



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК

H04N 19/523 (2019.02); H04N 19/593 (2019.02)

(21)(22) Заявка: 2017104967, 21.08.2015

(24) Дата начала отсчета срока действия патента:
21.08.2015

Дата регистрации:
19.08.2019

Приоритет(ы):

(30) Конвенционный приоритет:
22.08.2014 US 62/040,985;
20.08.2015 US 14/831,644

(43) Дата публикации заявки: 25.09.2018 Бюл. № 27

(45) Опубликовано: 19.08.2019 Бюл. № 23

(85) Дата начала рассмотрения заявки РСТ на
национальной фазе: 22.03.2017

(86) Заявка РСТ:
US 2015/046351 (21.08.2015)

(87) Публикация заявки РСТ:
WO 2016/029144 (25.02.2016)

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

(72) Автор(ы):

ПАН Чао (US),
РАПАКА Кришнакантх (US),
ВАН Е-Куй (US),
СОЛЕ РОХАЛЬС Джоэль (US),
КАРЧЕВИЧ Марта (US)

(73) Патентообладатель(и):

КВЭЛКОММ ИНКОРПОРЕЙТЕД (US)

(56) Список документов, цитированных в отчете
о поиске: B. LI et al. "Non-SCCE1: Unification
of intra BC and inter modes", JCTVC-R0100
(version 2), опубл. 29.06.2014 на 28 страницах
[найден 05.03.2019], размещено в Интернет
по адресу URL: http://phenix.it-sudparis.eu/jct/doc_end_user/current_document.php?id=9170. R.
JOSHI et al. "HEVC Screen Content Coding Draft
Text 1", JCTVC-R1005 (version 1), (см. прод.)

(54) ОБЪЕДИНЕННОЕ ПРЕДСКАЗАНИЕ МЕЖКАДРОВОЕ И С ВНУТРИКАДРОВОЙ КОПИЕЙ БЛОКА

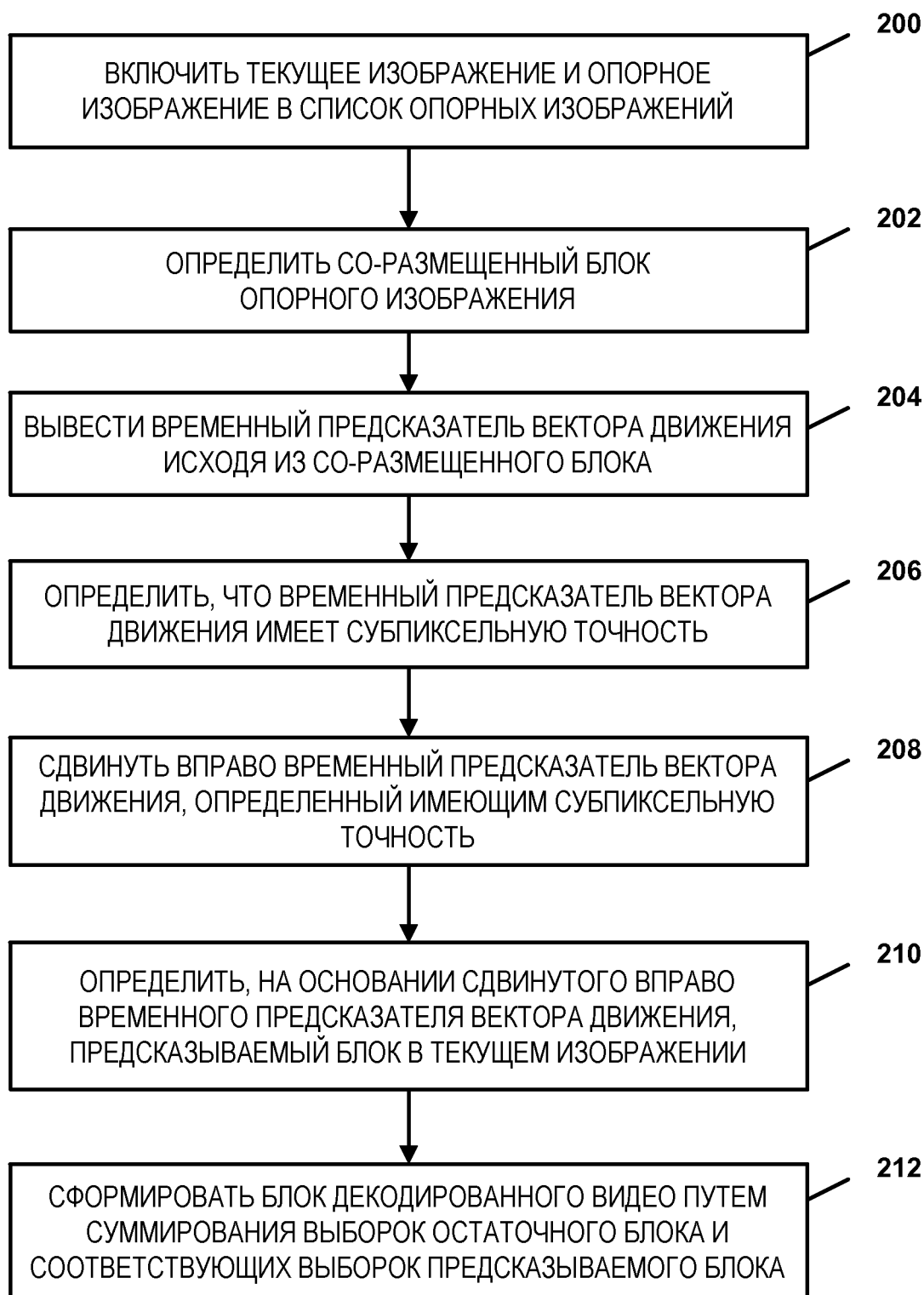
(57) Реферат:

Изобретение относится к вычислительной технике. Технический результат заключается в повышении эффективности декодирования. Способ декодирования видеоданных с использованием режима с внутрикадровой копией блока содержит: включение текущего изображения и опорного изображения в список опорных изображений, причем опорное изображение является отличным от текущего изображения; определение со-размещенного

блока опорного изображения, причем со-размещенный блок является со-размещаемым с текущим блоком текущего изображения; выведение временного предсказателя вектора движения, исходя из со-размещенного блока; определение, что временный предсказатель вектора движения имеет субпиксельную точность; сдвиг вправо временного предсказателя вектора движения, определенного имеющим субпиксельную точность; определение, на

основании сдвинутого вправо временного предсказателя вектора движения, предсказываемого блока в текущем изображении; формирование остаточного блока путем вычитания пиксельных значений предсказываемого блока из пиксельных значений текущего блока; и формирование

декодированного видеоблока путем суммирования выборок остаточного блока и соответствующих выборок предсказываемого блока, причем выборки остаточного блока соответствуют разности между текущим блоком и предсказываемым блоком. 6 н. и 22 з.п. ф-лы, 7 ил.



ФИГ. 6

(56) (продолжение):

опубл. 09.08.2014 на 358 страницах [найдено 05.03.2019], размещено в Интернет по адресу URL:http://phenix.it-sudparis.eu/jct/doc_end_user/current_document.php?id=9471. US 2014/0226721 A1, 14.08.2014. US 2014/0185682 A1, 03.07.2014. RU 2493669 C1, 20.09.2013.

R U 2 6 9 7 7 4 4 C 2

R U 2 6 9 7 7 4 4 C 2



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(12) **ABSTRACT OF INVENTION**

(52) CPC

H04N 19/523 (2019.02); H04N 19/593 (2019.02)(21)(22) Application: **2017104967, 21.08.2015**(24) Effective date for property rights:
21.08.2015Registration date:
19.08.2019

Priority:

(30) Convention priority:
22.08.2014 US 62/040,985;
20.08.2015 US 14/831,644(43) Application published: **25.09.2018 Bull. № 27**(45) Date of publication: **19.08.2019 Bull. № 23**(85) Commencement of national phase: **22.03.2017**(86) PCT application:
US 2015/046351 (21.08.2015)(87) PCT publication:
WO 2016/029144 (25.02.2016)Mail address:
129090, Moskva, ul. B.Spasskaya, 25, stroenie 3,
OOO "Yuridicheskaya firma Gorodisskij i
Partnery"

(72) Inventor(s):

PANG Chao (US),
RAPAKA Krishnakanth (US),
WANG Ye-Kui (US),
SOLE ROJALS Joel (US),
KARCZEWICZ Marta (US)

(73) Proprietor(s):

QUALCOMM INCORPORATED
INTERNATIONAL IP ADMINISTRATION
GROUP (US)(54) **COMBINED PREDICTION INTERFRAME AND INTRAFRAME COPY OF BLOCK**

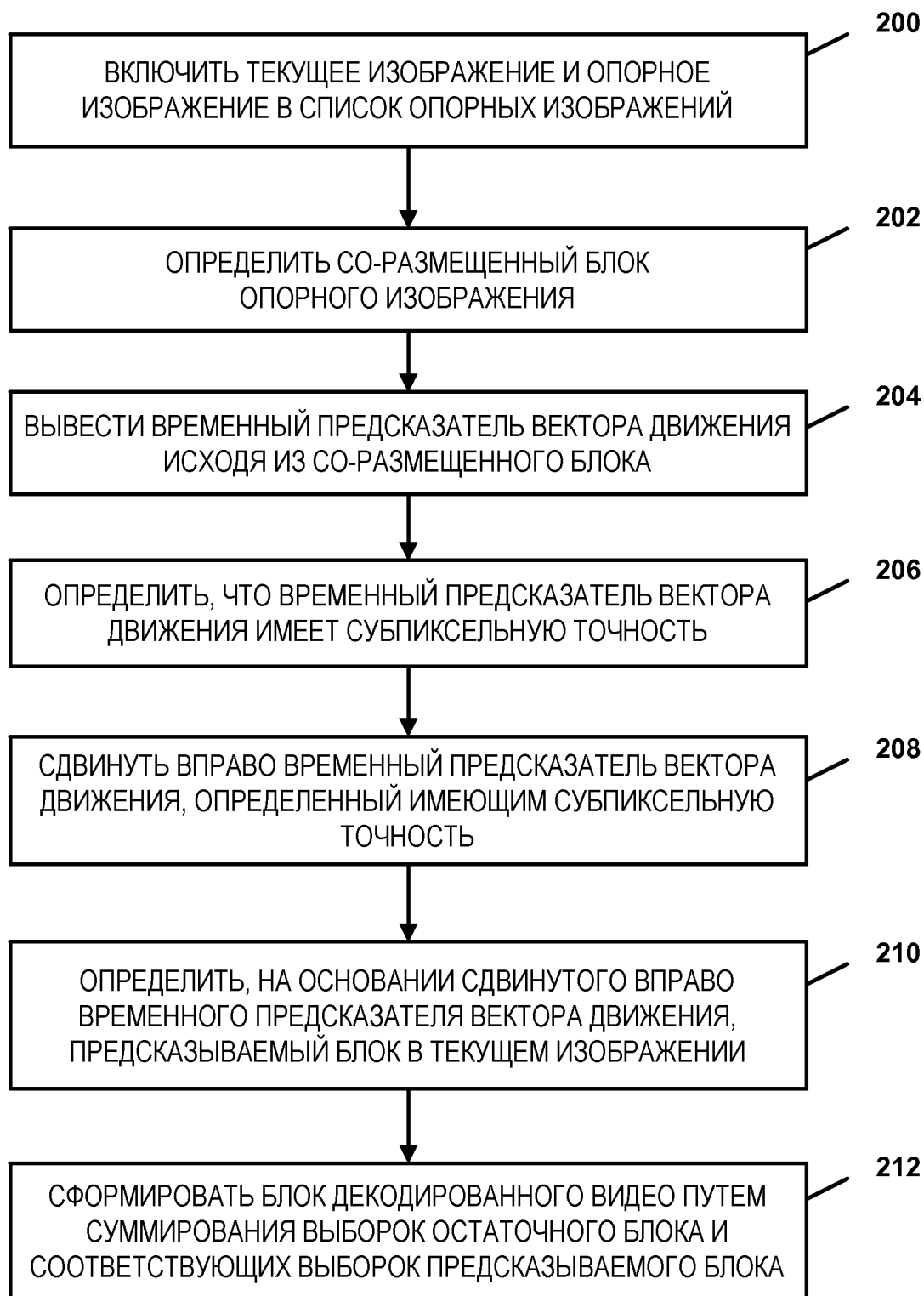
(57) Abstract:

FIELD: physics.

SUBSTANCE: invention relates to the computer equipment. Method of decoding video data using a mode with an intraframe copy of a unit comprises: including a current image and a reference image in a list of reference images, wherein the reference image is different from the current image; determining co-arranged reference image block, wherein co-arranged unit is co-arranged with current block of current image; deriving temporal predictor of motion vector, based on co-arranged unit; determining that temporal predictor of motion vector has subpixel accuracy; shifting to the

right a temporal predictor of a motion vector determined having sub-pixel accuracy; determining, based on the right-shifted temporary predictor of the motion vector, the predicted block in the current image; formation of residual block by subtraction of pixel values of predicted block from pixel values of current block; and formation of decoded video block by summation of residual block samples and corresponding samples of predicted block, wherein residual unit samples correspond to difference between current unit and predicted unit.

EFFECT: technical result consists in improvement



ФИГ. 6

РОДСТВЕННЫЕ ЗАЯВКИ

[0001] По настоящей заявке на патент испрашивается приоритет предварительной заявки на патент США с порядковым номером №62/040985, поданной 22 августа 2014, которая тем самым полностью включена в настоящий документ путем ссылки.

ОБЛАСТЬ ТЕХНИКИ, К КОТОРОЙ ОТНОСИТСЯ ИЗОБРЕТЕНИЕ

[0002] Настоящее раскрытие относится к кодированию видео и декодированию видео.
УРОВЕНЬ ТЕХНИКИ

[0003] Возможности цифрового видео могут быть включены в широкий спектр устройств, включающих в себя цифровые телевизоры, системы цифрового прямого вещания, беспроводные вещательные системы, персональные цифровые ассистенты (PDA), переносные или настольные компьютеры, планшетные компьютеры, устройства для чтения электронных книг, цифровые фотоаппараты, цифровые записывающие устройства, цифровые мультимедийные проигрыватели, видеоигровые устройства, видеоигровые консоли, сотовые или спутниковые радиотелефоны, так называемые "смартфоны", видео устройства видеоконференцсвязи, устройства потоковой передачи видео, и т.п.. Цифровые видеоустройства реализуют способы сжатия видео, например, описанные в стандартах, определенных в MPEG 2, MPEG 4, ITU-T H.263, ITU-T H.264/MPEG-4, Часть 10, Усовершенствованное кодирование видео (AVC), стандарте High Efficiency Video Coding (высокоэффективное кодирование видео, HEVC), разрабатываемом в настоящее время, и расширениях этих стандартов. Реализуя эти способы сжатия видео, видеоустройства могут более эффективно передавать, принимать, кодировать, декодировать и/или сохранять цифровую видеоинформацию.

[0004] В способах сжатия видео выполняется пространственное (внутрикадровое) предсказание и/или временное (межкадровое) предсказание для уменьшения или удаления избыточности, свойственной видеопоследовательностям. Для блочного кодирования видео «слайс» (slice) видео (то есть видеокادر или участок видеокадра) может быть разделен на видеоблоки, которые также могут называться древовидными блоками (treeblock), элементами кодирования (CU) и/или узлами кодирования. Видеоблоки во внутренне (в режиме intra) кодированном (I) слайсе изображения кодируются с использованием пространственного предсказания относительно опорных выборок (отсчетов) в соседних блоках в том же изображении. Видеоблоки в межкадрово (в режиме inter) кодированном (P или B) слайсе изображения могут использовать пространственное предсказание относительно опорных выборок в соседних блоках в том же изображении или временное предсказание относительно опорных выборок в других опорных изображениях. Изображения могут называться кадрами, и опорные изображения могут называться опорными кадрами.

[0005] Пространственное или временное предсказание приводит к использованию получаемого предсказанием блока для кодирования текущего блока. Остаточные данные представляют пиксельные разности между исходным блоком, подлежащим кодированию, и формируемым предсказанием (предсказываемым) блоком. Например, каждая соответствующая выборка из блока остаточных данных может быть равной разности между соответствующей выборкой исходного блока и соответствующей соответствующей выборкой предсказываемого блока. Межкадрово кодированный блок кодируется в соответствии с вектором движения, который указывает блок опорных выборок, образующих предсказываемый блок. Внутренне кодированный блок кодируется в соответствии с кодирование режимом внутрикадрового (intra) кодирования и остаточными данными. Для дополнительной компрессии остаточные данные могут быть преобразованы из пиксельной области в область преобразования, результатом

чего являются остаточные коэффициенты преобразования, которые затем могут быть квантованы. Квантованные коэффициенты преобразования, которые первоначально организованы в двумерный массив, можно сканировать, чтобы создать одномерный вектор коэффициентов преобразования, и для достижения еще большей компрессии

5 может применяться энтропийное кодирование.

СУЩНОСТЬ ИЗОБРЕТЕНИЯ

[0006] В общем, настоящее раскрытие описывает способы для предсказания с использованием внутрикадровой копии блока. В частности, настоящее раскрытие описывает различные способы для реализации объединенной схемы с внутрикадровой

10 копией блока и межкадрового предсказания. Например, кодер видео может декодировать или кодировать текущий блок видеоданных текущего изображения путем определения со-размещенного блока опорного изображения, причем со-размещенного блока, являющегося совместно размещенным с текущим блоком текущего изображения. Кодер видео может выводить временный предсказатель вектора движения исходя из

15 со-размещенного блока и может определять, что временный предсказатель вектора движения имеет субпиксельную точность. По определению, что временный предсказатель вектора движения имеет субпиксельную точность, кодер видео может сдвигать вправо временный предсказатель вектора движения. Сдвиг вправо временного предсказателя вектора движения может удалить младший значащий бит(ы),

20 представляющий субпиксельную точность временного вектора движения, эффективно преобразовывая временный вектор движения из имеющего субпиксельную степень точности к имеющему степень точности целого пикселя. Поскольку режим с внутрикадровой копией блока использует степень точности целого пикселя, преобразование временного предсказателя вектора движения может повысить

25 эффективность кодирования, поскольку кодер видео может использовать временный вектор движения, который иначе мог быть неприменимым или неэффективным для режима с внутрикадровой копией блока из-за субпиксельной точности. После сдвига вправо временного предсказателя вектора движения кодер видео может, на основании сдвинутого вправо временного предсказателя вектора движения, определять

30 предсказываемый блок внутри текущего изображения.

[0007] В одном примере настоящее раскрытие описывает способ декодирования видеоданных с использованием режима с внутрикадровой копией блока, который содержит включение текущего изображения и опорного изображения в список опорных изображений, опорное изображение является отличным от текущего изображения;

35 определение со-размещенного блока опорного изображения, со-размещенный блок, совместно размещаемый с текущим блоком текущего изображения; выведение временного предсказателя вектора движения исходя из со-размещенного блока; определение, что временный предсказатель вектора движения имеет субпиксельную точность; сдвиг вправо временного предсказателя вектора движения, определенного

40 имеющим субпиксельную точность; определение, на основании сдвинутого вправо временного предсказателя вектора движения, предсказываемого блока внутри текущего изображения; и формирование декодированного видеоблока путем суммирования выборок остаточного блока и соответствующих выборок предсказываемого блока.

[0008] В другом примере настоящее раскрытие описывает способ кодирования видеоданных с использованием режима с внутрикадровой копией блока, который содержит включение текущего изображения и опорного изображения в список опорных изображений, опорное изображение является отличным от текущего изображения;

45 определение со-размещенного блока опорного изображения, со-размещенный блок,

со-располагаемый с текущим блоком текущего изображения; выведение временного предсказателя вектора движения исходя из со-размещенного блока; определение, что временный предсказатель вектора движения имеет субпиксельную точность; сдвиг вправо временного предсказателя вектора движения, определенного имеющим субпиксельную точность; определение, на основании сдвинутого вправо временного предсказателя вектора движения, предсказываемого блока внутри текущего изображения; и формирование остаточных данных, представляющих разности между предсказываемым блоком, который был определен на основе сдвинутого вправо временного предсказателя вектора движения, и текущим блоком.

[0009] В другом примере настоящее раскрытие описывает устройство для кодирования видеоданных с использованием режима с внутрикадровой копией блока, которое содержит память, сконфигурированную для сохранения видеоданных изображения; и один или несколько процессоров, сконфигурированных чтобы: включать текущее изображение и опорное изображение в список опорных изображений, опорное изображение является отличным от текущего изображения; определять со-размещенный блок опорного изображения, со-размещенный блок, является совместно размещаемым с текущим блоком текущего изображения; выводить временный предсказатель вектора движения исходя из со-размещенного блока; определять, что временный предсказатель вектора движения имеет субпиксельную точность; сдвигать вправо временный предсказатель вектора движения, определенный имеющим субпиксельную точность; и определять, на основании сдвинутого вправо временного предсказателя вектора движения, предсказываемый блок внутри текущего изображения.

[0010] В другом примере настоящее раскрытие описывает устройство для декодирования видеоданных с использованием режима с внутрикадровой копией блока, устройство содержит: память, сконфигурированную для сохранения видеоданных опорного изображения; и один или несколько процессоров, сконфигурированных, чтобы: включать текущее изображение и опорное изображение в список опорных изображений, опорное изображение является отличным от текущего изображения; определять со-размещенный блок опорного изображения, со-размещенный блок является совместно размещаемым с текущим блоком текущего изображения; выводить временный предсказатель вектора движения исходя из со-размещенного блока; определять, что временный предсказатель вектора движения имеет субпиксельную точность; сдвигать вправо временный предсказатель вектора движения, определенный имеющим субпиксельную точность; определять на основании сдвинутого вправо временного предсказателя вектора движения предсказываемый блок внутри текущего изображения; и формировать блок декодированного видео путем суммирования выборок остаточного блока и соответствующих выборок предсказываемого блока.

[0011] В другом примере настоящее раскрытие описывает устройство для кодирования видеоданных с использованием режима с внутрикадровой копией блока, устройство содержит: память, сконфигурированную для сохранения видеоданных опорного изображения; и один или несколько процессоров, сконфигурированных чтобы: включать текущее изображение и опорное изображение в список опорных изображений, опорное изображение, является отличным от текущего изображения; определять со-размещенный блок опорного изображения, со-размещенный блок является совместно размещаемым с текущим блоком текущего изображения; выводить временный предсказатель вектора движения исходя из со-размещенного блока; определять, что временный предсказатель вектора движения имеет субпиксельную точность; сдвигать вправо временный предсказатель вектора движения, определенный имеющим субпиксельную точность;

определять на основании сдвинутого вправо временного предсказателя вектора движения предсказываемый блок внутри текущего изображения внутри текущего изображения; и генерировать остаточные данные, представляющие разности между предсказываемым блоком, который был определен на основании сдвинутого вправо временного предсказателя вектора движения, и текущим блоком.

[0012] В другом примере настоящее раскрытие описывает устройство для кодирования видеоданных с использованием режима с внутрикадровой копией блока, которое содержит средство для включения текущего изображения и опорного изображения в список опорных изображений, опорное изображение является отличным от текущего изображения; средство для определения со-размещенного блока опорного изображения, со-размещенный блок является совместно размещаемым с текущим блоком текущего изображения; средство для вывода временного предсказателя вектора движения исходя из со-размещенного блока; средство для определения, что временный предсказатель вектора движения имеет субпиксельную точность; средство для сдвига вправо временного предсказателя вектора движения, определенного имеющим субпиксельную точность; и средство для определения, на основании сдвинутого вправо временного предсказателя вектора движения, предсказываемого блока внутри текущего изображения.

[0013] В другом примере настоящее раскрытие описывает невременный читаемый компьютером носитель данных с наличием сохраненных на нем инструкций, которые при исполнении побуждают один или несколько процессоров вычислительного устройства, сконфигурированного, чтобы: включать текущее изображение и опорное изображение в список опорных изображений, опорное изображение является отличным от текущего изображения; определять со-размещенный блок опорного изображения, со-размещенный блок является совместно размещаемым с текущим блоком текущего изображения; выводить временный предсказатель вектора движения исходя из со-размещенного блока; определять, что временный предсказатель вектора движения имеет субпиксельную точность; сдвигать вправо временный предсказатель вектора движения, определенный имеющим субпиксельную точность; и определять на основании сдвинутого вправо временного предсказателя вектора движения предсказываемый блок внутри текущего изображения.

[0014] Подробности одного или нескольких примеров изложены на сопроводительных чертежах и описании ниже. Другие признаки, объекты и преимущества изобретения будут очевидными из описания, чертежей и пунктов формулы изобретения.

КРАТКОЕ ОПИСАНИЕ ЧЕРТЕЖЕЙ

[0015] Фиг.1 - блок-схема, иллюстрирующая примерную систему кодирования и декодирования видео, которая может использовать способы, описанные в настоящем раскрытии.

[0016] Фиг.2 - концептуальная схема, иллюстрирующая кандидаты-векторы движения для режимов слияния и усовершенствованного предсказания вектора движения (AMVP).

[0017] Фиг.3 - концептуальная схема, иллюстрирующая примерный способ для осуществления предсказания текущего блока видеоданных внутри текущего изображения согласно режиму (предсказания) с внутрикадровой копией блока (intra BC).

[0018] Фиг.4 - блок-схема, иллюстрирующая примерный кодер видео, который может реализовать способы, описанные в настоящем раскрытии.

[0019] Фиг.5 - блок-схема, иллюстрирующая примерный видеодекoder, который может реализовать способы, описанные в настоящем раскрытии.

[0020] Фиг.6 - структурная схема, иллюстрирующая примерный процесс для

декодирования видеоданных в соответствии со способами настоящего раскрытия.

[0021] Фиг.7 - структурная схема, иллюстрирующая примерный процесс для кодирования видеоданных в соответствии со способами настоящего раскрытия.

ПОДРОБНОЕ ОПИСАНИЕ СУЩНОСТИ ИЗОБРЕТЕНИЯ

5 [0022] Аспекты настоящего раскрытия направлены на способы для кодирования видео и декодирования видео. Например, кодер видео может выводить временный предсказатель вектора движения при выполнении Усовершенствованного предсказания вектора движения (AVMP). В свою очередь, кодер видео может сдвигать вправо временный предсказатель вектора движения, чтобы удалять какую-либо субпиксельную
10 точность. Так как режим с внутрикадровой копией блока использует целочисленную пиксельную точность, преобразование временного предсказателя вектора движения может повысить эффективность кодирования, поскольку кодер видео может использовать временный предсказатель вектора движения, который иначе мог быть неприменимым или неэффективным для режима с внутрикадровой копией блока из-за
15 субпиксельной точности. Аспекты настоящего раскрытия могут быть направлены на способы для усовершенствования кодирования вектора блока для режима intra Block Copy (BC). Аспекты настоящего раскрытия могут быть направлены на способы для выполнения intra BC в качестве части объединенного процесса межкадрового предсказания для кодирования видео и декодирования видео. Способы, описанные
20 здесь, могут применяться для кодирования контента экрана (SCC) или другого контента, где одно или несколько традиционных инструментальных средств кодирования являются неэффективными.

[0023] Межкадровое предсказание является способом предсказания элементов данных на основании элементов данных изображений отличных от текущего изображения.
25 Например, кодер видео может использовать межкадровое предсказание, чтобы определить предсказываемый блок для элемента (блока) предсказания (PU) для текущего изображения. Предсказываемый блок является блоком выборок, основанным на выборках в опорном изображении (то есть изображении, отличном от текущего изображения). Вектор движения для PU может указывать пространственное смещение
30 между блоком предсказания для PU и предсказываемым блоком. В этом примере видеокодер может сформировать остаточные данные (то есть ошибку предсказания), указывающие разность между предсказываемым блоком и блоком предсказания для PU. Видеокодер может включать данные, представляющие остаточные данные, в битовый поток, который содержит кодированное представление видеоданных.
35 Видеодекодер может использовать вектор движения, чтобы определить предсказываемый блок. Впоследствии, видеодекодер может суммировать соответствующие выборки остаточных данных и предсказываемого блока, чтобы восстановить блок предсказания.

[0024] В другом примере межкадрового предсказания, кодер видео может определять
40 вектор движения для текущего PU текущего изображения на основании временного предсказателя вектора движения (TMVP) в опорном изображении. Во многих примерах TMVP является вектором движения для PU, имеющим блок предсказания, который, по меньшей мере, частично со-размещен с блоком предсказания для PU. Например, кодер видео может принять TMVP в качестве вектора движения для текущего PU или
45 определить вектор движения для PU на основании вектора движения TMVP. Кодер видео затем может использовать вектор движения текущего PU, чтобы определить предсказываемый блок.

[0025] Внутрикадровое (Intra) предсказание является способом предсказания

элементов данных на основании элементов данных, таких как значения выборок, из того же декодированного слайса. Внутрикадровая копия блока (то есть intra BC) является вновь разработанной формой внутрикадрового предсказания, используемого для кодирования видеоблоков, таких как элементы кодирования (CU) или элементы PU.

- 5 Чтобы кодировать текущий видеоблок текущего изображения, используя intra BC, видеокодер осуществляет поиск в области поиска внутри того же слайса текущего изображения, что и текущий видеоблок, для предсказываемого блока. В некоторых примерах предсказываемый блок может быть блоком, содержащим массив выборок, которые наиболее близко соответствуют массиву выборок из текущего видеоблока.
- 10 Видеокодер генерирует вектор блока, указывающий смещение между текущим видеоблоком и предсказываемым блоком. Почти таким же образом, как и в межкадровом предсказании, видеокодер может использовать предсказываемый блок, чтобы генерировать остаточные данные. Видеокодер может использовать вектор блока текущего видеоблока, чтобы определить предсказываемый блок в текущем
- 15 изображении. Видеокодер может суммировать соответствующие выборки остаточных данных и предсказываемого блока, чтобы восстановить блок предсказания для текущего видеоблока.

- [0026] В общем, векторы движения, используемые для межкадрового предсказания, имеют субпиксельную точность. Например, вектор движения, используемый для
- 20 межкадрового предсказания, может иметь точность в четверть пикселя или половину пикселя. Таким образом, вектор движения, используемый для межкадрового предсказания, может указывать местоположение между двумя фактическими пикселями изображения. Фактические пиксели изображения могут называться целочисленными пикселями, поскольку фактическим пикселям присвоены целочисленные координаты
- 25 в массиве пикселей изображения. В случаях, где вектор движения указывает местоположение между двумя целочисленными пикселями, кодер видео (то есть видеокодер или видеодекодер) может интерполировать значения выборок. Напротив, векторы блоков, используемые для intra BC, ограничены точностью целого числа (то есть целочисленной точностью, точностью целого числа пикселей). Таким образом,
- 30 векторы блоков, используемые для intra BC, не указывают местоположения между целочисленными пикселями.

- [0027] Из-за сходств между межкадровым предсказанием и intra BC, имелись предложения, которые пробовали объединить межкадровое предсказание и intra BC. В некоторых таких попытках кодер видео формирует список опорных изображений.
- 35 Традиционно, список опорных изображений является списком изображений, доступных, чтобы использоваться текущим изображением для межкадрового предсказания. Несколько предложений объединить межкадровое предсказание и intra BC предлагают включение текущего изображения в список опорных изображений с тем, чтобы текущее изображение могло использоваться таким же образом, как другие опорные изображения.

- 40 [0028] Однако, несколько проблем появляются в таких предложениях. Например, видеокодер может определить, что использование intra BC приводит для текущего блока к лучшей метрике искажения в зависимости от скорости передачи, когда TMVP, который может быть вектором движения или вектором блока, используется в качестве вектора блока для текущего блока. Следовательно, в этом примере, видеокодер может
- 45 сигнализировать индекс кандидата, идентифицирующий TMVP, и может сигнализировать опорный индекс, указывающий позицию текущего изображения в списке опорных изображений. Видеокодер может также сигнализировать разность векторов движения (MVD), указывающую разность, если таковая имеется, между вектором блока для

текущего блока и TMVP (то есть вектором движения или вектором блока со-размещенного блока). Однако если TMVP оказывается вектором движения субпиксельной точности, вектор движения не может использоваться для intra BC или является неэффективным для использования в intraBC, поскольку intraBC имеет векторы
 5 блоков целочисленной точности. Различие в точности (то есть субпиксельная точность против целочисленной пиксельной точности) между межкадровым предсказанием и intra BC также означает, что эти два режима предсказания работают различно, поскольку межкадровое предсказание работает с использованием субпиксельной точности, и intra BC работает с использованием целочисленной пиксельной точности. Результатом
 10 является несовместимость, разрушающая обоснование для объединения межкадрового предсказания и intra BC.

[0029] Конкретные способы настоящего раскрытия обеспечивают решения таких проблем. Например, кодер видео может включать текущее изображение и опорное изображение в список опорных изображений. Кроме того, кодер видео может определять
 15 со-размещенный блок опорного изображения. Со-размещенный блок со-размещен с текущим блоком текущего изображения. Блок может быть со-размещен с другим блоком, если блоки соответствуют, по меньшей мере, частично перекрывающимся позициям внутри соответственных блоков в блоках. Вдобавок, в этом примере, кодер видео может выводить TMVP исходя из со-размещенного блока. Кодер видео затем
 20 может определять, имеет ли TMVP субпиксельную точность. В ответ на определение, что TMVP имеет субпиксельную точность, кодер видео может сдвигать вправо TMVP. Другими словами, кодер видео может сдвигать вправо значения, указывающие TMVP, так что удаляются какие-либо биты (например, два младших значащих бита), представляющие дробные части TMVP, имея в результате TMVP с целочисленной
 25 степенью точности. Например, если были удалены два младших значащих бита, это приравнивается смещению вправо на два. В качестве другого примера, смещение вправо TMVP, имеющего в HEVC точность в четверть пикселя, на два преобразует TMVP из имеющего субпиксельную точность (или точность в четверть пикселя в этом примере) в имеющий целочисленную пиксельную точность. Таким образом, кодер видео может
 30 эффективно преобразовывать TMVP из субпиксельной точности в целочисленную пиксельную точность. Кроме того, в этом примере, кодер видео может определить, на основании сдвинутого вправо временного предсказателя вектора движения, предсказываемый блок внутри текущего изображения и сформировать блок декодированного видео путем суммирования выборок остаточного блока и
 35 соответствующих выборок из предсказываемого блока. Таким образом, путем преобразования TMVP из субпиксельной точности к целочисленной точности, кодер видео может быть способным использовать TMVP в качестве вектора блока для использования в intra BC. Таким образом, этот способ из раскрытия может содействовать объединению межкадрового предсказания и intra BC.

[0030] Фиг.1 является блок-схемой, иллюстрирующей примерную систему 10 кодирования и декодирования видео, которая может использовать один или несколько способов из настоящего раскрытия. Как используется здесь, термин "кодер видео" относится в целом и к видеокодерам, и к видеodeкодерам. Подобным образом в настоящем раскрытии термины "кодирование видео" или "кодирование" в общем
 45 относятся к кодированию видео или декодированию видео. Как изложено по всему настоящему раскрытию, видеокодер 20 и видеodeкодер 30 системы 10 кодирования видео представляют различные примеры кодеров и декодеров, которые могут быть сконфигурированы для выполнения одного или большего числа способов из настоящего

раскрытия.

[0031] Как показано в примере по Фиг.1, система 10 кодирования видео включает в себя устройство-источник 12 и устройство-получатель 14. Устройство-источник 12 генерирует кодированные видеоданные. Соответственно, устройство-источник 12 может называться устройством кодирования видео или аппаратурой кодирования видео. Устройство-получатель 14 может декодировать кодированные видеоданные, сгенерированные устройством-источником 12. Соответственно, устройство-получатель 14 может называться устройством декодирования видео или аппаратурой декодирования видео. Устройство-источник 12 и устройство-получатель 14 могут быть примерами устройств кодирования видео или аппаратурой кодирования видео. Устройство-источник 12 и устройство-получатель 14 могут содержать любое устройство из широкого диапазона устройств, включающих в себя настольные компьютеры, блокнотные (то есть переносные) компьютеры, планшетные компьютеры, телевизионные абонентские приставки, устройства беспроводной связи, микротелефонные трубки, например, так называемые "смартфоны", так называемые "смарт-клавиатуры", телевизоры, камеры, устройства отображения, цифровые мультимедийные проигрыватели, видеоигровые консоли, устройство потокового видео, или подобное. В некоторых примерах устройство-источник 12 и устройство-получатель 14 оснащены для беспроводной связи.

[0032] Устройство-получатель 14 может принимать кодированные видеоданные от устройства-источника 12 через линию связи 16. Линия связи 16 может содержать один или несколько носителей или устройств, способных перемещать кодированные видеоданные из устройства-источника 12 в устройство-получатель 14. В одном примере, линия связи 16 содержит одну или несколько сред связи, которые позволяют устройству-источнику 12 передавать кодированные видеоданные непосредственно на устройство-получатель 14 в реальном времени. В этом примере устройство-источник 12 модулирует или передает кодированные видеоданные в соответствии со стандартом связи, например, протоколом беспроводной связи или проводной связи, и передает модулированные (или немодулированные) видеоданные на устройство-получатель 14. Одна или несколько сред связи могут включать в себя беспроводные и/или проводные среды связи, такие как радиочастотный (RF) спектр, или одну или большее число физических линий передачи. Одна или несколько сред связи могут составлять часть сети с пакетной передачей, такой как локальная сеть, широкомасштабная сеть или глобальная сеть (например, сеть Интернет). Одна или несколько сред связи могут включать в себя маршрутизаторы, коммутаторы, базовые станции или другое оборудование, которые содействуют передаче информации от устройства-источника 12 на устройство-получатель 14.

[0033] В некоторых примерах кодированные данные выводятся из выходного интерфейса 22 в запоминающее устройство 33. В таких примерах к кодированным данным доступ осуществляется из запоминающего устройства 33 посредством входного интерфейса 28. Запоминающее устройство 33 может включать в себя любой из множества распределенных или локально доступных носителей данных, например, накопитель на жестком диске, диски по технологии Blu-ray, DVD, CD-ROM, флэш-память, энергозависимую или энергонезависимую память или любые другие подходящие цифровые носители информации для хранения кодированных видеоданных.

[0034] Запоминающее устройство 33 может содержать файловый сервер или другое промежуточное запоминающее устройство, которое может хранить кодированное видео, сгенерированное устройством-источником 12. Устройство-получатель 14 может осуществлять доступ к сохраненным видеоданным из запоминающего устройства 33

посредством потоковой передачи или загрузки по сети. Файловый сервер может быть любым типом сервера, способным сохранять кодированные видеоданные и передавать эти кодированные видеоданные на устройство-получатель 14. Примерные файловые серверы включает в себя веб-сервер (например, для веб-сайта), сервер по протоколу передачи файлов (FTP), устройства сетевой системы хранения данных (NAS) или локальный накопитель на дисках.

[0035] Устройство-получатель 14 может осуществлять доступ к кодированным видеоданным через любое стандартное информационное соединение, включая Интернет-соединение. Примерные типы информационных соединений могут включать в себя канал беспроводной связи (например, соединение Wi-Fi), проводное соединение (например, DSL, кабельный модем и т.д.) или комбинацию обоих, которая является подходящей для осуществления доступа к кодированным видеоданным, сохраненным на файловом сервере. Передача кодированных видеоданных из запоминающего устройства 33 может быть потоковой передачей, передачей посредством загрузки по сети или комбинацией обеих.

[0036] Способы настоящего раскрытия для объединенного предсказания межкадрового и с внутренней копией блока не обязательно ограничиваются беспроводными применениями или установками. Способы могут применяться к кодированию видео в поддержку любого из множества мультимедийных приложений, таких как эфирное телевизионное вещание, передачи кабельного телевидения, передачи спутникового телевидения, передачи потокового видео, например, через сеть Интернет, кодирование цифрового видео для сохранения на носителе данных, декодирование цифрового видео, сохраненного на носителе данных, или других приложений. В некоторых примерах система 10 сконфигурирована с возможностью поддержки односторонней или двусторонней передачи видео для поддержки приложений, таких как потоковая передача видео, воспроизведение видео, видеовещание и/или видеотелефония.

[0037] Система 10 кодирования видео, иллюстрируемая на Фиг.1, является просто примером, и способы настоящего раскрытия могут применяться к установкам кодирования видео (например, кодирования видео или декодирования видео), которые не обязательно включают в себя какую-либо передачу данных между устройствами кодирования и декодирования. В других примерах данные извлекаются из локальной памяти, передаются потоком по сети или подобное. Устройство кодирования видео может кодировать и сохранять данные в памяти, и/или устройство декодирования видео может извлекать и декодировать данные из памяти. Во многих примерах, кодирование и декодирование выполняется устройствами, которые не осуществляют связь друг с другом, а просто кодируют данные в память и/или извлекают и декодируют данные из памяти.

[0038] В примере по Фиг.1 устройство-источник 12 включает в себя источник 18 видео, видеокодер 20 и выходной интерфейс 22. В некоторых примерах выходной интерфейс 22 включает в себя модулятор/демодулятор (модем) и/или передатчик. Источник 18 видео может включать в себя источник, такой как устройство захвата видео, например, видеокамеру, видеоархив, содержащий ранее захваченное видео, интерфейс для внешнего видео для приема видео от поставщика видеоконтента и/или систему компьютерной график, чтобы генерировать данные компьютерной графики в качестве источника видео, или комбинацию таких источников. В одном примере, где источник 18 видео является видеокамерой, устройство-источник 12 и устройство-получатель 14 составляют так называемые «камерофоны» или видеотелефоны. Однако,

способы, описанные в настоящем раскрытии, могут быть применимыми к кодированию видео вообще, и могут применяться к приложениям беспроводной и/или проводной связи.

[0039] Захваченное, предварительно захваченное, или машинно-генерируемое видео может кодироваться посредством видеокодера 20. Кодированные видеоданные могут передаваться непосредственно на устройство-получатель 14 через выходной интерфейс 22 устройства-источника 12. Кодированные видеоданные могут к тому же (или альтернативно) сохраняться на запоминающем устройстве 33 для последующего доступа посредством целевого устройства 14 или других устройств для декодирования и/или воспроизведения.

[0040] Устройство-получатель 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 устройства-получателя 14 принимает кодированные видеоданные по линии 16 связи. Кодированные видеоданные, переданные по линии 16 связи, или обеспеченные на запоминающем устройстве 33, могут включать в себя множество синтаксических элементов, генерируемых видеокодером 20 для использования видеodeкодером, например, видеodeкодером 30, в ходе декодирования видеоданных. Такие синтаксические элементы могут быть включены в кодированные видеоданные, передаваемые по среде связи, сохраняемые на носителе данных или сохраняемые в файловом сервере.

[0041] Устройство 32 отображения может быть интегрировано с устройством-получателем 14 или являться внешним по отношению к нему. В некоторых примерах устройство-получатель 14 может включать в себя интегрированное устройство отображения, а также является сконфигурированным для обеспечения интерфейса с внешним устройством отображения. В других примерах устройство-получатель 14 может быть устройством отображения. В общем, устройство 32 отображения отображает декодированные видеоданные пользователю, и может содержать любое из множества устройств отображения, например, жидкокристаллическое устройство отображения (LCD), плазменное устройство отображения, устройство отображения на органических светодиодах (OLED) или другой тип устройства отображения.

[0042] Каждый из видеокодера 20 и видеodeкодера 30 может быть реализован в виде любого из множества подходящих схемных решений, например, один или несколько микропроцессоров, цифровые процессоры сигналов (DSP), специализированные интегральные схемы (ASIC), программируемые вентильные матрицы (FPGA), дискретная логика, аппаратные средства или любая их комбинация. Если способы реализованы частично в виде программного обеспечения, устройство может сохранять инструкции для программного обеспечения в подходящем, не-временном читаемом компьютером носителе данных и может исполнять инструкции в аппаратных средствах с использованием одного или нескольких процессоров для выполнения способов настоящего раскрытия. Любое из вышеупомянутого (включая аппаратные средства, программное обеспечение, комбинацию аппаратных средств и программного обеспечения, и т.д.), можно рассматривать как один или несколько процессоров. Каждый из видеокодера 20 и видеodeкодера 30 может быть включен в один или несколько кодеров или декодеров, любой из которых может быть интегрирован в виде части объединенного кодера/декодера (CODEC) в соответствующем устройстве.

[0043] Видеокодер 20 и видеodeкодер 30 может работать в соответствии со стандартом кодирования видеоизображений, например, недавно завершенным стандартом Высокоэффективного кодирования видеоизображений (HEVC), а также стандартом

Расширения области HEVC, разработанным Объединенной группой специалистов по кодированию видео (JCT-VC). Альтернативно, видеокодер 20 и видеодекодер 30 могут работать согласно другим собственным или промышленным стандартам, например, стандарт H.264 Международного союза электросвязи - сектора телекоммуникаций (ITU-T), альтернативно называемый MPEG 4, Часть 10, Усовершенствованное кодирование видео (AVC), или расширениям таких стандартов. Однако способы настоящего раскрытия не ограничиваются каким-либо конкретным стандартом кодирования. Другие примеры стандартов кодирования видео включают в себя MPEG-2 и H.263 ITU-T.

[0044] Недавний проект стандарта HEVC, называемый "HEVC Working Draft 10" или "WD10", описывается в документе JCTVC-L1003v34, авторов Bross и др., "High efficiency video coding (HEVC) text specification draft 10 (for FDIS & Last Call)" Объединенной совместной группы по вопросам кодирования видео (JCT-VC) в ITU-T SG16 WP3 и ISO/IEC JTC1/SC29/WG11, 12th Meeting: Geneva, CH, 14-23 января 2013, который с 30 августа 2013 является загружаемым по адресу: http://phenix.int-evry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v34.zip. Другой проект стандарта HEVC называется здесь "WD10 revisions", описанным в документе Bross и др., "Editors' proposed corrections to HEVC version 1", Объединенной совместной группы по вопросам кодирования видео (JCT-VC) ITU-T SG16 WP3 и ISO/IEC JTC1/SC29/WG11, 13th Meeting, Incheon, KR, апрель 2013, который с 30 августа 2013 является доступным по адресу: http://phenix.int-evry.fr/jct/doc_end_user/documents/13_Incheon/wg11/JCTVC-M0432-v3.zip.

[0045] Другой новый проект текстовых спецификаций HEVC описывается в проекте технических условий HEVC, и называемый в дальнейшем HEVC WD, является доступным по адресу http://phenix.int-evry.fr/jct/doc_end_user/documents/15_Geneva/wg11/JCTVC-O1003-v2.zip. Материал авторов Wang и др., "High Efficiency Video Coding (HEVC) Defect Report 4", Объединенная группа специалистов по кодированию видео (JCT-VC) ITU-T SG16 WP3 и ISO/IEC JTC1/SC29/WG11, 17th Meeting, Valencia, ES, 27 марта - 4 апреля 2014, документ: JCTVC-Q1003 (в дальнейшем, "JCTVC-Q1003") также описывает стандарт HEVC.

[0046] Каждый из видеокодера и видеодекодера 30 представляет пример кодера видео, сконфигурированного для выполнения любой комбинации способов, описанных в настоящем раскрытии. В некоторых примерах один или несколько способов, описанных здесь, сконфигурированы для использования с одним или несколькими стандартами кодирования видео. Примеры стандартов кодирования видео включает 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 его расширения Масштабируемого кодирования видео (Scalable Video Coding, SVC) и Многовидовое кодирование видео (Multiview Video Coding, MVC). Высокоэффективное кодирование видеоизображений (HEVC) является новым стандартом кодирования видео, разработанным Объединенной группой специалистов по кодированию видео (JCT-VC) Экспертной группы по кодированию видео (Video Coding Experts Group, VCEG) ITU-T и Экспертной группы по вопросам движущегося изображения (MPEG) ISO/IEC. Усилия по стандартизации HEVC являются/были основанными на эволюционирующей модели устройства кодирования видео, называемой Тестовой Моделью HEVC (НМ). НМ предполагает несколько дополнительных возможностей для устройств кодирования видео относительно существующих устройств согласно, например, H.264/AVC ITU-T. Например, тогда как H.264 обеспечивает девять режимов кодирования с внутрикадровым предсказанием, НМ может обеспечить целых

тридцать три режима кодирования с внутрикадровым предсказанием.

[0047] Хотя не показано на Фиг.1, в некоторых примерах, каждый из видеокодера 20 и видеodeкодера 30 интегрирован с аудиокодером и аудиodeкодером, и может включать в себя соответствующие блоки MUX-DEMUX (мультиплексор/демультиплексор), или другое аппаратное и программное обеспечение, чтобы обрабатывать кодирование и аудио, и видео в общем потоке данных или отдельных потоках данных. Если применимо, в некоторых примерах, блоки MUX-DEMUX удовлетворяют разработанному ITU протоколу H.223 мультиплексора или другим протоколам, таким как протокол дейтаграмм пользователя (UDP).

[0048] Настоящее раскрытие может в общем обращаться к "сигнализации" или "передаче" некоторой информации посредством видеокодера 20 на другое устройство, такое как видеodeкодер 30. Термин "сигнализация" или "передача" может в общем относиться к передаче синтаксических элементов и/или других данных, используемых для декодирования сжатых видеоданных. Такая передача может произойти в реальном или почти в реальном времени. В свою очередь, такая передача может происходить за промежуток времени, например, может происходить при сохранении синтаксических элементов на читаемом компьютером носителе данных в кодированном битовом потоке в момент кодирования, которые затем может быть извлечено посредством декодера в любое время после сохранения в этом носителе.

[0049] Видео последовательность обычно включает в себя серию видео кадров или изображений. Например, группа изображений (GOP) обычно содержит серию из одного или нескольких видеоизображений. GOP может включать синтаксические данные в заголовок GOP, заголовок одного или нескольких изображений, или в другом месте, которые описывают ряд изображений, включенных в GOP. Каждый слайс изображения может включать в себя синтаксические данные слайса, которые описывают режим кодирования для соответственного слайса. Видеокодер 20 обычно оперирует блоками видео внутри отдельных видеослайсов для того, чтобы кодировать видеоданные.

[0050] В HEVC и других стандартах кодирования видео, видео последовательность обычно включает в себя серию изображений. Изображения могут также называться "кадрами". Изображение может включать три массива выборок, обозначенных S_L , S_{Cb} и S_{Cr} . S_L является двумерным массивом (то есть блоком) выборок яркости. S_{Cb} является двумерным массивом выборок сигналов цветности Cb . S_{Cr} является двумерным массивом выборок сигналов цветности Cr . Выборки сигналов цветности могут также называться здесь выборками "цветности" (chroma). В других примерах изображение может быть монохромным и может включать только массив выборок яркости (luma).

[0051] Чтобы сгенерировать кодированное представление изображения, видеокодер 20 может генерировать набор элементов дерева кодирования (CTU). Каждый соответственный CTU может быть блоком выборок яркости дерева кодирования, двумя соответствующими блоками выборок цветности дерева кодирования и синтаксическими структурами, используемыми для кодирования выборок блоков дерева кодирования. В монохромном изображении или изображении, которое имеет три отдельные цветовые плоскости, CTU может содержать одиночный блок дерева кодирования и синтаксические структуры, используемые для кодирования выборок блока дерева кодирования. Блок дерева кодирования может быть блоком из $N \times N$ выборок. Блок дерева кодирования может быть блоком из $N \times N$ выборок. CTU может также называться "древовидным блоком" или "наибольшим элементом кодирования" (LCU). CTU в HEVC могут в общих чертах походить на макроблоки из других стандартов, например, H.264/AVC. Однако

CTU не обязательно ограничивается конкретным размером и может включать один или несколько элементов кодирования (CU). Синтаксические данные в битовом потоке могут задавать размер для элементов CTU.

[0052] В HEVC наибольший элемент кодирования в слайсе называют блоком дерева кодирования (CTB). CTB содержит квадрадерево, узлы которого являются элементами кодирования. Размер CTB может находиться в пределах от 16×16 до 64×64 в основном профиле параметров HEVC (хотя технически могут поддерживаться размеры CTB 8×8). Каждый CU кодируется с помощью одного режима.

[0053] Видеокадр или изображение могут быть разделены на один или несколько слайсов. Слайс может включать в себя целое число CTU, упорядоченных последовательно в порядке кодирования, например, порядке растрового сканирования.

[0054] Каждый CTU может быть разложен на один или несколько элементов кодирования (CU) согласно квадрадереву. В общем, структура данных квадрадерева включает один узел на один CU, с корневым узлом, соответствующим CTU. Если CU разложен на четыре суб-CU, узел, соответствующий CU, включает в себя четыре дочерних узла, каждый из которых соответствует одному из суб-CU. Элемент кодирования (CU) может иметь одинаковый размер с CTB и может быть в пределах 8×8.

[0055] Каждый узел структуры данных квадрадерева может обеспечивать синтаксические данные для соответствующего CU. Например, узел в квадрадереве может включать в себя флаг разложения, указывающий, разложен ли CU, соответствующий узлу, на суб-CU. Синтаксические элементы для CU могут задаваться рекурсивно, и могут зависеть от того, разложен ли CU на суб-CU. Если CU далее не разлагается, CU называется CU-листом.

[0056] Видеокодер 20 может рекурсивно выполнять разбиение квадрадерева на блоках CTU дерева кодирования, чтобы разделить блоки дерева кодирования на блоки кодирования, отсюда название "блоки дерева кодирования". Блок кодирования может быть блоком из NxN выборов. В некоторых примерах CU содержит блок кодирования из выборов яркости и два соответствующих блока кодирования из выборов цветности для изображения, которое имеет массив выборов яркости, массив выборов Cb и массив выборов Cr, и синтаксические структуры, используемые для кодирования выборов блоков кодирования. В монохромном изображении или изображении, которое имеет три отдельные цветовые плоскости, CU может содержать одиночный блок кодирования и синтаксические структуры, используемые для кодирования выборов блока кодирования.

[0057] CU имеет такое же назначение, как и макроблок по стандарту H.264, за исключением того, что CU не имеет различия по размеру. Размер, которому CU соответствует, может быть квадратным или прямоугольным по форме. Размер CU может находиться в диапазоне от 8×8 пикселей до размера древовидного блока максимально в 64×64 пикселей или больше. Древовидный блок может быть разложен на четыре дочерних узла (также называемых суб-CU), и каждый дочерний узел в свою очередь может быть родительским узлом и разложенным еще на четыре дочерних узла. Заключительный, неразлагаемый дочерний узел, называемый листовым узлом квадрадерева, содержит узел кодирования, также называемый CU-листом.

Синтаксические данные, связанные с кодированным битовым потоком, могут задавать максимальное количество раз, которое древовидный блок можно расщеплять, называемое максимальной глубиной CU, и может также задавать минимальный размер узлов кодирования. Соответственно, битовый поток может также задавать наименьший

элемент кодирования (SCU). Настоящее раскрытие использует термин "элемент", чтобы обращаться к любому из CU, который может далее включать один или несколько элементов предсказания (PU), или элементам преобразования (TU) в контексте HEVC, или подобным структурам данных в контексте других стандартов (например,

5 макроблоки и субблоки таковых в H.264/AVC).

[0058] CU включает в себя один или несколько элементов предсказания (PU) и один или несколько элементов преобразования (TU). Синтаксические данные, связанные с CU, могут описывать, например, разделение CU на один или несколько PU. Режимы разделения могут различаться между тем, является ли CU кодированным в режиме с пропуском или прямо, кодированным в режиме с внутрикадровым предсказанием,

10 или кодированным в режиме с межкадровым предсказанием. CU может быть разделен так, что PU в CU могут быть неквадратными по форме. Синтаксические данные, связанные с CU, могут также описывать, например, разделение CU на один или несколько TU согласно квадрадеру.

15 [0059] Таким образом, когда CU кодируется межкадрово, он может быть далее разделен на два элемента предсказания (PU) или становится только одним PU, когда дальнейшее разделение не применяют. Когда два PU присутствует в одном CU, PU могут быть прямоугольниками половинного размера или двумя прямоугольниками, при одном - в одну четверть размера CU и другом - в три четверти размера CU. В HEVC

20 наименьшими размерами PU являются 8×4 и 4×8. Когда CU является кодированным межкадрово, один набор информации движения присутствует для каждого PU в CU. Кроме того, каждый PU может быть кодирован с уникальным режимом межкадрового предсказания для вывода набора информации движения.

[0060] Видеокодер 20 может разделить блок кодирования в CU на один или несколько

25 блоков предсказания. Блок предсказания может быть прямоугольным (то есть квадратным или неквадратным) блоком выборок, на которых применяют одинаковое предсказание. PU в CU может содержать блок предсказания из выборок яркости, два соответствующих блока предсказания из выборок цветности для изображения и синтаксические структуры, используемые для предсказания выборок блока предсказания.

30 В монохромном изображении или изображении, которое имеет три отдельные цветовые плоскости, PU может содержать единственный блок предсказания и синтаксические структуры, используемые для предсказания выборок блока предсказания.

[0061] В общем, PU представляет пространственную область, соответствующую всему или участку соответствующего CU, и может включать данные для извлечения

35 опорной выборки для PU. Кроме того, PU включает данные, относящиеся к предсказанию. В некоторых примерах PU кодируется с использованием режима intra (с внутрикадровым предсказанием), или режима inter (с межкадровым предсказанием). В качестве другого примера, когда PU кодируется в межкадровом режиме, PU может включать данные, задающие один или несколько векторов движения для PU. Данные,

40 задающие вектор движения для PU, могут описывать, например, горизонтальный компонент вектора движения, вертикальный компонент вектора движения, разрешающую способность для вектора движения (например, точность в одну четвертую пикселя или точность в одну восьмую пикселя), опорное изображение, на которое указывает вектор движения и/или список опорных изображений (например, Список 0,

45 Список 1 или Список C) для вектора движения.

[0062] В качестве примера, HEVC поддерживает предсказание в PU различных размерах. Полагая, что размером конкретного CU является 2N×2N, HEVC поддерживает межкадровое предсказание в PU с размерами 2N×2N или N×N и межкадровое

предсказание в симметричных PU с размерами $2N \times 2N$, $2N \times N$, $N \times 2N$ или $N \times N$. HEVC также поддерживает асимметричное разделение для межкадрового предсказания в PU с размерами $2N \times nU$, $2N \times nD$, $nL \times 2N$ и $nR \times 2N$. В асимметричном разделении одно направление CU не разделяется, тогда как другое направление разделено на 25% и 75%.

- 5 Участок CU, соответствующий разделу в 25%, обозначен "n", за которым следует указание "Вверху" (U), "Внизу" (D), "Слева" (L) или "Справа" (R). Таким образом, например, " $2N \times nU$ " относится к CU $2N \times 2N$, который разделен по горизонтали с PU $2N \times 0,5N$ вверху и PU $2N \times 1,5N$ внизу.

- [0063] В настоящем раскрытии " $N \times N$ " и "N на N" может использоваться
10 взаимозаменяемо, чтобы обращаться к пиксельным размерностям видеоблока в терминах размерностей по вертикали и горизонтали, например, 16×16 пикселей или 16 на 16 пикселей. В общем, блок 16×16 имеет 16 пикселей в вертикальном направлении ($y=16$) и 16 пикселей в горизонтальном направлении ($x=16$). Аналогично, блок $N \times N$ обычно имеет N пикселей в вертикальном направлении и N пикселей в горизонтальном
15 направлении, где N представляет неотрицательное целое значение. Пиксели в блоке могут быть организованы в строки и столбцы. Кроме того, блоки не обязательно должны иметь такое же число пикселей в горизонтальном направлении, как и в вертикальном направлении. Например, блоки могут содержать $N \times M$ пикселей, где M не обязательно равно N. Блок преобразования может быть прямоугольным блоком
20 выборки, на которых применяется одинаковое преобразование. Элемент преобразования (TU) в CU может содержать блок преобразования из выборки яркости, два соответствующих блока преобразования из выборки цветности и синтаксические структуры, используемые для преобразования выборки блока преобразования. Таким образом, каждый TU в CU может иметь блок преобразования яркости, блок
25 преобразования S_b , и блок преобразования S_r . Блок преобразования яркости в TU может быть суб-блоком остаточного блока яркости для CU. Блок преобразования S_b может быть суб-блоком остаточного блока S_b для CU. Блок преобразования S_r может быть суб-блоком остаточного блока S_r для CU. В монохромном изображении или изображении, которое имеет три отдельные цветовые плоскости, TU может содержать
30 одиночный блок преобразования и синтаксические структуры, используемые для преобразования выборки блока преобразования. TU может быть квадратным или неквадратным (например, прямоугольным) по форме. Другими словами, блок преобразования, соответствующий TU, может быть квадратным или неквадратным по форме.

- [0064] Стандарт HEVC допускает преобразования согласно TU, которые могут быть
35 различными для различных CU. Размеры TU обычно определяют на основании размера PU внутри данного CU, заданного для разделенного LCU, хотя это не всегда может иметь место. TU обычно имеют такой же или меньший размер, чем PU. В некоторых примерах остаточные выборки, соответствующие CU, подразделяют на меньшие модули, используя структуру квадрадерева, известную как "остаточное квадрадерево" (RQT).
40 Листовые узлы RQT могут называться единицами TU. Значения пиксельной разности, связанные с единицами TU, могут быть преобразованы, чтобы создать коэффициенты преобразования, которые могут быть квантованы.

- [0065] TU могут указываться с использованием RQT (также называемой структурой
45 квадрадерева TU), как обсуждено выше. Например, флаг разложения может указывать, разложен ли CU-лист на четыре TU. Затем, каждый TU может быть разложен далее на последующие суб-TU. Когда TU далее не разлагается, он может называться TU-листом. Обычно, для внутреннего кодирования, все TU-листы, принадлежащие CU-листу,

совместно использует одинаковый режим внутрикадрового предсказания. Таким образом, такой же режим внутрикадрового предсказания обычно применяется для вычисления предсказанных значений для всех TU CU-листа. Для внутреннего кодирования видеокодер 20 может вычислять остаточное значение для каждого TU-листа, используя режим внутрикадрового предсказания, в виде разности между участком CU, соответствующего TU, и исходным блоком. TU не обязательно ограничивается размером PU. Таким образом, TU могут быть больше или меньше чем PU. Для внутреннего кодирования PU может быть со-размещен с соответствующим TU-листом для того же CU. В некоторых примерах максимальный размер TU-листа соответствует размеру соответствующего CU-листа.

[0066] Кроме того, TU в CU-листах также могут быть связаны с соответственными структурами данных квадрадерева, называемыми RQT. То есть CU-лист может включать в себя квадрадерево, указывающее, каким образом CU-лист разделен на TU. Корневой узел квадрадерева TU обычно соответствует CU-листу, тогда как корневой узел квадрадерева CU обычно соответствует древовидному блоку. В общем, настоящее раскрытие использует термины CU и TU для обращения к CU-листу и TU-листу, соответственно, если не указано иное.

[0067] Единицы и PU, и TU могут содержать (то есть представлять собой) один или несколько блоков выборки, соответствующих каждому из каналов цветового пространства, связанного с этим блоком. Блоки единиц PU могут включать в себя выборки предсказываемого блока, и блоки единиц TU могут быть блоками, которые включают в себя остаточные выборки, соответствующие разности между исходным блоком и предсказываемым блоком. Для блоков, связанных с цветовым пространством YCbCr, блоки выборки яркости могут соответствовать каналу "Y", и два различных канала блоков цветности могут соответствовать каналам Cb и Cr, соответственно. Следуя кодированию с внутрикадровым предсказанием или межкадровым предсказанием с использованием PU в CU, видеокодер 20 или видеодекодер 30 могут вычислять остаточные данные для элементов TU из CU. PU могут содержать синтаксические данные, описывающие способ или режим формирования предсказываемых пиксельных данных в пространственной области (также называемой пиксельной областью), и TU могут содержать коэффициенты в области преобразования, являющиеся результатом применения к остаточным видеоданным преобразования, например, дискретного косинусного преобразования (DCT), целочисленного преобразования, вейвлет-преобразования или концептуально подобного преобразования. Остаточные данные могут соответствовать пиксельным разностям между пикселями некодированного изображения и значениями по предсказанию, соответствующими единицам PU. Видеокодер 20 или видеодекодер 30 могут создать единицы TU, включающие остаточные данные для CU, и затем преобразовать единицы TU, чтобы создать коэффициенты преобразования для CU. Другими словами, видеокодер 20 может применить преобразование к блоку преобразования для TU, чтобы сформировать блок коэффициентов преобразования для TU. Видеодекодер 30 может применить обратное преобразование к блоку коэффициентов преобразования для TU, чтобы восстановить блок преобразования для TU.

[0068] После применения преобразований (если таковые имеются), чтобы создать коэффициенты преобразования, видеокодер 20 может выполнять квантование коэффициентов преобразования. Другими словами, видеокодер 20 может квантовать коэффициенты преобразования из блока коэффициентов преобразования. Квантование обычно относится к процессу, в котором квантуют коэффициенты преобразования,

чтобы по возможности уменьшить объем данных, используемых для представления коэффициентов, обеспечивая дополнительное сжатие. Процесс квантования может уменьшать битовую глубину, связанную с некоторыми или всеми коэффициентами. Например, n -битовое значение может быть округлено в меньшую сторону к m -битовому значению в ходе квантования, где n больше, чем m . Обратное квантование (то есть деквантование) может повышать битовые глубины некоторых или всех из коэффициентов.

[0069] После квантования, видеокодер 20 может осуществлять просмотр коэффициентов преобразования, создавая одномерный вектор из двумерной матрицы, включающей квантованные коэффициенты преобразования. Просмотр может быть спроектирован, чтобы помещать коэффициенты более высокой энергии (и, следовательно, более низкой частоты) в начале массива и помещать коэффициенты более низкой энергии (и, следовательно, более высокой частоты) в конце массива. В некоторых примерах видеокодер 20 использует predetermined порядок просмотра, чтобы просматривать квантованные коэффициенты преобразования для создания сериализованного вектора, который может быть энтропийно кодирован. В других примерах видеокодер 20 выполняет адаптивный просмотр. После просматривания квантованных коэффициентов преобразования, чтобы сформировать одномерный вектор, видеокодер 20 может энтропийно кодировать одномерный вектор, например, согласно контекстно-зависимому адаптивному двоичному арифметическому кодированию (CABAC), контекстно-зависимому кодированию с переменной длиной (CAVLC), синтаксически-ориентированному контекстно-адаптивному двоичному арифметическому кодированию (SBAC), энтропийному кодированию с разделением на интервалы вероятности (Probability Interval Partitioning Entropy, PIPE) или другой методологии энтропийного кодирования. Видеокодер 20 может также энтропийно кодировать синтаксические элементы, связанные с кодированными видеоданными, для использования видеодекодером 30 при декодировании видеоданных.

[0070] Для выполнения CABAC, видеокодер 20 может в рамках контекстной модели назначать контекст символу, подлежащему передаче. Контекст может относиться, например, к тому, являются ли соседние значения символа ненулевыми или нет. Для выполнения CAVLC, видеокодер 20 может выбирать код переменной длины для символа, подлежащего передаче. Кодовые слова в кодировании с переменной длиной слова (VLC) могут строиться так, что относительно более короткие коды соответствуют более вероятным символам, тогда как более длинные коды соответствуют менее вероятным символам. Таким образом, с использованием VLC можно добиться экономии битов, например, над использованием кодовых слов равной длины для каждого символа, подлежащего передаче. Определение вероятности может основываться на контексте, назначенном символу.

[0071] Видеокодер 20 может кроме того посылать синтаксические данные, такие как ориентированные на блоки синтаксические данные, ориентированные на кадры синтаксические данные и ориентированные на GOP синтаксические данные на видеодекодер 30, например, в заголовке кадра, заголовке блока, заголовке слайса или заголовке GOP. Синтаксические данные для GOP могут описывать ряд кадров в соответственной GOP, и синтаксические данные для кадра могут указывать режим кодирования/предсказания, используемый для кодирования соответствующего кадра.

[0072] В соответствии со стандартом HEVC, при использовании меж- или внутри-кадрового предсказания, информация движения может быть доступной видеокодеру 20 или видеодекодеру 30. Для каждого блока набор данных движения может быть

доступным. Набор данных движения может содержать данные движения для прямого и обратного направлений предсказания. Термины прямое и обратное направления предсказания предназначены для указания двух направлений предсказания, связанных с режимом двунаправленного предсказания. Термины "прямое" и "обратное" не обязательно имеют геометрическое значение. Предпочтительнее "прямое" и "обратное" соответствуют списку 0 (List 0) опорных изображений (RefPicList0), и списку 1 (List 1) опорных изображений (RefPicList1) для текущего изображения. Если только один список опорных изображений является доступным для изображения или слайса, то является доступным только RefPicList0, и информацией движения каждого блока слайса всегда является «прямое».

[0073] Для каждого направления предсказания данные движения должны содержать опорный индекс и вектор движения. Опорный индекс является индексом для конкретного изображения в списке опорных изображений. В некоторых примерах, для простоты, видеокодер полагает, что движение имеет связанный опорный индекс. Опорный индекс используется, чтобы идентифицировать опорное изображение в текущем списке опорных изображений (RefPicList0 или RefPicList1). Вектор движения имеет горизонтальную и вертикальную составляющую. Если CU кодируется межкадрово, один набор данных движения присутствует для каждого PU. Видеокодер использует режим межкадрового предсказания, связанный с PU для вывода набора данных движения для этого PU.

[0074] Счетчик очередности изображения (POC) широко используется в стандартах кодирования видео для идентификации очередности отображения изображения. Когда множественные кодированные видеопоследовательности присутствуют в битовом потоке, изображения с одинаковым значением POC могут быть ближе друг к другу в терминах очередности декодирования. Значения POC для изображений обычно используются для построения списка опорных изображений, вывода набора опорных изображений как в HEVC и масштабирования вектора движения.

[0075] В стандарте HEVC имеются два режима межкадрового предсказания, режимы слияния (merge) (каковое включает в себя режим пропуска, особый случай режима слияния) и усовершенствованного предсказания вектора движения (AMVP), соответственно, для элемента предсказания (PU). Либо в AMVP, либо в режиме слияния, кодер видео поддерживает список кандидатов-векторов движения (MV) множества предсказателей векторов движения. В некоторых примерах список кандидатов MV содержит до 5 кандидатов для режима слияния и только два кандидата для режима AMVP. Кодер видео формирует вектор(ы) движения, а также опорные индексы, подлежащие использованию при кодировании PU с использованием режима слияния, путем выбора одного кандидата из списка кандидатов MV.

[0076] Например, когда видеокодер 20 сигнализирует данные движения текущего PU, используя режим слияния, видеокодер 20 может формировать список кандидатов на слияние, который включает в себя один или несколько кандидатов слияния. Каждый из кандидатов слияния задает данные движения пространственного предсказателя вектора движения или временного предсказателя вектора движения (то есть TMVP). Пространственным предсказателем вектора движения может быть PU в текущем изображении (то есть изображении, которое включает в себя текущий PU). TMVP может быть PU во временном опорном изображении (то есть изображении, которое имеет место в другой момент времени от текущего изображения). Кандидат слияния, который указывает данные движения для TMVP, может называться "временным кандидатом слияния".

[0077] После формирования списка кандидатов слияния видеокодер 20 может

выбирать один из кандидатов слияния и включать, в битовый поток, синтаксический элемент, который указывает позицию внутри списка кандидатов слияния для выбранного кандидата слияния. Когда видеodeкодер 30 декодирует текущий PU, видеodeкодер 30 формирует такой же список кандидатов слияния. Кроме того, видеodeкодер 30 может
 5 получать синтаксический элемент из битового потока и может использовать синтаксический элемент для определения выбранного кандидата слияния в списке кандидатов слияния. Видеodeкодер 30 может затем использовать данные движения, указанные выбранным кандидатом слияния, в качестве данных движения для текущего PU. Таким образом, кандидат слияния может содержать набор данных движения,
 10 например, векторы движения, соответствующие обоим спискам опорных изображений (списку 0 и списку 1), и соответствующие опорные индексы. Если кодер видео идентифицирует кандидата слияния посредством индекса слияния, видео кодер использует идентифицированные опорные изображения для предсказания текущих блоков. Кoder видео также определяет связанные векторы движения для текущих
 15 блоков.

[0078] Режим AMVP является подобным режиму слияния в том, что видеodeкодер 20 формирует список кандидатов и выбирает кандидата из списка кандидатов. Кандидаты в списке кандидатов AMVP могут называться кандидатами предсказателя вектора движения (MVP). Кандидаты в списке кандидатов AMVP могут включать в себя
 20 кандидаты, указывающие векторы движения для пространственных предсказателей векторов движения и/или временных предсказателей векторов движения. Когда видеodeкодер 20 сигнализирует данные движения текущего PU, используя режим AMVP, видеodeкодер 20 может сигнализировать разность векторов движения (MVD) для текущего PU и опорный индекс в дополнение к сигнализации позиции выбранного кандидата
 25 MVP в списке кандидатов. MVD для текущего PU может указывать разность между вектором движения для текущего PU и вектором движения для выбранного кандидата из списка кандидатов AMVP. В однонаправленном предсказании видеodeкодер 20 может сигнализировать один MVD и один опорный индекс для текущего PU. В
 30 двунаправленном предсказании видеodeкодер 20 может сигнализировать два MVD и два опорных индекса для текущего PU. Таким образом, при кодировании блока с использованием AMVP для каждого потенциального направления предсказания или из списка 0, или из списка 1, видео кодер явно сигнализирует каждый опорный индекс наряду с индексом MVP согласно некоторым примерам. Индекс MVP является индексом к списку кандидатов MV, поскольку кандидат AMVP содержит только вектор движения.
 35 В режиме AMVP кодер видео может дополнительно уточнять предсказанные векторы движения.

[0079] Кроме того, когда данные движения текущего PU сигнализируются с использованием режима AMVP, видеodeкодер 30 может получать, из битового потока, MVD для текущего PU и индекс списка кандидатов. Видеodeкодер 30 может формировать
 40 такой же список кандидатов AMVP и может определять выбранного кандидата на основании указания позиции выбранного кандидата MVP в списке кандидатов AMVP. Видеodeкодер 30 может восстанавливать вектор движения для текущего PU путем сложения MVD с вектором движения, указанным посредством выбранного кандидата MVP. Таким образом, видеodeкодер 30 может определять, на основании, по меньшей
 45 мере частично, вектора движения, указанного посредством выбранного кандидата MVP и MVD, вектор движения для текущего PU. Видеodeкодер 30 может затем использовать восстановленный вектор движения или векторы движения текущего PU, чтобы генерировать предсказываемые блоки для текущего PU.

[0080] Соответственно, как описано выше, кодер видео осуществляет аналогичным образом вывод кандидатов для обоих режимов AMVP и слияния на основании одних и тех же пространственных и временных соседних блоков. Однако режимы слияния и AMVP отличаются в том, что кандидат слияния соответствует полному набору данных движения, тогда как кандидат AMVP содержит только один вектор движения для конкретного направления предсказания и опорный индекс. Кандидаты для обоих режимов выводят аналогичным образом из одних и тех же пространственных и временных соседних блоков. Фиг.2 является концептуальной схемой, иллюстрирующей кандидаты пространственных векторов движения для режимов слияния и AMVP. Таким образом, Фиг.2 иллюстрирует кандидаты пространственных MV для режимов слияния и AMVP

[0081] Кандидаты пространственных MV выводят исходя из соседних блоков, показанных на Фиг.2, для конкретного PU (PU_0), хотя способы формирования кандидатов на основе блоков отличаются для режимов слияния и AMVP. Например, в режиме слияния, имеются пять позиций возможных кандидатов пространственных MV: $\{a_0, a_1, b_0, b_1, b_2\}$, как проиллюстрировано на Фиг.2. Кодер видео определяет для каждой позиции кандидата MV, является ли кандидат MV в этой позиции доступным. Кандидаты могут не являться доступными, если блок в этом местоположении еще не был декодирован, как пример. Кодер видео может проверять, доступный ли кандидат MV, в следующем порядке: $\{a_1, b_1, b_0, a_0, b_2\}$.

[0082] В режиме AMVP видеокодер разделяет соседние блоки на две группы: левая группа, состоящая из блоков a_0 и a_1 , и соседняя-сверху группа, состоящая из блоков b_0, b_1 , и b_2 , как показано на Фиг.2. Для соседней слева группы кодер видео проверяет наличие блоков согласно следующему порядку в некоторых примерах: $\{a_0, a_1\}$. Для соседней поверх группы, кодер видео проверяет наличие соседних верхних блоков согласно следующему порядку в некоторых примерах: $\{b_0, b_1, b_2\}$. Для каждой группы потенциальный кандидат, который ссылается на то же опорное изображение, что и изображение, указанное сигнализированным опорным индексом, имеет высший приоритет, подлежащий выбору кодером видео, чтобы сформировать окончательный кандидат для группы. Возможно, что соседние блоки не содержат вектор движения, который указывает на такое же опорное изображение. В этом случае, если видеокодер не может найти вектор движения, который указывает на то же опорное изображение, что и изображение, указанное сигнализированным индексом, видеокодер выбирает первый доступный кандидат. Видеокодер масштабирует первый доступный кандидат, который образует окончательный кандидат. Масштабирование кандидата-вектора движения компенсирует временные разности в расстоянии в этом случае.

[0083] Как описано выше, PU или CU могут иметь блок выборки значений яркости, а также множественные соответствующие блоки выборки значений цветности. В некоторых примерах могут быть множественные выборки яркости, которые соответствуют каждой выборке цветности (называемой цветовой субдискретизацией). В некоторых примерах вектор движения выводят для компонента яркости текущего PU/CU. В таких примерах, до использования вектора движения для компенсации движения цветности, вектор движения масштабируют на основании формата дискретизации цветности. Таким образом, при определении вектора движения, кодер видео сначала выводит составляющую вектора движения яркости для PU или CU. Кодер видео может масштабировать вектор движения, чтобы определить вектор движения цветности на основании формата субдискретизации цветности. Например, вектор

движения может быть масштабирован различным образом в зависимости от того, кодируется ли текущее изображение с использованием формата 4:4:4, 4:2:2 или 4:2:0.

[0084] Вдобавок, в HEVC, кодер видео может делить LCU на параллельные области оценки движения (MER). Когда LCU является частью MER, кодер видео допускает, что только соседние PU, которые принадлежат различным MER, от текущего PU подлежат включению в процесс построения списка MVP слияния/пропуска. Кодер видео сигнализирует размер MER в наборе параметров изображения в виде синтаксического элемента (например, "log2_parallel_merge_level_minus2").

[0085] В некоторых примерах, например, когда размер MER больше чем NxN выборки, и когда наименьший размер возможного CU установлен в 2N×2N выборки, MER может оказывать влияние на доступность пространственно соседнего блока по отношению к оценке движения. Например, в вышеупомянутом случае, если пространственно соседний блок находится внутри того же MER, что и PU в текущий момент, видеокодер считает, что пространственно соседний блок является недоступным.

[0086] Предложения, относящиеся к способам Intra Block Copying (BC), были недавно приняты для введения в стандарт Расширений области (HEVC). Материал авторов D. Flynn, и др., "High Efficiency Video Coding (HEVC) Range Extensions text specification: Draft 7", Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 и ISO/IEC JTC 1/SC 29/WG 11, 17th Meeting: Valencia, ES, 27 марта - 4 апреля 2014, документ: JCTVC-Q1005_v4 (в дальнейшем, "JCTVC-Q1005_v4"), доступный по адресу <http://phenix.int-evry.fr/jct/doc_end_user/documents/17_Valencia/wg11/JCTVC-Q1005-v4.zip>, является проектом Расширений области HEVC. Расширения области HEVC также являются разрабатываемыми JCT-VC.

[0087] В этом документе текст технических условий HEVC, как в JCTVC-Q1003, может называться как "версия 1 HEVC". Технические условия расширения области могут стать версией 2 HEVC. Однако, в значительной степени, что касается предложенных способов, например, предсказание вектора движения, версия 1 HEVC и технические условия расширения области являются технически подобными. Следовательно, ссылка на изменения к версии 1 HEVC означает, что такие же изменения могут применяться к техническим условиям расширения области.

[0088] В последнее время началось исследование новых инструментальных средств кодирования для материала контента экрана, такого как текст и графика с движением, и исследователи предложили технологии, которые повышают эффективность кодирования для контента экрана. Поскольку имеется основание, что существенные улучшения эффективности кодирования могут быть получены путем использования характеристик контента экрана вместе с новыми специализированными инструментальными средствами кодирования, был выпущен Запрос предложений (CfP). Варианты использования и требования этого CfP описаны в документе N14174 MPEG.

В ходе 17^{ой} встречи JCT-VC исследователи установили тестовую модель SCC (SCM). Материал авторов Joshi и др., "High Efficiency Video Coding (HEVC) Screen Content Coding: Draft 1," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 и ISO/IEC JTC 1/SC 29/WG 11, 18th Meeting, Sapporo, JP, 30 июня - 9 июля 2014, документ JCTVC-R1005 (в дальнейшем, "JCTVC-R1005"), доступный по адресу: http://phenix.int-evry.fr/jct/doc_end_user/documents/18_Sapporo/wg11/JCTVC-R1005-v1.zip, является новым Рабочим Проектом (WD) по SCC.

[0089] Многие приложения, такие как удаленный рабочий стол, удаленные игры, беспроводные дисплеи, автомобильная информационно-развлекательная среда,

«облачные» вычисления, и т.д., становятся обычными в частом употреблении.

Видеоконтент в этих приложениях может состоять из комбинаций из естественного контента, текста, искусственной (машинной) графики, и т.д. В областях текста и

искусственной графики, часто существуют повторяющиеся образы (такие как знаки, пиктограммы, символы, и т.д.). Intra BC является способом, который дает возможность устранения избыточности в кодировании текста и искусственной графики и повышает эффективность внутреннего кодирования кадра, как описано в материале авторов С. Pang, и др., "Non-RCE3 Intra Motion Compensation with 2-D MVs", Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 и ISO/IEC JTC 1/SC 29/WG 11, 14th Meeting: Vienna, AT, 25 июля - 2 августа 2013, документ JCTVC-N0256 (в дальнейшем, "JCTVC-N0256").

[0090] Intra BC является способом, который дает возможность удаления пространственной избыточности некоторого вида и повышения эффективности внутреннего кодирования кадра. Больше подробностей относительно способов intra BC можно найти в JCTVC-N0256. Режим внутрикадрового предсказания Intra Block Copy (BC) была включен в действующее кодирование контента экрана (SCC). Пример Intra BC показан на Фиг.3, причем текущий блок (например, текущий CU или PU) предсказаны исходя из уже декодированного блока текущего изображения или слайса. Сигнал предсказания может быть восстановлен без внутриконтурной фильтрации артефактов, включая деблокирование и Адаптивное к выборке смещение (Sample Adaptive Offset, SAO). Для компонента яркости или компонентов цветности, которые кодируют с помощью Intra BC, блочную компенсацию выполняют с целочисленной блочной компенсацией согласно некоторым примерам; поэтому какая-либо интерполяция не требуется.

[0091] Фиг.3 является концептуальной схемой, иллюстрирующей пример способа с внутрикадровой копией блока. В примере по Фиг.3 текущее изображение 103 включает в себя текущий блок 102. В некоторых примерах текущим блоком 102 является CU. В других примерах текущим блоком 102 является PU. Кроме того, Фиг.3 иллюстрирует предсказываемый блок 104 (то есть сигнал предсказания) внутри текущего изображения 103. Кодер видео может использовать предсказываемый блок 104, чтобы предсказать текущий видеоблок 102 в соответствии с режимом intra BC.

[0092] Видеокодер 20 определяет предсказываемый блок 104 из набора ранее восстановленных блоков видеоданных. В примере по Фиг.3 область 108 поиска внутри изображения 103 включает в себя набор ранее восстановленных видеоблоков. Видеокодер 20 может определять предсказываемый видеоблок 104, чтобы предсказать текущий блок 102 из числа видеоблоков в области 108 поиска на основании анализа относительной эффективности и точности предсказания и кодирования текущего блока 102 на основании различных видеоблоков внутри области 108 поиска.

[0093] Видеокодер 20 определяет двумерный вектор 106 смещения, представляющий местоположение или смещение предсказываемого видеоблока 104 относительно текущего видеоблока 102. Настоящее раскрытие может также относиться к векторам, используемым в режиме внутрикадровой копии блока в качестве "векторов блоков". Вектор 106 смещения включает в себя составляющую 112 горизонтального смещения и составляющую 110 вертикального смещения, которые соответственно представляют горизонтальное и вертикальное смещение предсказываемого блока 104 относительно текущего блока 102. Видеокодер 20 может включать один или несколько синтаксических элементов, которые идентифицируют или задают двумерный вектор 106 смещения, например, которые задают составляющую 112 горизонтального смещения и

составляющую 110 вертикального смещения, в битовый поток кодированного видео. Видеодекодер 30 может декодировать один или несколько синтаксических элементов, чтобы определять вектор 106 смещения, и может использовать определенный вектор, чтобы идентифицировать предсказываемый блок 104.

5 [0094] Текущим блоком 102 может быть CU, или PU из CU. В некоторых примерах кодер видео (например, видеокодер 20 и/или видеодекодер 30) разлагает CU, который предсказан согласно intra BC, на ряд PU. В таких примерах кодер видео может определять соответственный (например, другой) двумерный вектор 106 смещения для каждого из PU в CU. Например, кодер видео может разложить CU $2N \times 2N$ на два PU $2N \times N$, два PU $N \times 2N$ или четыре PU $N \times N$. В качестве других примеров, кодер видео может разложить CU $2N \times 2N$ на PU $((N/2) \times N + (3N/2) \times N)$, PU $((3N/2) \times N + (N/2) \times N)$, $(N \times (N/2) + N \times (3N/2))$ PU, $(N \times (3N/2) + N \times (N/2))$ PU, четыре PU $(N/2) \times 2N$, или четыре PU $2N \times (N/2)$. В некоторых примерах видеокодер предсказывает CU $2N \times 2N$, используя PU $2N \times 2N$.

15 [0095] При кодировании блока с использованием режима intra BC, кодер видео определяет, что вектор 106 смещения имеет целочисленную пиксельную точность для обоих компонентов и яркости, и цветности. Использование целочисленной пиксельной точности для вектора движения избегает интерполяции опорного блока для формирования предсказываемого блока. Вектор блока предсказывают и сигнализируют с целочисленной пиксельной точностью. Согласно JCTVC-R1005, кодер видео первоначально устанавливает предсказатель вектора блока в $(-w, 0)$ в начале каждого СТВ, где "w" - ширина CU. Используя предсказатель вектора блока, видеокодер делает попытку предсказать вектор блока для текущего блока 102 на основании значения предварительно кодированного CU или PU, кодированного с использованием режима intra BC.

25 [0096] Кодер видео обновляет предсказатель вектора блока после того, как каждый CU или PU кодируют в intra BC. Если CU или PU не кодируют с использованием intra BC, видеокодер не изменяет предсказатель вектора блока. После определения предсказателя вектора блока для CU или PU, видеокодер 20 определяет разность между предсказателем и фактическим вектором блока, чтобы определить разность векторов блока. Видеокодер 20 может кодировать разность векторов блока, используя способ кодирования разности векторов движения по HEVC.

30 [0097] В некоторых примерах, intra BC разрешен на уровнях и CU, и PU. Для режима intra BC уровня PU, размеры разбиения PU $2N \times N$ и $N \times 2N$ поддерживаются для всех размеров CU. Когда наименьший возможный CU имеет размер $N \times N$ выборки, поддерживается разбиение PU.

35 [0098] Кодер видео может затем обрабатывать Intra BC подобно режиму Inter. Кроме того, авторы Bin Li и др., "Non-SCCE1: Unification of intra BC and inter modes", Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 18th Meeting: Sapporo, JP, 30 июня - 9 июля 2014, документ JCTVC-R0100 (в дальнейшем, "JCTVC-R0100") предложили способ для объединения режимов intra BC и межкадрового предсказания. В соответствии со способом из JCTVC-R0100, кодер видео добавляет текущее изображение в список опорных изображений. Поскольку текущее изображение находится в списке опорных изображений, кодер видео может использовать текущее изображение для Intra BC в качестве опорного изображения таким же образом, как видеокодер будет использовать временное опорное изображение (то есть опорное изображение, имеющее значение РОС, отличное от значения РОС текущего изображения). В соответствии со способом из JCTVC-R0100, кодер видео помечает текущее изображение как долгосрочное опорное изображение до декодирования текущего изображения, и

помечает текущее изображение как краткосрочное опорное изображение после декодирования текущего изображения. Кроме того, в соответствии со способом из JCTVC-R0100, когда разрешен режим Intra BC, кодер видео следует процессу синтаксического разбора и процессу декодирования P-слайса для I-слайса. Как
 5 определено в JCTVC-Q1003, I-слайс является слайсом, который декодируют, используя только внутрикадровое предсказание. Кроме того, как определено в JCTVC-Q1003, внутрикадровое предсказание является предсказанием, выведенным исходя только из элемента данных (например, значений выборок) того же декодированного слайса. Как
 10 определено в JCTVC-Q1003, P-слайс является слайсом, который можно декодировать с использованием внутрикадрового предсказания или межкадрового предсказания, используя не более чем один вектор движения и опорный индекс для предсказания значений выборок каждого блока. B-слайс является слайсом, который может быть декодирован с использованием внутрикадрового предсказания или межкадрового предсказания с использованием самое большее два вектора движения и опорные
 15 индексы, чтобы предсказать значения выборок для каждого блока.

[0099] Когда режим Intra BC обрабатывают как режим межкадрового предсказания, некоторые проблемы могут существовать в выводе временного предсказания вектора движения для (режима) AMVP или слияния, взаимодействии с ограниченным внутренним предсказанием, построении опорного списка, и так далее. Способы настоящего
 20 раскрытия улучшают объединение Intra BC и Межкадрового предсказания в случае, что кодер видео обрабатывает режим Intra BC как режим Межкадрового предсказания. В этом случае, кодер видео добавляет текущее изображение в опорный список 0 (то есть список 0 опорных изображений). Хотя режим Intra BC подобен режиму Межкадрового предсказания, кодер видео может определять, является ли блок
 25 кодированным в Intra BC в сопоставлении с кодированным в межкадровом режиме, путем проверки, является ли опорное изображение, связанное с текущим блоком, текущим изображением (то есть опорное изображение имеет такое же значение РОС, как и текущее изображение). Если опорное изображение блока является текущим изображением, блок относится к режиму Intra BC. Иначе, текущий блок является
 30 кодированным с использованием традиционного режима Межкадрового предсказания. Различные примеры в соответствии со способами настоящего раскрытия описаны здесь. Каждый из последующих примеров может применяться отдельно или совместно с одним или несколькими из других.

[0100] В некоторых примерах, когда кодер видео (например, видеокодер 20 и/или
 35 видеодекодер 30) добавляет текущее изображение к опорному списку 0, кодер видео помечает текущее опорное изображение как «долгосрочное» до декодирования текущего изображения. В этом случае, кодер видео может кодировать вектор блока для режима Intra BC, используя целочисленную пиксельную точность или субпиксельную точность, например, точность одной четвертой пикселя, как описано в стандарте HEVC. Когда
 40 кодер видео использует временное предсказание вектора движения для блока, и кодер видео кодирует вектор движения для Intra BC с целочисленной пиксельной точностью, кодер видео может использовать временный предсказатель вектора движения (TMVP) для режима Intra BC, режима AMVP или режима слияния с целочисленной пиксельной точностью. Кодер видео может использовать TMVP с целочисленной пиксельной
 45 точностью, когда со-размещенный блок является блоком Intra BC, или с субпиксельной точностью, когда со-размещенный блок является обычным блоком режима Inter.

[0101] В некоторых примерах кодер видео может определить, что TMVP имеет субпиксельную точность. В этом случае, кодер видео может сначала сдвигать вправо

вектор движения и затем использовать результирующий сдвинутый-вправо вектор движения в качестве TMVP. Например, кодер видео может сдвигать вправо двоичные значения в TMVP, так что какие-либо биты (например, два младших значащих бита), представляющие дробные части TMVP, удаляются, приводя к TMVP, имеющему

5 целочисленную степень точности. Другими словами, кодер видео может определять, что TMVP имеет субпиксельную точность, и сдвигать вправо TMVP, определенный имеющим субпиксельную точность. В этом примере смещение вправо TMVP преобразовывает TMVP из субпиксельной точности (предсказания с субпиксельной точностью?) в целочисленную пиксельную точность, тем самым делая TMVP

10 применимым или эффективным для кодирования текущего блока в режиме intra BC. В другом примере кодер видео может использовать вектор движения TMVP без сдвига вправо вектора движения, независимо от того, является ли точность вектора движения целочисленной точностью или субпиксельной точностью. При использовании вектора движения TMVP, без выполнения сдвига вправо, кодер видео интерпретирует вектор

15 движения субпиксельной точности как вектор движения, имеющий целочисленную пиксельную точность.

[0102] Таким образом, в примерах настоящего раскрытия там, где кодер видео выполняет процесс кодирования, который предусматривает смещение вправо TMVP, кодер видео определяет со-размещенный блок видеоданных опорного изображения,

20 отличного от текущего изображения. Кроме того, в таких примерах, кодер видео может выводить TMVP исходя из со-размещенного блока видеоданных. Кодер видео может сдвигать вправо TMVP. Кодер видео может кодировать текущий блок видеоданных текущего изображения, используя режим с внутрикадровой копией блока и сдвинутый вправо временный предсказатель вектора движения. В качестве части этого процесса

25 кодирования, кодер видео может определять, является ли со-размещенный блок видеоданных кодированным в режиме intra BC, или кодированным в режиме межкадрового предсказания. Поскольку видеоблоки, кодированные с использованием режима intra BC, имеют векторы движения целочисленной пиксельной точности, и видеоблоки, кодированные с использованием межкадрового предсказания, имеют

30 векторы движения субпиксельной точности, кодер видео может быть способным определять на основании того, кодирован ли видеоблок с использованием режима intra BC или межкадрового предсказания, имеет ли TMVP субпиксельную точность или целочисленную пиксельную точность. Таким образом, кодер видео может быть сконфигурирован для сдвига вправо TMVP только в ответ на определение, что со-

35 размещенный блок видеоданных является кодированным в режиме межкадрового предсказания.

[0103] В некоторых примерах видеокодер 20 и/или видеодекодер 30 могут определять, имеет ли временный предсказатель вектора движения целочисленную пиксельную точность или субпиксельную точность. Субпиксельная точность может также называться

40 дробной пиксельной точностью или нецелочисленной пиксельной точностью. Видеокодер 20 и/или видеодекодер 30 могут быть сконфигурированы для сдвига вправо временного предсказателя вектора движения только в ответ на определение, что временный предсказатель вектора движения имеет субпиксельную точность.

[0104] Как указано выше, кодер видео может сдвигать вправо TMVP, чтобы

45 преобразовать TMVP из имеющего субпиксельную точность в имеющий целочисленную пиксельную точность. Например, кодер видео может сдвинуть вправо TMVP, чтобы удалить субпиксельную точность, таким образом, преобразовывая пиксельную точность TMVP из субпиксельной точности в целочисленную пиксельную точность. Например,

TMVP может первоначально иметь значения (3,25, -3,5) для горизонтальных и вертикальных составляющих, означая, что вектор движения указывает местоположение 3,25 пикселя вправо и 3,5 пикселя выше текущего блока. В этом примере сдвиг вправо для удаления субпиксельной точности дает в результате TMVP, имеющий значения (3,-3) для горизонтальной и вертикальной составляющих, означая, что вектор движения указывает местоположение в 3 пикселя вправо и 3 пикселя выше текущего блока. Путем преобразования временного предсказателя вектора движения, имеющего целочисленную степень точности, можно избегать интерполяцию опорного блока для формирования предсказываемого блока.

[0105] В качестве другого примера, видеodeкодер 30 может быть сконфигурирован, чтобы принимать, кодированный в режиме с внутрикадровой копией блока, блок видеоданных текущего изображения. Видеodeкодер 30 может определять со-размещенный блок видеоданных опорного изображения, которое отличается от текущего изображения. Видеodeкодер 30 может выводить временный предсказатель вектора движения исходя из со-размещенного блока видеоданных. Видеodeкодер 30 может сдвигать вправо TMVP. Видеodeкодер 30 может определять предсказываемый блок в текущем изображении на основании сдвинутого вправо TMVP. Видеodeкодер 30 может декодировать блок видеоданных текущего изображения, используя сдвинутый вправо временный предсказатель вектора движения.

[0106] В качестве другого примера, видеodeкодер 30 может быть сконфигурирован для включения текущего изображения и опорного изображения в список опорных изображений при опорном изображении, являющимся отличным от текущего изображения. Кроме того, в этом примере, видеodeкодер 30 определяет со-размещенный блок опорного изображения. Со-размещенный блок со-размещен с текущим блоком текущего изображения. Видеodeкодер 30 может выводить временный предсказатель вектора движения исходя из со-размещенного блока. Например, видеodeкодер 30 может определять, что TMVP является равным (-5,25, 4,00), значения которого являются идентифицируемыми двоичным представлением со значениями субпиксельной точности, соответствующими, например, двум младших значащим битам двоичного представления. Видеodeкодер 30 может быть сконфигурирован для определения, что TMVP имеет субпиксельную точность. В некоторых примерах видеodeкодер 30 определяет, что TMVP имеет субпиксельную точность, путем определения, является ли со-размещенный блок видеоданных кодированным в режиме внутрикадровой копии блоком или кодированным в режиме межкадрового предсказания. Определение, что со-размещенный блок видеоданных является кодированным в режиме межкадрового предсказания, может означать, что со-размещенный блок видеоданных имеет субпиксельную точность (например, нецелочисленную точность или дробную точность).

[0107] В этом примере, в ответ на определение, что TMVP имеет субпиксельную точность, видеodeкодер 30 сдвигает вправо TMVP. В некоторых примерах видеodeкодер 30 сдвигает TMVP вправо только по определению, что со-размещенный блок является кодированным в режиме межкадрового предсказания. Видеodeкодер 30 может быть сконфигурирован для определения на основании сдвинутого вправо TMVP предсказываемого блока внутри текущего изображения. Например, в примере по Фиг.3, видеodeкодер 30 может использовать сдвинутый вправо TMVP, чтобы определить предсказываемый сигнал 104. Видеodeкодер 30 может формировать остаточные данные на основании предсказываемого блока, который был определен на основании сдвинутого вправо временного предсказателя вектора движения.

[0108] В качестве другого примера, видеodeкодер 20 может быть сконфигурирован для

включения текущего изображения и опорного изображения в список опорных изображений, причем опорное изображение является отличным от текущего изображения. Видеокодер 20 может быть сконфигурирован для определения со-

- 5 размещенного блока опорного изображения видеоданных, со-размещенный блок является со-размещаемым с текущим блоком текущего изображения, опорное изображение является отличным от текущего изображения. Видеокодер 20 может быть сконфигурирован для вывода временного предсказателя вектора движения исходя из со-размещенного блока. Видеокодер 20 может быть сконфигурирован для определения, что временный предсказатель вектора движения имеет субпиксельную точность.
- 10 Например, видеокодер 20 может быть сконфигурирован для определения, что временный предсказатель вектора движения имеет субпиксельную точность, являясь конфигурируемым для определения, является ли со-размещенный блок видеоданных кодированным в режиме с внутрикадровой копией блока, или кодированным в режиме межкадрового предсказания. Определение, что со-размещенный блок видеоданных
- 15 является кодированным в режиме межкадрового предсказания, может означать, что со-размещенный блок видеоданных имеет субпиксельную точность (например, нецелочисленную точность или дробную точность).

[0109] Видеокодер 20 может быть сконфигурирован для сдвига вправо временного предсказателя вектора движения, определенного имеющим субпиксельную точность.

- 20 В некоторых примерах видеокодер 20 может быть сконфигурирован для сдвига вправо временного предсказателя вектора движения только после определения, что со-размещенный блок является кодированным в режиме межкадрового предсказания. Видеокодер 20 может быть сконфигурирован, чтобы на основании сдвинутого вправо временного предсказателя вектора движения определять предсказываемый блок внутри
- 25 текущего изображения. Видеокодер 20 может быть сконфигурирован для формирования остаточных данных на основании предсказываемого блока.

[0110] Кроме того, в некоторых примерах настоящего раскрытия, относительно I-слайсов, кодер видео выводит значение `initType`. Значение `initType` указывает тип контекста, который кодер видео использует для определения, каким образом

30 синтаксический элемент кодируется по CABAC. Значение `initType` может зависеть от значения в `initType`, которое может далее зависеть от синтаксического элемента `cabac_init_flag`. Как определено в JCTVC-Q1003, синтаксический элемент `cabac_init_flag` задает способ для определения таблицы инициализации, используемой в процессе инициализации для переменных контекста. Кодер видео может использовать значение

35 `initType`, чтобы определять значение для значения `ctxIdx`, которое может указывать контекст CABAC. В одном примере в соответствии со способами настоящего раскрытия выведение `initType` модифицировано, как изложено в следующем псевдокоде:

```

    if (slice_type == I && !intra_block_copy_enabled_flag)
        initType=0
    else if (slice_type == I && intra_block_copy_enabled_flag)
        initType=cabac_init_flag ? 2: 1
        else if (slice_type == P)
            initType=cabac_init_flag ? 2: 1
        else
45  initType=cabac_init_flag ? 1: 2

```

Секция 1 кода: Выведение `initType`

Предшествующий код Секции 1 модифицирует Раздел 9.3.2.2 в JCTVC-Q1003.

Подчеркнутый текст указывает добавленный к JCTVC-Q1003 текст. Как определено в

JCTVC-Q1003, синтаксический элемент `slice_type` указывает тип кодирования для слайса. Например, синтаксический элемент `slice_type` указывает, является ли слайс I-слайсом, P-слайсом или B-слайсом. Кроме того, как определено в JCTVC-R1005, флаг `intra_block_copy_enabled_flag` является синтаксическим элементом в наборе параметров последовательности. Флаг `intra_block_copy_enabled_flag` указывает, разрешен ли intra BC для слайсов в кодированных видео последовательностях, для которых набор параметров последовательности является активным. Таким образом, в примере Секции 1 кода, переменная `initType` имеет различные значения при кодировании I-слайса в зависимости от того, разрешен ли intra BC для I-слайса.

[0111] В другом примере кодер видео определяет значение `initType`, используемое для синтаксических элементов кода CABAC, которые не присутствуют в битовом потоке для I-слайса и не используют intra BC, согласно Секции кода 1. Другими словами, только для синтаксических элементов, которые не присутствуют для традиционного I-слайса, то есть без Intra BC, соответствующую переменную `initType` выводят, как в Секции 1 кода. Для других синтаксических элементов кодер видео может установить значение `initType`, равное 0.

[0112] В другом примере, когда intra-BC разрешен для CU или PU, кодер видео может допускать режимы и внутри-, и меж-кадрового предсказания в I-слайсе. Следовательно, видеodecoder 30 может декодировать первый блок I-слайса изображения, используя внутрикадровое предсказание, и может декодировать второй блок I-слайса, используя режим Intra BC. Подобным образом, видеокодер 20 может кодировать первый блок I-слайса изображения, используя внутрикадровое предсказание, и может кодировать второй блок I-слайса, используя режим Intra BC. Последующий пример псевдокода иллюстрирует эту возможность:

Когда флаг `pred_mode_flag` не присутствует, кодер видео выводит переменную `CuPredMode[x][y]`, как изложено ниже для $x=x_0 \dots x_0+nCbS-1$ и $y=y_0 \dots y_0+nCbS-1$:

- Если `slice_type` равен, I и флаг `intra_block_copy_enabled_flag` равен 0, заключают, что `CuPredMode[x][y]` будет равным `MODE_INTRA`.

Секция 2 кода: Модификация CUPredMode

Предшествующий псевдокод модифицирует Раздел 7.4.9.5 в JCTVC-Q1003. Подчеркнутый текст указывает добавленный к JCTVC-Q1003 текст. Как определено в JCTVC-Q1003, `pred_mode_flag` является синтаксическим элементом, указывающим режим предсказания (например, межкадровое предсказание (`MODE_INTER`) или внутрикадровое предсказание (`MODE_INTRA`)) для CU. Поскольку режимы предсказания для единиц CU могут выводиться в некоторых обстоятельствах, кодер видео загружает режимы предсказания в имеющую тип «массив» переменную `CuPredMode`, индексированную на основании местоположений пикселей, а не просто основываться на значениях `pred_mode_flag`. Таким образом, в Секции 2 кода, `CuPredMode[x][y]` указывает режим предсказания для CU в позиции (x,y) в изображении. `nCbS` указывает размер блока кодирования для CU. Следовательно, в примере Секции 2 кода, кодер видео способен заключить, что значением `CuPredMode[x][y]` (то есть режим предсказания для CU в позиции (x,y)), является `MODE_INTRA`, когда CU находится в I-слайсе, и intra BC разрешен для I-слайса. Следовательно, видеокодеру 20 не требуется сигнализировать синтаксический элемент `pred_mode_flag` для CU. Это может уменьшить цифровой поток кодированных видеоданных.

[0113] В качестве другого примера, в котором кодер видео обрабатывает Intra BC как межкадровое предсказание, используя объединенные способы настоящего раскрытия, кодированный в Intra BC блок может использовать один и тот же список

кандидатов для режимов слияния или AMVP совместно с межкадрово-кодированным блоком. В этом примере список кандидатов может включать в себя и кодированные в Intra BC блоки, и межкадрово-кодированные блоки. Следовательно, видеокодер 20 и/или видеодекодер 30 могут формировать список кандидатов-векторов движения, который включает в себя, по меньшей мере, один кандидат-вектор движения для кодированного в режиме внутрикадровой копии блока, и, по меньшей мере, один кандидат для кодированного с межкадровым предсказанием блока. В некоторых примерах блок Intra BC может иметь отдельный от кодированного в Inter блока список кандидатов для AMVP или слияния. Хотя режимы предсказания являются одинаковыми, кодер видео может различать кодированный в Intra BC блок от традиционного Inter блока, проверяя, является ли опорное изображение для текущего блока текущим изображением.

[0114] Кроме того, конкретные способы настоящего раскрытия могут также улучшить объединенный режим intra BC и межкадрового предсказания, модифицируя процесс построения опорного списка. Последующий формулировка и псевдокод модифицируют процесс декодирования относительно построения списков опорных изображений в Разделе 8.3.4 в JCTVC-Q1003 подчеркнутым текстом, указывающим добавленный к JCTVC-Q1003 текст:

Этот процесс активизируется в начале процесса декодирования для каждого P или В слайса или для I-слайса, когда флаг `intra_block_copy_enabled_flag` равен 1

Опорные изображения адресуются посредством опорных индексов, как предписано в пункте 8.5.3.3.2 (Процесс выбора опорного изображения). Опорный индекс является индексом в списке опорных изображений. При декодировании P-слайса, имеется единственный список `RefPicList0` опорных изображений. При декодировании В-слайса имеется второй независимый список `RefPicList1` опорных изображений в дополнение к `RefPicList0`.

[0115] Таким образом, предшествующая формулировка используется, чтобы допускать модификацию процесса для опорного изображения, когда режим intra BC разрешен для I-слайса. Следовательно, в этом примере, видеокодер 20 может строить список опорных изображений для I-слайса изображения, когда intra BC разрешен для I-слайса, список опорных изображений включает изображение. В подобном примере видеодекодер 30 может получить из битового потока синтаксический элемент (например, `intra_block_copy_enabled_flag`), указывающий, разрешен ли intra BC для изображения. В этом примере, в ответ на определение на основании синтаксического элемента, что intra BC разрешен для изображения, видеодекодер 30 может строить список опорных изображений, включающий изображение.

[0116] В одном примере, при декодировании I-слайса с флагом `intra_block_copy_enabled_flag`, равным 1, видеодекодер 30 может использовать только один список опорных изображений. В другом примере, при декодировании I-слайса с флагом `intra_block_copy_enabled_flag`, равным 1, может быть второй независимый список `RefPicList1` опорных изображений в дополнение к `RefPicList0`.

[0117] В некоторых примерах, в начале процесса декодирования для каждого слайса, списки опорных изображений, `RefPicList0`, и для В слайсов, `RefPicList1`, выводят согласно последующему псевдокоду с подчеркнутым текстом, указывающим добавленный к JCTVC-Q1003 текст:

`incrementRef=(intra_block_copy_enabled_flag)? 1: 0`

Переменная `NumRpsCurrTempList0` устанавливается равной `Max (num_ref_idx_l0_active_minus1+1, NumPocTotalCurr)+incrementRef`, и список `RefPicListTemp0`

строится, как изложено ниже:

```

    rIdx=0
    while(rIdx<NumRpsCurrTempList0) {
    for(i=0; i<NumPocStCurrBefore && rIdx<NumRpsCurrTempList0; rIdx++, i++)
5   RefPicListTemp0[rIdx]=RefPicSetStCurrBefore[i]
    for (i=0; i<NumPocStCurrAfter && rIdx<NumRpsCurrTempList0; rIdx++, i++) (8 8)
    RefPicListTemp0[rIdx]=RefPicSetStCurrAfter[i]
    for (i=0; i<NumPocLtCurr&&rIdx<NumRpsCurrTempList0; rIdx++, i++)
    RefPicListTemp0[rIdx]=RefPicSetLtCurr[i]
10   if (incrementRef)
    RefPicListTemp0[rIdx]=currPic (8-8)
    }

```

Секция кода 3: Модификация RefPicList0

[0118] Таким образом, предшествующий псевдокод модифицирует переменную RefPicListTemp0 так, что RefPicListTemp0[rIdx] является равным текущему изображению, где rIdx является максимальным индексом в последнем индексе в массиве RefPicListTemp0. В некоторых примерах кодер видео добавляет текущее изображение в начало RefPicListTemp0 или перед долгосрочными опорными изображениями. В некоторых примерах кодер видео добавляет текущее изображение в RefPicListTemp0 на основании параметра slice_type. Таким образом, в этих примерах, кодер видео может включать, в списке опорных изображений, текущее изображение перед долгосрочными изображениями во втором списке опорных изображений; или включать, в списке опорных изображений, текущее изображение в положение в списке опорных изображений на основании того, является ли слайс I-слайсом, P-слайсом или B-слайсом.

25 Поскольку меньшие по значению опорные индексы могут быть представлены с использованием меньшего числа битов, желательно, чтобы более часто используемые опорные изображения имели меньшие по значению опорные индексы. Таким образом, включение текущего изображения в более раннее положение в списке опорных изображений относительно других опорных изображений в списке опорных изображений

30 отражает предположение, что текущее изображение будет использоваться в intra BC более часто, чем другие опорные изображения.

[0119] В качестве другого примера, например, для I_slice, кодер видео добавляет текущее изображение в RefPicListTemp0 в начале RefPicListTemp0, и для P или B слайса - в конце долгосрочных опорных изображений. Последующий псевдокод иллюстрирует

35 этот пример подчеркнутым текстом, указывающим текст, добавленный к JCTVC-Q1003:

```

    while(rIdx<NumRpsCurrTempList0) {
    if (incrementRef && Slice_type==I_Slice)
    RefPicListTemp0[rIdx]=currPic
    for(i=0; i<NumPocStCurrBefore && rIdx<NumRpsCurrTempList0; rIdx++, i++)
40   RefPicListTemp0[rIdx]=RefPicSetStCurrBefore[i]
    for(i=0; i<NumPocStCurrAfter && rIdx<NumRpsCurrTempList0; rIdx++, i++) (8-8)
    RefPicListTemp0[rIdx]=RefPicSetStCurrAfter[i]
    for(i=0; i<NumPocLtCurr && rIdx<NumRpsCurrTempList0; rIdx++, i++)
    RefPicListTemp0[rIdx]=RefPicSetLtCurr[i]
45   if (incrementRef&&(slice_type==P_Slice || slice_type==B_Slice))
    RefPicListTemp0[rIdx]=currPic
    }

```

Секция кода 4: Модификация построения списка опорных изображений

[0120] В качестве другого примера, для I-слайса кодер видео добавляет текущее изображение к RefPicListTemp0 в начале, и для P или B слайса добавляет текущее изображение перед долгосрочными опорными изображениями. В качестве другого примера, позиция добавления текущего изображения в RefPicListTemp0 основана на slice_type текущего изображения.

[0121] В некоторых примерах кодер видео осуществляет построение RefPicList0 на основании RefPicListTemp0 согласно следующему псевдокоду с подчеркнутым текстом, указывающим добавленный к JCTVC-Q1003 текст:

Список RefPicList0 строится, как изложено ниже:

```
for(rIdx=0; rIdx <= num_ref_idx_l0_active_minus1; rIdx++)
  RefPicList0[rIdx]=ref_pic_list_modification_flag_l0 ? RefPicListTemp0[list_entry_l0[rIdx]]:
  RefPicListTemp0[rIdx] (8-9)
```

Когда слайс является B-слайсом, переменную NumRpsCurrTempList1 устанавливают равной $\text{Max}(\text{num_ref_idx_l1_active_minus1}+1, \text{NumPocTotalCurr})+\text{incrementRef}$, и список

RefPicListTemp1 строится, как изложено ниже:

```
rIdx=0
while(rIdx<NumRpsCurrTempList1) {
  for(i=0; i<NumPocStCurrAfter && rIdx<NumRpsCurrTempList1; rIdx++, i++)
    RefPicListTemp1[rIdx]=RefPicSetStCurrAfter[i]
  for(i=0; i<NumPocStCurrBefore && rIdx<NumRpsCurrTempList1; rIdx++, i++) (8 10)
    RefPicListTemp1[rIdx]=RefPicSetStCurrBefore[i]
  for(i=0; i<NumPocLtCurr && rIdx<NumRpsCurrTempList1; rIdx++, i++)
    RefPicListTemp1[rIdx]=RefPicSetLtCurr[i] (8-10)
  if (incrementRef)
    RefPicListTemp1[rIdx]=currPic
}
```

[0122] В некоторых примерах, подобных вышеприведенному процессу для RefPicListTemp1, кодер видео может добавлять текущее изображение в RefPicListTemp1, основываясь на slice_type текущего изображения согласно следующему псевдокоду и формулировке стандартов JCTVC-Q1003, Раздел 8.3.4 с подчеркнутым текстом, показывающим добавленный к JCTVC-Q1003 текст:

Когда слайс является B слайсом, список RefPicList1 строится, как изложено ниже:

```
for(rIdx=0; rIdx <= num_ref_idx_l1_active_minus1; rIdx++) (8-11)
```

```
RefPicList1[rIdx]=ref_pic_list_modification_flag_l1 ? (8-11)
```

```
RefPicListTemp1[list_entry_l1[rIdx]]: RefPicListTemp1[rIdx]
```

где currPic - текущее декодированное изображение, как в 8.5.3.3.2, и должно иметь такой же временный ID текущего изображения.

[0123] В некоторых примерах процесс выбора опорного изображения для определения индексов опорного изображения в Разделе 8.5.3.3.2 в JCTVC-Q1003 может быть изменен, как изложено ниже, с подчеркнутым текстом, показывающим добавленный к JCTVC-Q1003 текст:

Вход в этот процесс - опорный индекс refIdxLX.

Выход этого процесса - опорное изображение, состоящее из двумерного массива refPicLX_L выборок яркости и двух двумерных массивов выборок refPicLX_{Cb} и refPicLX_{Cr} цветности.

Выходное опорное изображение RefPicListX[refIdxLX] состоит из массива pic_width_in_яркости_samples pic_height_in_яркости_samples выборок refPicLX_L яркости

и двух массивов `PicWidthInSamplesC` на `PicHeightInSamplesC` выборки цветности, `refPicLXСb` и `refPicLXСr`.

Массивы выборок `refPicLXL`, `refPicLXСb` и `refPicLXСr` опорных изображений соответствуют массивам `SL`, `SСb`, и `SСr` декодированных выборок, выведенных в п. 8.7 или как в пункте 8.7 без внутриконтурного фильтра (деблокирующего и SAO) для предварительно декодированного изображения, или текущего декодированного изображения.

[0124] Когда разрешены и Intra BC, и ограниченное внутрикадровое предсказание, предсказание для блоков Intra или Intra BC может только быть исходя из блоков Intra или блока Inter с любыми опорными изображениями, являющимися текущим изображением. В ограниченном внутрикадровом предсказании, внутрикадровое предсказание использует только остаточные данные и декодированные выборки из соседних блоков кодирования, кодированных с использованием режимов внутрикадрового предсказания. Как описано в JCTVC-Q1003, синтаксический элемент (например, `constrained_intra_pred_flag`) сигнализируется в наборе параметров изображения, чтобы указать, используется ли ограниченное внутрикадровое предсказание в ходе декодирования изображений, для которых набор параметров изображения является активным. Таким образом, этот пример, когда и Intra BC, и ограниченное внутрикадровое предсказание разрешены для слайса, кодер видео может использовать межкадровое предсказание, чтобы определять вектор блока для блока слайса на основании вектора блока или вектора движения для блока в другом изображении.

[0125] JCTVC-Q1003 дает определение синтаксического элемента `cu_skip_flag`. В соответствии с примером из настоящего раскрытия, семантика `cu_skip_flag`, определенного в JCTVC-Q1003, модифицирована, как изложено ниже:

cu_skip_flag[x0][y0], равный 1, предписывает, что для текущего элемента кодирования, при декодировании Р или В-слайса или I-слайса с флагом `intra_block_copy_enabled_flag`, равным 1, синтаксические элементы, кроме индекса `merge_idx[x0][y0]` кандидата слияния, более синтаксически не разбираются после `cu_skip_flag[x0][y0]`.

Таким образом, в этом примере, когда `cu_skip_flag` является 1 для текущего CU в I-слайсе, в котором разрешен Intra BC, видеокодер 20 действительно сигнализирует синтаксический элемент «флаг слияния», синтаксические элементы MVD, синтаксические элементы «флаг предсказателя вектора движения» или синтаксические элементы направления межкадрового предсказания. Вдобавок, когда `cu_skip_flag` является 1 для текущего CU в I-слайсе, в котором разрешен Intra BC, видеокодер 20 не сигнализирует дерево преобразования для текущего CU. Предпочтительнее кодер видео принимает, что предсказываемый блок, указанный информацией движения кандидата слияния, указанного индексом кандидата слияния, будет восстановленным блоком кодирования для текущего CU.

[0126] JCTVC-Q1003 задает `slice_temporal_mvp_enabled_flag` в заголовке сегмента слайса. Флаг `slice_temporal_mvp_enabled_flag` предписывает, могут ли временные предсказатели вектора движения использоваться для межкадрового предсказания. Когда флаг `slice_temporal_mvp_enabled_flag` равен 0 в I-слайсе, когда флаг `intra_block_copy_enabled_flag` равен 0, он не оказывает влияния на нормативный процесс декодирования изображения, а просто выражает ограничение битового потока.

[0127] Хотя способы настоящего раскрытия, относящиеся к способам объединенного режима с внутрикадровой копией блока и межкадрового предсказания, описаны

относительно HEVC, способы настоящего раскрытия не ограничиваются HEVC, и могут быть применимыми в любой системе кодирования видео, в которой используются межкадровое предсказание и intra BC, или их аналоги.

[0128] Фиг.4 является блок-схемой, иллюстрирующей примерный видеокодер 20, который может реализовать способы объединенного intra BC и межкадрового предсказания, описанные в настоящем раскрытии. Видеокодер 20 может выполнять внутри- и меж-кадровое кодирование видеоблоков внутри видео слайсов.

Внутрикадровое кодирование основывается на пространственном предсказании для снижения или устранения пространственной избыточности в видео в рамках данного видеокadra или изображения. Меж-кадровое кодирование основывается на временном предсказании для снижения или устранения временной избыточности в видео внутри смежных кадров или изображений видеопоследовательности. Режим с внутренним кодированием (режим I) может относиться к любому из нескольких режимов сжатия на основе пространственного предсказания. Как обсуждено выше, видеокодер 20 может быть сконфигурирован для кодирования видеоданных в одном или нескольких из множества «intra» режимов, включая режим intra DC, intra angular (углового предсказания) режим, режим intra planar (планарного предсказания) и режим intra BC. Режимы Inter (с межкадровым предсказанием), такие как однонаправленное предсказание (P режим) или двунаправленное предсказание (B режим), могут относиться к любому из нескольких режимов сжатия на основе временного предсказания.

[0129] В примере по Фиг.4 видеокодер 20 включает в себя память 40 видеоданных, блок 41 обработки предсказания, буфер 64 декодированных изображений, сумматор 50, блок 52 обработки преобразования, блок 54 квантования и блок 56 энтропийного кодирования. Блок 41 обработки предсказания включает в себя блок 35 разбиения, блок 42 оценки движения, блок 44 компенсации движения и блок 46 обработки внутрикадрового предсказания. Для восстановления видеоблока видеокодер 20 также включает в состав блок 58 обратного квантования, блок 60 обработки обратного преобразования и сумматор 62. Деблокирующий фильтр (не показан на Фиг.4) также может включаться в состав, чтобы фильтровать границы блоков для удаления артефактов блочности изображения из восстановленного видео. Если желательно, деблокирующий фильтр фильтрует выход сумматора 62. Дополнительные контурные фильтры (внутриконтурные или постконтурные) также могут использоваться в дополнение к деблокирующему фильтру.

[0130] Память 40 видеоданных может сохранять видеоданные, подлежащие кодированию компонентами видеокодера 20. Видеоданные, сохраненные в памяти 40 видеоданных, могут быть получены, например, от источника 18 видео. Буфер (DPB) 64 декодированных изображений является буфером, который сохраняет опорные видеоданные для использования в кодировании видеоданных видеокодером 20 (например, в режимах внутри- или меж-кадрового кодирования, также называемых режимами кодирования с внутри- или меж-кадровым предсказанием). Память 40 видеоданных и DPB 64 может составлять любое из множества запоминающих устройств, такое как динамическая оперативная память (DRAM), включая синхронную DRAM (SDRAM), магниторезистивная RAM (MRAM), резистивная RAM (RRAM), или другие типы запоминающих устройств. Память 40 видеоданных и DPB 64 могут обеспечиваться одним и тем же запоминающим устройством или отдельными запоминающими устройствами. В различных примерах память 40 видеоданных может быть на кристалле с другими компонентами видеокодера 20, или вне кристалла относительно тех компонентов.

[0131] Как показано на Фиг.4, видеокодер 20 принимает видеоданные, и блок 35 разбиения разделяет данные на блоки видео. Это разделение может также включать разделение на слайсы, фрагменты, или другие более крупные единицы, а также разбиение видеоблоков, например, согласно структуре квадрадерева для элементов LCU и CU.

5 Видеокодер 20 обычно иллюстрирует компоненты, которые кодируют видеоблоки внутри видеослайса, подлежащего кодированию. Слайс может быть разделен на множественные видеоблоки (и возможно на наборы видеоблоков, называемые (мозаичными) фрагментами). Блок 41 обработки предсказания может выбирать один из ряда возможных режимов кодирования, таких как один из множества режимов
10 внутрикадрового кодирования или один из множества режимов межкадрового кодирования, для текущего видеоблока на основании результатов ошибок (например, скорость кодирования и уровень искажения). Блок 41 обработки предсказания может быть сконфигурирован для реализации способов настоящего раскрытия, описанного здесь, для выполнения объединенных режимов intra BC, внутри- и меж-кадрового
15 предсказания. Блок 41 обработки предсказания может предоставлять результирующий внутри- или меж-кадрово кодированный блок на сумматор 50, чтобы сформировать остаточный блок данных, и на сумматор 62, чтобы восстановить кодированный блок для использования в качестве опорного изображения.

[0132] Блок 46 обработки внутрикадрового предсказания в блоке 41 обработки
20 предсказания может выполнять внутреннее кодирование с предсказанием для текущего видеоблока относительно одного или нескольких соседних блоков в том же кадре или слайсе в качестве текущего блока, подлежащего кодированию, чтобы обеспечивать пространственное сжатие. Блок 42 оценки движения и блок 44 компенсации движения в блоке 41 обработки предсказания выполняют межкадровое кодирование с
25 предсказанием для текущего видеоблока относительно одного или нескольких предсказываемых блоков в одном или нескольких опорных изображениях, чтобы обеспечивать временное сжатие.

[0133] Блок 42 оценки движения может быть сконфигурирован для определения режима межкадрового предсказания для видеослайса согласно predetermined шаблону для видео последовательности. Предetermined шаблон может обозначать
30 видеослайсы в последовательности как P-слайсы, B-слайсы или, в некоторых примерах, GPB-слайсы. Блок 42 оценки движения и блок 44 компенсации движения могут быть высоко интегрированными, но проиллюстрированы отдельно с понятийными целями. Оценка движения, выполняемая блоком 42 оценки движения, является процессом
35 формирования векторов движения, которые оценивают движение для видеоблоков. Вектор движения, например, может указывать смещение PU для видеоблока внутри текущего видеокadra или изображения относительно предсказываемого блока внутри опорного изображения.

[0134] Предсказываемый блок является блоком, который считают близко
40 соответствующим PU видеоблока, подлежащего кодированию, в терминах пиксельной разности, которая может быть определена суммой абсолютной разности (SAD), суммой квадратичной разности (SSD) или другими метриками разности. В некоторых примерах видеокодер 20 может вычислять значения для субцелочисленных пиксельных позиций опорных изображений, сохраненных в буфере 64 декодированных изображений.
45 Например, видеокодер 20 может интерполировать значения позиций одной четвертой пикселя, позиций одной восьмой пикселя или других дробных пиксельных позиций опорного изображения. Следовательно, блок 42 оценки движения может выполнять поиск движения относительно полных пиксельных позиций и дробных пиксельных

позиций и выводить вектор движения с дробной пиксельной точностью.

[0135] Блок 42 оценки движения вычисляет вектор движения для PU видеоблока в межкадрово кодированном слайсе путем сравнения позиции PU с позицией предсказываемого блока опорного изображения. Опорное изображение может
5 выбираться из первого списка опорных изображений (Список 0) или второго списка опорных изображений (Список 1), каждый из которых идентифицирует одно или несколько опорных изображений, сохраненных в буфере 64 декодированных изображений. Блок 42 оценки движения посылает вычисленный вектор движения на блок 56 энтропийного кодирования и блок 44 компенсации движения.

[0136] Компенсация движения, выполняемая блоком 44 компенсации движения, может подразумевать извлечение или формирование предсказываемого блока на основании вектора движения, определенного согласно оценке движения, возможно выполняя интерполяции по отношению к субпиксельной точности. По приему вектора движения для PU текущего видеоблока блок 44 компенсации движения может определить
15 местоположение предсказываемого блока, на который вектор движения указывает, в одном из списков опорных изображений. Видеокодер 20 формирует остаточный блок видео путем вычитания пиксельных значений предсказываемого блока из пиксельных значений текущего кодируемого видеоблока, формируя пиксельные значения разности. Пиксельные значения разности образуют остаточные данные для блока, и могут
20 включать разностные компоненты и яркости, и цветности. Сумматор 50 представляет компонент или компоненты, которые выполняют эту операцию вычитания. Блок 44 компенсации движения может также формировать синтаксические элементы, связанные с видеоблоками и видео слайсом, для использования видеодекодером 30 при декодировании видеоблоков видео слайса.

[0137] Блок 46 обработки внутрикадрового предсказания может внутренне предсказывать текущий блок в качестве альтернативы межкадровому предсказанию, выполняемому блоком 42 оценки движения и блоком 44 компенсации движения, как описано выше. В частности блок 46 обработки внутрикадрового предсказания может определять режим внутрикадрового предсказания, включая режим Intra BC, чтобы
30 использовать для кодирования текущего блока. В некоторых примерах блок 46 обработки внутрикадрового предсказания может кодировать текущий блок, используя различные режимы внутрикадрового предсказания, например, в ходе отдельных проходах кодирования, и блок 46 обработки внутрикадрового предсказания (или блок выбора режима, в некоторых примерах) может выбирать для использования надлежащий
35 режим внутрикадрового предсказания из испытываемых режимов.

[0138] Например, блок 46 обработки внутрикадрового предсказания может вычислять значения искажения в зависимости от скорости передачи, используя анализ искажения в зависимости от скорости передачи для различных испытываемых режимов внутрикадрового предсказания, и выбирать режим внутрикадрового предсказания,
40 имеющий наилучшие характеристики искажения в зависимости от скорости передачи из числа испытываемых режимов. Анализ искажения в зависимости от скорости передачи обычно определяет величину искажения (или ошибку) между кодированным блоком и исходным, некодированным блоком, который был кодирован, чтобы создать кодированный блок, а также расход битов (то есть число битов), используемый для создания кодированного блока. Блок 46 обработки внутрикадрового предсказания может вычислять коэффициенты исходя из искажений и скоростей для различных кодированных блоков, чтобы определить, какой режим внутрикадрового предсказания
45 показывает для блока наилучшее значение искажения в зависимости от скорости

передачи.

[0139] В соответствии со способами настоящего раскрытия, как описано выше, видеокодер 20 может быть сконфигурирован с возможностью определять вектор движения для кодированного в режиме с внутрикадровой копией блока (intra BC) блока, кодированного с использованием временного предсказания вектора движения (TMVP), определять со-размещенный блок на основании вектора TMVP, в ответ на определение, что со-размещенный блок кодирован в Intra BC с использованием вектора TMVP с целочисленной пиксельной точностью, чтобы определить опорный блок, и в ответ на определение, что со-размещенный блок является межкадрово кодированным, использовать вектор TMVP с субпиксельной точностью для определения опорного блока.

[0140] В соответствии со способами настоящего раскрытия, видеокодер может быть сконфигурирован для выполнения обычно взаимнообратного процесса для любого процесса, описанного здесь относительно кодера видео. Аналогично, кодер видео может быть сконфигурирован для выполнения обычно взаимнообратного процесса для любого процесса, описанного здесь относительно декодера видео.

[0141] В любом случае, после выбора режима внутрикадрового предсказания для блока, блок 46 обработки внутрикадрового предсказания может предоставлять информацию, указывающую выбранный режим внутрикадрового предсказания для блока, на блок 56 энтропийного кодирования. Блок 56 энтропийного кодирования может кодировать информацию, указывающую выбранный режим внутрикадрового предсказания, в соответствии со способами настоящего раскрытия. Видеокодер 20 может включать в передаваемый битовый поток данные конфигурации, которые могут включать в себя множество индексных таблиц режима внутрикадрового предсказания и множество модифицированных индексных таблиц режима внутрикадрового предсказания (также называемых таблицами отображения кодовых слов), определения контекстов кодирования для различных блоков и указатели относительно наиболее вероятного режима внутрикадрового предсказания, индексные таблицы режима внутрикадрового предсказания и модифицированные индексные таблицы режима внутрикадрового предсказания для использования для каждого из контекстов.

[0142] После того, как блок 41 обработки предсказания сгенерирует предсказываемый блок для текущего блока при помощи либо межкадрового предсказания, либо при помощи внутрикадрового предсказания, видеокодер 20 формирует остаточный блок путем вычитания предсказываемого блока из текущего блока. Остаточные видеоданные в остаточном блоке могут включаться в один или несколько TU и применяться к блоку 52 обработки преобразования. Блок 52 обработки преобразования преобразовывает остаточные видеоданные в остаточные коэффициенты преобразования, используя преобразование, например, дискретное косинусное преобразование (DCT) или концептуально подобное преобразование. Блок 52 обработки преобразования может преобразовать остаточные видеоданные из пиксельной области в область преобразования, такую как частотная область.

[0143] Блок 52 обработки преобразования может посылать результирующие коэффициенты преобразования на блок 54 квантования. Блок 54 квантования квантует коэффициенты преобразования, чтобы дополнительно снизить расход битов. Процесс квантования может снизить битовую глубину, связанную с некоторыми или всеми коэффициентами. Степень квантования может быть модифицирована корректировкой параметра квантования. В некоторых примерах блок 54 квантования может затем выполнять просмотр матрицы, включающей квантованные коэффициенты

преобразования. Альтернативно, блок 56 энтропийного кодирования может выполнять просмотр.

[0144] После квантования, блок 56 энтропийного кодирования энтропийно кодирует квантованные коэффициенты преобразования. Например, блок 56 энтропийного кодирования может выполнять контекстно-зависимое адаптивное кодирование с переменной длиной кодового слова (CAVLC), контекстно-зависимое адаптивное двоичное арифметическое кодирование (CABAC), синтаксически-ориентированное контекстно-адаптивное двоичное арифметическое кодирование (SBAC), энтропийное кодирование с разделением на интервалы вероятности (PIPE) или другую методологию или способ энтропийного кодирования. После энтропийного кодирования блоком 56 энтропийного кодирования, кодированный битовый поток может передаваться на видеodeкодер 30 или архивироваться для более поздней передачи или извлечения видеodeкодером 30. Блок 56 энтропийного кодирования может также энтропийно кодировать векторы движения и другие синтаксические элементы для текущего кодируемого видеослайса.

[0145] Блок 58 обратного квантования и блок 60 обработки обратного преобразования применяют обратное квантование и обратное преобразование, соответственно, чтобы восстановить остаточный блок в пиксельной области для более позднего использования в качестве опорного блока опорного изображения. Блок 44 компенсации движения может вычислять опорный блок путем добавления остаточного блока к предсказываемому блоку одного из опорных изображений в одном из списков опорных изображений. Блок 44 компенсации движения может также применять один или несколько интерполяционных фильтров к восстановленному остаточному блоку, чтобы вычислить значения субпиксельной точности для использования в оценке движения. Сумматор 62 добавляет восстановленный остаточный блок к предсказываемому блоку с компенсацией движения, созданному блоком 44 компенсации движения, чтобы создать опорный блок для сохранения в буфере 64 декодированных изображений. Опорный блок может использоваться блоком 42 оценки движения и блоком 44 компенсации движения в качестве опорного блока, чтобы осуществлять межкадровое предсказание блока в последующем видеокадре или изображении.

[0146] В соответствии с аспектами настоящего раскрытия видеodeкодер 20 может быть сконфигурирован для выполнения любой комбинации способов, описанных в настоящем раскрытии.

[0147] Фиг.5 является блок-схемой, иллюстрирующей примерный видеodeкодер 30, который может реализовать способы для объединенного intra VC и межкадрового предсказания, описанного в настоящем раскрытии. В примере по Фиг.5 видеodeкодер 30 включает в себя память 79 видеоданных, блок 80 энтропийного декодирования, блок 81 обработки предсказания, блок 86 обратного квантования, блок 88 обработки обратного преобразования, сумматор 90 и буфер 92 декодированных изображений. Блок 81 обработки предсказания включает в себя блок 82 компенсации движения и блок 84 обработки внутрикадрового предсказания. Видеodeкодер 30 в некоторых примерах может выполнять проход декодирования, обычно взаимнообратный проходу кодирования, описанному относительно видеodeкодера 20 по Фиг.4.

[0148] Память 79 видеоданных может сохранять видеоданные, такие как битовый поток кодированного видео, подлежащий декодированию компонентами видеodeкодера 30. Видеоданные, сохраненные в памяти 79 видеоданных, могут быть получены, например, из запоминающего устройства 34, из локального источника видео, такого как камера, через проводную или беспроводную сеть передачи видеоданных, или

посредством доступа к физическим носителям данных. Память 79 видеоданных может составлять буфер кодированных изображений (CPB), который сохраняет кодированные видеоданные из битового потока кодированного видео. Буфер 92 декодированных изображений является одним примером буфера декодированных изображений (DPB), который сохраняет опорные видеоданные для использования при декодировании видеоданных видеodeкодером 30 (например, в режимах внутри- или меж-кадрового кодирования, также называемых режимами кодирования с внутри- или меж-кадровым предсказания). Память 79 видеоданных и DPB 92 может составлять любое из множества запоминающих устройств, такое как динамическая оперативная память (DRAM), включая синхронную DRAM (SDRAM), магниторезистивная RAM (MRAM), резистивная RAM (RRAM), или другие типы запоминающих устройств. Память 79 видеоданных и DPB 92 могут обеспечиваться одним и тем же запоминающим устройством или отдельными запоминающими устройствами. В различных примерах память 79 видеоданных может быть на кристалле с другими компонентами видеodeкодера 30 или вне кристалла относительно этих компонентов.

[0149] В ходе процесса декодирования видеodeкодер 30 принимает от видеокодера 20 битовый поток кодированного видео, который представляет видеоблоки кодированного видеослайса и связанные синтаксические элементы. Блок 80 энтропийного декодирования видеodeкодера 30 энтропийно декодирует битовый поток, чтобы сформировать квантованные коэффициенты, векторы движения и другие синтаксические элементы. Блок 80 энтропийного декодирования пересылает векторы движения и другие синтаксические элементы на блок 81 обработки предсказания. Видеodeкодер 30 может принимать синтаксические элементы на уровне видеослайса и/или видеоблока.

[0150] Когда видеослайс кодирован в виде внутренне кодированного (I) слайса, блок 84 обработки внутрикадрового предсказания в блоке 81 обработки предсказания может формировать данные предсказания для видеоблока текущего видеослайса на основании сигнализированного режима внутрикадрового предсказания и данных из предварительно декодированных блоков текущего кадра или изображения. Блок 81 обработки предсказания может быть сконфигурирован для реализации способов настоящего раскрытия для выполнения объединенного intra BC и межкадрового предсказания. Когда видеокادر кодирован в виде межкадрово кодированного (то есть B, или P) слайса, блок 82 компенсации движения в блоке 81 обработки предсказания создает предсказываемые блоки для видеоблока текущего видеослайса на основании векторов движения и других синтаксических элементов, принятых от блока 80 энтропийного декодирования. Предсказываемые блоки могут создаваться из одного из опорных изображений в одном из списков опорных изображений. Видеodeкодер 30 может строить списки опорных кадров, Список 0 и Список 1, используя способы построения по умолчанию, на основании опорных изображений, сохраненных в буфере 92 декодированных изображений.

[0151] Блок 82 компенсации движения определяет информацию предсказания для видеоблока текущего видеослайса путем разбора векторов движения и других синтаксических элементов, и использует информацию предсказания, чтобы создавать предсказываемые блоки для текущего декодируемого видеоблока. Например, блок 82 компенсации движения использует некоторые из принятых синтаксических элементов для определения режима предсказания (например, внутри- или меж-кадрового предсказания), используемого для кодирования видеоблоков видеослайса, типа слайса межкадрового предсказания (например, B-слайс, P-слайс или GPB-слайс), информации

построения для одного или нескольких списков опорных изображений для слайса, векторов движения для каждого, межкадрово кодированного видеоблока слайса, состояния межкадрового предсказания для каждого, межкадрово кодированного видеоблока слайса, и другой информации, чтобы декодировать видеоблоки в текущем видеослайсе.

[0152] Блок 82 компенсации движения может также выполнять интерполяцию на основании интерполяционных фильтров. Блок 82 компенсации движения может использовать интерполяционные фильтры, как использованы видеокодером 20 в ходе кодирования видеоблоков, чтобы вычислять интерполированные значения для субцелочисленных пикселей опорных блоков. В этом случае, блок 82 компенсации движения может определять интерполяционные фильтры, используемые видеокодером 20, исходя из принятых синтаксических элементов и использовать интерполяционные фильтры, чтобы создавать предсказываемые блоки.

[0153] Блок 86 обратного квантования обращенно квантует, то есть де-квантует, квантованные коэффициенты преобразования, обеспеченные в битовом потоке и декодированные блоком 80 энтропийного декодирования. Процесс обратного квантования может включать в себя использование параметра квантования, вычисляемого видеокодером 20 для каждого видеоблока в видеослайсе, чтобы определять степень квантования и, подобным образом, степень обратного квантования, которую следует применить. Блок 88 обработки обратного преобразования применяет обратное преобразование, например, обратное DCT, обратное целочисленное преобразование или концептуально подобный процесс обратного преобразования, к коэффициентам преобразования, чтобы создать остаточные блоки в пиксельной области.

[0154] После того, как блок 82 компенсации движения сгенерирует предсказываемый блок для текущего видеоблока на основании векторов движения и других синтаксических элементов, видеодекодер 30 формирует блок декодированного видео, суммируя остаточные блоки из блока 88 обработки обратного преобразования с соответствующими предсказываемыми блоками, сгенерированными блоком 82 компенсации движения. Сумматор 90 представляет компонент или компоненты, которые выполняют эту операцию суммирования. Если желательно, деблокирующий фильтр также может применяться, чтобы фильтровать декодированные блоки для того, чтобы удалить артефакты блочности изображения. Другие контурные фильтры (или в цикле кодирования, или после цикла кодирования) также могут использоваться, чтобы сглаживать пиксельные переходы, или иным образом повысить качество видео.

Декодированные видеоблоки в данном кадре или изображении затем сохраняются в буфере 92 декодированных изображений, который сохраняет опорные изображения, используемые для последующей компенсации движения. Буфер 92 декодированных изображений также сохраняет декодированное видео для более позднего воспроизведения на устройстве отображения, например, устройстве 32 отображения по Фиг.1.

[0155] Согласно аспектам настоящего раскрытия, видеодекодер 30 может быть сконфигурирован для выполнения любой комбинации способов, описанных в настоящем раскрытии.

[0156] Фиг.6 является структурной схемой, иллюстрирующей примерный процесс для декодирования видеоданных, соответствующий способам настоящего раскрытия. Процесс по Фиг.6 в общем описывается в виде выполняемого видеокодером 30 с целями иллюстрации, хотя множество других процессоров также могут выполнять процесс, показанный на Фиг.6.

[0157] В примере по Фиг.6 видеодекодер 30 может включать (200) текущее изображение и опорное изображение в список опорных изображений, опорное изображение является отличным от текущего изображения. Видеодекодер 30 может определять (202) со-размещенный блок опорного изображения, со-размещенный блок является со-размещаемым с текущим блоком текущего изображения. Видеодекодер 30 может выводить (204) временный предсказатель вектора движения исходя из со-размещенного блока. Видеодекодер 30 может определять (206), что временный предсказатель вектора движения имеет субпиксельную точность. Видеодекодер 30 может сдвигать вправо (208) временный предсказатель вектора движения, определенный имеющим субпиксельную точность. Видеодекодер 30 может определять (210), на основании сдвинутого вправо временного предсказателя вектора движения, предсказываемый блок внутри текущего изображения. Видеодекодер 30 может сформировать (212) блок декодированного видео путем суммирования выборок остаточного блока и соответствующих выборок предсказываемого блока.

[0158] Фиг.7 является структурной схемой, иллюстрирующей примерный процесс для кодирования видеоданных, соответствующий способам настоящего раскрытия. Процесс по Фиг.7 в целом описывается в виде выполняемого видеокодером 20 с целью иллюстрации, хотя множество других процессоров также могут выполнять процесс, показанный на Фиг.7.

[0159] В примере по Фиг.7 видеокодер 20 может включать (220) текущее изображение и опорное изображение в список опорных изображений, опорное изображение является отличным от текущего изображения. Видеокодер 20 может определять (222) со-размещенный блок опорного изображения, со-размещенный блок является со-размещаемым с текущим блоком текущего изображения. Видеокодер 20 может выводить (224) временный предсказатель вектора движения исходя из со-размещенного блока. Видеокодер 20 может определять (226), что временный предсказатель вектора движения имеет субпиксельную точность. Видеокодер 20 может сдвигать вправо (228) временный предсказатель вектора движения, определенный имеющим субпиксельную точность. Видеокодер 20 может определять (230), на основании сдвинутого вправо временного предсказателя вектора движения, предсказываемый блок внутри текущего изображения. Видеокодер 20 может формировать (232) остаточные данные, представляющие разности между предсказываемым блоком, который был определен на основании сдвинутого вправо временного предсказателя вектора движения, и текущим блоком.

[0160] Теперь здесь описываются примеры в соответствии со способами настоящего раскрытия. Следует понимать, что кодер видео, такой как видеокодер 20 или видеодекодер 30, может выполнять последующие примеры.

[0161] Пример 1. Способ кодирования или декодирования видеоданных, способ содержит: определение вектора движения для кодированного с внутрикадровой копией блока (intra BC) для блока, кодированного с использованием временного предсказания вектора движения (TMVP); определение со-размещенного блока на основании вектора TMVP; в ответ на определение, что со-размещенный блок является кодированным в Intra BC, использование вектора TMVP с целочисленной пиксельной точностью для определения опорного блока; и в ответ на определение, что со-размещенный блок является кодированным межкадрово, использование вектора TMVP с субцелочисленной пиксельной точностью, чтобы определить опорный блок.

[0162] Пример 2. Способ по примеру 1, дополнительно содержащий: кодирование блока, кодированного с внутрикадровой копией блока, на основании опорного блока или декодирование блока, кодированного с внутрикадровой копией блока, на основании

опорного блока (на который была ссылка?).

[0163] Пример 3. Способ по примеру 1, дополнительно содержащий: когда вектор TMVP имеет субцелочисленную пиксельную точность, смещение вправо вектора движения TMVP.

5 [0164] Пример 4. Способ по примеру 1, дополнительно содержащий:
в ответ на определение, что данный со-размещенный блок является кодированным в Intra BC, использование вектора TMVP с субцелочисленной пиксельной точностью.

[0165] Пример 5. Способ кодирования или декодирования видеоданных, способ
10 содержит: кодирование I-слайса видеоданных; и выведение значения `initType` для синтаксического элемента исходя из I-слайса на основании того, разрешен ли режим с внутрикадровой копией блока для блока I-слайса; и использование переменной `initType` для определения значения `ctxIdx` для CABAC кодирования синтаксического элемента.

[0166] Пример 6. Способ по примеру 5, дополнительно содержащий: определение, связан ли синтаксический элемент только с режимом `intra BC`; и ответ на определение,
15 что синтаксический элемент связан только с режимом `intra BC`, установку `initType`, равным 0.

[0167] Пример 7. Способ по примеру 5, дополнительно содержащий CABAC кодирование синтаксического элемента.

[0168] Пример 8. Способ по примеру 5, дополнительно содержащий CABAC
20 декодирование синтаксического элемента.

[0169] Пример 9. Способ по примеру 5, дополнительно содержащий:
допущение обоих режимов и внутреннего, и межкадрового предсказания для I-слайса, когда режим `intra BC` разрешен для I-слайса.

[0170] Пример 10. Способ кодирования или декодирования видеоданных, способ
25 содержит: кодированием слайса изображения, в ответ на определение, что режим с внутренней копией блока (`intra BC`) разрешен и слайс является I-слайсом, или в ответ на определение, что слайс является P-слайсом или B-слайсом: модифицирование построения списка опорных изображений, когда режим `intra BC` разрешен для слайса.

[0171] Пример 11. Способ по примеру 10, дополнительно содержащий:
30 использование только одного списка опорных изображений, когда слайс является I-слайсом и когда режим `intra BC` разрешен для слайса.

[0172] Пример 12. Способ по примеру 10, дополнительно содержащий:
использование второго независимого списка опорных изображений, когда слайс является I-слайсом и когда режим `intra BC` разрешен для слайса.

35 [0173] Пример 13. Способ по примеру 10, дополнительно содержащий:
вставку текущего изображения слайса в конец списка опорных изображений.

[0174] Пример 14. Способ по примеру 10, дополнительно содержащий:
добавление текущего изображения слайса в начало или перед долгосрочными опорными изображениями в списке опорных изображений.

40 [0175] Пример 15. Способ по примеру 14, в котором слайс является P-слайсом или B-слайсом.

[0176] Пример 16. Способ по примеру 10, дополнительно содержащий:
добавление текущего изображения в позицию в списке опорных изображений на основании типа слайса для слайса.

45 [0177] Пример 17. Способ по примеру 10, дополнительно содержащий:
добавление текущего изображения в позицию в списке опорных изображений на основании типа предсказания текущего изображения.

[0178] Пример 18. Способ по примеру 10, дополнительно содержащий:

определение опорного изображения исходя из списка опорных изображений; и вышеизложенную внутриконтурную фильтрацию, деблокирование, и адаптивное к выборке смещение опорного изображения, когда опорное изображение является предварительно декодированным изображением, или изображения.

5 [0179] Пример 19. Способ по примеру 10, дополнительно содержащий: определение для элемента кодирования изображения, что режим intra BC разрешен для элемента кодирования и режим пропуска разрешен для элемента кодирования; в ответ на определение, что режим пропуска разрешен для элемента кодирования, разбор только синтаксических элементов кандидатов слияния в ходе построения списка опорных
10 изображений.

[0180] Пример 20. Способ по примеру 10, дополнительно содержащий: определение, что режим intra BC разрешен для слайса; и в ответ на определение, что режим intra BC разрешен для слайса, определение, что процесс декодирования для изображения не изменяется.

15 [0181] Пример 21. Способ по примеру 10, дополнительно содержащий: кодирование списка опорных изображений.

[0182] Пример 22. Способ по примеру 10, дополнительно содержащий: декодирование списка опорных изображений. В одном примере из раскрытия режим внутрикадрового предсказания содержит один или несколько режимов из режима intra BC, режима intra
20 Angular, режима intra Planar или режима intra DC, и текущий блок видеоданных находится в любом из I-слайса, P-слайса или B-слайса.

[0183] Пример 23. Любая комбинация любого из примеров, описанных в настоящем раскрытии.

[0184] В другом примере раскрытия видеокодер 20 может быть дополнительно
25 сконфигурирован для определения конкретного опорного блока из одного или нескольких опорных блоков, подлежащих использованию в качестве предсказываемого блока для текущего блока видеоданных, причем кодирование текущего блока видеоданных с использованием режима intra BC содержит предсказание текущего блока видеоданных исходя из предсказываемого блока.

30 [0185] Должно быть признано, что в зависимости от примера, некоторые действия или события любого из способов, описанных здесь, могут выполняться в другой последовательности, могут добавляться, объединяться или пропускаться в целом (например, не все описанные действия или события необходимы для осуществления способов на практике). Кроме того, в некоторых примерах, действия или события могут
35 выполняться параллельно, например, посредством многопоточной обработки, обработки прерываний или множества процессоров, а не последовательно. Кроме того, тогда как некоторые аспекты настоящего раскрытия описаны выполняемыми единственным модулем или блоком с целью ясности, следует понимать, что способы настоящего раскрытия могут выполняться комбинацией блоков или модулей, связанных
40 с кодером видео.

[0186] В соответствии с этим раскрытием, термин "или" можно интерпретировать как "и/или", где контекст не предписывает иное. Вдобавок, тогда как фразы такие как "один или несколько" или "по меньшей мере, один" или подобные, могли использоваться для некоторых признаков, раскрытых здесь, но не для других; признаки, для которых
45 такая формулировка не использовалась, можно интерпретировать имеющими такое подразумеваемое значение, если контекст не предписывает иное.

[0187] Некоторые аспекты настоящего раскрытия были описаны относительно одного или нескольких стандартов кодирования видео (например, стандарта HEVC) с

целью иллюстрации. Однако, способы, описанные в настоящем раскрытии, могут быть полезными для других процессов кодирования видео, включая другие стандартные или составляющие собственность процессы кодирования видео, еще не разработанные или находящиеся в разработке.

5 [0188] Способы, описанные здесь, могут исполняться видеокодером 20 (например, фигуры Фиг. 1 и 4) и/или декодером 30 (например, фигуры Фиг. 1 и 5), оба из которых могут в общем называться кодером видео. Аналогично, кодирование видео может относиться к кодированию видео или декодированию видео, как применимо.

10 [0189] Хотя выше описаны конкретные комбинации различных аспектов способов, эти комбинации приведены просто для иллюстрации примеров способов, описанных в настоящем раскрытии. Соответственно, способы настоящего раскрытия не должны ограничиваться этими примерными комбинациями и могут охватывать любую мыслимую комбинацию различных аспектов способов, описанных в настоящем раскрытии. Способы, описанные в настоящем раскрытии, могут использоваться вместе
15 в любой комбинации.

[0190] В одном или нескольких примерах функции, описанные в настоящем раскрытии, могут быть реализованы в виде аппаратных средств, программного обеспечения, микропрограммного обеспечения или любой комбинации таковых. Если реализованы программно, функции могут сохраняться, в виде одной или нескольких инструкций или
20 кода, на читаемом компьютером носителе или передаваться на нем, и исполняться посредством аппаратного блока обработки. Читаемые компьютером носители могут охватывать читаемые компьютером носители данных, каковое соответствует материальному носителю, такому как среда хранения данных, или среда связи, включая любой носитель, который содействует передаче компьютерной программы из одного
25 места в другое, например, согласно протоколу связи. Таким образом, читаемые компьютером носители обычно могут соответствовать (1) материальному читаемому компьютером носителю данных, который является не-временным, или (2) среде связи, такой как сигнал или несущая. Носители для хранения данных могут быть любыми доступными носителями, к которым могут осуществлять доступ один или несколько
30 компьютеров, или один или несколько процессоров, чтобы извлекать инструкции, код и/или структуры данных для реализации способов, описанных в настоящем раскрытии. Компьютерный программный продукт может включать в себя читаемый компьютером носитель.

[0191] В качестве примера, и не ограничения, такие читаемые компьютером носители
35 могут содержать RAM, ROM, EEPROM, CD-ROM или другой накопитель на оптическом диске, накопитель на магнитном диске, или другие магнитные запоминающие устройства, флэш-память или любой другой носитель, который может использоваться для сохранения требуемого кода программы в форме инструкций или структур данных, и к которому может осуществляться доступ посредством компьютера. Кроме того, любое
40 соединение соответственно называют читаемым компьютером носителем. Например, если инструкции передают с веб-сайта, сервера, или от другого удаленного источника, используя коаксиальный кабель, оптический кабель, витую пару, цифровую абонентскую линию (DSL) или беспроводные технологии, такие как инфракрасная, радио и микроволновая связь, то коаксиальный кабель, оптический кабель, витая пара, DSL
45 или беспроводные технологии, такие как инфракрасная, радио и микроволновая связь включаются в определение носителя. Однако следует понимать, что читаемые компьютером носители данных и накопители данных не включают в себя соединения, несущие, сигналы или другие временные носители, а вместо этого ориентированы на

не-временные, материальные носители. Диск и немагнитный диск, как используется здесь, включает в себя компакт-диск (CD), лазерный диск, оптический диск, цифровой многофункциональный диск (DVD), гибкий диск и диск по технологии Blu-ray, где диски обычно воспроизводят данные магнитным способом, тогда как немагнитные диски
 5 воспроизводят данные оптически с помощью лазеров. Комбинации вышеуказанного также могут включаться в рамки читаемых компьютером носителей.

[0192] Инструкции могут исполняться одним или несколькими процессорами, такими как один или несколько цифровых процессоров сигналов (DSP), универсальные микропроцессоры, специализированные интегральные схемы (ASIC), программируемые
 10 вентиляемые матрицы (FPGA), или другой эквивалентной интегральной или дискретной логической схемой. Соответственно, термин "процессор", как используется здесь, может относиться к любой вышеизложенной структуре или любой другой структуре, подходящей для реализации способов, описанных здесь. Кроме того, в некоторых аспектах, функциональность, описанная здесь, может обеспечиваться в рамках
 15 специализированных аппаратных средств и/или программных модулей, сконфигурированных для кодирования и декодирования, или встраиваться в комбинированный кодек. Кроме того, способы могут быть полностью реализованы в одной или нескольких схемах или логических элементах.

[0193] Способы настоящего раскрытия могут быть реализованы в широком спектре устройств или аппаратуры, включая беспроводной телефон, интегральную схему (IC)
 20 или набор IC (например, набор микросхем). Различные компоненты, модули или блоки описываются в настоящем раскрытии, чтобы подчеркнуть функциональные аспекты устройств, сконфигурированных для выполнения раскрытых способов, но не требуют обязательной реализации различными аппаратно-реализованными блоками.

Предпочтительнее, как описано выше, различные блоки могут быть объединены в аппаратно-реализованный блок кодека или обеспечиваться набором
 25 межвзаимодействующих аппаратных модулей, включая один или несколько процессоров, как описано выше, вместе с подходящим программным обеспечением и/или микропрограммным обеспечением.

[0194] Были описаны различные примеры. Эти и другие примеры находятся в рамках
 30 объема последующей формулы изобретения.

(57) Формула изобретения

1. Способ декодирования видеоданных с использованием режима с внутрикадровой
 35 копией блока, причем способ содержит:

включение текущего изображения и опорного изображения в список опорных изображений, причем опорное изображение является отличным от текущего изображения;

определение со-размещенного блока опорного изображения, причем со-размещенный
 40 блок является со-размещаемым с текущим блоком текущего изображения;

выведение временного предсказателя вектора движения, исходя из со-размещенного блока;

определение, что временный предсказатель вектора движения имеет субпиксельную
 45 точность;

сдвиг вправо временного предсказателя вектора движения, определенного имеющим
 45 субпиксельную точность;

определение, на основании сдвинутого вправо временного предсказателя вектора движения, предсказываемого блока в текущем изображении;

формирование остаточного блока путем вычитания пиксельных значений предсказываемого блока из пиксельных значений текущего блока; и

формирование декодированного видеоблока путем суммирования выборок остаточного блока и соответствующих выборок предсказываемого блока, причем
5 выборки остаточного блока соответствуют разности между текущим блоком и предсказываемым блоком.

2. Способ по п.1, в котором список опорных изображений является первым списком опорных изображений, причем способ дополнительно содержит:

10 построение второго списка опорных изображений, который включает текущее изображение для В-слайса текущего изображения, на основании режима с внутрикадровой копией блока, разрешаемого для В-слайса.

3. Способ по п.1, в котором текущее изображение является первым изображением, и список опорных изображений является первым списком опорных изображений, причем способ дополнительно содержит:

15 получение, из битового потока, синтаксического элемента, указывающего, разрешена ли внутрикадровая копия блока для второго изображения; и

в качестве части процесса декодирования для I-слайса второго изображения, построение в ответ на определение на основании синтаксического элемента, что внутрикадровая копия блока разрешена для I-слайса, второго списка опорных
20 изображений, причем второй список опорных изображений включает второе изображение.

4. Способ по п.1, в котором включение текущего изображения и опорного изображения в список опорных изображений содержит одно из:

25 включения в список опорных изображений текущего изображения до долгосрочных изображений в списке опорных изображений; и

включения в список опорных изображений текущего изображения в позицию в списке опорных изображений на основании того, является ли слайс I-слайсом, Р-слайсом или В-слайсом.

5. Способ по п.1, дополнительно содержащий формирование списка кандидатов-векторов движения, который включает в себя, по меньшей мере, один кандидат-вектор
30 движения для кодированного в режиме внутрикадровой копии блока и, по меньшей мере, один кандидат для кодированного с межкадровым предсказанием блока.

6. Способ кодирования видеоданных с использованием режима с внутрикадровой копией блока, причем способ содержит:

35 включение текущего изображения и опорного изображения в список опорных изображений, причем опорное изображение является отличным от текущего изображения;

определение со-размещенного блока опорного изображения, причем со-размещенный блок является со-размещаемым с текущим блоком текущего изображения;

40 выведение временного предсказателя вектора движения, исходя из со-размещенного блока;

определение, что временный предсказатель вектора движения имеет субпиксельную точность;

45 сдвиг вправо временного предсказателя вектора движения, определенного имеющим субпиксельную точность;

определение, на основании сдвинутого вправо временного предсказателя вектора движения, предсказываемого блока в текущем изображении; и

формирование остаточных данных, представляющих разности между

предсказываемым блоком, который был определен на основе сдвинутого вправо временного предсказателя вектора движения, и текущим блоком.

7. Способ по п.6, в котором список опорных изображений является первым списком опорных изображений, причем способ дополнительно содержит:

- 5 построение второго списка опорных изображений, который включает текущее изображение для В-слайса текущего изображения, на основании режима с внутрикадровой копией блока, разрешаемого для В-слайса.

8. Способ по п.6, в котором текущее изображение является первым изображением, список опорных изображений является первым списком опорных изображений, причем
10 способ дополнительно содержит:

построение второго списка опорных изображений для I-слайса второго изображения, когда внутрикадровая копия блока разрешена для I-слайса, причем второй список опорных изображений включает второе изображение.

9. Способ по п.6, в котором включение текущего изображения и опорного
15 изображения в список опорных изображений содержит одно из:

включения в список опорных изображений текущего изображения до долгосрочных изображений в списке опорных изображений; и

- включения в список опорных изображений текущего изображения в позицию в списке опорных изображений на основании того, является ли слайс I-слайсом, Р-слайсом или
20 В-слайсом.

10. Способ по п.6, дополнительно содержащий формирование списка кандидатов-векторов движения, который включает, по меньшей мере, один кандидат-вектор движения для кодированного в режиме внутрикадровой копии блока и, по меньшей мере, один кандидат для кодированного с межкадровым предсказанием блока.

- 25 11. Устройство для декодирования видеоданных с использованием режима с внутрикадровой копией блока, причем устройство содержит:

память, сконфигурированную с возможностью сохранения видеоданных опорного изображения; и

- один или более процессоров, сконфигурированных с возможностью:
30 включать текущее изображение и опорное изображение в список опорных изображений, причем опорное изображение является отличным от текущего изображения;

определять со-размещенный блок опорного изображения, причем со-размещенный блок является со-размещаемым с текущим блоком текущего изображения;

- 35 выводить временный предсказатель вектора движения, исходя из со-размещенного блока;

определять, что временный предсказатель вектора движения имеет субпиксельную точность;

- сдвигать вправо временный предсказатель вектора движения, определенный имеющим
40 субпиксельную точность;

определять на основании сдвинутого вправо временного предсказателя вектора движения предсказываемый блок в текущем изображении;

формировать остаточный блок путем вычитания пиксельных значений предсказываемого блока из пиксельных значений текущего блока; и

- 45 формировать блок декодированного видео путем суммирования выборок остаточного блока и соответствующих выборок предсказываемого блока, причем выборки остаточного блока соответствуют разности между текущим блоком и предсказываемым блоком.

12. Устройство по п.11, в котором список опорных изображений является первым списком опорных изображений, причем один или более процессоров дополнительно сконфигурированы с возможностью:

5 осуществлять построение второго списка опорных изображений, который включает текущее изображение для В-слайса текущего изображения, на основании режима с внутрикадровой копией блока, разрешаемого для В-слайса.

13. Устройство по п.11, в котором текущее изображение является первым изображением, список опорных изображений является первым списком опорных изображений, причем один или более процессоров дополнительно сконфигурированы
10 с возможностью:

получать из битового потока синтаксический элемент, указывающий, разрешена ли внутрикадровая копия блока для второго изображения; и

15 в качестве части процесса декодирования для I-слайса второго изображения, создавать в ответ на определение на основании синтаксического элемента, что внутрикадровая копия блока разрешена для I-слайса, второй список опорных изображений, причем второй список опорных изображений включает второе изображение.

14. Устройство по п.11, в котором один или более процессоров сконфигурированы с возможностью выполнять одно из:

20 включать, в список опорных изображений, текущее изображение перед долгосрочными изображениями в списке опорных изображений; и

включать, в список опорных изображений, текущее изображение в позицию в списке опорных изображений на основании того, является ли слайс I-слайсом, Р-слайсом или В-слайсом.

15. Устройство по п.11, в котором один или более процессоров дополнительно
25 сконфигурированы с возможностью формировать список кандидатов-векторов движения, который включает в себя, по меньшей мере, один кандидат-вектор движения для кодированного в режиме внутрикадровой копии блока и, по меньшей мере, один кандидат для кодированного с межкадровым предсказанием блока.

16. Устройство по п.11, причем устройство содержит, по меньшей мере, одно из:
30 интегральной схемы;

микропроцессора; или

35 устройства беспроводной связи, содержащего одно или оба из устройства отображения, сконфигурированного с возможностью отображения текущего изображения, или камеры, сконфигурированной с возможностью захвата текущего изображения.

17. Устройство по п.11, причем устройство дополнительно содержит устройство отображения, сконфигурированное с возможностью отображения декодированных видеоданных.

40 18. Устройство по п.11, в котором один или более процессоров дополнительно сконфигурированы с возможностью кодирования видеоданных, причем устройство дополнительно содержит камеру, сконфигурированную с возможностью захвата видеоданных.

19. Устройство для кодирования видеоданных с использованием режима с внутрикадровой копией блока, причем устройство содержит:

45 память, сконфигурированную с возможностью сохранения видеоданных опорного изображения; и

один или более процессоров, сконфигурированных с возможностью:

включать текущее изображение и опорное изображение в список опорных

изображений, причем опорное изображение является отличным от текущего изображения;

определять со-размещенный блок опорного изображения, причем со-размещенный блок является со-размещаемым с текущим блоком текущего изображения;

5 выводить временный предсказатель вектора движения, исходя из со-размещенного блока;

определять, что временный предсказатель вектора движения имеет субпиксельную точность;

10 сдвигать вправо временный предсказатель вектора движения, определенный имеющим субпиксельную точность;

определять на основании сдвинутого вправо временного предсказателя вектора движения предсказываемый блок в текущем изображении; и

15 формировать остаточные данные, представляющие разности между предсказываемым блоком, который был определен на основе сдвинутого вправо временного предсказателя вектора движения, и текущим блоком.

20. Устройство по п.19, в котором список опорных изображений является первым списком опорных изображений, причем один или более процессоров дополнительно сконфигурированы с возможностью:

20 осуществлять построение второго списка опорных изображений, который включает текущее изображение для В-слайса текущего изображения, на основании режима с внутрикадровой копией блока, разрешаемого для В-слайса;

кодировать первый блок I-слайса второго изображения с использованием внутрикадрового предсказания; и

25 кодировать второй блок I-слайса с использованием режима с внутрикадровой копией блока.

21. Устройство по п.19, в котором текущее изображение является первым изображением, список опорных изображений является первым списком опорных изображений, причем один или более процессоров дополнительно сконфигурированы с возможностью:

30 создавать второй список опорных изображений для I-слайса второго изображения, когда внутрикадровая копия блока разрешена для I-слайса, причем второй список опорных изображений включает в себя второе изображение.

22. Устройство по п.19, в котором один или более процессоров сконфигурированы с возможностью выполнять одно из:

35 включать в список опорных изображений текущее изображение перед долгосрочными изображениями в списке опорных изображений; и

включать в список опорных изображений текущее изображение в позиции в списке опорных изображений на основании того, является ли слайс I-слайсом, P-слайсом или В-слайсом.

40 23. Устройство по п.19, в котором один или более процессоров дополнительно сконфигурированы с возможностью формировать список кандидатов-векторов движения, который включает, по меньшей мере, один кандидат-вектор движения для кодированного в режиме внутрикадровой копии блока и, по меньшей мере, один кандидат для кодированного с межкадровым предсказанием блока.

45 24. Устройство по п.19, причем устройство содержит, по меньшей мере, одно из: интегральной схемы;

микропроцессора; или

устройства беспроводной связи, содержащего одно или оба из устройства

отображения, сконфигурированного с возможностью отображения текущего изображения, или камеры, сконфигурированной с возможностью захвата текущего изображения.

25. Устройство по п.19, причем устройство дополнительно содержит устройство
5 отображения, сконфигурированное с возможностью отображения декодированных видеоданных.

26. Устройство по п.19, в котором один или более процессоров дополнительно сконфигурированы с возможностью декодирования видеоданных.

27. Устройство для кодирования видеоданных с использованием режима с
10 внутрикадровой копией блока, причем устройство содержит:

средство для включения текущего изображения и опорного изображения в список опорных изображений, причем опорное изображение является отличным от текущего изображения;

15 средство для определения со-размещенного блока опорного изображения, причем со-размещенный блок является со-размещаемым с текущим блоком текущего изображения;

средство для вывода временного предсказателя вектора движения, исходя из со-размещенного блока;

20 средство для определения, что временный предсказатель вектора движения имеет субпиксельную точность;

средство для сдвига вправо временного предсказателя вектора движения, определенного имеющим субпиксельную точность; и

средство для определения, на основании сдвинутого вправо временного предсказателя вектора движения, предсказываемого блока в текущем изображении.

28. Постоянный машиночитаемый носитель данных, имеющий инструкции, хранящиеся на нем, которые при исполнении предписывают одному или более процессорам вычислительного устройства, сконфигурированного для кодирования видеоданных:

30 включать текущее изображение и опорное изображение в список опорных изображений, причем опорное изображение является отличным от текущего изображения;

определять со-размещенный блок опорного изображения, причем со-размещенный блок является со-размещаемым с текущим блоком текущего изображения;

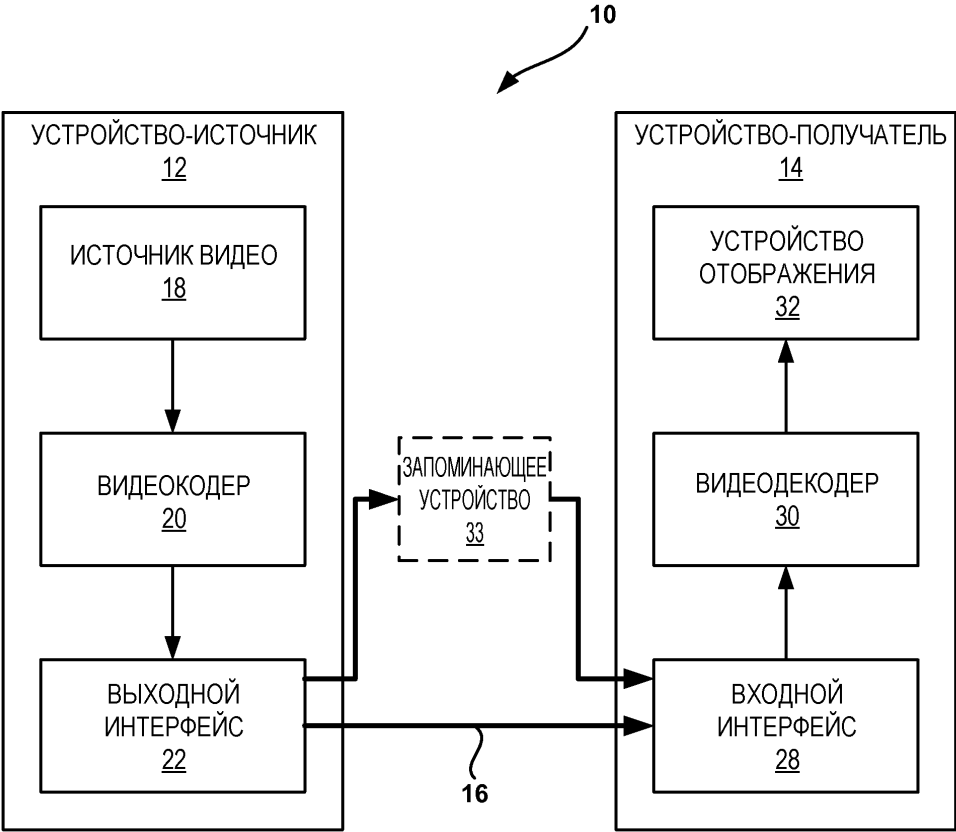
35 выводить временный предсказатель вектора движения, исходя из со-размещенного блока;

определять, что временный предсказатель вектора движения имеет субпиксельную точность;

сдвигать вправо временный предсказатель вектора движения, определенный имеющим субпиксельную точность; и

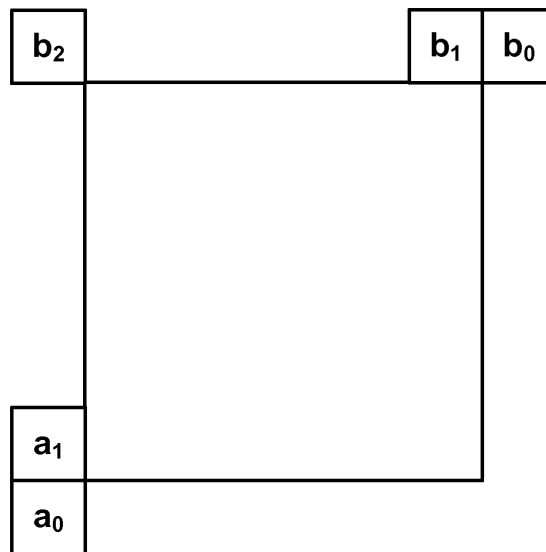
40 определять на основании сдвинутого вправо временного предсказателя вектора движения предсказываемый блок в текущем изображении.

1/7

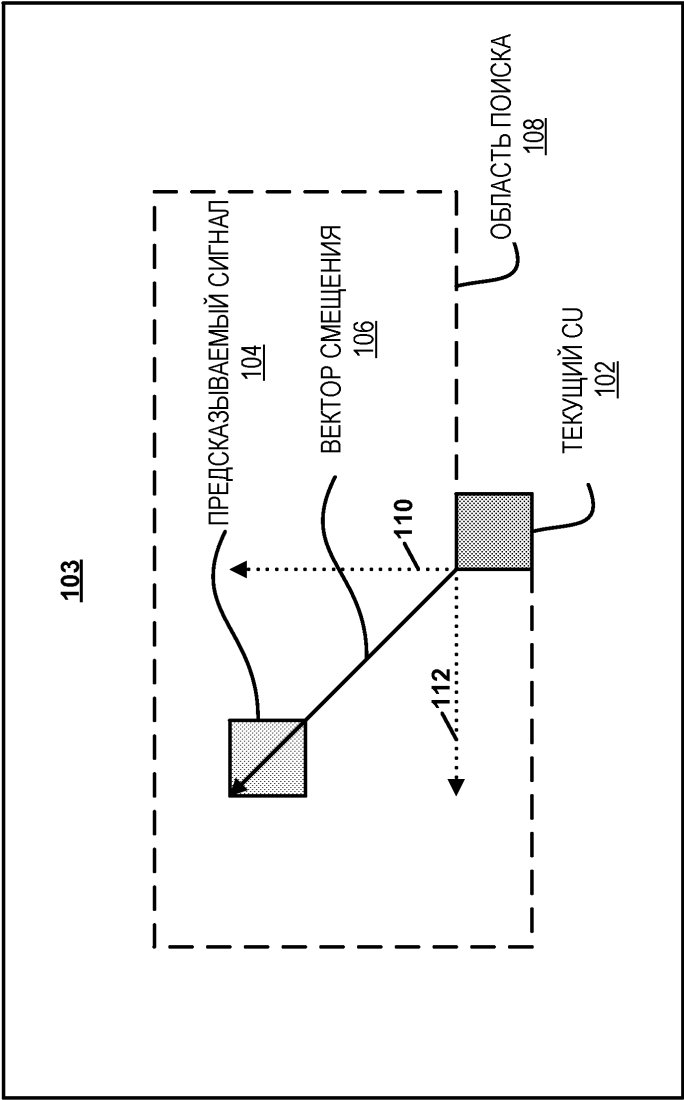


ФИГ. 1

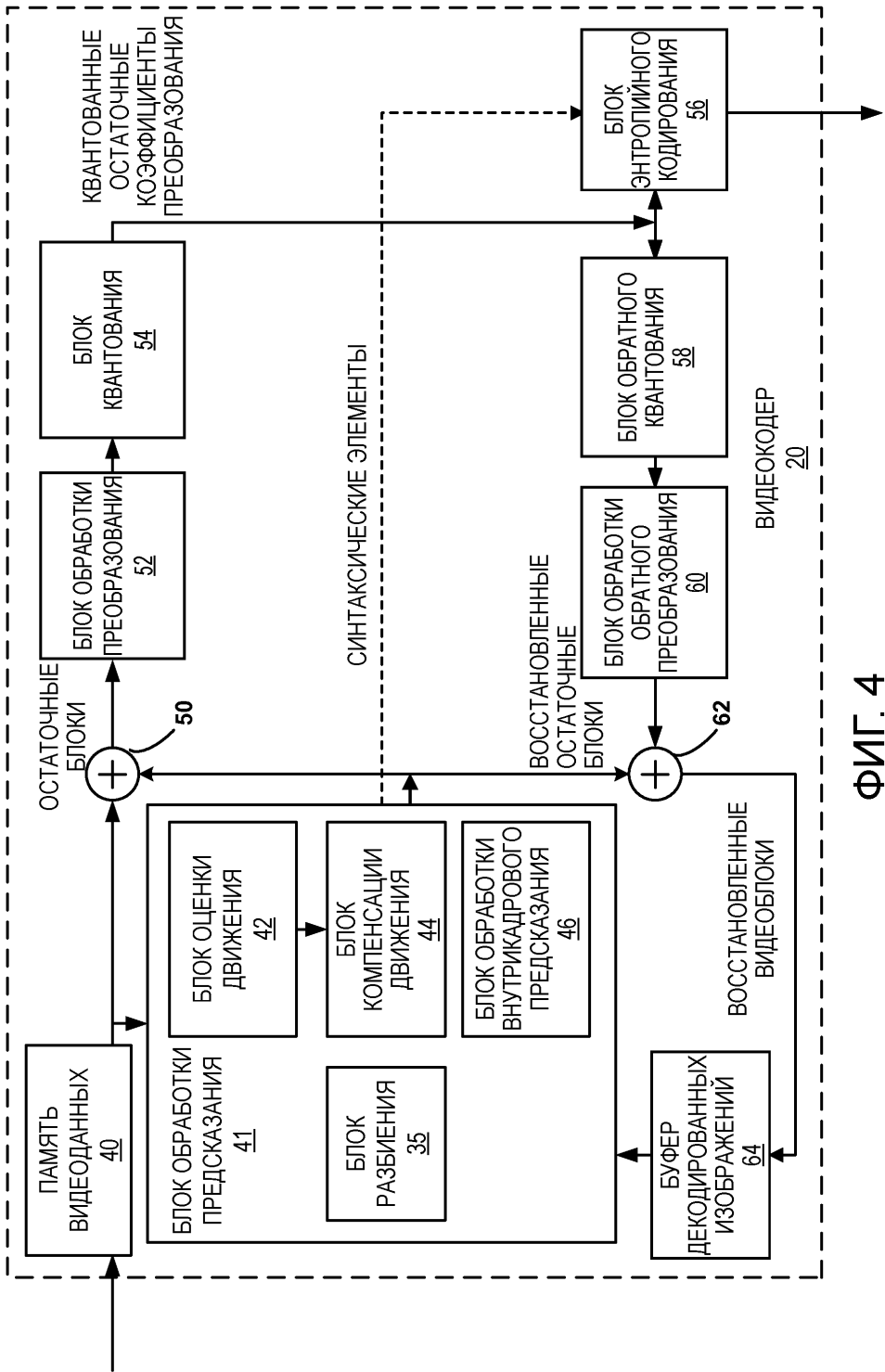
2/7



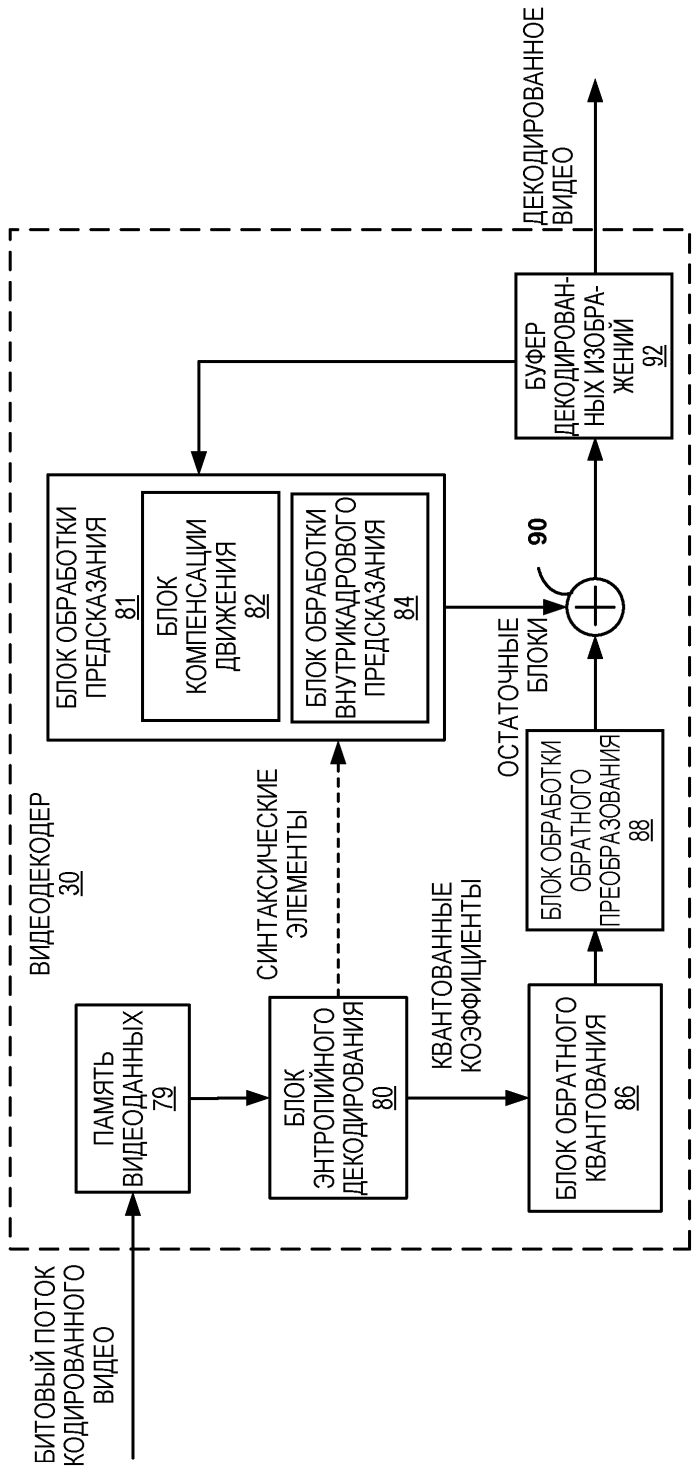
ФИГ. 2



ФИГ. 3

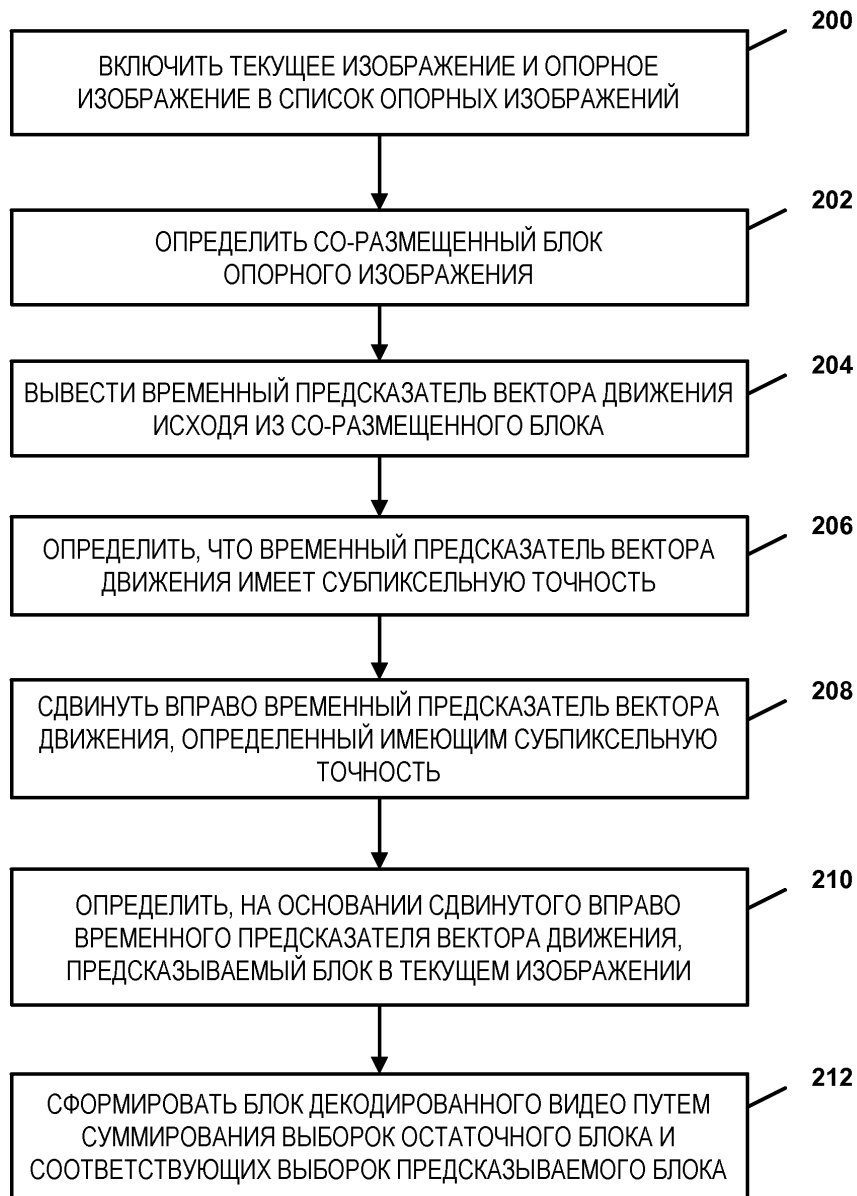


5/7



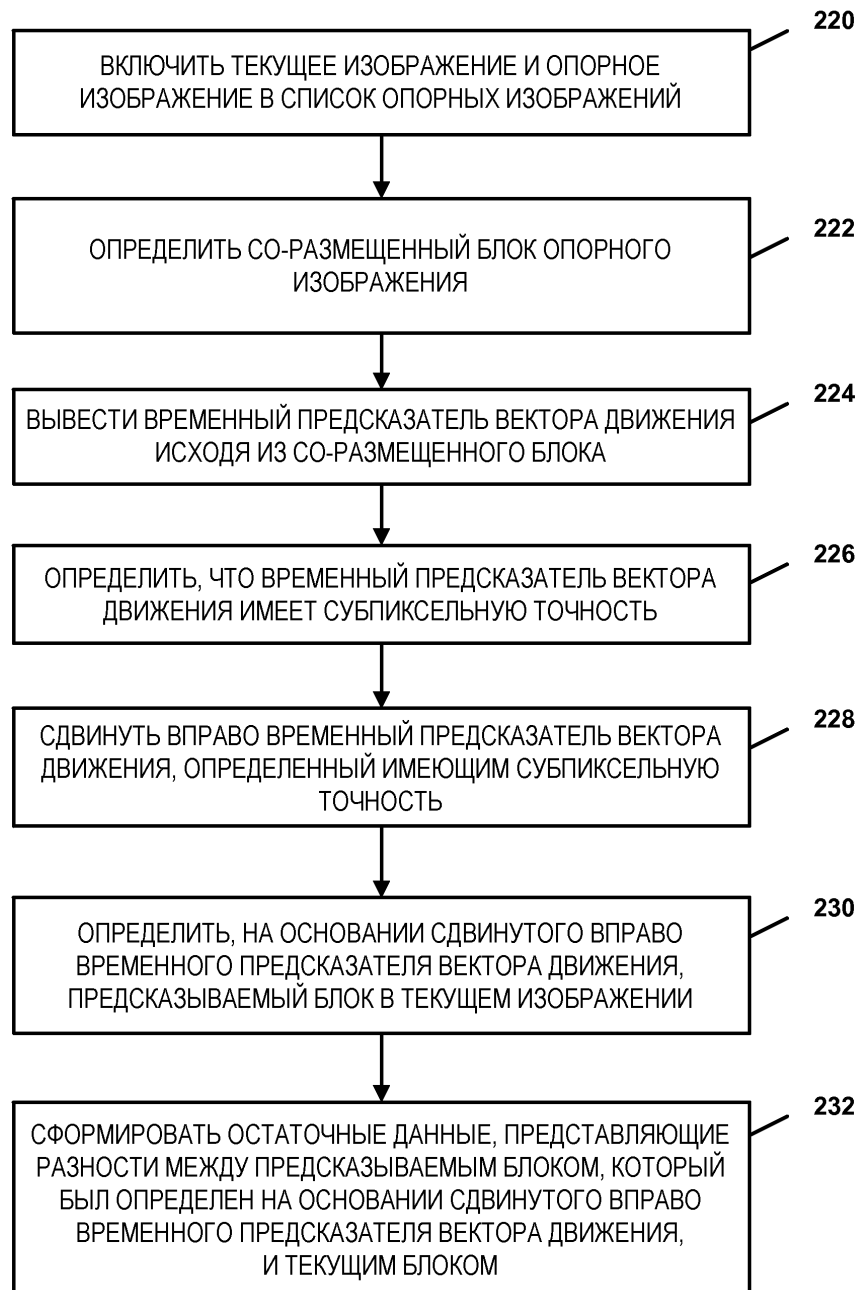
ФИГ. 5

6/7



ФИГ. 6

7/7



ФИГ. 7