок кодирования для PU, CU и STU слева направо и сверху вниз. Блок 126 обработки внутреннего предсказания может использовать любой из нескольких разных режимов внутреннего предсказания, например, 33 режимов направленного внутреннего предсказания. В некоторых примерах количество режимов внутреннего предсказания может зависеть от размера области, связанной с PU.

Блок 100 обработки предсказания может выбирать предсказательные данные для PU CU из предсказательных данных, генерируемых блоком 120 обработки внешнего предсказания для PU или предсказательных данных, генерируемых блоком 126 обработки внутреннего предсказания для PU. В некоторых примерах блок 100 обработки предсказания выбирает предсказательные данные для PU CU на основании метрик скорости/искажения наборов предсказательных данных. Блоки предсказательных выборок выбранных предсказательных данных могут именоваться здесь выбранными блоками предсказательных выборок.

Блок 102 генерации остатка может генерировать, на основании блоков кодирования (например, блоков кодирования яркости, Cb и Cr) CU и выбранных блоков предсказательных выборок (например, предсказательных блоков яркости, Cb и Cr) PU CU, остаточных блоков (например, остаточных блоков яркости, Cb и Cr) CU. Например, блок 102 генерации остатка может генерировать остаточные блоки CU таким образом, что каждая выборка в остаточных блоках имеет значение, равное разности между выборкой в блоке кодирования CU и соответствующей выборкой в соответствующем выбранном блоке предсказательных выборок PU CU.

Блок 104 обработки преобразования может осуществлять разбиение квадрадерева для разбиения остаточных блоков, связанных с CU, на блоки преобразования, связанные с TU CU. Таким образом, в некоторых примерах, TU может быть связана с блоком преобразования яркости и двумя блоками преобразования цветности. Размеры и позиции блоков преобразования яркости и цветности TU CU могут основываться или не основываться на размерах и позициях блоков предсказания PU CU. Структура квадрадерева, известная как "остаточное квадрадерево" (RQT), может включать в себя узлы, связанные с каждой из областей. TU CU могут соответствовать краевым узлам RQT.

Блок 104 обработки преобразования может генерировать блоки коэффициентов преобразования для каждой TU CU путем применения одного или более преобразований к блокам преобразования TU. Блок 104 обработки преобразования может применять различные преобразования к блоку преобразования, связанному с TU. Например, блок 104 обработки преобразования может применять дискретное косинусное преобразование (DCT), направленное преобразование или принципиально аналогичное преобразование к блоку преобразования. В некоторых примерах блок 104 обработки преобразования не применяет преобразования к блоку преобразования. В таких примерах блок преобразования можно рассматривать как блок преобразования коэффициентов.

Блок 106 квантования может квантовать коэффициенты преобразования в блоке коэффициентов. Процесс квантования может уменьшать битовую глубину, связанную с некоторыми или всеми из коэффициентов преобразования. Например, n-битовый коэффициент преобразования можно округлять в сторону уменьшения до m-битового коэффициента преобразования при квантовании, где n больше m. Блок 106 квантования может квантовать блок коэффициентов, связанный с TU CU на основании значения параметра квантования (QP), связанного с CU. Видеокодер 20 может регулировать степень квантования, применяемую к блокам коэффициентов, связанным с CU, путем регулировки значения QP, связанного с CU. Квантование может вносить потерю информации, из-за чего квантованные коэффициенты преобразования могут иметь более низкую точность, чем исходные.

Блок 108 обратного квантования и блок 110 обработки обратного преобразования может применять обратное квантование и обратные преобразования к блоку коэффициентов, соответственно, для реконструкции остаточного блока из блока коэффициентов. Блок 112 реконструкции может добавлять реконструированный остаточный блок к соответствующим выборкам из одного или более блоков предсказательных выборок, генерируемых блоком 100 обработки предсказания для создания реконструированного блока преобразования, связанного с TU. Реконструируя таким образом блоки преобразования для каждой TU CU, видеокодер 20 может реконструировать блоки кодирования CU.

Блок 114 фильтрации может осуществлять одну или более операций ликвидации блочности для уменьшения артефактов блочности в блоках кодирования, связанных с CU. В буфере 116 декодированных изображений могут сохраняться реконструированные блоки кодирования после осуществления блоком 114 фильтрации одной или более операций ликвидации блочности на реконструированных блоках кодирования. Блок 120 обработки внешнего предсказания может использовать опорное изображение, которое содержит реконструированные блоки кодирования для осуществления внешнего предсказания

PU других изображений. Кроме того, блок 126 обработки внутреннего предсказания может использовать реконструированные блоки кодирования в буфере 116 декодированных изображений для осуществления внутреннего предсказания на других PU в том же изображении, что и CU.

Блок 118 энтропийного кодирования может принимать данные из других функциональных компонентов видеокодера 20. Например, блок 118 энтропийного кодирования может принимать блоки коэффициентов от блока 106 квантования и может принимать элементы синтаксиса от блока 100 обработки предсказания. Блок 118 энтропийного кодирования может осуществлять одну или более операций энтропийного кодирования на данных для генерации энтропийно кодированных данных. Например, блок 118 энтропийного кодирования может осуществлять на данных операцию CABAC, операцию контекстно-адаптивного кодирования с переменной длиной серии (CAVLC), операцию кодирования от переменной длины к переменной длине (V2V), операцию контекстно-адаптивного двоичного арифметического кодирования на основе синтаксиса (SBAC), операцию энтропийного кодирования с разбиением интервала вероятности (PIPE), операцию экспоненциально-голомбовского кодирования или операцию энтропийного кодирования другого типа. Видеокодер 20 может выводить битовый поток, который включает в себя энтропийно кодированные данные, генерируемые блоком 118 энтропийного кодирования. Например, битовый поток может включать в себя данные, представляющие RQT для CU.

В некоторых примерах остаточное кодирование не осуществляется с кодированием палитры. Соответственно видеокодер 20 может не осуществлять преобразование или квантование при кодировании с использованием режима кодирования палитры. Кроме того, видеокодер 20 может энтропийно кодировать данные, генерируемые с использованием режима кодирования палитры, отдельно от остаточных данных.

Согласно одному или более из методов этого изобретения видеокодер 20 и, в частности, блок 122 кодирования на основе палитры может осуществлять кодирование на основе палитры видеоданных предсказанных видеоблоков. Как описано выше, палитра, генерируемая видеокодером 20, может явно кодироваться и отправляться на видеодекoder 30, предсказываться из предыдущих записей палитры, предсказываться из предыдущих пиксельных значений или получаться комбинированным способом.

В соответствии с одним или более методами этого изобретения блок 122 кодирования на основе палитры может применять методы этого изобретения для осуществления преобразования значения выборки в индекс для кодирования видеоданных с использованием одного или более режимов кодирования палитры, причем режимы кодирования палитры не включают в себя режим совместного использования палитры. Методы этого изобретения включают в себя блок 122 кодирования на основе палитры видеокодера 20, выполненный с возможностью определения первого бина первого элемента синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре. Блок 122 кодирования на основе палитры видеокодера 20 может быть дополнительно выполнен с возможностью кодирования битового потока. Битовый поток может включать в себя первый элемент синтаксиса. Битовый поток также может не включать в себя второй элемент синтаксиса, который указывает режим совместного использования палитры. В некоторых примерах определение первого бина первого элемента синтаксиса содержит определение первого бина первого элемента синтаксиса с использованием контекстно-адаптивного двоичного арифметического кодирования. В других примерах определение первого бина первого элемента синтаксиса содержит определение первого бина первого элемента синтаксиса с использованием одного или более контекстов. В некоторых примерах использования одного или более контекстов один или более контекстов могут базироваться на по меньшей мере одном из предсказанного количества записей кодирования палитры и размера блока.

Кроме того, методы этого изобретения включают в себя блок 122 кодирования на основе палитры видеокодера 20, выполненный с возможностью определения, что текущий пиксель является первым пикселем в линии в порядке сканирования. Блок 122 кодирования на основе палитры видеокодера 20 может дополнительно определять наличие соседнего пикселя, расположенного над текущим пикселем. В случае определения, что текущий пиксель является первым пикселем в линии в порядке сканирования, и определения наличия соседнего пикселя, расположенного над текущим пикселем, блок 122 кодирования на основе палитры видеокодера 20 может быть дополнительно выполнен с возможностью обходить кодирование первого элемента синтаксиса в битовом потоке, причем первый элемент синтаксиса указывает тип серии, и кодировать оставшуюся часть битового потока.

Кроме того, методы этого изобретения включают в себя блок 122 кодирования на основе палитры видеокодера 20, выполненный с возможностью определения первого элемента синтаксиса, который указывает максимально допустимый размер палитры и имеет минимальное значение нуль. Блок 122 кодирования на основе палитры видеокодера 20 также может быть выполнен с возможностью кодирования битового потока, который включает в себя первый элемент синтаксиса. В некоторых примерах битовый поток дополнительно включает в себя второй элемент синтаксиса, который указывает максимальный размер палитры предсказателя и имеет минимальное значение нуль. В некоторых примерах первый элемент синтаксиса имеет максимальное значение 4096 и второй элемент синтаксиса имеет максимальное значение 8192. В других примерах первый элемент синтаксиса имеет максимальное значение 4095 и второй элемент синтаксиса имеет максимальное значение 4095. В других примерах первый элемент синтаксиса имеет максимальное значение 4095 и второй элемент синтаксиса имеет максимальное значение

8191. В прочих примерах первый элемент синтаксиса имеет максимальное значение, равное количеству пикселей в наибольшей единице кодирования, и второй элемент синтаксиса имеет максимальное значение, равное положительной постоянной, например 2, умноженной на максимальное значение первого элемента синтаксиса. В других примерах битовый поток включает в себя другой элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре. В некоторых примерах этого этот элемент синтаксиса представляется одним из кода Голомба-Райса, экспоненциального кода Голомба, усеченного кода Райса или унарного кода. В других примерах этого этот элемент синтаксиса представляется одним из усеченного кода Голомба-Райса, усеченного экспоненциального кода Голомба, усеченного кода Райса, усеченного унарного кода или кода, который также используется для кодирования третьего элемента синтаксиса, включенного в кодированный битовый поток, который указывает, копируется ли индекс палитры из индекса палитры в строке над текущим пикселем или явно кодируется в кодированном битовом потоке. В некоторых примерах этот элемент синтаксиса представляется режимом усеченного Райса. В некоторых примерах элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре, имеет максимальное значение, равное количеству пикселей в текущем блоке видеоданных.

На фиг. 3 показана блок-схема, демонстрирующая пример видеodeкодера 30, который выполнен с возможностью реализации методов этого изобретения. Videodecoder 30 может действовать обратно videodecoderу 20, описанному со ссылкой на фиг. 2. Фиг. 3 обеспечена в целях объяснения, но не ограничения методов, в целом, представленных и описанных в этом изобретении. В целях объяснения это изобретение описывает videodecoder 30 в контексте кодирования HEVC. Однако методы этого изобретения можно применять к другим стандартам или способам кодирования, где используется кодирование в режиме палитры.

В примере, показанном на фиг. 3, videodecoder 30 включает в себя память 148 видеоданных, блок 150 энтропийного декодирования, блок 152 обработки предсказания, блок 154 обратного квантования, блок 156 обработки обратного преобразования, блок 158 реконструкции, блок 160 фильтрации и буфер 162 декодированных изображений. Блок 152 обработки предсказания включает в себя блок 164 компенсации движения и блок 166 обработки внутреннего предсказания. Videodecoder 30 также включает в себя блок 165 декодирования на основе палитры, выполненный с возможностью осуществления различных аспектов методов кодирования на основе палитры, описанных в этом изобретении. В других примерах videodecoder 30 может включать в себя больше, меньше структурных компонентов или другие структурные компоненты.

В памяти 148 видеоданных могут храниться видеоданные, например битовый поток кодированного видеосигнала, подлежащие декодированию компонентами videodecoderа 30. Videоданные, хранящиеся в памяти 148 видеоданных, можно получать, например, из канала 16, например из локального источника видеосигнала, например камеры, посредством проводной или беспроводной сетевой передачи видеоданных или посредством осуществления доступа к физическим носителям данных. Память 148 видеоданных может образовывать буфер кодированных изображений (CPB), где хранятся данные кодированного видеосигнала из битового потока кодированного видеосигнала. Буфер 162 декодированных изображений может представлять собой память опорных изображений, где хранятся опорные видеоданные для использования при декодировании видеоданных videodecoderом 30, например, в режимах внутреннего или внешнего кодирования. Память 148 видеоданных и буфер 162 декодированных изображений могут быть образованы любым из различных запоминающих устройств, например динамической оперативной памятью (DRAM), включающей в себя синхронную DRAM (SDRAM), магниторезистивную RAM (MRAM), резистивную RAM (RRAM) или другие типы запоминающих устройств. Память 148 видеоданных и буфер 162 декодированных изображений может обеспечиваться одним и тем же запоминающим устройством или отдельными запоминающим устройствами. В различных примерах память 148 видеоданных может располагаться на одном кристалле с другими компонентами videodecoderа 30 или вне кристалла относительно этих компонентов.

Память 148 видеоданных, например CPB, может принимать и сохранять данные кодированного видеосигнала (например, единицы NAL) битового потока. Блок 150 энтропийного декодирования может принимать данные кодированного видеосигнала (например, единицы NAL) из памяти 148 видеоданных и может разбирать единицы NAL для декодирования элементов синтаксиса. Блок 150 энтропийного декодирования может энтропийно декодировать энтропийно кодированные элементы синтаксиса в единицах NAL. Блок 152 обработки предсказания, блок 154 обратного квантования, блок 156 обработки обратного преобразования, блок 158 реконструкции и блок 160 фильтрации могут генерировать декодированные видеоданные на основании элементов синтаксиса, полученных (например, извлеченных) из битового потока.

Единицы NAL битового потока могут включать в себя единицы NAL кодированного слайса. В порядке декодирования битового потока блок 150 энтропийного декодирования может извлекать и энтропийно декодировать элементы синтаксиса из единиц NAL кодированного слайса. Каждый из кодированных слайсов может включать в себя заголовок слайса и данные слайса. Заголовок слайса может содержать элементы синтаксиса, относящиеся к слайсу. Элементы синтаксиса в заголовке слайса могут вклю-

чать в себя элемент синтаксиса, который идентифицирует PPS, связанный с изображением, которое содержит слайс.

Помимо декодирования элементов синтаксиса из битового потока видеodeкодер 30 может осуществлять операцию реконструкции на CU, не подвергнутой разбиению. Для осуществления операции реконструкции на CU, не подвергнутой разбиению, видеodeкодер 30 может осуществлять операцию реконструкции на каждой TU CU. Осуществляя операцию реконструкции для каждой TU CU, видеodeкодер 30 может реконструировать остаточные блоки CU.

В порядке осуществления операции реконструкции на TU CU блок 154 обратного квантования может обратно квантовать, например деквантовать, блоки коэффициентов, связанные с TU. Блок 154 обратного квантования может использовать значение QP, связанное с CU TU для определения степени квантования и, аналогично, степени обратного квантования для блока 154 обратного квантования для применения. Таким образом, коэффициентом сжатия, например отношением количества битов, используемых для представления исходной последовательности и сжатой последовательности, можно управлять путем регулировки значений QP, используемого при квантовании коэффициентов преобразования. Коэффициент сжатия также может зависеть от используемого способа энтропийного кодирования.

После обратного квантования блока коэффициентов блоком 154 обратного квантования блок 156 обработки обратного преобразования может применять одно или более обратных преобразований к блоку коэффициентов для генерации остаточного блока, связанного с TU. Например, блок 156 обработки обратного преобразования может применять к блоку коэффициентов обратное DCT, обратное целочисленное преобразование, обратное преобразование Карунена-Лева (KLT), обратное вращательное преобразование, обратное направленное преобразование или другое обратное преобразование.

Если PU кодируется с использованием внутреннего предсказания, блок 166 обработки внутреннего предсказания может осуществлять внутреннее предсказание для генерации предсказательных блоков для PU. Блок 166 обработки внутреннего предсказания может использовать режим внутреннего предсказания для генерации предсказательных блоков яркости, Cb и Cr для PU на основании блоков предсказания соседствующих в пространстве PU. Блок 166 обработки внутреннего предсказания может определять режим внутреннего предсказания для PU на основании одного или более элементов синтаксиса, декодированных из битового потока.

Блок 152 обработки предсказания может строить первый список опорных изображений (RefPicList0) и второй список опорных изображений (RefPicList1) на основании элементов синтаксиса, извлеченных из битового потока. Если же PU кодируется с использованием внешнего предсказания, блок 150 энтропийного декодирования может извлекать информацию движения для PU. Блок 164 компенсации движения может определять на основании информации движения PU одну или более опорных областей для PU. Блок 164 компенсации движения может генерировать, на основании блоков выборки на одном или более опорных блоков для PU предсказательные блоки (например, предсказательные блоки яркости, Cb и Cr) для PU.

Блок 158 реконструкции может использовать блоки преобразования (например, блоки преобразования яркости, Cb и Cr), связанные с TU CU, и предсказательные блоки (например, блоки яркости, Cb и Cr) PU CU, например, либо данные внутреннего предсказания, либо данные внешнего предсказания, в зависимости от того, что применимо, для реконструкции блоков кодирования (например, блоков кодирования яркости, Cb и Cr) CU. Например, блок 158 реконструкции может добавлять выборки блоков преобразования (например, блоков преобразования яркости, Cb и Cr) к соответствующим выборкам предсказательных блоков (например, предсказательных блоков яркости, Cb и Cr) для реконструкции блоков кодирования (например, блоков кодирования яркости, Cb и Cr) CU.

Блок 160 фильтрации может осуществлять операцию ликвидации блочности для уменьшения артефактов блочности, связанных с блоками кодирования (например, блоками кодирования яркости, Cb и Cr) CU. Видеodeкодер 30 может сохранять блоки кодирования (например, блоки кодирования яркости, Cb и Cr) CU в буфере 162 декодированных изображений. Буфер 162 декодированных изображений может обеспечивать опорные изображения для последующих компенсации движения, внутреннего предсказания и представления на устройстве отображения, например, устройстве 32 отображения, показанном на фиг. 1. Например, видеodeкодер 30 может осуществлять, на основании блоков (например, блоков яркости, Cb и Cr) в буфере 162 декодированных изображений, операции внутреннего предсказания или внешнего предсказания на PU других CU. Таким образом, видеodeкодер 30 может извлекать из битового потока уровни коэффициентов преобразования значительного блока коэффициентов, обратно квантовать уровни коэффициентов преобразования, применять преобразование к уровням коэффициентов преобразования для генерации блока преобразования, генерировать, по меньшей мере, частично на основании блока преобразования, блок кодирования, и выводить блок кодирования для отображения.

В соответствии с различными примерами этого изобретения видеodeкодер 30 может быть выполнен с возможностью осуществления кодирования на основе палитры. Блок 165 декодирования на основе палитры, например, может осуществлять декодирование на основе палитры, когда выбран режим декодирования на основе палитры, например, для CU или PU. Например, блок 165 декодирования на основе палитры может быть выполнен с возможностью генерации палитры, имеющей записи, указывающие

пиксельные значения. Кроме того, в этом примере блок 165 декодирования на основе палитры может принимать информацию, связывающую, по меньшей мере, некоторые позиции блока видеоданных с записями в палитре. В этом примере блок 165 декодирования на основе палитры может выбирать пиксельные значения в палитре на основании информации. Дополнительно в этом примере блок 165 декодирования на основе палитры может реконструировать пиксельные значения блока на основании выбранных пиксельных значений. Хотя различные функции описаны как осуществляемые блоком 165 декодирования на основе палитры, некоторые или все из таких функций могут осуществляться другими блоками обработки или комбинацией разных блоков обработки.

В соответствии с одним или более методами этого изобретения блок 165 декодирования на основе палитры может принимать информацию режима кодирования палитры и осуществлять вышеупомянутые операции, когда информация режима кодирования палитры указывает, что режим кодирования палитры применяется к блоку. Когда информация режима кодирования палитры указывает, что режим кодирования палитры не применяется к блоку, или когда другая информация режима указывает использование другого режима, блок 165 декодирования на основе палитры декодирует блок видеоданных с использованием режима кодирования не на основе палитры, например такого HEVC с внешним предсказанием или режима кодирования с внутренним предсказанием, когда информация режима кодирования палитры указывает, что режим кодирования палитры не применяется к блоку. Блок видеоданных может представлять собой, например, CU или PU, генерируемую согласно процессу кодирования HEVC. Видеодекодер 30 может декодировать некоторые блоки в режимах временного предсказания с внешним предсказанием или пространственного кодирования с внутренним предсказанием и декодировать другие блоки в режиме кодирования на основе палитры. Режим кодирования на основе палитры может содержать один из множества различных режимов кодирования на основе палитры, или может существовать единственный режим кодирования на основе палитры.

Согласно одному или более из методов этого изобретения видеодекодер 30 и, в частности, блок 165 декодирования на основе палитры может осуществлять декодирование на основе палитры видеоблоков, кодированных на основе палитры. Как описано выше, палитра, декодированная видеодекодером 30, может явно кодироваться и сигнализироваться видеокодером 20, реконструироваться видеодекодером 30 относительно принятого блока, кодированного на основе палитры, предсказываться из предыдущих записей палитры, предсказываться из предыдущих пиксельных значений или получаться комбинированным способом.

Блок 165 декодирования на основе палитры может применять методы этого изобретения для осуществления преобразования значения выборки в индекс, декодировать видеоданные с использованием одного или более режимов кодирования палитры, причем режимы кодирования палитры не включают в себя режим совместного использования палитры. Кроме того, методы этого изобретения включают в себя блок 165 декодирования на основе палитры видеодекодера 30, выполненный с возможностью приема кодированного битового потока. В этом примере, Кодированный битовый поток не включает в себя первый элемент синтаксиса, который указывает режим совместного использования палитры. Кроме того, кодированный битовый поток включает в себя второй элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре. Блок 165 декодирования на основе палитры видеодекодера 30 может быть дополнительно выполнен с возможностью декодирования первого бина второго элемента синтаксиса. В некоторых примерах декодирование первого бина второго элемента синтаксиса содержит декодирование первого бина второго элемента синтаксиса с использованием блока контекстно-адаптивного двоичного арифметического кодирования (CABAC). В других примерах декодирование первого бина второго элемента синтаксиса содержит декодирование первого бина второго элемента синтаксиса с использованием одного или более контекстов. В некоторых примерах использования одного или более контекстов, один или более контекстов могут базироваться на, по меньшей мере, одном из предсказанного количества записей кодирования палитры и размера блока.

Кроме того, методы этого изобретения включают в себя блок 165 декодирования на основе палитры видеодекодера 30, выполненный с возможностью приема кодированного битового потока. Кодированный битовый поток может включать в себя первый элемент синтаксиса, который указывает тип серии. Блок 165 декодирования на основе палитры видеодекодера 30 может быть дополнительно выполнен с возможностью определения, что текущий пиксель является первым пикселем в линии в порядке сканирования. Блок 165 декодирования на основе палитры видеодекодера 30 может дополнительно определять наличие соседнего пикселя, расположенного над текущим пикселем. В случае определения, что текущий пиксель является первым пикселем в линии в порядке сканирования, и определения наличия соседнего пикселя, расположенного над текущим пикселем, блок 165 декодирования на основе палитры видеодекодера 30 может обходить декодирование первого элемента синтаксиса.

Кроме того, методы этого изобретения включают в себя блок 165 декодирования на основе палитры видеодекодера 30, выполненный с возможностью приема кодированного битового потока, который включает в себя первый элемент синтаксиса, который указывает максимально допустимый размер палитры и имеет минимальное значение нуля. Блок 165 декодирования на основе палитры видеодекодера 30 может быть дополнительно выполнен с возможностью декодирования кодированного битового пото-

ка. В некоторых примерах, кодированный битовый поток дополнительно включает в себя второй элемент синтаксиса, который указывает максимальный размер палитры предсказателя и имеет минимальное значение нуль. В некоторых примерах первый элемент синтаксиса имеет максимальное значение 4096 и второй элемент синтаксиса имеет максимальное значение 8192. В других примерах первый элемент синтаксиса имеет максимальное значение 4095 и второй элемент синтаксиса имеет максимальное значение 4095. В других примерах первый элемент синтаксиса имеет максимальное значение 4095 и второй элемент синтаксиса имеет максимальное значение 8191. В прочих примерах первый элемент синтаксиса имеет максимальное значение, равное количеству пикселей в наибольшей единице кодирования, и второй элемент синтаксиса имеет максимальное значение, равное положительной постоянной, например 2, умноженной на максимальное значение первого элемента синтаксиса. В других примерах кодированный битовый поток включает в себя другой элемент синтаксиса, например третий элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре. В некоторых примерах этого изобретения элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре, представляется одним из кода Голомба-Райса, экспоненциального кода Голомба, усеченного кода Райса или унарного кода. В других примерах этого изобретения элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре, представляется одним из усеченного кода Голомба-Райса, усеченного экспоненциального кода Голомба, усеченного кода Райса, усеченного унарного кода или кода, который также используется для кодирования третьего элемента синтаксиса, включенного в кодированный битовый поток, который указывает, копируется ли индекс палитры из индекса палитры в строке над текущим пикселем или явно кодируется в кодированном битовом потоке. В некоторых примерах элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре, представляется режимом усеченного Райса. В некоторых примерах элемент синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре, имеет максимальное значение, равное количеству пикселей в текущем блоке видеоданных.

Следует понимать, что в зависимости от примера, некоторые действия или события любого из описанных здесь методов могут осуществляться в разной последовательности, могут добавляться, объединяться или вовсе исключаться (например, не все описанные действия или события необходимы для практического применения методов). Кроме того, в некоторых примерах действия или события могут осуществляться одновременно, например при многопоточной обработке, обработке прерываний или при использовании нескольких процессоров, а не последовательно. Кроме того, хотя некоторые аспекты этого изобретения описаны для простоты как осуществляемые единственным модулем или блоком, следует понимать, что методы этого изобретения могут осуществляться комбинацией блоков или модулей, связанных с видеокодером.

Некоторые аспекты этого изобретения в целях иллюстрации были описаны в отношении разрабатываемого стандарта HEVC. Однако методы, описанные в этом изобретении, могут быть полезны для других процессов кодирования видеоданных, в том числе, других стандартных или еще не разработанных авторских процессов кодирования видеоданных.

Вышеописанные методы могут осуществляться видеокодером 20 (фиг. 1 и 2) и/или видеодекодером 30 (фиг. 1 и 3), которые оба могут, в целом, именоваться видеокодером. Аналогично, кодирование видеоданных может относиться к кодированию видеосигнала или декодированию видеосигнала, в зависимости от того, что применимо.

В некоторых примерах методы кодирования на основе палитры можно конфигурировать для использования в одном или более режимах кодирования стандарта HEVC или стандарта SCC HEVC. В других примерах методы кодирования на основе палитры можно использовать независимо или как часть других существующих или будущих систем или стандартов. В некоторых примерах методы кодирования на основе палитры видеоданных можно использовать с одним или более другими методами кодирования, например методами кодирования с внешним предсказанием или кодирования видеоданных с внутренним предсказанием. Например, как более подробно описано ниже, кодер или декодер, или объединенный кодер-декодер (кодек), может быть выполнен с возможностью осуществления кодирования с внешним и с внутренним предсказанием, а также кодирования на основе палитры.

В отношении инфраструктуры HEVC, в порядке примера, методы кодирования на основе палитры могут быть сконфигурированы для использования в режиме единиц кодирования (CU). В других примерах методы кодирования на основе палитры могут быть сконфигурированы для использования в режиме единиц предсказания (PU) в инфраструктуре HEVC. Соответственно все раскрытые ниже процессы, описанные в контексте режима CU могут, дополнительно или альтернативно, применяться к PU. Однако эти примеры на основе HEVC не следует рассматривать как ограничение описанных здесь методов кодирования на основе палитры, что позволяет применять методы работы независимо или как часть других существующих или еще не разработанных систем/стандартов. В этих случаях единицей для кодирования палитры могут быть квадратные блоки, прямоугольные блоки или даже области непрямоугольной формы.

Основная идея кодирования на основе палитры состоит в том, что для каждой CU выводится палитра, которая содержит (и может состоять из) наиболее преобладающие пиксельные значения в текущей

CU. Сначала размер и элементы палитры передаются от видеокодера на декодер. Размер и/или элементы палитры могут непосредственно кодироваться или кодироваться с предсказанием с использованием размера и/или элементов палитры в соседних CU (например, выше и/или левее кодированной CU). После этого пиксельные значения в CU кодируются на основании палитры согласно некоторому порядку сканирования. Для каждого пикселя сначала передается положение в CU, флаг, например `palette_flag`, для указания, включено ли пиксельное значение в палитру. В некоторых примерах такой флаг называется `copy_above_palette_indices_flag`. Для пиксельных значений, которые отображаются в запись в палитре, индекс палитры, связанный с этой записью, сигнализируется для положения данного пикселя в CU. Для пиксельных значений, не существующих в палитре, пикселю может назначаться особый индекс, и фактическое пиксельное значение (в ряде случаев, квантованное пиксельное значение) передается для положения данного пикселя в CU. Эти пиксели именуются "выходными пикселями". Выходной пиксель может кодироваться с использованием любого существующего способа энтропийного кодирования, например кодирования с фиксированной длиной серии, унарного кодирования и т.д.

В других примерах для явного указания, является ли пиксель "выходным" пикселем, не используется никакого флага. Вместо этого флаг или другой элемент синтаксиса можно использовать для указания типа серии. Элемент синтаксиса, указывающий тип серии, может указывать, копируются ли следующие индексы из позиции над текущим пикселем, или существует ли серия сигнализируемых значений индекса. Если выведенное значение индекса конкретного пикселя соответствует "выходному индексу" (например, заранее определенному индексу в палитре, указывающему использование выходного пикселя), то декодер 30 может определять, что такой пиксель является выходным пикселем.

Для повышения эффективности кодирования содержания экрана было предложено несколько способов, расширяющих режим палитры. Например, такие способы можно найти в JCTVC-S0114 (Kim, J., et al., "CE6-related: Enabling copy above mode prediction at the boundary of CU," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 19th Meeting: Strasbourg, FR, 17-24 October 2014); JCTVC-S0120 (Ye, J., et al., "Non-CE6: Copy previous mode," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 19th Meeting: Strasbourg, FR, 17-24 October 2014); and JCTVC-S0151 (Wang, W., et al., "Non-CE6: 2-D Index Map Coding of Palette Mode in HEVC SCC" Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 19th Meeting: Strasbourg, FR, 17-24 October 2014).

В документе X. Guo and A. Saxena, "RCE4: Summary report of HEVC Range Extension Core Experiments 4 (RCE4) on palette coding for screen content," JCTVC-P0035, San Jose, US, 9-17 Jan. 2014 описано два результата испытаний режимов на основе палитры, которые были сообщены для достижения значительного снижения искажения скорости по Бьентегору (BD-rate), в особенности для содержимого экрана. Ниже кратко описаны два способа.

В одном иллюстративном способе, описанном, например, в документе X. Guo, Y. Lu and S. Li, "RCE4: Test 1. Major-color-based screen content coding," JCTVC-P0108, San Jose, US, 9-17 Jan. 2014, для классификации пикселей используется алгоритм на основе гистограммы. В частности, N наиболее значительных пиковых значений в гистограмме выбираются в качестве основных цветов для кодирования. Пиксельные значения, близкие к основному цвету, будут квантоваться к основному цвету. Другие пиксели, не принадлежащие никакому набору основных цветов, являются выходными пикселями, которые также квантуются до кодирования. Для беспотерного кодирования квантование не используется.

Классификация позволяет преобразовывать пиксели единицы кодирования (CU) в индексы цвета. После этого количество и значения основных цветов кодируются. Затем индексы цвета кодируются следующим образом: для каждой линии пикселей сигнализируется флаг, указывающий режим кодирования. Существует три режима: горизонтальный режим, вертикальный режим и нормальный режим.

Если режим является горизонтальным режимом, вся линия (т.е. все пиксели в линии) совместно использует один и тот же индекс цвета. В этом случае передается индекс цвета.

Если режим является вертикальным режимом, вся линия идентична вышележащей линии. В этом случае ничего не передается. Текущая линия копирует индексы цвета вышележащей линии.

Если режим является нормальным режимом, сигнализируется флаг для позиции каждого пикселя для указания, идентичен ли он одному из левого и вышележащего пикселей. Если нет, передается сам индекс.

Кроме того, если пиксель является выходным пикселем, передается пиксельное значение.

В другом иллюстративном способе, описанном, например, в документе L. Guo, W. Pu, M. Karczewicz, J. Sole, R. Joshi, and F. Zou, "RCE4: Results of Test 2 on Palette Mode for Screen Content Coding," JCTVC-P0198, San Jose, US, 9-17 Jan. 2014, режим кодирования на основе палитры включен в качестве режима CU. Процесс кодирования второго способа может включать в себя следующие:

передачу палитры: схема предсказания по всем записям используется для кодирования текущей палитры на основании палитры левой CU (CU, соседствующей слева с кодируемой в данный момент CU). После этого передаются не предсказанные записи палитры.

передачу пиксельных значений: пиксели в CU кодируются в порядке растрового сканирования с использованием следующих трех режимов:

"режима серии": сначала сигнализируется индекс палитры, затем "palette_run" (M). Следующие M индексов палитры идентичны первому сигнализированному индексу палитры.

"режима копирования сверху": значение "copy_run" (N) передается для указания, что следующие N индексов палитры идентичны своим соответствующим соседям сверху.

"режима пикселя": сначала передается флаг предсказания. Значение флага, равное 1, указывает остаток предсказания с использованием реконструированного верхнего соседнего пикселя при передаче предсказателя. Если значение этого флага равно 0, пиксельное значение передается без предсказания.

Палитра может занимать довольно значительную часть битов для кодированного блока палитры (например, CU). Соответственно видеокодер может предсказывать одну или более записей палитры на основании одной или более записей ранее кодированной палитры (например, как упомянуто выше в отношении "передачи палитры").

В некоторых примерах видеокодер может генерировать список предсказателей палитры при прогнозировании записей палитры. Например, в документ C. Gisquet, G. Laroche, and P. Onno, "AhG10: Palette predictor stuffing," JCTVC-Q0063 раскрыт один иллюстративный процесс для определения предсказателей палитры. В некоторых примерах видеокодер может использовать логический вектор для указания, используется ли (или не используется) каждый элемент в списке предсказателей палитры для прогнозирования одной или более записей в палитре для кодируемого в данный момент блока.

В некоторых примерах все элементы в списке предсказателей палитры выводятся из ранее кодированной палитры (например, палитры, кодированной ранее кодированным блоком). Однако такие палитры могут быть сильно пространственно разнесены от текущей CU, из-за чего корреляция с палитрой оказывается довольно слабой. В общем случае расширение таблицы предсказателей палитры может быть полезно (например, может обеспечивать более точные предсказатели, что может приводить к выигрышу в эффективности). Однако определение и использование сравнительно большой таблицы предсказателей палитры приводит к относительному удлинению логического вектора.

В одном примере кодирования палитры видеокодер 20 может генерировать элемент синтаксиса, например флаг "PLT_Mode_flag", который указывает, используется ли режим кодирования на основе палитры для конкретной области кадра видео. Например, PLT_Mode_flag может генерироваться на уровне слайса, уровне CU, уровне PU или любом другом уровне кадра видео. Например, видеокодер 20 может генерировать PLT_Mode_flag на уровне CU и сигнализировать PLT_Mode_flag в битовом потоке кодированного видеосигнала. Затем, после декодирования битового потока кодированного видеосигнала, видеodeкодер 30 может разбирать PLT_Mode_flag. В этом примере значение этого PLT Mode flag, равное 1, указывает, что текущая CU кодируется с использованием режима палитры. В этом случае видеodeкодер 30 может применять режим кодирования на основе палитры для декодирования CU. В некоторых примерах элемент синтаксиса может указывать один из множества разных режимов палитры для CU.

Значение этого PLT_Mode_flag, равное 0, указывает, что текущая CU кодируется с использованием режима, отличного от режима палитры. Например, можно использовать любой из различных режимов кодирования с внешним предсказанием, с внутренним предсказанием или других. Когда значение PLT_Mode_flag равно 0, дополнительная информация может передаваться для сигнализации конкретного режима, используемого для кодирования соответствующей CU, причем такой конкретный режим обычно может быть режимом кодирования HEVC (например, внутреннего кодирования или внешнего кодирования). Использование PLT_Mode_flag описано в порядке примера. В других примерах другие элементы синтаксиса, например многобитовые коды, можно использовать для указания, подлежит ли использованию режим кодирования на основе палитры для CU (или PU в других примерах) или для указания, какие из множества режимов подлежат использованию.

PLT_Mode_flag или другой элемент синтаксиса также может передаваться на более высоком уровне. Например, PLT_Mode_flag может передаваться на уровне слайса. В этом случае значение флага, равное 1, означает, что все CU в слайсе будут кодироваться с использованием режима палитры (что означает отсутствие необходимости в передаче информации режима, например, для режима палитры или других режимов, на уровне CU). Аналогично, этот флаг может сигнализироваться на уровне набора параметров изображения (PPS), набора параметров последовательности (SPS) или набора параметров видео (VPS). Кроме того, на одном из этих уровней может отправляться флаг, указывающий, разрешен или запрещен режим палитры для конкретного изображения, слайса и т.д., тогда как PLT_Mode_flag указывает, используется ли режим кодирования на основе палитры для каждой CU. В этом случае, если флаг или другой элемент синтаксиса, отправленный на уровне слайса, PPS, SPS или VPS, указывает, что режим кодирования палитры запрещен, в некоторых примерах может не требоваться сигнализировать PLT_Mode_flag для каждой CU. Альтернативно, если флаг или другой элемент синтаксиса, отправленный на уровне слайса, PPS, SPS или VPS, указывает, что режим кодирования палитры разрешен, PLT Mode flag может дополнительно сигнализироваться для указания, подлежит ли использованию режим кодирования на основе палитры для каждой CU. Опять же, как упомянуто выше, применение этих методов для указания кодирования на основе палитры CU можно дополнительно или альтернативно использовать для указания кодирования на основе палитры PU.

Флаг, например PLT_Mode_flag, может дополнительно или альтернативно условно передаваться

или выводиться. Условия для передачи `PLT_Mode_flag` или вывода флага могут представлять собой, например, один или более из размера CU, типа кадра, цветового пространства, цветовой компоненты, размера кадра, частоты кадров, `id` уровня при кодировании масштабируемых видеоданных или `id` вида в многовидовом кодировании.

Теперь рассмотрим методы генерации и передачи палитры. Вideoкодер 20 может быть выполнен с возможностью генерации и сигнализации одного или более элементов синтаксиса и значений, которые видеодекодер 30 может использовать для построения и/или реконструкции палитры, используемой видеокодером 20 для кодирования конкретного уровня кадра видео (например, CU). В некоторых примерах видеокодер 20 может указывать или иначе сигнализировать палитру для каждой CU. В других примерах, видеокодер 20 может указывать или иначе сигнализировать палитру, которая может совместно использоваться несколькими CU.

Размер палитры, например, в отношении количества включенных пиксельных значений, может быть фиксированным значением или может сигнализироваться видеокодером 20 в битовом потоке кодированного видеосигнала. Видеодекодер 30 может принимать и декодировать указание размера палитры из битового потока кодированного видеосигнала. Сигнализация может различаться для разных компонент, или для всех компонент может сигнализироваться единый размер. Разные компоненты могут представлять собой, например, компоненты яркости и цветности. В сигнализации можно использовать унарные коды или усеченные унарные коды (например, усеченные на максимальном размере палитры). Также можно использовать экспоненциально-голомбовские коды или коды Райса-Голомба. В некоторых примерах сигнализация размера может осуществляться следующим образом: после сигнализации записи в палитре сигнализируется флаг "stop". Значение этого флага, равное 1, указывает, что текущая запись является последней в палитре. Значение этого флага, равное 0, указывает, что в палитре присутствуют и другие записи. Флаг "stop" может не передаваться кодером, если построенная на данный момент палитра достигала максимального размера палитры. В некоторых примерах размер палитры также может условно передаваться или выводиться на основании вспомогательной информации таким же образом, как описано выше для передачи флага `PLT_Mode_flag`.

Палитра может передаваться отдельно для каждой цветовой компоненты в CU. Например, может существовать палитра для компоненты Y этой CU, другая палитра для компоненты U этой CU и еще одна палитра для компоненты V этой CU. Для палитры Y запись (предположительно) может представлять значение Y в этой CU. То же самое справедливо для компонент U и V. Возможно также, что палитра передается для всех цветовых компонент в CU. В этом примере i-я запись в палитре представляет собой триплет (Y_i , U_i , V_i). В этом случае палитра включает в себя значения для каждой из компонент.

Предсказание палитры является альтернативным подходом к вышеописанной "передаче палитры". В некоторых примерах методы предсказания палитры можно использовать совместно с методами сигнализации палитры. Таким образом, видеокодер 20 может быть выполнен с возможностью сигнализации элементов синтаксиса, которые могут использоваться видеодекодером 30 для предсказания части суммарного количества записей палитры. Кроме того, видеокодер 20 может быть выполнен с возможностью явной сигнализации другой части записей палитры.

В одном примере подхода к предсказанию палитры для каждой CU передается один флаг "pred_palette_flag". Значение этого флага, равное 1, указывает, что палитра для текущей CU будет предсказана из прошлых данных и поэтому нет необходимости передавать палитру. Значение этого флага, равное 0, означает, что палитру текущей CU необходимо передавать. Флаг может различаться для разных цветовых компонент (например, таким образом, что в формате видео YUV необходимо передавать 3 флага для CU), или для всех цветовых компонент может сигнализироваться единый флаг. Например, единый флаг может указывать, передаются ли палитры для всех компонент, или будут ли предсказаны палитры для всех компонент.

В некоторых примерах предсказание может осуществляться следующим образом. Если значение флага предсказания равно 1, для текущей CU, видеокодер 20 копирует палитру одной или более из уже кодированных соседних CU. Палитра уже кодированных соседних CU может передаваться или предсказываться. Например, скопированная соседняя CU может быть соседней слева CU. В случае отсутствия палитры левой CU (как в случае, когда левая CU не кодируется с использованием режима палитры, или текущая CU находится в первом столбце изображения), палитра может копироваться из CU над текущей CU. Скопированная палитра также может быть комбинацией палитр нескольких соседних CU. Например, одну или более формул, функций, правил и т.п. можно применять для генерации палитры на основании палитр одной или комбинации множества соседних CU.

Возможно также, что можно построить список кандидатов, и индекс передается видеокодером 20 для указания CU-кандидата, из которой текущая CU копирует палитру. Видеодекодер 30 может строить один и тот же список кандидатов и затем использовать индекс для выбора палитры соответствующей CU для использования с текущей CU. Например, список кандидатов может включать в себя одну CU над и одну CU слева, относительно текущей CU, подлежащей кодированию в слайсе или изображении. В этом примере флаг или другой элемент синтаксиса может сигнализироваться для указания выбора кандидата. Например, передаваемый флаг, равный 0, означает копирование из левой CU, и передаваемый флаг, рав-

ный 1, означает копирование из верхней CU. Видеодекодер 30 выбирает палитру, подлежащую копированию из соответствующей соседней CU и копирует ее для использования при декодировании текущей CU. Предсказание также может выводиться с использованием наиболее частых значений выборов в причинных соседях текущей CU.

Предсказание палитр также может производиться по всем записям. Для каждой записи в палитре видеодекодер 20 генерирует и сигнализирует флаг. Значение флага, равное 1, для данной записи указывает, что предсказанное значение (например, соответствующая запись из выбранного CU-кандидата, например, левой CU) используется как значение этой записи. Значение флага, равное 0, указывает, что эта запись не предсказана и это значение будет передаваться на видеодекодер 30 от видеокодера 20, например, сигнализироваться в битовом потоке, кодированном видеокодером 20 для дальнейшего декодирования видеодекодером 30.

Значение "pred_palette_flag", CU-кандидат, палитра которой используется для предсказания палитры текущей CU, или правила построения кандидатов, также могут условно передаваться или выводиться на основании вспомогательной информации таким же образом, как описано выше для передачи флага PLT_Mode_flag.

Затем видеодекодер 20 может генерировать и сигнализировать карту, которая указывает, какая соответствующая запись палитры связана с каждым пикселем в CU. i -я запись в карте соответствует i -ой позиции в CU. Значение i -ой записи, равное 1 указывает, что пиксельное значение в этом i -ом положении в CU является одним из значений в палитре и индекс палитры дополнительно передается таким образом, что видеодекодер 30 может реконструировать пиксельное значение (в случае наличия только одной записи в палитре, передачу индекса палитры можно пропустить). Значение i -ой записи, равное 0, указывает, что пиксельное значение в i -ой позиции в CU не находится в палитре, и поэтому пиксельное значение будет явно передаваться на видеодекодер 30.

Если пиксельное значение в одной позиции в CU является значением в палитре, наблюдается, что с высокой вероятностью соседние позиции в CU имеют одно и то же пиксельное значение. Поэтому после кодирования индекса палитры (например, j , который соответствует пиксельному значению s) для позиции видеодекодер 20 может передавать элемент синтаксиса "run" для указания количества последовательных значений одного и того же пиксельного значения s в CU, прежде чем сканирование достигнет другого пиксельного значения. Например, если непосредственно следующее имеет значение, отличное от s , то передается $run=0$. Если следующее равно s , но следующее после него не равно s , то $run=1$.

В случае, когда серия не передается (например, неявный вывод серии), значение серии может быть постоянным, например 4, 8, 16 и т.д., или значение серии также может зависеть от вспомогательной информации. Например, значение серии может зависеть от размера блока, например серия равна ширине текущего блока или высоте текущего блока или полуширине (или полувысоте) текущего блока или доле ширины и высоты блока или кратному высоте/ширины блока. Значение серии также может зависеть от QP, типа кадра, цветовой компоненты, формата цвета (например, 444, 422, 420) и/или цветового пространства (например, YUV, RGB). Значение серии также может зависеть от направления сканирования. В других примерах значение серии может зависеть от других типов вспомогательной информации.

Значение серии также может сигнализироваться с использованием синтаксиса высокого уровня (например, PPS, SPS).

В некоторых примерах передача карты может не требоваться. Серия может начинаться только в некоторых положениях. Например, серия может начинаться только в начале каждой строки, или в начале каждых N строк. Начальное положение может различаться для разных направлений сканирования. Например, если используется вертикальное сканирование, серия может начинаться только в начале столбца или в начале каждых N столбцов. Начальное положение может зависеть от вспомогательной информации. Например, начальным положением может быть средняя точка каждой строки или каждого столбца, или $1/n$, $2/n$, ..., $(n-1)/n$ (т.е. доли) каждой/ого строки/столбца. Начальное положение также может зависеть от QP, типа кадра, цветовой компоненты, формата цвета (например, 444, 422, 420) и/или цветового пространства (например, YUV, RGB). В других примерах, начальная позиция серии может зависеть от других типов вспомогательной информации. Начальная позиция также может сигнализироваться с использованием синтаксиса высокого уровня (например, PPS, SPS и т.д.).

Возможно также объединение неявного вывода начальной позиции и неявного вывода серии. Например, серия равна расстоянию между двумя соседними начальными позициями. В случае, когда начальная точка является началом (т.е. первой позицией) каждой строки, длина серии равна строке.

Направление сканирования может быть вертикальным или горизонтальным. Возможно, что флаг передается для каждой CU для указания направления сканирования. Флаги могут передаваться отдельно для каждой компоненты или может передаваться единый флаг, и указанное направление сканирования применяется ко всем цветовым компонентам. Можно также использовать другие направления сканирования, например, 45 градусов или 135 градусов. Порядок сканирования может быть фиксированным или может зависеть от вспомогательной информации таким же образом, как описано выше для передачи флага PLT_Mode_flag.

Выше было объяснено, как передавать палитру. Альтернативой вышеописанным примерам является

ся построение палитры в оперативном режиме. В этом случае в начале CU в палитре не существует записей, и когда видеокодер 20 сигнализирует новые значения пикселей для позиций в CU, эти значения включаются в палитру. Таким образом, видеокодер 20 добавляет пиксельные значения в палитру по мере их генерации и передачи для позиций в CU. Тогда дальнейшие позиции в CU, имеющие одинаковые значения, могут относиться к пиксельным значениям в палитре, например, со значениями индекса, вместо того, чтобы видеокодер 20 передавал пиксельные значения. Аналогично, когда видеодекодер 30 принимает новое пиксельное значение (например, сигнализируемое кодером) для позиции в CU, он включает пиксельное значение в палитру, построенную видеодекодером 30. Когда дальнейшие позиции в CU имеют пиксельные значения, добавленные в палитру, видеодекодер 30 может принимать информацию, например, значения индекса, которые идентифицируют соответствующие пиксельные значения в палитре для реконструкции пиксельных значений в CU.

Если достигается максимальный размер палитры, например, при динамическом построении палитры в оперативном режиме, кодер и декодер совместно используют один и тот же механизм для удаления записи палитры. Один способ предусматривает удаление старейшей записи в палитре (очередь FIFO). Другой способ предусматривает удаление наименее используемой записи в палитре. Еще один предусматривает взвешивание обоих способов (времени в палитре и частоты использования) для принятия решения на замену записи. В порядке одного примера при удалении записи пиксельного значения из палитры и повторном появлении пиксельного значения в более поздней позиции в палитре кодер может передавать пиксельное значение вместо того, чтобы включать запись в палитру. Дополнительно или альтернативно, такое пиксельное значение можно повторно вводить в палитру после удаления, например, когда кодер и декодер сканируют позиции в CU.

Это изобретение также предусматривает объединение начальной сигнализации палитры с выводом палитры в оперативном режиме. В одном примере начальная палитра будет обновляться с кодированием пикселей. Например, после передачи начальной палитры видеокодер 20 может добавлять значения в начальную палитру или изменять значения в начальной палитре по мере сканирования пиксельных значений дополнительных положений в CU. Аналогично, приняв начальную палитру, видеодекодер 30 может добавлять значения в начальную палитру или изменять значения в начальной палитре по мере сканирования пиксельных значений дополнительных положений в CU. Аналогично, кодер может сигнализировать, использует ли текущая CU передачу всей палитры, или генерацию палитры в оперативном режиме, или комбинацию передачи начальной палитры с обновлением начальной палитры путем вывода в оперативном режиме. В некоторых примерах, начальная палитра может быть полной палитрой при максимальном размере палитры, и в этом случае значения в начальной палитре могут изменяться, или палитрой уменьшенного размера, и в этом случае значения добавляются в начальную палитру, и, в необязательном порядке, значения в начальной палитре изменяются.

Выше было описано, как передавать карту путем идентификации пиксельного значения. Согласно вышеописанному способу, передача карты может осуществляться путем сигнализации копирования линий. В одном примере, копирование линий сигнализируется видеокодером 20 таким образом, что пиксельное значение для записи равно пиксельному значению записи вышележащей линии (или в столбце слева при вертикальном сканировании). Затем может сигнализироваться 'серия' записей, которые копируются из линии. Также может указываться линия, из которой она копируется; с этой целью можно буферизовать несколько вышележащих линий. Например, сохраняются четыре предыдущих строки, и копируемая строка может сигнализироваться усеченным унарным кодом или другими кодами и затем может сигнализироваться количество копируемых записей этой строки, т.е. серия. Поэтому в некоторых примерах пиксельное значение для записи может сигнализироваться равным пиксельному значению записи в строке, находящейся непосредственно над или в двух или более строках над текущей строкой.

В случае отсутствия сигнализации серии, значение серии может быть постоянным/фиксированным или может зависеть от вспомогательной информации (и выводиться декодером) с использованием вышеописанного способа.

Возможно также, что карту не обязательно передавать. Например, серия может начинаться только в некоторых позициях. Начальная позиция может быть фиксированной или может зависеть от вспомогательной информации (и выводиться декодером), поэтому сигнализацию начальной позиции можно пропустить. Вместо этого, можно применять один или более вышеописанных методов. Неявный вывод начальной позиции и неявный вывод серии также можно объединять тем же способом, как описано выше.

Если используются оба способа передачи карты, то флаг или другой элемент синтаксиса может указывать, получается ли пиксель из палитры или из предыдущих линий, и затем индекс указывает запись в палитре или строку и, наконец, 'серию'.

Это изобретение описывает способы, устройства и методы для упрощения кодирования в режиме палитры и/или для повышения эффективности кодирования на основе палитры. Методы этого изобретения можно использовать совместно или отдельно для повышения эффективности кодирования и/или упрощения кодера. В общем случае, согласно методам этого изобретения, устройство кодирования видеоданных может быть выполнено с возможностью кодирования и декодирования видеоданных с использованием одного или более режимов кодирования палитры, причем режимы кодирования палитры не

включают в себя режим совместного использования палитры.

В одном примере режима палитры флаг, например `palette_share_flag`, может сигнализироваться в битовый поток для указания, что палитра для или более блоков видеоданных совместно используется или объединяется из палитры другого блока видеоданных. Блок видеоданных, из которого получают совместно используемую палитру, может базироваться на заранее определенных правилах (например, использовать палитру блока слева или над текущим блоком) или может иначе указываться в битовом потоке кодированного видеосигнала. Как описано в R. Joshi and J. Xu, "High efficient video coding (HEVC) screen content coding: Draft 2," JCTVC-S1005, Section 7.4.9.6, семантика `palette_share_flag` предусматривает, что "`palette_share_flag[x0][y0]`", равный 1, указывает, что палитра для текущей единицы кодирования выводится копированием первых записей `PreviousPaletteSize` из палитры предсказателя. Переменная `PreviousPaletteSize` выводится согласно подпункту 8.4.5.2.8. `palette_share_flag[x0][y0]`, равный 0, указывает, что палитра для текущей единицы кодирования задается как комбинация записей палитры из предыдущих единиц кодирования и новых записей палитры, которые явно сигнализируются".

В одном примере, когда значение `palette_share_flag` равно 1, `palette_share_flag` указывает, что текущий блок может повторно использовать последнюю кодированную палитру из ранее кодированного блока. Этот способ также известен как совместное использование палитры. Однако новые результаты исследований указывают, что этот флаг, совместно со способом совместного использования палитры, который он представляет, неэффективен для повышения эффективности кодирования, также дополнительно усложняя разбор и декодирование.

Кроме того, выявляются некоторые избыточности в процессе сигнализации для элемента синтаксиса, который указывает тип серии, например `palette_run_type_flag`. В частности, когда текущий пиксель является первым пикселем в линии в порядке сканирования и при наличии пикселя рядом с текущим пикселем и над ним текущий пиксель не может пребывать в режиме "копирования сверху". Выражение "имеется вышележащий пиксель" означает, что сосед сверху находится в текущем блоке для горизонтального сканирования или сосед слева находится в блоке для вертикального порядка сканирования, если способы "копирования снаружи" не разрешены. Когда способы "копирования снаружи" разрешены, "вышележащий пиксель" всегда может иметься для каждого пикселя в блоке. Примеры способов "копирования снаружи" описаны в Y.-C. Sun, J. Kim, T.-D. Chuang, Y.-W. Chen, S. Liu, Y.-W. Huang, and S. Lei, "Non-CE6: Cross-CU palette colour index prediction," JCTVC-S0079 и J. Kim, Y.-C. Sun, S. Liu, T.-D. Chuang, Y.-W. Chen, Y.-W. Huang, and S. Lei, "CE6-related: Enabling copy above mode prediction at the boundary of CU," JCTVC-S0114.

Если текущий пиксель кодируется согласно режиму "копирования сверху", то индекс текущего пикселя равен индексу соседа сверху текущего пикселя. Напротив, согласно правилу, гласящему, что после режима "копирования сверху" не может непосредственно следовать другой режим "копирования сверху", сосед сверху должен быть концом серии "индекс копии". Таким образом, серию "индекс копии" соседа сверху можно удлинить по меньшей мере на 1 путем добавления текущего пикселя в серию "индекс копии" вместо того, чтобы делать текущий пиксель первым пикселем серии "копирования сверху". Таким образом, можно нормативно отключать режим "копирования сверху", если текущий пиксель является первым пикселем в линии в порядке сканирования. Это приводит к экономии битов, поскольку для такого пикселя тип серии можно выводить как "индекс копии", что избавляет от необходимости сигнализировать такой индекс.

Кроме того, текущая бинаризация для элемента синтаксиса `palette_num_signalled_entries` обеспечивается усеченным унарным кодом. Элемент синтаксиса `palette_num_signalled_entries` указывает количество записей в текущей палитре (например, палитре, подлежащей использованию для кодирования текущего блока видеоданных), которые явно сигнализируются. Количество явно сигнализируемых выборок может определяться разностью между количеством записей в палитре, вычитаемым из количества записей в палитре, предсказанных из палитры другого блока видеоданных (включающего в себя любые записи палитры, которые указывают использование выходной выборки). В некоторых примерах элемент синтаксиса `palette_num_signalled_entries` может именоваться элементом синтаксиса `num_signalled_palette_entries`.

В некоторых примерах кодовое слово, используемое для кодирования значения элемента синтаксиса `palette_num_signalled_entries`, может быть нежелательно длинным, что может приводить к кодовым словам длиной более 32. Например, в HEVC1 все кодовые слова имеют длину 32 или менее. Такая ситуация может возникать при кодировании значения элемента синтаксиса `palette_predictor_run`. Элемент синтаксиса `palette predictor run` указывает количество нулей перед ненулевой записью в матрице `predictor_palette_entry_reuse_flag`.

`predictor_palette_entry_reuse_flag` указывает, используется ли повторно конкретная запись палитры из одной или более ранее использовавшихся палитр для текущей палитры. Значение `palette_predictor_run` может составлять от 0 до максимального размера предсказателя палитры включительно.

Ввиду этих недостатков в одном примере изобретения это изобретение предусматривает, что видеокoder 20 и видеокoder 30 выполнены с возможностью осуществления режима кодирования на основе палитры без методов совместного использования палитры. В частности, видеокoder 20 и видеокoder 30 могут быть выполнены с возможностью осуществления кодирования на основе палитры без

использования элемента синтаксиса `palette_share_flag[x0][y0]`, как указано ниже:

<code>palette_share_flag[x0][y0]</code>	<code>ae(v)</code>
<code>if(!palette_share_flag[x0][y0]) {</code>	<code>1}</code>

Вместо использования методов совместного использования палитры видеокодер 20 и видеodeкодер 30 могут быть выполнены с возможностью кодирования палитры для использования с одним или более блоками видеоданных с использованием других методов, например вышеописанных методов предсказания палитры. В других примерах видеокодер 20 и/или видеodeкодер 30 могут быть выполнены с возможностью осуществления предсказания палитры с использованием следующих методов.

На фиг. 4 показана блок-схема, более подробно демонстрирующая блок 122 кодирования на основе палитры видеокодера 20. Блок 122 кодирования на основе палитры может быть выполнен с возможностью осуществления одного или более из иллюстративных методов этого изобретения для кодирования видеоданных на основе палитры.

Как описано выше, блок 122 кодирования на основе палитры может быть выполнен с возможностью кодирования блока видеоданных (например, CU или PU) в режиме кодирования на основе палитры. В режиме кодирования на основе палитры палитра может включать в себя записи, пронумерованные индексом и представляющие значения цветовой компоненты (например, RGB, YUV и т.д.) или интенсивности, которые можно использовать для указания пиксельных значений. Блок 203 генерации палитры может быть выполнен с возможностью приема пиксельных значений 212 для текущего блока видеоданных и генерации палитры значений цвета для текущего блока видеоданных. Блок 203 генерации палитры может использовать любые методы генерации палитры для текущего блока видеоданных, в том числе рассмотренный выше метод на основе гистограммы. Блок 203 генерации палитры может быть выполнен с возможностью генерации палитры любого размера. В одном примере блок 203 генерации палитры может быть выполнен с возможностью генерации 32 записей палитры, где каждая запись палитры включает в себя пиксельные значения для компонент Y, C_r и C_b пикселя. В первом примере предполагается, что каждая запись палитры указывает значения для всех цветовых компонент выборки (пикселя). Однако принципы, изложенные в этом документе, применимы к использованию отдельной палитры для каждой цветовой компоненты.

После того, как блок 203 генерации палитры генерирует палитру, картографический блок 204 может генерировать карту для текущего блока видеоданных, которая указывает, можно ли представить конкретный пиксель в текущем блоке видеоданных записью в палитре, сгенерированной блоком 203 генерации палитры. Картографический блок 204 может создавать карту 214, которая включает в себя элементы синтаксиса, которые указывают, как каждый пиксель использует (или не использует) записи из палитры. Как рассмотрено выше, в некоторых примерах выходные пиксели не сигнализируются отдельным элементом синтаксиса, но могут указываться заранее определенным зарезервированным индексом в палитре. Если в палитре не найдено значения для пикселя в текущем блоке видеоданных, картографический блок 204 может указывать использование выходного пикселя с зарезервированным индексом в палитре и явно передавать пиксельное значение для этого конкретного пикселя. В некоторых примерах картографический блок 204 может предсказывать явное пиксельное значение из одной из записей, найденных в палитре. В некоторых других примерах картографический блок 204 может квантовать пиксель и передавать квантованные значения.

Помимо сигнализации элементов синтаксиса, которые указывают значения цвета, используемые для каждого пикселей в блоке, блок 122 кодирования на основе палитры также может быть выполнен с возможностью сигнализации палитры, которая подлежит использованию для текущего блока видеоданных. В соответствии с методами этого изобретения блок 122 кодирования на основе палитры может быть выполнен с возможностью применять методы предсказания палитры для уменьшения объема данных, сигнализируемых для указания значений палитры для конкретного блока видеоданных.

В порядке одного примера предсказания палитры, описанного в JCTVC-Q0094, который можно найти с 20 июня 2014 г. по адресу http://phenix.int-evry.fr/jct/doc_end_user/documents/17_Valencia/wg11/JCTVC-Q0094-v1.zip, палитра может включать в себя записи, которые копируются из палитры предсказателя. Палитра предсказателя может включать в себя записи палитры из ранее кодированных блоков, которые используют режим палитры, или из других реконструированных выборок. Как показано на фиг. 4, блок 122 кодирования на основе палитры может включать в себя буфер 210 палитры предсказателя. Буфер 210 палитры предсказателя может быть выполнен с возможностью хранения нескольких ранее использовавшихся записей палитры из ранее кодированных блоков. В порядке одного примера буфер 210 палитры предсказателя можно конфигурировать как буфер, действующий по принципу "первым вошел, первым вышел" (FIFO) заранее определенного размера. Буфер 210 палитры предсказателя может иметь любой размер. В одном примере буфер 210 палитры предсказателя включает в себя до 64 ранее использовавшихся записей палитры.

В некоторых примерах блок 122 кодирования на основе палитры может быть выполнен с возможностью отсекаать записи в буфере 210 палитры предсказателя таким образом, чтобы все записи палитры в буфере 210 палитры предсказателя были уникальными. Таким образом, для каждой новой записи палит-

ры, подлежащей добавлению в буфер 210 палитры предсказателя, блок 122 кодирования на основе палитры может быть выполнен с возможностью сначала удостовериться в том, что в буфере 210 палитры предсказателя уже не хранятся другие такие же записи. В отсутствие одинаковых записей новая запись палитры добавляется в буфер 210 палитры предсказателя. Если новая запись дублирует существующую запись, новая запись палитры добавляется в буфер 210 палитры предсказателя и дублирующие записи удаляются из буфера 210 палитры предсказателя.

Блок 122 кодирования на основе палитры может включать в себя блок 206 генерации двоичного вектора предсказания, который выполнен с возможностью генерации и сигнализации двоичного флага (например, `predictor_palette_entry_reuse_flag`). Для каждой записи в палитре для текущего блока видеоданных, генерируемой блоком 203 генерации палитры, для указания, копируется (или повторно используется) ли запись палитры в буфере 210 палитры предсказателя для одной из записей в палитре для текущего блока видеоданных (что, например, указано флагом=1). Таким образом, флаг со значением 1 в двоичном предсказательном векторе указывает, что соответствующая запись в буфере 210 палитры предсказателя повторно используется для палитры для текущего блока, тогда как флаг со значением 0 в двоичном векторе предсказания указывает, что соответствующая запись в буфере 210 палитры предсказателя повторно не используется для палитры для текущего блока. Дополнительно блок 122 кодирования на основе палитры может быть выполнен с возможностью явной сигнализации некоторых значений для текущей палитры, которые нельзя скопировать из записей в буфере 210 палитры предсказателя. Также может сигнализироваться количество новых записей. В связи с этим видеокодер 20 и/или декодер 30 может быть выполнен с возможностью сигнализации количества явно сигнализируемых записей палитры с использованием элемента синтаксиса `palette_num_signalled_entries`.

При использовании режима кодирования на основе палитры, в котором используются методы предсказания палитры, видеокодер 20 и декодер 30 могут быть выполнены с возможностью кодирования, помимо других элементов синтаксиса, элемента синтаксиса, который указывает количество записей палитры, которые явно сигнализируются для текущей палитры, подлежащей использованию для кодирования текущего блока видеоданных (например, `palette_num_signalled_entries`). Это изобретение предусматривает методы повышения эффективности кодирования или ограничения длины кодового слова при кодировании такого элемента синтаксиса.

В одном примере изобретения блок 122 кодирования на основе палитры может быть выполнен с возможностью кодирования первого бина элемента синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре, например элемента синтаксиса `palette_num_signalled_entries`, с использованием контекста САВАС. Блок 122 кодирования на основе палитры может кодировать другие бины `palette_num_signalled_entries` с использованием других методов кодирования. В другом примере изобретения блок 122 кодирования на основе палитры может быть выполнен с возможностью использования более одного контекста для кодирования первого бина элемента синтаксиса `palette_num_signalled_entries`. В одном примере, блок 122 кодирования на основе палитры может быть выполнен с возможностью определения контекстов на основании размера блока кодируемого текущего блока видео и/или на основании значения других элементов синтаксиса.

В соответствии с одним примером изобретения блок 122 кодирования на основе палитры может быть выполнен с возможностью определения первого бина первого элемента синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре. Видеокодер 20 может быть дополнительно выполнен с возможностью кодирования битового потока, который включает в себя первый элемент синтаксиса. Битовый поток также может не включать в себя второй элемент синтаксиса, который указывает режим совместного использования палитры. В некоторых примерах блок 122 кодирования на основе палитры может быть выполнен с возможностью кодирования первого бина первого элемента синтаксиса с использованием контекстно-адаптивного двоичного арифметического кодирования. В других примерах блок 122 кодирования на основе палитры может быть выполнен с возможностью кодирования первого бина первого элемента синтаксиса с использованием одного или более контекстов. В некоторых примерах использования одного или более контекстов один или более контекстов могут базироваться на по меньшей мере одном из предсказанного количества записей кодирования палитры и размера блока.

В другом примере изобретения, чтобы длина кодового слова `palette_num_signalled_entries` не увеличивалась свыше 32 битов, предложено вносить нормативные семантические изменения в современные методы кодирования палитры (например, R. Joshi and J. Xu, "High efficient video coding (HEVC) screen content coding: Draft 2," JCTVC-S1005). Например, возможные значения элемента синтаксиса, который указывает максимально допустимый размер палитры, например `palette_max_size`, и элемента синтаксиса, который указывает максимальный размер палитры предсказателя, например `palette_max_predictor_size`, можно ограничивать порогом.

Такой порог может быть заранее определенным и храниться в памяти (например, в памяти 98 видеоданных на фиг. 2 или памяти 148 видеоданных на фиг. 3), к которой может обращаться блок 122 кодирования на основе палитры. В частности, значение `palette_max_size` может быть любым значением от 0 до T1 включительно, где T1 является порогом. В его отсутствие блок 122 кодирования на основе палитры

может быть выполнен с возможностью делать вывод, что значение `palette_max_size` равно 0. Кроме того, значение `palette_max_predictor_size` может быть любым значением от 0 до T2, включительно, где T2 является порогом. В его отсутствие, блок 122 кодирования на основе палитры может быть выполнен с возможностью делать вывод, что значение `palette_max_predictor_size` равно 0.

В одном примере T1 равен 4096, и T2 равен 8192. В другом примере T1 равен 4095 и T2 равен 4095. В еще одном примере T1 равен 4095 и T2 равен 8191.

В порядке другого примера это изобретение предусматривает, что значение `palette_max_size` равно количеству пикселей в единице кодирования наибольшего размера. Такое значение может быть заранее определенным и храниться в памяти, к которой может обращаться блок 122 кодирования на основе палитры. В некоторых примерах значение `palette_max_predictor_size` может быть меньше или равно $K \cdot \text{palette_max_size}$, где K - положительная постоянная. В некоторых примерах K=2.

В другом примере блок 122 кодирования на основе палитры (например, использующий блок 209 сжатия двоичного вектора предсказания или другой структурный компонент видеокодера 20, например, блок 118 энтропийного кодирования) может быть выполнен с возможностью кодирования значения элемента синтаксиса `palette_num_signalled_entries` с использованием одного или более методов кодирования из семейства кодов Голомба (например, кода Голомба-Райса, экспоненциального кода Голомба, усеченного кода Райса, унарного кода и т.д.). В одном примере изобретения блок 122 кодирования на основе палитры выполнен с возможностью кодирования значения элемента синтаксиса `palette_num_signalled_entries` с использованием экспоненциального кода Голомба порядка 0. В другом примере изобретения блок 122 кодирования на основе палитры выполнен с возможностью кодирования значения элемента синтаксиса `palette_num_signalled_entries` с использованием объединения усеченного кода Райс (TR) и экспоненциального кода Голомба, например, используемого в кодировании коэффициентов для кодирования элемента синтаксиса `coeff_abs_level_remaining` в HEVC1.

Ниже приведен пример объединения кода TR и экспоненциального кода Голомба для параметра Голомба-Райса, равного 0:

Символ	Бинаризация
0	0
1	10
2	110
3	1110
[4, 5]	11110x
[6, 9]	111110xx
[10, 17]	1111110xxx
...	

Здесь x может принимать значение 0 или 1. Аналогично, нижеследующая таблица демонстрирует пример объединенной бинаризации, используемой при кодировании элемента синтаксиса `paletteRun`. Она получается объединением усеченного кода Райса и усеченного экспоненциального кода Голомба порядка 0 для максимального значения серии 7.

Символ	Бинаризация
0	0
1	10
[2, 3]	110x
[4, 7]	111xx

Здесь x может принимать значение 0 или 1.

Использование одного или более кодов Голомба (например, экспоненциального кода Голомба или объединения кода TR и экспоненциального кода Голомба) для кодирования элемента синтаксиса `palette_num_signalled_entries` обеспечивает преимущество по сравнению с предыдущими методами кодирования значения элемента синтаксиса `palette_num_signalled_entries`. Предыдущие методы кодирования значения элемента синтаксиса `palette_num_signalled_entries` использовали унарный код.

Использование унарного кода приводило к тому, что кодированная длина элемента синтаксиса `palette_num_signalled_entries` в ряде случаев превышала 32 бита. За счет использования одного или более кодов Голомба для кодирования элемента синтаксиса `palette_num_signalled_entries` методы этого изобретения позволяют блоку 122 кодирования на основе палитры кодировать значение элемента синтаксиса `palette_num_signalled_entries` таким образом, что кодированная длина остается меньшей или равной некоторому заранее определенному количеству битов (например, 32 битам).

В другом примере блок 122 кодирования на основе палитры может быть выполнен с возможностью кодирования значения элемента синтаксиса `palette_num_signalled_entries` с использованием усеченной версии семейства кодов Голомба (например, усеченного кода Голомба-Райса, усеченного экспоненциального кода Голомба, усеченного кода Райса, усеченного унарного кода и т.д.). В другом примере изо-

бренения блок 122 кодирования на основе палитры может быть выполнен с возможностью кодирования значения элемента синтаксиса `palette_num_signalled_entries` с использованием того же кода, который используется для кодирования элемента синтаксиса `paletteRun`. В другом примере блок 122 кодирования на основе палитры может быть выполнен с возможностью кодирования значения элемента синтаксиса `palette_num_signalled_entries` с использованием способа, применяемого для кодирования элемента синтаксиса `coeff_abs_level_remaining` в кодировании коэффициентов (например, объединения усеченного кода Райса (TR) и экспоненциального кода Голомба). В соответствии с этим примером предпочтительно, чтобы параметр TR был равен 0. В каждом из этих примеров конкретный усеченный код выбирается таким образом, чтобы кодированная длина элемента синтаксиса `palette_num_signalled_entries` оставалась меньшей или равной 32 битам.

В другом примере на битовый поток предлагается накладывать ограничение, согласно которому `palette_num_signalled_entries` равен количеству пикселей в блоке. Таким образом, блок 122 кодирования на основе палитры может быть выполнен с возможностью ограничения возможного значения элемента синтаксиса `palette_num_signalled_entries` количеством пикселей в кодируемом на данный момент блоке. В другом примере блок 122 кодирования на основе палитры может быть выполнен с возможностью ограничения возможного значения `palette_num_signalled_entries` количеством пикселей в наибольшем возможном блоке конкретного изображения (например, блоке большого размера, определяемом конкретным стандартом кодирования видеоданных).

В другом примере блок 122 кодирования на основе палитры может быть выполнен с возможностью обходить сигнализацию элемента синтаксиса, который указывает тип серии, например `palette_run_type_flag`, если текущий пиксель является первым пикселем в линии в порядке сканирования, и над текущим пикселем имеется пиксель, соседствующий с текущим пикселем. В одном примере блок 122 кодирования на основе палитры может быть выполнен с возможностью определения, что текущий пиксель является первым пикселем в линии в порядке сканирования. Блок 122 кодирования на основе палитры может дополнительно определять наличие соседнего пикселя, расположенного над текущим пикселем. В случае определения, что текущий пиксель является первым пикселем в линии в порядке сканирования, и определения наличия соседнего пикселя, расположенного над текущим пикселем, блок 122 кодирования на основе палитры может быть дополнительно выполнен с возможностью обходить кодирование первого элемента синтаксиса в битовом потоке, причем первый элемент синтаксиса указывает тип серии, и кодировать оставшуюся часть битового потока.

Согласно фиг. 4 и методам предсказания палитры этого изобретения в заявке США № 14/667411, поданной 24 марта 2015 г., опубликованной как патентная публикация США № 2015/0281728, предложены способ сигнализации на основе двоичного дерева и способы сигнализации на основе конечной позиции для кодирования двоичного предсказательного вектора палитры. В предварительной патентной заявке США № 62/002741, поданной 23 мая 2014 г., предложен способ сигнализации на групповой основе. Это изобретение предусматривает дополнительные методы генерации, кодирования и декодирования двоичного вектора предсказания.

Некоторые описанные здесь примеры относятся к способам кодирования вектора предсказания палитры для повышения эффективности кодирования. Например, предположим, что двоичный вектор предсказания, генерируемый блоком 206 генерации двоичного вектора предсказания, обозначен как

$$b = [b_0, b_1, \dots, b_{N-1}], \quad N \geq 0, \quad b_i \in \{0, 1\}, \quad 0 \leq i < N$$

Согласно вышеприведенному уравнению $b_i \in \{0, 1\}$, $0 \leq i < N$ обозначает флаг предсказания (также именуемый двоичным флагом или двоичным флагом предсказания). Если $N=0$, $b=\phi$ (т.е. b является пустым вектором), который не нужно сигнализировать. Таким образом, в нижеследующем описании можно предположить, что $N>0$.

На фиг. 5 показан один пример буфера 210 палитры предсказателя и текущей палитры 220. Как можно видеть на фиг. 5, текущая палитра 220 повторно использует пиксельные значения из буфера 210 палитры предсказателя, связанной с индексами записей 1, 2, 5 и 9. Таким образом, двоичный предсказательный вектор, сформированный блоком 206 генерации двоичного вектора предсказания, показанным на фиг. 4, будет иметь вид $b=[110010001000]$. Как можно видеть в этом примере, двоичный вектор предсказания b включает в себя флаги со значением 1, соответствующие 1-му, 2-му, 5-му и 9-му индексам в буфере 210 палитры предсказателя. Таким образом, 1-я, 2-я, 5-я и 9-я записи в буфере 210 палитры предсказателя являются только записями, повторно используемыми для текущей палитры 220. Для индексов записей 5-8 в текущей палитре 220 блок 122 кодирования на основе палитры может быть выполнен с возможностью сигнализации значений записей палитры в битовом потоке кодированного видеосигнала (например, с использованием явной сигнализации или другого метода предсказания).

В соответствии с одним или более методами этого изобретения видеокодер 20 может быть выполнен с возможностью кодировать или, в целом, кодировать двоичный предсказательный вектор b для уменьшения объема данных, необходимого для сигнализации палитры в битовом потоке кодированного видеосигнала. Как показано на фиг. 4, блок 209 сжатия двоичного вектора предсказания может быть выполнен с возможностью генерации и сигнализации кодированного двоичного вектора 215 предсказания. Однако следует понимать, что методы сжатия двоичного вектора предсказания согласно этому изобрете-

нию можно реализовать в других структурах видеокодера 20, включающих в себя блок 118 энтропийного кодирования, показанный на фиг. 2. В одном примере изобретения блок 209 сжатия двоичного вектора предсказания может быть выполнен с возможностью кодирования двоичного вектора предсказания с использованием методов кодирования на основе длины серии. Например, блок 209 сжатия двоичного вектора предсказания может быть выполнен с возможностью кодирования двоичного вектора предсказания путем сигнализации количества последовательных '0' между '1' в двоичном векторе предсказания с использованием экспоненциального кода Голомба. Опять же в порядке примера предположим, что $b=[110010001000]$. В этом примере, как показано на фиг. 6, двоичный вектор предсказания (т.е. b) можно выразить как: 'нуль последовательных 0'- '1'- 'нуль последовательных 0'- '1'- 'два последовательных 0'- '1'- 'три последовательных 0'- '1'- и 'четыре последовательных 0'. Поскольку известно, что $b_i \in \{0,1\}$, за исключением последней группы 'последовательных 0', за каждой группой 'последовательных 0' должна следовать '1'. Таким образом, блок 209 сжатия двоичного вектора предсказания может использовать методы кодирования по длине серии на основе нулей для представления двоичного вектора предсказания b как 'нуль последовательных 0'- 'нуль последовательных 0'- 'два последовательных 0'- 'три последовательных 0'- 'четыре последовательных 0', который можно выразить как последовательность длин серий '0-0-2-3-4'.

В соответствии с одним или более примерами этого изобретения, связанными с сигнализацией на основе длины серии, для кодирования последовательности длин серий можно использовать код Голомба-Райса, экспоненциальный код Голомба любого порядка, усеченный экспоненциальный код Голомба, усеченный код Райса или любые другие бинаризации, включающие в себя усеченные бинаризации. В одном примере блок 209 сжатия двоичного вектора предсказания использует экспоненциальный код Голомба 0-го порядка в качестве метода кодирования по длине серии.

Для усеченной бинаризации maxsymbol может быть максимально возможным значением серии в зависимости от позиции '1' в двоичном векторе и размера двоичного вектора, поскольку после перемещения в конец двоичного вектора, максимально возможное значение серии уменьшается от размера вектора до 0 в зависимости от позиции в векторе. Например, maxsymbol может быть длиной двоичного вектора или длиной двоичного вектора минус позиция '1', от которой отсчитывается серия. Другими словами, это оставшаяся длина, измеренная от конца двоичного вектора. Для вышеприведенного примера с двоичным вектором b конкретного размера, например 13, последовательность длин серий '0-0-2-3-4' может кодироваться усеченной бинаризацией '0[13]-0[12]-2[11]-3[8]-4[4]', где maxsymbol указан в квадратных скобках.

Кроме того, в некоторых примерах бинаризация может зависеть от позиции или индекса элемента (0 или 1) в двоичном векторе. В порядке конкретного примера, если позиция меньше некоторого порога, используется бинаризация одного типа; в противном случае используется бинаризация другого типа. В некоторых примерах тип бинаризации может представлять собой разные коды бинаризации или одно и то же семейство кодов, но с разным порядком, например экспоненциальный код Голомба.

В одном примере порогом может быть длина палитры из предыдущего блока или предыдущего кодированного блока палитры. В другом примере порог может быть задан равным некоторому значению, принятому по умолчанию, или сигнализироваться для каждого блока, слайса, изображения или в другом месте. Следует понимать, что, в необязательном порядке, можно использовать соответствующий метод для задания контекста САВАС для кодирования значений серий. Дополнительно блок 122 кодирования на основе палитры (см. фиг. 2) может быть выполнен с возможностью остановки сигнализации длины серии, когда количество сигнализируемых элементов '1' (т.е. количество записей палитры из буфера 210 палитры предсказателя, указанных как повторно используемые для текущей палитры 220) достигает максимально возможной величины. В некоторых примерах максимально возможная величина является максимально возможным размером палитры.

Некоторые примеры этого изобретения относятся к кодированию конечной позиции последовательности длин серий, указывающей двоичный вектор предсказания b . В одном или более примерах этого изобретения блок 209 сжатия двоичного вектора предсказания может быть выполнен с возможностью кодирования двоичного вектора предсказания b с использованием зарезервированной длины серии L для кодирования конечной позиции двоичного вектора предсказания. В одном примере в качестве зарезервированной длины серии используется $L=1$. На видеокодере 20, если длина серии больше или равна L , блок 209 сжатия двоичного вектора предсказания выполнен с возможностью прибавления 1 к длине серии. Если фактическая длина серии меньше L , блок 209 сжатия двоичного вектора предсказания выполнен с возможностью сигнализации первоначальной длины серии. Блок 209 сжатия двоичного вектора предсказания может сигнализировать длину серии конечной позиции зарезервированной длиной серии L .

Аналогично, на видеокодере 30, если декодированное значение длины серии больше L , 1 вычитается из фактической длины серии. Если декодированное значение или длина серии меньше L , декодированное значение используется в качестве фактической длины серии. Если декодированное значение равно L , все оставшиеся позиции в двоичном векторе предсказания b равны 0. Поэтому, если декодированное значение равно L , сигнализация серии уже не требуется.

Согласно вышеприведенному примеру (т.е. $b=[110010001000]$) и исходя из того, что $L=1$, блок 209 сжатия двоичного вектора предсказания выполнен с возможностью сигнализации последовательности

длин серий '0-0-2-3-4', показанной на фиг. 6 как '0-0-3-4-1'. Тогда с применением вышеупомянутых правил видеodeкодер 30 может быть выполнен с возможностью восстановления последовательности длин серий как '0-0-2-3-конец'. Таким образом, первое значение длины серии 0 декодируется как 0, и следующая последовательность длин серий 0 декодируется как 0, поскольку обе последовательности длин серий 0 меньше зарезервированного значения длины серии $L=1$. Следующая последовательность длин серий равна 3, что позволяет видеodeкодеру 30 вычитать 1 из значения 3, получая 2, поскольку принятое значение 3 больше зарезервированного значения длины серии $L=1$. Аналогично, видеodeкодер 30 способен вычитать 1 из принятого значения 4 для получения 3 для следующей последовательности длин серий, поскольку принятое значение 4 больше зарезервированного значения длины серии $L=1$. Наконец, последнее принятое значение длины серии 1 равно зарезервированному значению длины серии $L=1$. Соответственно видеodeкодер 30 может определять, что в двоичном векторе предсказания больше нет значений '1'.

На фиг. 7 показана блок-схема, демонстрирующая пример блока 165 декодирования на основе палитры видеodeкодера 30. Блок 165 декодирования на основе палитры может быть выполнен с возможностью осуществления действия обратного блоку 122 кодирования на основе палитры, показанному на фиг. 4. Блок 165 декодирования на основе палитры может быть выполнен с возможностью приема карты 312, которая указывает для каждого пикселя в текущем блоке, будут ли использоваться записи для палитры для пикселей в текущем блоке. Кроме того, карта 312 может дополнительно указывать, какие записи палитры подлежат использованию для данного пикселя. Картографический блок 302 может декодировать текущий блок видеоданных с использованием карты 312 и палитры, генерируемой блоком 304 генерации палитры для создания декодированных видеоданных 314.

В соответствии с методами этого изобретения блок 165 декодирования на основе палитры также может принимать кодированный двоичный вектор 316 предсказания. Как рассмотрено выше, двоичный вектор 316 предсказания может кодироваться с использованием метода кодирования по длине серии, который кодирует последовательность длин серий, указывающую серию нулевых значений в двоичном векторе предсказания. Блок 306 снятия сжатия двоичного вектора предсказания может быть выполнен с возможностью декодирования кодированного двоичного вектора 316 предсказания с использованием любой комбинации методов кодирования по длине серии, описанных выше со ссылкой на фиг. 4-6. После восстановления двоичного вектора предсказания блоком 306 снятия сжатия двоичного вектора предсказания блок 304 генерации палитры может генерировать палитру для текущего блока видеоданных на основании двоичного вектора предсказания и ранее использовавшихся записей палитры, хранящихся в буфере 310 палитры предсказателя. Блок 165 декодирования на основе палитры может быть выполнен с возможностью сохранения ранее использовавшихся записей палитры в буфере 310 палитры предсказателя таким же образом, как блок 122 кодирования на основе палитры (см. фиг. 2) сохраняет ранее использовавшиеся записи палитры в буфере 210 палитры предсказателя.

В одном примере изобретения блок 165 декодирования на основе палитры может быть выполнен с возможностью декодирования первого бина элемента синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре, например элемента синтаксиса `palette_num_signalled_entries`, с использованием контекста CABAC. Блок 165 декодирования на основе палитры может декодировать другие бины `palette_num_signalled_entries` с использованием других методов декодирования. В другом примере изобретения блок 165 декодирования на основе палитры может быть выполнен с возможностью использования более одного контекста для декодирования первого бина элемента синтаксиса `palette_num_signalled_entries`. В одном примере блок 165 декодирования на основе палитры может быть выполнен с возможностью определения контекстов на основании размера блока декодируемого на данный момент блока видео и/или на основании значения других элементов синтаксиса.

В соответствии с одним примером изобретения блок 165 декодирования на основе палитры может быть выполнен с возможностью определения первого бина первого элемента синтаксиса, который указывает количество явно сигнализируемых записей в текущей палитре. Видеodeкодер 30 может быть дополнительно выполнен с возможностью декодирования битового потока, который включает в себя первый элемент синтаксиса. Битовый поток также может не включать в себя второй элемент синтаксиса, который указывает режим совместного использования палитры. В некоторых примерах блок 165 декодирования на основе палитры может быть выполнен с возможностью декодирования первого бина первого элемента синтаксиса с использованием контекстно-адаптивного двоичного арифметического кодирования. В других примерах блок 165 декодирования на основе палитры может быть выполнен с возможностью декодирования первого бина первого элемента синтаксиса с использованием одного или более контекстов. В некоторых примерах использования одного или более контекстов один или более контекстов могут базироваться на по меньшей мере одном из предсказанного количества записей кодирования палитры и размера блока.

В другом примере изобретения, чтобы длина кодового слова `palette_num_signalled_entries` не увеличивалась свыше 32 битов, предложено вносить нормативные семантические изменения в современные методы кодирования палитры. Например, возможные значения элемента синтаксиса, который указывает максимально допустимый размер палитры, например `palette_max_size`, и элемента синтаксиса, который

указывает максимальный размер палитры предсказателя, например `palette_max_predictor_size`, можно ограничивать порогом. Такой порог может быть заранее определенным и храниться в памяти (например, память 148 видеоданных на фиг. 3), к которой может обращаться блок 165 декодирования на основе палитры. В частности, значение `palette_max_size` может быть любым значением от 0 до T1, включительно, где T1 является порогом. В его отсутствие блок 165 декодирования на основе палитры может быть выполнен с возможностью делать вывод, что значение `palette_max_size` равно 0. Кроме того, значение `palette_max_predictor_size` может быть любым значением от 0 до T2, включительно, где T2 является порогом. В его отсутствие, блок 165 декодирования на основе палитры может быть выполнен с возможностью делать вывод, что значение `palette_max_predictor_size` равно 0.

В одном примере T1 равен 4096 и T2 равен 8192. В другом примере T1 равен 4095 и T2 равен 4095. В еще одном примере T1 равен 4095 и T2 равен 8191.

В порядке другого примера это изобретение предусматривает, что значение `palette_max_size` равно количеству пикселей в единице кодирования наибольшего размера. Такое значение может быть заранее определенным и храниться в памяти, к которой может обращаться блок 165 декодирования на основе палитры. В некоторых примерах значение `palette_max_predictor_size` может быть меньше или равно $K \cdot \text{palette_max_size}$, где K - положительная постоянная. В некоторых примерах K=2.

В другом примере блок 165 декодирования на основе палитры, показанный на фиг. 3 (например, использующий блок 306 снятия сжатия двоичного вектора предсказания или другой структурный компонент видеodeкодера 30, например, блок 150 энтропийного декодирования, показанный на фиг. 3), может быть выполнен с возможностью декодирования значения элемента синтаксиса `palette_num_signalled_entries` с использованием одного или более методов декодирования из семейства кодов Голомба (например, кода Голомба-Райса, экспоненциального кода Голомба, усеченного кода Райса, унарного кода и т.д.). В одном примере изобретения блок 165 декодирования на основе палитры выполнен с возможностью декодирования значения элемента синтаксиса `palette_num_signalled_entries` с использованием объединения усеченного кода Райса и экспоненциального кода Голомба.

В другом примере блок 165 декодирования на основе палитры может быть выполнен с возможностью декодирования значения элемента синтаксиса `palette_num_signalled_entries` с использованием усеченной версии семейства кодов Голомба (например, усеченного кода Голомба-Райса, усеченного экспоненциального кода Голомба, усеченного кода Райса, усеченного унарного кода и т.д.). В другом примере изобретения блок 165 декодирования на основе палитры может быть выполнен с возможностью декодирования значения элемента синтаксиса `palette_num_signalled_entries` с использованием того же кода, который используется для кодирования элемента синтаксиса `paletteRun`. В другом примере блок 165 декодирования на основе палитры может быть выполнен с возможностью декодирования значения элемента синтаксиса `palette_num_signalled_entries` с использованием способа декодирования элемента синтаксиса `coeff_abs_level_remaining` при декодировании коэффициентов (например, объединения усеченного кода Райса (TR) и экспоненциального кода Голомба). В соответствии с этим примером предпочтительно, что-бы параметр TR был равен 0.

В другом примере на битовый поток предлагается накладывать ограничение, согласно которому `palette_num_signalled_entries` равен количеству пикселей в блоке. Таким образом, блок 165 декодирования на основе палитры может быть выполнен с возможностью ограничения возможного значения элемента синтаксиса `palette_num_signalled_entries` количеством пикселей в кодируемом на данный момент блоке. В другом примере блок 165 декодирования на основе палитры может быть выполнен с возможностью ограничения возможного значения `palette_num_signalled_entries` количеством пикселей в наибольшем возможном блоке конкретного изображения (например, блоке большого размера, определяемом конкретным стандартом кодирования видеоданных).

В другом примере блок 165 декодирования на основе палитры может быть выполнен с возможностью делать вывод, что значение элемента синтаксиса, который указывает тип серии, например `palette_run_type_flag`, если текущий пиксель является первым пикселем в линии в порядке сканирования, и над текущим пикселем имеется пиксель, соседствующий с текущим пикселем. В одном примере блок 165 декодирования на основе палитры может быть выполнен с возможностью определения, что текущий пиксель является первым пикселем в линии в порядке сканирования, блок 165 декодирования на основе палитры может дополнительно определять наличие соседнего пикселя, расположенного над текущим пикселем. В случае определения, что текущий пиксель является первым пикселем в линии в порядке сканирования, и определения наличия соседнего пикселя, расположенного над текущим пикселем, блок 165 декодирования на основе палитры может быть дополнительно выполнен с возможностью делать вывод, что значение первого элемента синтаксиса в битовом потоке, причем первый элемент синтаксиса указывает тип серии, и кодировать оставшуюся часть битового потока.

На фиг. 8 показана блок-схема операций, демонстрирующая пример способа кодирования видеосигнала согласно методам изобретения. Методы, представленные на фиг. 8, можно реализовать одной или более аппаратными структурами видеокодера 20, включающими в себя блок 122 кодирования на основе палитры и/или блок 118 энтропийного кодирования (см. фиг. 2).

В одном примере изобретения видеокодер 20 может быть выполнен с возможностью кодирования

блока видеоданных с использованием режима кодирования на основе палитры и палитры (800), и генерации множества элементов синтаксиса, которые указывают палитру, которая использовалась для кодирования блока видеоданных, причем множество элементов синтаксиса включает в себя первый элемент синтаксиса, который указывает количество значений палитры для палитры, которые явно сигнализируются в битовом потоке кодированного видеосигнала (802). Видеокодер 20 может быть дополнительно выполнен с возможностью кодирования первого элемента синтаксиса с использованием одного или более кодов Голомба таким образом, что длина кодированного первого элемента синтаксиса меньше или равна заранее определенному количеству битов (804), и включения множества элементов синтаксиса в битовый поток кодированного видеосигнала (806).

В одном примере изобретения первый элемент синтаксиса является элементом синтаксиса `palette_num_signalled_entries`. В другом примере изобретения множество элементов синтаксиса включает в себя значения палитры, указанные как явно сигнализируемые первым элементом синтаксиса.

В одном примере изобретения заранее определенное максимальное количество битов равно 32 и один или более кодов Голомба являются экспоненциальным кодом Голомба порядка 0. В другом примере изобретения заранее определенное максимальное количество битов равно 32, и один или более кодов Голомба получены объединением усеченного кода Райса и экспоненциального кода Голомба.

В другом примере изобретения максимальное значение первого элемента синтаксиса задается относительно второго элемента синтаксиса, который указывает максимальный размер палитры, и третьего элемента синтаксиса, который указывает максимальный размер предсказателя палитры. В этом примере видеокодер 20 может быть дополнительно выполнен с возможностью задания второго элемента синтаксиса в качестве значения от 0 до первого порога и задания третьего элемента синтаксиса в качестве значения от 0 до второго порога. В одном примере первый порог принимает одно из значений 4095 и 4096 и второй порог принимает одно из значений 4095, 8191 и 8192.

В другом примере изобретения максимальное значение первого элемента синтаксиса задается относительно второго элемента синтаксиса, который указывает максимальный размер палитры, и третьего элемента синтаксиса, который указывает максимальный размер предсказателя палитры. В этом примере видеокодер 20 может быть дополнительно выполнен с возможностью задания второго элемента синтаксиса меньшим или равным количеству пикселей в наибольшем возможном блоке в битовом потоке кодированного видеосигнала, и задания третьего элемента синтаксиса меньшим или равным $K \cdot \text{значение второго элемента синтаксиса}$, где K - положительная постоянная. В одном примере, K равно 2.

В другом примере изобретения видеокодер 20 может быть дополнительно выполнен с возможностью сигнализировать элемент синтаксиса, указывающий тип серии палитры, в случае, когда текущий пиксель не является первым пикселем в порядке сканирования, и не сигнализировать элемент синтаксиса, указывающий тип серии палитры, в случае, когда текущий пиксель является первым пикселем в порядке сканирования и при наличии предыдущего/ей пикселя/выборки.

На фиг. 9 показана блок-схема операций, демонстрирующая пример способа декодирования видеосигнала согласно методам изобретения. Методы, представленные на фиг. 9, можно реализовать одной или более аппаратными структурами видеodeкодера 30, включающими в себя блок 165 декодирования на основе палитры и/или блок 150 энтропийного декодирования (см. фиг. 3).

В одном примере изобретения видеodeкодер 30 может быть выполнен с возможностью приема блока видеоданных в битовом потоке кодированного видеосигнала, причем блок видеоданных кодируется с использованием режима кодирования на основе палитры (900), и приема множества элементов синтаксиса, которые указывают палитру, которая использовалась для кодирования блока видеоданных, причем множество элементов синтаксиса включает в себя первый элемент синтаксиса, который указывает количество значений палитры для палитры, которые явно сигнализируются в битовом потоке кодированного видеосигнала, причем первый элемент синтаксиса кодируется с использованием одного или более кодов Голомба таким образом, что длина кодированного первого элемента синтаксиса меньше или равна заранее определенному максимальному количеству битов (902). Видеodeкодер 30 может быть дополнительно выполнен с возможностью декодировать множество элементов синтаксиса, в том числе декодировать первый элемент синтаксиса с использованием одного или более кодов Голомба (904), реконструировать палитру на основании декодированного множества элементов синтаксиса (906), и декодировать блок видеоданных с использованием реконструированной палитры (908). Видеodeкодер 30 может быть дополнительно выполнен с возможностью отображения декодированного блока видеоданных.

В одном примере изобретения первый элемент синтаксиса является элементом синтаксиса `palette_num_signalled_entries`. В другом примере изобретения множество элементов синтаксиса включает в себя значения палитры, указанные как явно сигнализируемые первым элементом синтаксиса.

В одном примере изобретения, заранее определенное максимальное количество битов равно 32, и один или более кодов Голомба являются экспоненциальным кодом Голомба порядка 0. В другом примере изобретения, заранее определенное максимальное количество битов равно 32 и один или более кодов Голомба получены объединением усеченного кода Райса и экспоненциального кода Голомба.

В другом примере изобретения максимальное значение первого элемента синтаксиса задается относительно второго элемента синтаксиса, который указывает максимальный размер палитры, и третьего

элемента синтаксиса, который указывает максимальный размер предсказателя палитры. В этом примере видеodecoder 30 может быть дополнительно выполнен с возможностью задания второго элемента синтаксиса в качестве значения от 0 до первого порога и задания третьего элемента синтаксиса в качестве значения от 0 до второго порога. В одном примере первый порог принимает одно из значений 4095 и 4096 и второй порог принимает одно из значений 4095, 8191 и 8192.

В другом примере изобретения максимальное значение первого элемента синтаксиса задается относительно второго элемента синтаксиса, который указывает максимальный размер палитры, и третьего элемента синтаксиса, который указывает максимальный размер предсказателя палитры. В этом примере видеodecoder 30 может быть дополнительно выполнен с возможностью задания второго элемента синтаксиса меньшим или равным количеству пикселей в наибольшем возможном блоке в битовом потоке кодированного видеосигнала, и задания третьего элемента синтаксиса меньшим или равным $K \cdot \text{значение второго элемента синтаксиса}$, где K - положительная постоянная. В одном примере K равно 2.

В другом примере изобретения видеodecoder 30 может быть дополнительно выполнен с возможностью принимать элемент синтаксиса, указывающий тип серии палитры, в случае, когда текущий пиксель не является первым пикселем в порядке сканирования, и выводить элемент синтаксиса, указывающий тип серии палитры, в случае, когда текущий пиксель является первым пикселем в порядке сканирования.

Хотя выше описаны конкретные комбинации различных аспектов методов, эти комбинации призваны лишь иллюстрировать примеры методов, описанных в этом изобретении. Соответственно методы этого изобретения не подлежат ограничению этими иллюстративными комбинациями и могут охватывать любые допустимые комбинации различных аспектов методов, описанных в этом изобретении.

В одном или более примерах описанные функции можно реализовать аппаратными средствами, программными средствами, программно-аппаратными средствами или любой их комбинацией. В случае реализации программными средствами функции могут храниться на или передаваться по, в виде одной или более инструкций или кода, компьютерно-считываемому носителю и выполняться аппаратным блоком обработки. Компьютерно-считываемые носители могут включать в себя компьютерно-считываемые носители данных, которые соответствуют материальному носителю, например носителям данных, или среды передачи данных, включающие в себя любую среду, которая позволяет переносить компьютерную программу с места на место, например, согласно протоколу связи. Таким образом, компьютерно-считываемые носители, в целом, могут соответствовать (1) материальным компьютерно-считываемым носителям данных, которые являются нетранзиторными, или (2) среде передачи данных, например сигналу или несущей волне. Носителями данных могут быть любые доступные носители, к которым могут обращаться один или более компьютеров или один или более процессоров для извлечения инструкций, кода и/или структур данных для реализации методов, описанных в этом изобретении. Компьютерный программный продукт может включать в себя компьютерно-считываемый носитель.

В порядке примера, но не ограничения, такие компьютерно-считываемые носители данных могут содержать RAM, ROM, EEPROM, CD-ROM или другое запоминающее устройство на основе оптического диска, запоминающее устройство на основе магнитного диска или другие магнитные запоминающие устройства, флеш-память, или любой другой носитель, который можно использовать для хранения нужного программного кода в форме инструкций или структур данных и к которым может обращаться компьютер. Кроме того, компьютерно-считываемым носителем можно называть любое соединение. Например, если инструкции передаются с веб-сайта, сервера или другого удаленного источника с использованием коаксиального кабеля, оптоволоконного кабеля, витой пары, цифровой абонентской линии (DSL), или беспроводных технологий, например инфракрасной, радио и микроволновой, то коаксиальный кабель, оптоволоконный кабель, витая пара, DSL или беспроводные технологии, например инфракрасная, радио и микроволновая включаются в определение носителя. Однако следует понимать, что компьютерно-считываемые носители данных и носители данных не включают в себя соединения, несущие волны, сигналы или другие переходные среды, но, напротив, относятся к непереходным, материальным носителям данных. Используемый здесь термин "диск" или "диск" включает в себя компакт-диск (CD), лазерный диск, оптический диск, цифровой универсальный диск (DVD), флоппи-диск и диск Blu-ray, где диски обычно воспроизводят данные магнитно, тогда как диски воспроизводят данные оптически с помощью лазеров. Комбинации вышеперечисленного также подлежат включению в понятие компьютерно-считываемых носителей.

Инструкции могут выполняться одним или более процессорами, например одним или более цифровыми сигнальными процессорами (DSP), микропроцессорами общего назначения, специализированными интегральными схемами (ASIC), вентильными матрицами, программируемыми пользователем (FPGA), или другими эквивалентными интегральными или дискретными логическими схемами. Соответственно, используемый здесь термин "процессор" может относиться к любой из вышеупомянутых структур или любой другой структуре, пригодной для реализации описанных здесь методов. Кроме того, в некоторых аспектах описанные здесь функциональные возможности могут обеспечиваться в специализированных аппаратных и/или программных модулях, выполненных с возможностью кодирования и декодирования, или включенных в объединенный кодек. Кроме того, методы можно полностью реализовать в одной или более схемах или логических элементах.

Методы этого изобретения можно реализовать в разнообразных устройствах, включающих в себя беспроводную телефонную трубку, интегральную схему (ИС) или набор ИС (например, набор микросхем). Различные компоненты, модули или блоки описаны в этом изобретении, чтобы подчеркнуть функциональные аспекты устройств, выполненных с возможностью осуществления раскрытых методов, но не обязательно требуют реализации разными аппаратными блоками. Напротив, как описано выше, различные блоки могут объединяться в аппаратном блоке кодека или обеспечиваться совокупностью взаимодействующих друг с другом аппаратных блоков, включающих в себя один или более вышеописанных процессоров, совместно с пригодным программным обеспечением и/или программно-аппаратным обеспечением.

Были описаны различные примеры изобретения. Возможна любая комбинация описанных систем, операций или функций. Эти и другие примеры включены в объем нижеследующей формулы изобретения.

ФОРМУЛА ИЗОБРЕТЕНИЯ

1. Способ декодирования видеоданных, причем способ содержит этапы, на которых
 - принимают блок видеоданных в битовом потоке кодированного видеосигнала, причем блок видеоданных кодируется с использованием режима кодирования на основе палитры;
 - принимают множество элементов синтаксиса, которые указывают палитру, которая использовалась для кодирования блока видеоданных, причем множество элементов синтаксиса включает в себя первый элемент синтаксиса, который указывает количество значений палитры для палитры, которые явно сигнализируются в битовом потоке кодированного видеосигнала, причем максимальное значение первого элемента синтаксиса задается относительно одного или более из второго элемента синтаксиса, который указывает максимальный размер палитры, и третьего элемента синтаксиса, который указывает максимальный размер предсказателя палитры, причем второй элемент синтаксиса имеет значение от 0 до первого порога, и третий элемент синтаксиса имеет значение от 0 до второго порога, и кодируется с использованием одного или более кодов Голомба таким образом, что длина кодированного первого элемента синтаксиса меньше или равна заранее определенному максимальному количеству битов;
 - декодируют множество элементов синтаксиса, в том числе декодируют первый элемент синтаксиса с использованием одного или более кодов Голомба;
 - реконструируют палитру на основании декодированного множества элементов синтаксиса и декодируют блок видеоданных с использованием реконструированной палитры.
2. Способ по п.1, дополнительно содержащий этап, на котором отображают декодированный блок видеоданных.
3. Способ по п.1, дополнительно содержащий этапы, на которых
 - принимают элемент синтаксиса, указывающий тип серии палитры, в случае, когда текущий пиксель блока видеоданных не является первым пикселем в порядке сканирования блока видеоданных; и
 - делают вывод, что элемент синтаксиса указывает тип серии палитры, в случае, когда текущий пиксель является первым пикселем в порядке сканирования.
4. Устройство, выполненное с возможностью декодирования видеоданных, причем устройство содержит
 - средство для приема блока видеоданных в битовом потоке кодированного видеосигнала, причем блок видеоданных кодируется с использованием режима кодирования на основе палитры;
 - средство для приема множества элементов синтаксиса, которые указывают палитру, которая использовалась для кодирования блока видеоданных, причем множество элементов синтаксиса включает в себя первый элемент синтаксиса, который указывает количество значений палитры для палитры, которые явно сигнализируются в битовом потоке кодированного видеосигнала, причем максимальное значение первого элемента синтаксиса задается относительно одного или более из второго элемента синтаксиса, который указывает максимальный размер палитры, и третьего элемента синтаксиса, который указывает максимальный размер предсказателя палитры, причем второй элемент синтаксиса имеет значение от 0 до первого порога, и третий элемент синтаксиса имеет значение от 0 до второго порога, и кодируется с использованием одного или более кодов Голомба таким образом, что длина кодированного первого элемента синтаксиса меньше или равна заранее определенному максимальному количеству битов;
 - средство для декодирования множества элементов синтаксиса, в том числе декодирования первого элемента синтаксиса с использованием одного или более кодов Голомба;
 - средство для реконструкции палитры на основании декодированного множества элементов синтаксиса и
 - средство для декодирования блока видеоданных с использованием реконструированной палитры.
5. Способ кодирования видеоданных, причем способ содержит этапы, на которых
 - кодируют блок видеоданных с использованием палитры и режима кодирования на основе палитры;
 - генерируют множество элементов синтаксиса, которые указывают палитру, которая использовалась для кодирования блока видеоданных, причем множество элементов синтаксиса включает в себя первый

элемент синтаксиса, который указывает количество значений палитры для палитры, которые явно сигнализируются в битовом потоке кодированного видеосигнала;

кодируют первый элемент синтаксиса с использованием одного или более кодов Голомба таким образом, что длина кодированного первого элемента синтаксиса меньше или равна заранее определенному максимальному количеству битов, причем максимальное количество битов задается относительно одного или более из второго элемента синтаксиса, который указывает максимальный размер палитры, и третьего элемента синтаксиса, который указывает максимальный размер предсказателя палитры, причем второй элемент синтаксиса имеет значение от 0 до первого порога и третий элемент синтаксиса имеет значение от 0 до второго порога; и

включают множество элементов синтаксиса в битовый поток кодированного видеосигнала.

6. Способ по п.1 или 5, в котором первый элемент синтаксиса является элементом синтаксиса `num_signalled_palette_entries`.

7. Способ по п.1 или 5, в котором заранее определенное максимальное количество битов равно 32 и при этом один или более кодов Голомба является экспоненциальным кодом Голомба порядка 0.

8. Способ по п.1 или 5, в котором заранее определенное максимальное количество битов равно 32 и при этом один или более кодов Голомба получен объединением усеченного кода Райса и экспоненциального кода Голомба.

9. Способ по п.1 или 5, в котором множество элементов синтаксиса включает в себя значения палитры, указанные как явно сигнализируемые первым элементом синтаксиса.

10. Способ по п.1 или 5, в котором первый порог принимает одно из значений 4095 и 4096 и второй порог принимает одно из значений 4095, 8191 и 8192.

11. Способ по п.1 или 5, в котором максимальное значение первого элемента синтаксиса задается относительно как второго элемента синтаксиса, который указывает максимальный размер палитры, и третьего элемента синтаксиса, который указывает максимальный размер предсказателя палитры.

12. Способ по п.1 или 5, в котором K равно 2.

13. Способ по п.5, дополнительно содержащий этапы, на которых

сигнализируют элемент синтаксиса, указывающий тип серии палитры, в случае, когда текущий пиксель не является первым пикселем в порядке сканирования; и

не сигнализируют элемент синтаксиса, указывающий тип серии палитры, в случае, когда текущий пиксель блока видеоданных является первым пикселем в порядке сканирования блока видеоданных.

14. Устройство, выполненное с возможностью кодирования видеоданных, причем устройство содержит

средство для кодирования блока видеоданных с использованием палитры и режима кодирования на основе палитры;

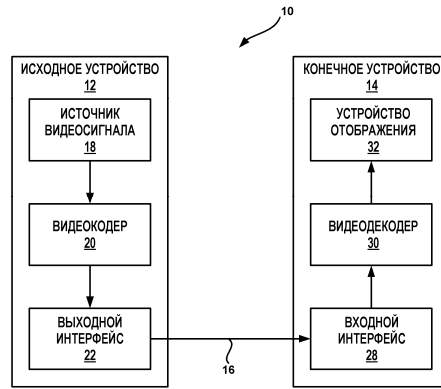
средство для генерации множества элементов синтаксиса, которые указывают палитру, которая использовалась для кодирования блока видеоданных, причем множество элементов синтаксиса включает в себя первый элемент синтаксиса, который указывает количество значений палитры для палитры, которые явно сигнализируются в битовом потоке кодированного видеосигнала;

средство для кодирования первого элемента синтаксиса с использованием одного или более кодов Голомба таким образом, что длина кодированного первого элемента синтаксиса меньше или равна заранее определенному максимальному количеству битов;

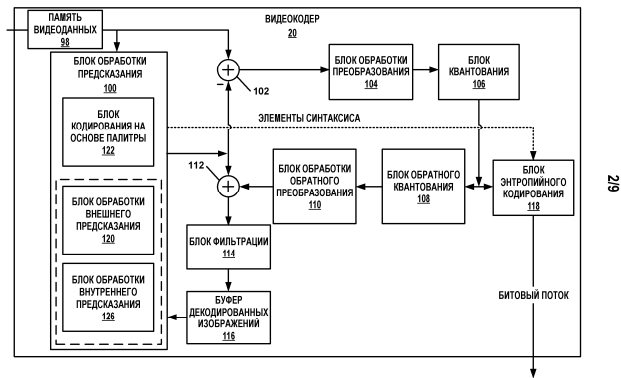
причем максимальное количество битов задается относительно одного или более из второго элемента синтаксиса, который указывает максимальный размер палитры, и третьего элемента синтаксиса, который указывает максимальный размер предсказателя палитры, причем второй элемент синтаксиса имеет значение от 0 до первого порога и третий элемент синтаксиса имеет значение от 0 до второго порога, и

средство для включения множества элементов синтаксиса в битовый поток кодированного видеосигнала.

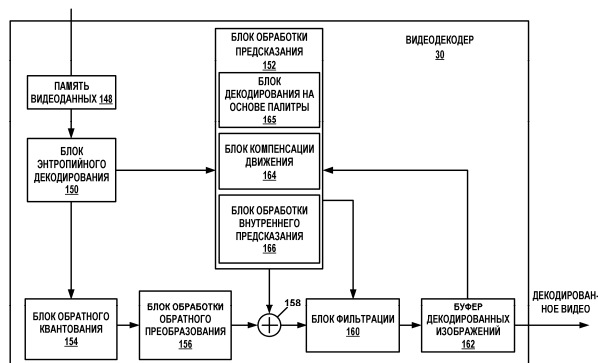
15. Компьютерно-считываемый носитель данных, хранящий инструкции, которые при выполнении предписывают одному или более процессорам устройства, выполненного с возможностью кодирования видеоданных, осуществлять способ по любому из пп.1-3 или 5-13.



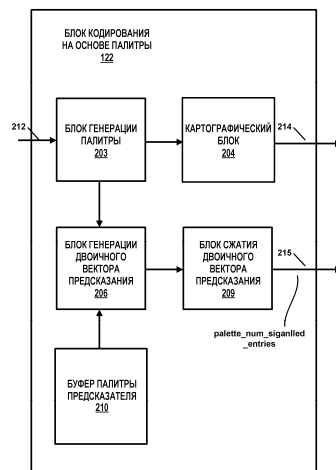
Фиг. 1



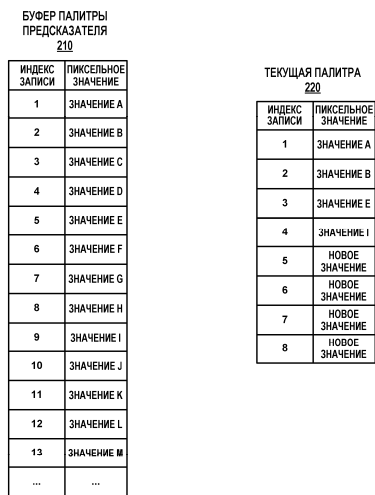
Фиг. 2



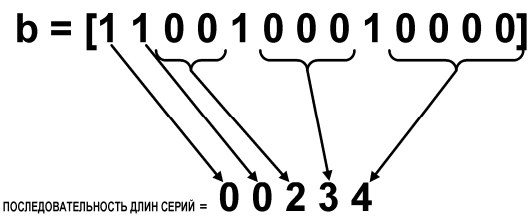
Фиг. 3



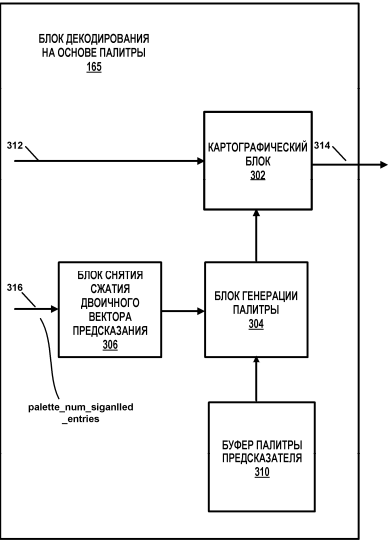
Фиг. 4



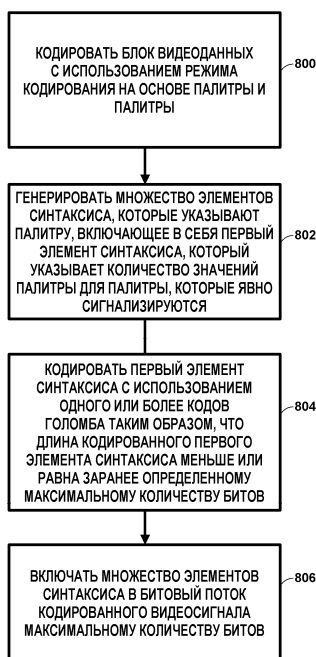
b = [1100100010000]
Фиг. 5



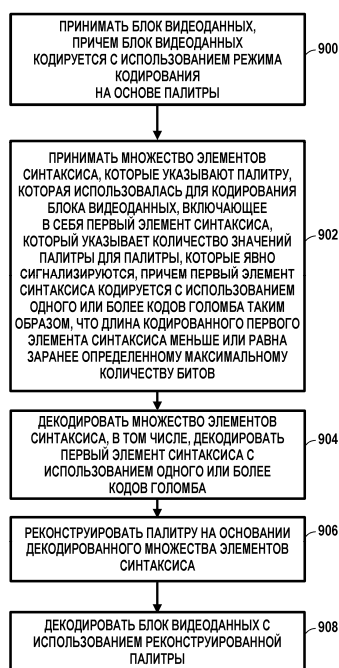
Фиг. 6



Фиг. 7



Фиг. 8



Фиг. 9

