



(51) МПК  
*H04N 19/91* (2014.01)  
*H04N 19/50* (2014.01)  
*G06T 9/00* (2006.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА  
 ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК

*H04N 19/13* (2006.01); *H04N 19/91* (2006.01); *H04N 19/124* (2006.01); *H04N 19/44* (2006.01); *H04N 19/61* (2006.01); *H04N 19/625* (2006.01)

(21)(22) Заявка: 2017103486, 20.06.2012

(24) Дата начала отсчета срока действия патента:  
20.06.2012

Дата регистрации:  
02.03.2018

Приоритет(ы):

(30) Конвенционный приоритет:  
24.06.2011 FR 11/55,606

Номер и дата приоритета первоначальной заявки,  
из которой данная заявка выделена:  
2014102226 24.06.2011

(45) Опубликовано: 02.03.2018 Бюл. № 7

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

(72) Автор(ы):

ГЕНРИ Феликс (FR),  
ПАТЁ Стефан (FR),  
КЛЭР Гордон (FR)

(73) Патентообладатель(и):

ДОЛБИ ИНТЕРНЭШНЛ АБ (NL)

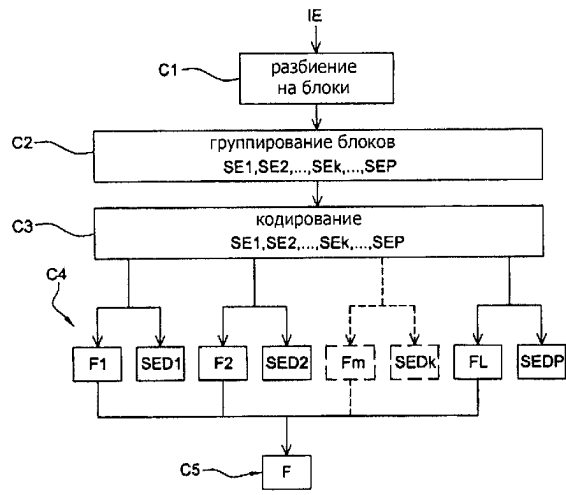
(56) Список документов, цитированных в отчете  
о поиске: US 2010/0135416 A1, 03.06.2010. US  
6480537 B1, 12.11.2002. US 2005/0038837 A1,  
17.02.2005. HEIKO SCHWARZ et al. статья  
"САВАС AND SLICES", опубликованная  
21.07.2002, 17 стр. [он-лайн] [найдено 2016-  
01-25], найдено в Интернет:  
[http://wftp3.itu.int/av-arch/jvt-site/  
2002\\_07\\_Klagenfurt/](http://wftp3.itu.int/av-arch/jvt-site/2002_07_Klagenfurt/). US 2008/0246637 A1,  
09.10.2008. RU 2330325 C2, 27.07.2008.

(54) СПОСОБ КОДИРОВАНИЯ И ДЕКОДИРОВАНИЯ ИЗОБРАЖЕНИЙ, УСТРОЙСТВО  
КОДИРОВАНИЯ И ДЕКОДИРОВАНИЯ И СООТВЕТСТВУЮЩИЕ КОМПЬЮТЕРНЫЕ  
ПРОГРАММЫ

(57) Реферат:

Группа изобретений относится к технологиям кодирования/декодирования видеоданных. Техническим результатом является повышение эффективности кодирования/декодирования изображения. Предложен способ кодирования по меньшей мере одного изображения. Способ содержит этапы, на которых осуществляют сегментирование изображения на множество блоков, группировку блоков в predetermined число подмножеств блоков. Далее способ содержит этап, на котором осуществляют кодирование, с использованием модуля энтропийного кодирования, текущего блока из

упомянутых подмножеств блоков. При этом кодирование содержит, когда текущий блок является первым блоком в порядке кодирования подмножества, которое не является первым подмножеством изображения в порядке кодирования: определение вероятности возникновения символа для текущего блока, причем вероятности являются теми, которые были определены посредством кодирования некоторого predetermined блока из по меньшей мере одного другого подмножества. б н. и б з.п. ф-лы, 14 ил.



Фиг. 2А

RU 2646345 C1

RU 2646345 C1



FEDERAL SERVICE  
FOR INTELLECTUAL PROPERTY

(51) Int. Cl.  
*H04N 19/91* (2014.01)  
*H04N 19/50* (2014.01)  
*G06T 9/00* (2006.01)

(12) **ABSTRACT OF INVENTION**

(52) CPC

*H04N 19/13* (2006.01); *H04N 19/91* (2006.01); *H04N 19/124* (2006.01); *H04N 19/44* (2006.01); *H04N 19/61* (2006.01); *H04N 19/625* (2006.01)

(21)(22) Application: **2017103486, 20.06.2012**

(24) Effective date for property rights:  
**20.06.2012**

Registration date:  
**02.03.2018**

Priority:

(30) Convention priority:  
**24.06.2011 FR 11/55,606**

Number and date of priority of the initial application,  
from which the given application is allocated:  
**2014102226 24.06.2011**

(45) Date of publication: **02.03.2018 Bull. № 7**

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

(72) Inventor(s):

**HENRY, Felix (FR),  
PATEUX, Stephane (FR),  
CLARE, Gordon (FR)**

(73) Proprietor(s):

**DOLBY INTERNATIONAL AB (NL)**

(54) **IMAGE ENCODING AND DECODING METHOD, DEVICE FOR ENCODING AND DECODING AND CORRESPONDING SOFTWARE**

(57) Abstract:

FIELD: image processing means.

SUBSTANCE: group of inventions relates to video encoding/decoding equipment. Method for encoding at least one image is proposed. Method comprises the stages of segmenting an image into a plurality of blocks, grouping the blocks into a predetermined number of block subsets. Method further comprises the stage of encoding the current block of the above subset of blocks with the use of the entropy encoding unit. In this case, when the current block is the first block in the encoding

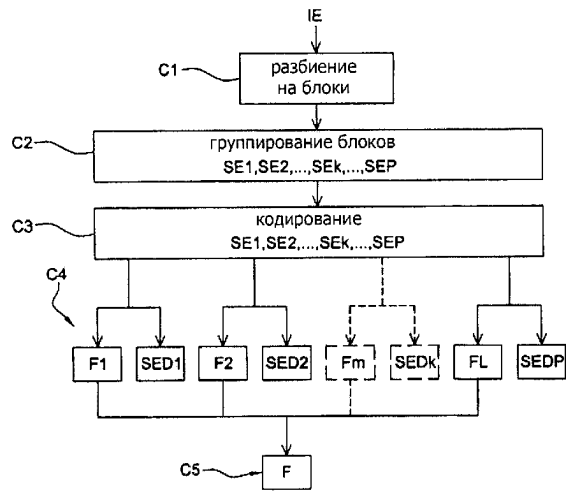
order of the subset that is not the first subset of the image in the encoding order, encoding comprises: determining the probability of occurrence of a symbol for the current block, the probabilities being those determined by encoding of a predetermined block from at least one other subset.

EFFECT: technical result is high efficiency of encoding/decoding images.

12 cl, 14 dwg

RU 2 646 345 C1

RU 2 646 345 C1



Фиг. 2А

RU 2646345 C1

RU 2646345 C1

**Область техники, к которой относится изобретение**

Настоящее изобретение относится в общем случае к области обработки изображений и более конкретно к кодированию и декодированию цифровых изображений и последовательностей цифровых изображений.

5 Настоящее изобретение, таким образом, может быть применено к способам видеокодирования, реализованным как в современных видеокодирующих устройствах (стандарты MPEG, H.264 и т.п.), так и в будущих видеокодирующих устройствах (стандарты ITU-T/VCEG (H.265) или ISO/MPEG (HVC)).

**Уровень техники**

10 Современные видеокодирующие устройства (стандарты MPEG, H.264 и т.п.) используют поблочное представление видеопоследовательности. Изображения разбивают на макроблоки, каждый макроблок в свою очередь разбивают на блоки и каждый блок или макроблок кодируют с применением внутрикадрового или межкадрового прогнозирования. Таким образом, некоторые изображения кодируют  
15 посредством пространственного прогнозирования (внутрикадровое прогнозирование), а другие изображения кодируют посредством временного прогнозирования (межкадровое прогнозирование) относительно одного или нескольких кодированных и/или декодированных опорных изображений с применением компенсации движения, известной специалистам в этой области. Более того, для каждого блока может быть  
20 кодирован остаточный блок, соответствующий исходному блоку, из которого вычтен результат прогнозирования. Коэффициенты этого блока квантуют, возможно, после преобразования, и затем кодируют посредством энтропийного кодирующего устройства.

Для применения внутрикадрового прогнозирования и межкадрового прогнозирования требуется, чтобы были доступны некоторые блоки, которые были ранее кодированы  
25 и декодированы, чтобы их можно было использовать в декодере или в кодирующем устройстве для прогнозирования текущего блока. Схематичный пример кодирования с прогнозированием, подобного рассматриваемому, представлен на фиг. 1, на которой изображение  $I_N$  разбито на блоки, текущий блок  $MB_i$  этого изображения подвергается  
30 кодированию с прогнозированием относительно заданного числа трех предварительно кодированных и декодированных блоков  $MB_{r_1}$ ,  $MB_{r_2}$  и  $MB_{r_3}$ , обозначенных заштрихованными стрелками. Упомянутые выше три блока, в частности, представляют собой блок  $MB_{r_1}$ , расположенный непосредственно слева от текущего блока  $MB_i$ , и два блока  $MB_{r_2}$  и  $MB_{r_3}$ , расположенные непосредственно сверху и сверху справа от текущего  
35 блока  $MB_i$ .

Более конкретный интерес здесь представляет энтропийное кодирующее устройство. Это энтропийное кодирующее устройство кодирует информацию в порядке ее поступления. Обычно блоки просматривают строка за строкой по схеме "растрового сканирования", как показано на фиг. 1 линией PRS, начиная от блока в верхнем левом  
40 углу изображения. Для каждого блока в энтропийное кодирующее устройство последовательно передают различные сегменты информации для представления этого блока (тип блока, режим прогнозирования, остаточные коэффициенты и т.п.).

На сегодня уже известно обладающее приемлемой сложностью эффективное устройство арифметического кодирования, именуемое устройством контекстно-ориентированного адаптивного двоичного арифметического кодирования ("САВАС"  
45 ("Context Adaptive Binary Arithmetic Coder")) и введенное в стандарт сжатия AVC (известный также под названиями (ISO-MPEG4 часть 10 и ITU-T H.264)).

Такое энтропийное кодирующее устройство реализует различные принципы:

- арифметическое кодирование: кодирующее устройство, такое как устройство, первоначально описанное в документе Й. Риссанен и Г.Г. Лэнгдон, мл. "Универсальное моделирование и кодирование" (J. Rissanen and G. G. Langdon Jr, "Universal modeling and coding") IEEE Trans. Inform. Theory, vol. IT-27, pp. 12-23, Jan. 1981, используется для

5 кодирования символа и вероятности появления этого символа;

- адаптация к контексту: здесь это охватывает адаптацию вероятности появления символа, подлежащего кодированию. С одной стороны "обучение" происходит прямо в процессе работы ("на лету"). С другой стороны, в зависимости от состояния ранее

10 кодированной информации для кодирования используется конкретный контекст. Каждому контексту здесь соответствует некая присущая ему собственная вероятность появления символа. Например, контекст соответствует типу кодируемого символа (представление коэффициента остатка, сигнализация о режиме кодирования и т.п.) согласно данной конфигурации или состоянию ближайшего окружения (соседства) (например, число ближайших соседствующих блоков, для которых выбран режим

15 внутрикадрового прогнозирования, и т.п.);

- бинаризация: осуществляется формирование последовательности битов символов, подлежащих кодированию. После этого полученные различные биты последовательно передают в двоичное энтропийное кодирующее устройство.

20 Таким образом, это энтропийное кодирующее устройство реализует, для каждого используемого контекста, систему для обучения вероятностям в процессе работы применительно к ранее кодированным символам для рассматриваемого контекста. Это обучение основано на порядке кодирования этих символов. Обычно изображение просматривают согласно описанному выше порядку типа "растрового сканирования".

25 В процессе кодирования данного символа  $b$ , который может быть равен 0 или 1, результат обучения вероятности  $p_i$  появления этого символа обновляют для текущего блока  $MB_i$  следующим образом:

$$p_i(b=0) = \alpha \cdot p_{i-1}(b=0) + \begin{cases} (1-\alpha) & \text{если кодируемый бит равен 0} \\ 0 & \text{иначе} \end{cases}$$

30 где  $\alpha$  представляет собой заданную величину, например 0,95, и  $p_{i-1}$  представляет собой вероятность появления символа, вычисленную после последнего появления этого символа.

На фиг. 1 представлен схематичный пример такого энтропийного кодирования, в котором такому энтропийному кодированию подвергается текущий блок  $MB_i$

35 изображения  $I_N$ . Когда начинается энтропийное кодирование блока  $MB_i$ , используют вероятности появления символов, полученные после кодирования ранее кодированного и декодированного блока, в качестве которого выбирают блок, непосредственно предшествующий текущему блоку  $MB_i$  в соответствии с описанной выше схемой просмотра блоков строка за строкой по принципу "растрового сканирования". Такое

40 обучение на основе взаимозависимости от блока к блоку показано на фиг. 1 для некоторых блоков (только для ясности картинки) посредством тонких стрелок.

Недостаток такого типа энтропийного кодирования состоит в том, что вероятности, используемые при кодировании символа, расположенного в начале строки, соответствуют главным образом вероятностям, наблюдаемым для символов,

45 находящихся в конце предшествующей строки, если применять просмотр блоков в порядке "растрового кодирования". Теперь, применительно к возможным пространственным вариациям вероятностей символов (например, для символов, связанных с сегментом информации движения, движение в правой части изображения

может отличаться от движения, наблюдаемого в левой части, и это, по всей видимости, отразится на вытекающих отсюда локальных вероятностях), можно наблюдать недостаток локальной конформности вероятностей, что возможно способно привести к потере эффективности во время кодирования.

5 Для ограничения этого явления были внесены предложения о модификации порядка просмотра блоков с целью обеспечить улучшенную локальную согласованность, но кодирование и декодирование остаются последовательными.

В этой области лежит другой недостаток энтропийного кодирующего устройства этого типа. Действительно, кодирование и декодирование символа зависят от состояния "выученной" для него вероятности, поэтому декодирование может быть произведено только в том же порядке, который был использован во время кодирования. Обычно декодирование в этом случае может быть только последовательным, что препятствует параллельному декодированию нескольких символов (например, чтобы получить выигрыш от многоядерных архитектур).

15 Документ: Томас Виганд, Гэри Дж. Салливан, Гисле Бьонтегаард и Аджай Лутра "Обзор стандарта видеокодирования H.264/AVC" (Thomas Wiegand, Gary J. Sullivan, Gisle Bjontegaard, and Ajay Luthra, "Overview of the H.264/AVC Video Coding Standard", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, No. 7, pp. 560-576, July 2003), указывает, что энтропийное кодирующее устройство стандарта САВАС имеет характерную особенность назначения нецелочисленного количества битов каждому символу текущего алфавита, подлежащему кодированию, это является предпочтительным для вероятностей появления символов свыше 0,5. Конкретнее, кодирующее устройство стандарта САВАС ожидает момента, когда оно прочитает несколько символов, и затем назначает этой группе прочитанных символов заданное 20 число битов, которые это кодирующее устройство записывает в сжатом потоке данных для передачи декодеру. Таким образом, такое предложение делает возможным "обобществить" биты нескольких символов и кодировать символ посредством дробного числа битов, это число отражает информацию, более близкую к информации, реально передаваемой символом. Другие биты, ассоциированные с прочитанными символами, не передают в составе сжатого потока, а сохраняют в режиме готовности, ожидая назначения одному или нескольким новым символам, прочитанным кодирующим устройством стандарта САВАС, что делает возможным снова обобществить эти другие символы. Указанное энтропийное кодирующее устройство предпринимает известным 25 способом в данный момент времени "опустошение" этих переданных битов. Иными словами, в этот данный момент времени кодирующее устройство извлекает биты, еще не переданные, и записывает в состав сжатого потока, адресованного декодеру. Такое опустошение имеет место, например, в момент, в который прочитан последний символ, подлежащий кодированию, чтобы обеспечить, что сжатый поток действительно содержит все биты, что позволит декодеру осуществить декодирование всех символов алфавита. В более общем смысле, момент, в который происходит указанное 30 опустошение, определяется в функции характеристик и функциональных возможностей, специфичных для рассматриваемого кодирующего устройства/декодера.

Документ, доступный в Интернете по адресу [http://research.microsoft.com/en-us/um/people/jinl/paper\\_2002/msgr\\_ipeg.htm](http://research.microsoft.com/en-us/um/people/jinl/paper_2002/msgr_ipeg.htm) на 15 апреля 2011 г., описывает способ кодирования неподвижных изображений в соответствии со стандартом сжатия JPEG2000. Согласно 35 этому способу данные неподвижного изображения подвергают дискретному вейвлетному преобразованию, за которым следует квантование, что делает возможным получение квантованных вейвлетных коэффициентов, с которыми соответственно

ассоциированы индексы квантования. Полученные индексы квантования кодируют посредством энтропийного кодирующего устройства. Квантованные коэффициенты предварительно группируют в прямоугольные блоки, именуемые кодовыми блоками, обычно размером 64×64 или 32×32. Затем осуществляют независимое кодирование каждого кодового блока посредством энтропийного кодирования. Таким образом, энтропийное кодирующее устройство во время кодирования текущего кодового блока не использует вероятности появления символов, вычисленные в ходе кодирования предшествующих кодовых блоков. Поэтому указанное энтропийное кодирующее устройство находится в инициализированном состоянии в начале кодирования каждого кодового блока. Преимуществом такого способа является декодирование данных кодового блока без необходимости декодировать соседние кодовые блоки. Таким образом, например, сегмент клиентского программного обеспечения может потребовать от сегмента серверного программного обеспечения предоставить только те сжатые кодовые блоки, которые нужны клиенту для декодирования выделенной части изображения. Преимуществом такого способа является также возможность параллельного кодирования и/или декодирования кодовых блоков. Таким образом, чем меньше размер кодового блока, тем выше уровень параллелизма. Например, при фиксированном уровне параллелизма, равном двум, два кодовых блока могут быть кодированы и/или декодированы параллельно. Теоретически величина уровня параллелизма равна числу кодовых блоков, подлежащих кодированию в изображении. Однако характеристики сжатия, получаемые таким способом, не являются оптимальными вследствие того факта, что такое кодирование не использует вероятности, соответствующие ближайшему окружению текущего кодового блока.

#### **Цель и краткое изложение существа изобретения**

Одной из целей настоящего изобретения является устранение недостатков, свойственных упомянутым выше известным способам.

Поэтому предмет настоящего изобретения относится к способу кодирования по меньшей мере одного изображения, содержащему этапы:

- разбиение изображения на несколько блоков, которые могут содержать символы, принадлежащие заданному множеству символов;
- группирование блоков в заданное число подмножеств блоков;
- кодирование, посредством модуля энтропийного кодирования, каждого из подмножества блоков путем ассоциирования цифровой информации с символами каждого блока из рассматриваемого подмножества, этап кодирования содержит, для первого блока изображения, подэтап инициализации переменных состояния для модуля энтропийного кодирования;
- генерация по меньшей мере одного подпотока данных, представляющего по меньшей мере одно из кодированных подмножеств блоков.

Способ согласно настоящему изобретению примечателен тем, что:

- в случае когда текущий блок является первым блоком, подлежащим кодированию, из состава рассматриваемого подмножества, в качестве вероятностей появления символов для первого текущего блока принимают вероятности, которые были определены для кодированного и декодированного заданного блока из состава по меньшей мере одного другого подмножества;
- если текущий блок представляет собой последний кодированный блок из состава рассматриваемого подмножества, тогда:
  - записывают в подпоток, представляющий рассматриваемое подмножество, всю полноту цифровой информации, ассоциированной с символами, во время кодирования



блоков рассматриваемого подмножества;

- выполняют подэтап инициализации.

Указанный выше этап записи содержит, как только будет закодирован последний блок какого-либо подмножества блоков, опустошение цифровой информации (битов), еще не переданной, как было объяснено в приведенном выше описании.

Связь между упомянутым выше этапом записи и этапом реинициализации модуля энтропийного кодирования делает возможным генерировать поток закодированных данных, содержащий различные подпотоки данных, соответствующие по меньшей мере одному закодированному подмножеству блоков, этот поток адаптирован для параллельного декодирования согласно различным уровням параллелизма, и это не зависит от типа кодирования - последовательного или параллельного, примененного к рассматриваемому подмножеству блоков. Таким образом, может быть получена большая степень свободы при кодировании с точки зрения уровня параллелизма в функции от ожидаемых характеристик кодирования/декодирования. Уровень параллелизма при декодировании является величиной переменной и может даже отличаться от уровня параллелизма при кодировании, поскольку в момент начала декодирования подмножества блоков декодер всегда находится в инициализированном состоянии.

Согласно первому примеру переменные состояния модуля энтропийного кодирования образуют границы внутреннего представления вероятности появления одного из символов из совокупности, составляющей заданное множество символов.

Согласно второму примеру переменные состояния модуля энтропийного кодирования представляют собой строки символов, содержащиеся в кодовой преобразовательной таблице энтропийного кодирующего устройства для кода LZW (Лемпеля-Зива-Уэлша), хорошо известного специалистам в рассматриваемой области и описанного по следующему интернет-адресу, 21 июня 2011 г.: <http://en.wikipedia.org/wiki/Lempel%E2%80%93Ziv%E2%80%93Welch>.

Основное преимущество от использования вероятностей появления символов, найденных для первого блока из указанного другого подмножества, в процессе энтропийного кодирования первого текущего блока из состава рассматриваемого подмножества блоков состоит в экономии буферной памяти кодирующего устройства путем сохранения в этой памяти исключительно обновлений вероятностей появления символов, не принимая в расчет вероятности появления символов, найденные другими последовательными блоками указанного другого подмножества.

Основное преимущество от использования вероятностей появления символов, найденных для какого-либо другого блока из указанного другого подмножества, отличного от первого блока, например второго блока, в процессе энтропийного кодирования первого текущего блока из состава рассматриваемого подмножества блоков состоит в получении более точных и потому лучше "изученных" вероятностей появления символов, что дает лучшие характеристики сжатия видеосигнала.

В конкретном варианте указанные подмножества блоков кодируют последовательно или еще и параллельно.

Тот факт, что кодирование подмножеств блоков осуществляется последовательно, имеет то преимущество, что способ кодирования согласно настоящему изобретению совместим со стандартом H.264/MPEG-4 AVC.

Тот факт, что кодирование подмножеств блоков осуществляется параллельно, имеет преимущество сокращения затрат времени на кодирование и использования выгод архитектуры, имеющей несколько платформ, для кодирования изображения.

В другом конкретном варианте, когда по меньшей мере два подмножества блоков кодируют параллельно по меньшей мере с одним другим подмножеством блоков, указанные по меньшей мере два закодированных множества блоков заключены в одном и том же подпотоке данных.

5 Такое предложение делает возможным, в частности, уменьшение объема сигнализации в подпотоках данных. Действительно, чтобы модуль декодирования мог произвести декодирование подпотока как можно раньше, необходимо указать в сжатом файле точку, в которой начинается интересующий подпоток. Когда в одном и том же подпотоке данных присутствуют несколько подмножеств блоков, необходим единый  
10 индикатор, что позволит уменьшить размер сжатого файла.

Еще в одном другом конкретном варианте, когда закодированные подмножества блоков предназначены для параллельного декодирования в заданном порядке, подпотоки данных, полученные после кодирования соответственно каждого подмножества блоков, сначала упорядочивают заданным образом, перед тем как  
15 передать, с учетом перспективы декодирования этих данных.

Такое предложение делает возможным адаптировать поток закодированных данных к конкретному виду декодирования без того, чтобы требовать декодировать и затем снова кодировать изображение.

Соответственно, настоящее изобретение относится далее к устройству для  
20 кодирования по меньшей мере одного изображения, содержащему:

- модуль для разбиения изображения на несколько блоков, которые могут содержать символы, принадлежащие заданному множеству символов;
- модуль для группирования блоков в заданное число подмножеств блоков;
- модуль для кодирования каждого из подмножеств блоков, так что этот модуль для  
25 кодирования содержит модуль энтропийного кодирования, способный ассоциировать цифровую информацию с символами каждого блока из рассматриваемого подмножества, этот модуль для кодирования содержит, для первого блока изображения, submodule для инициализации переменных состояния для модуля энтропийного кодирования;
- модуль для генерации по меньшей мере одного подпотока данных, представляющего  
30 по меньшей мере одно из закодированных подмножеств блоков.

Такое устройство для кодирования примечательно тем, что оно содержит:

- модуль для определения вероятностей появления символов для текущего блока, так что этот модуль в случае, когда текущий блок является первым блоком, подлежащим  
35 кодированию, из состава рассматриваемого подмножества, принимает в качестве вероятностей появления символов для первого текущего блока вероятности, которые были определены для закодированного и декодированного заданного блока из состава по меньшей мере одного другого подмножества;
- записывающий модуль, который в случае, когда текущий блок является последним закодированным блоком из состава рассматриваемого подмножества, активизируют для  
40 записи в подпоток, представляющий рассматриваемое подмножество, всей полноты цифровой информации, ассоциированной с символами, во время кодирования блоков рассматриваемого подмножества,

указанный submodule инициализации дополнительно активизируют, чтобы реинициализировать переменные состояния модуля энтропийного кодирования.

45 Соответствующим образом, настоящее изобретение относится к способу декодирования потока, представляющего по меньшей мере одно закодированное изображение, содержащему этапы:

- идентификация потока, содержащего заданное число подпотоков данных,

соответствующих по меньшей мере одному подмножеству блоков, подлежащих декодированию, так что эти блоки могут содержать символы, принадлежащие заданному множеству символов;

5 - декодирование идентифицированных подмножеств блоков, посредством модуля энтропийного декодирования, путем считывания, по меньшей мере в одном из идентифицированных подпотоков, цифровой информации, ассоциированной с символами каждого блока из подмножества, соответствующего указанному по меньшей мере одному идентифицированному подпотоку, этап декодирования содержит, для первого блока, подлежащего декодированию в изображении, подэтап инициализации переменных  
10 состояния для модуля энтропийного декодирования.

Такой способ декодирования примечателен тем, что:

- в случае когда текущий блок является первым блоком, подлежащим декодированию, из состава рассматриваемого подмножества, в качестве вероятностей появления символов для первого блока из состава рассматриваемого подмножества принимают  
15 вероятности, которые были определены для декодированного заданного блока из состава по меньшей мере одного другого подмножества;

- если текущий блок представляет собой последний декодированный блок из состава рассматриваемого подмножества, тогда выполняют подэтап инициализации.

В конкретном варианте указанные подмножества блоков декодируют  
20 последовательно или еще и параллельно.

В другом конкретном варианте, когда по меньшей мере два подмножества блоков декодируют параллельно по меньшей мере с одним другим подмножеством блоков, один из идентифицированных подпотоков данных представляет по меньшей мере два подмножества блоков.

25 Еще в одном другом конкретном варианте, когда кодированные подмножества блоков предназначены для параллельного декодирования в заданном порядке, подпотоки данных, соответствующие кодированным подмножествам блоков, сначала упорядочивают в указанном заданном порядке в составе потока, подлежащего декодированию.

30 Соответственно, настоящее изобретение относится также к устройству для декодирования потока, представляющего по меньшей мере одно кодированное изображение, содержащему:

- модуль для идентификации в потоке заданного числа подпотоков данных, соответствующих по меньшей мере одному подмножеству блоков, подлежащих  
35 декодированию, так что эти блоки могут содержать символы, принадлежащие заданному множеству символов;

- модуль декодирования идентифицированных подмножеств блоков, этот модуль декодирования содержит модуль энтропийного декодирования, способный считывать, по меньшей мере в одном из идентифицированных подпотоков, цифровую информацию,  
40 ассоциированную с символами каждого блока из подмножества, соответствующего указанному по меньшей мере одному идентифицированному подпотоку, этот модуль декодирования содержит, для первого блока, подлежащего декодированию в изображении, подмодуль для инициализации переменных состояния для модуля энтропийного декодирования.

45 Такое устройство для декодирования примечательно тем, что оно содержит модуль для определения вероятностей появления символов для текущего блока, так что этот модуль в случае, когда текущий блок является первым блоком, подлежащим кодированию, из состава рассматриваемого подмножества, принимает в качестве

вероятностей появления символов для первого текущего блока вероятности, которые были определены для декодированного заданного блока из состава по меньшей мере одного другого подмножества,

и тем, что если текущий блок представляет собой последний декодированный блок из состава рассматриваемого подмножества, тогда активизируется submodule инициализации для повторной инициализации переменных состояния для модуля энтропийного декодирования.

Настоящее изобретение направлено также на создание компьютерной программы, содержащей команды для реализации этапов описанного выше способа кодирования или декодирования, при выполнении этой программы компьютером.

Такая программа может использовать любой язык программирования и может быть в форме исходного кода, объектного кода или кода, занимающего промежуточное положение между исходным кодом и объектным кодом, например, в частично скомпилированной форме или в какой-либо другой нужной форме.

Еще один предмет настоящего изобретения также направлен на создание читаемого компьютером носителя записи, содержащего команды компьютерной программы, такой как упомянутая выше.

Такой носитель записи может представлять собой какой-либо объект или устройство, способное сохранять указанную программу. Например, подобный носитель может представлять собой запоминающее устройство, такое постоянное запоминающее устройство (ПЗУ (ROM)), например ПЗУ на основе компакт-диска (CD-ROM), или микроэлектронное ПЗУ (ROM) или также магнитный носитель записи, например дискета (гибкий диск) или накопитель на жестком магнитном диске.

Более того, такой носитель записи может быть передаваемым носителем, таким как электрический или оптический сигнал, который может быть передан по электрическому или по оптическому кабелю, по радио или каким-либо другим способом. Программу согласно настоящему изобретению можно, в частности, скачать из сети связи типа Интернет.

В качестве альтернативы такой носитель записи может представлять собой интегральную схему, в которую встроена указанная программа, эта схема может быть адаптирована для реализации рассматриваемого способа или может быть использована при его реализации.

Перечисленные выше устройство для кодирования, способ декодирования, устройство для декодирования и компьютерные программы обладают по меньшей мере такими же преимуществами, какие свойственны способу кодирования согласно настоящему изобретению.

#### **Краткое описание чертежей**

Другие характеристики и преимущества станут очевидными после прочтения описания двух предпочтительных вариантов, данного со ссылками на прилагаемые чертежи, на которых:

- фиг. 1 представляет схему кодирования изображения согласно известным способам;
- фиг. 2А представляет основные этапы способа кодирования согласно настоящему изобретению;
- фиг. 2В представляет подробно процесс кодирования, реализованный способом кодирования, показанным на фиг. 2А;
- фиг. 3А представляет первый вариант кодирующего устройства согласно настоящему изобретению;
- фиг. 3В представляет модуль кодирования в составе кодирующего устройства,

показанного на фиг. 3А;

- фиг. 3С представляет второй вариант кодирующего устройства согласно настоящему изобретению;

5 - фиг. 4А представляет схему кодирования/декодирования изображения согласно первому предпочтительному варианту;

- фиг. 4В представляет схему кодирования/декодирования изображения согласно второму предпочтительному варианту;

- фиг. 5А представляет основные этапы способа декодирования согласно настоящему изобретению;

10 - фиг. 5В представляет подробно процесс декодирования, реализованный способом декодирования, показанным на фиг. 5А;

- фиг. 6А представляет вариант декодирующего устройства согласно настоящему изобретению;

15 - фиг. 6В представляет модуль декодирования в составе декодирующего устройства, показанного на фиг. 6А;

- фиг. 7А представляет схему кодирования/декодирования изображения, реализующую кодирование последовательного типа и декодирование параллельного типа;

- фиг. 7В представляет схему кодирования/декодирования изображения, реализующую кодирование/декодирование параллельного типа при различных уровнях параллелизма.

#### 20 **Подробное описание первого варианта кодирующей части**

Теперь будет описан один из вариантов настоящего изобретения, в котором способ кодирования согласно настоящему изобретению используется для кодирования последовательности изображений в соответствии с двоичным потоком, таким же, как поток, получаемый в результате кодирования по стандарту H.264/MPEG-4 AVC. В  
25 рассматриваемом варианте способ кодирования согласно настоящему изобретению реализован, например, посредством программного обеспечения или аппаратным способом путем модификации кодирующего устройства, первоначально совместимого со стандартом H.264/MPEG-4 AVC. Способ кодирования согласно настоящему изобретению представлен в форме алгоритма, содержащего этапы с С1 по С5 и  
30 показанного на фиг. 2А.

Способ кодирования согласно одному из вариантов настоящего изобретения реализован в кодирующем устройстве СО, два варианта которого представлены соответственно на фиг. 3А и 3С.

Как показано на фиг. 2А, первый этап С1 кодирования представляет собой разбиение  
35 изображения ИЕ в последовательности, изображений, подлежащих кодированию, на несколько блоков или макроблоков МВ, как это изображено на фиг. 4А и 4В. Указанные макроблоки могут содержать один или несколько символов, составляющих часть заданного множества символов. В представленных примерах эти макроблоки МВ обладают квадратной формой и все имеют одинаковый размер. В зависимости от  
40 размера изображения, который необязательно кратен размеру блоков, последние блоки слева и последние блоки внизу изображения могут не быть квадратными. В альтернативном варианте блоки, например, могут быть прямоугольной формы и/или могут быть не совмещены один с другим.

Каждый блок или макроблок может, далее, сам быть разбит на субблоки, которые  
45 в свою очередь могут быть разбиты на части.

Такое разбиение осуществляется посредством модуля РСО разбиения, показанного на фиг. 3А и использующего, например, хорошо известный алгоритм разбиения.

Как показано на фиг. 2А, второй этап С2 кодирования представляет собой

группирование указанных выше блоков в заданное число  $P$  подмножеств  $SE_1, SE_2, \dots, SE_k, \dots, SE_P$  последовательных блоков, которые должны быть кодированы последовательно или параллельно. В примерах, показанных на фиг. 4А и 4В, число подмножеств  $P=6$ , но для большей ясности чертежа изображены только четыре подмножества  $SE_1, SE_2, SE_3$  и  $SE_4$ . Каждое из этих четырех подмножеств блоков показано в заштрихованном виде, а все четыре подмножества образованы первыми четырьмя строками (каждое - своей) блоков изображения ИЕ.

Такое группирование осуществляется посредством вычислительного модуля GRCO разбиения, показанного на фиг. 3А, с использованием хорошо известного алгоритма.

Как показано на фиг. 2А, третий этап СЗ кодирования состоит в кодировании каждого из подмножеств блоков с  $SE_1$  по  $SE_6$ , так что блоки из состава любого рассматриваемого подмножества кодируют в соответствии с заданным порядком PS прохождения через эти блоки, например для последовательного кодирования. В примерах, представленных на фиг. 4А и 4В, блоки текущего подмножества  $SE_k$  ( $1 \leq k \leq P$ ) кодируют один за другим слева направо, как показано стрелкой PS.

Согласно первому варианту такое кодирование относится к последовательному типу кодирования и реализуется в одном кодирующем модуле UC, как представлено на фиг. 3А. Согласно известным технологиям кодирующее устройство СО содержит буферную память МТ, адаптированную для хранения вероятностей появления символов, постоянно обновляемых в процессе кодирования текущего блока.

Как показано более подробно на фиг. 3В, кодирующий модуль UC содержит:

- модуль для кодирования с прогнозированием применительно к текущему блоку с учетом по меньшей мере одного из ранее кодированных и декодированных блоков, обозначенный МСР;
- модуль для энтропийного кодирования текущего блока с использованием по меньшей мере одной вероятности появления символа, вычисленной для указанного ранее кодированного и декодированного блока, обозначенный МСЕ.

Модуль МСР кодирования с прогнозированием представляет собой программный модуль, способный осуществлять кодирование с прогнозированием применительно к текущему блоку в соответствии с обычными способами прогнозирования, такими как, например, режим внутрикадрового и/или межкадрового прогнозирования.

Указанный модуль МСЕ энтропийного кодирования представляет собой модуль стандарта САВАС, но модифицированный согласно настоящему изобретению, как это будет рассмотрено далее в этом описании.

В качестве варианта указанное энтропийное кодирующее устройство МСЕ может представлять собой известное кодирующее устройство в коде Хаффмена.

В примерах, представленных на фиг. 4А и 4В, модуль UC кодирует блоки первой строки  $SE_1$  слева направо. После достижения последнего блока в первой строке  $SE_1$  модуль переходит к первому блоку второй строки  $SE_2$ . После достижения последнего блока во второй строке  $SE_2$  модуль переходит к первому блоку третьей строки  $SE_3$ . После достижения последнего блока в третьей строке  $SE_3$  модуль переходит к первому блоку четвертой строки  $SE_4$  и т.д. до тех пор, пока не будет закодирован последний блок изображения ИЕ.

Безусловно, возможны и другие варианты просмотра изображения, отличные от того варианта, который был только что описан выше. Таким образом, можно разбить изображение ИЕ на несколько поды изображений и затем применить разбиение такого типа к каждому поды изображению независимо. Кроме того, кодирующий модуль может обрабатывать не последовательность строк, как было рассмотрено выше, а

последовательность столбцов. Можно также просматривать строки или столбцы в любом направлении.

Согласно второму варианту такое кодирование представляет собой кодирование параллельного типа и отличается от первого варианта последовательного кодирования исключительно тем фактом, что оно реализовано заданным числом  $R$  кодирующих модулей  $UC_k$  ( $1 \leq k \leq R$ ), при  $R=2$ , как в примере, представленном на фиг. 3С. Такое параллельное кодирование, как известно, вызывает значительное ускорение способа кодирования.

Каждый из кодирующих модулей  $UC_k$  идентичен кодирующему модулю  $UC$ , представленному на фиг. 3В. Соответствующим образом кодирующий модуль  $UC_k$  содержит модуль  $MCP_k$  кодирования с прогнозированием и модуль энтропийного кодирования  $MCE_k$ .

Снова со ссылками на фиг. 4А и 4В первый модуль  $UC_1$  кодирует, например, блоки строк с нечетными номерами, тогда как второй модуль  $UC_2$  кодирует, например, блоки строк с четными номерами. Более точно, первый модуль  $UC$  кодирует блоки первой строки  $SE_1$  слева направо. После достижения последнего блока в первой проке  $SE_1$  этот модуль переходит к первому блоку  $(2n+1)$ -й строки, иными словами, к третьей строке  $SE_3$  и т.д. Параллельно с обработкой, выполняемой первым модулем  $UC_1$ , второй модуль  $UC_2$  кодирует блоки второй строки  $SE_2$  слева направо. После достижения последнего блока во второй строке  $SE_2$  этот модуль переходит к первому блоку  $(2n)$ -й строки, здесь это четвертая строка  $SE_4$ , и т.д. Указанные выше два просматривающих прохода повторяют до тех пор, пока не будет закодирован последний блок изображения  $IE$ .

Как показано на фиг. 2А, на четвертом этапе  $C_4$  кодирования генерируют  $L$  подпоток  $F_1, F_2, \dots, F_m, \dots, F_L$  ( $1 \leq m \leq L \leq P$ ) битов, представляющих обработанные блоки, сжатые посредством указанного выше кодирующего модуля  $UC$  или каждого из указанных выше кодирующих модулей  $UC_k$ , равно как и декодированную версию обработанных блоков из состава каждого подмножества  $SE_k$ . Декодированные обработанные блоки из состава рассматриваемого подмножества, обозначенные  $SED_1, SED_2, \dots, SED_k, \dots, SED_P$ , могут быть повторно использованы кодирующим модулем  $UC$ , представленным на фиг. 3А, или каждым из кодирующих модулей  $UC_k$ , представленных на фиг. 3С, в соответствии с механизмом синхронизации, который будет подробно рассмотрен в настоящем описании дальше.

Как показано на фиг. 3В, этап генерации  $L$  подпотоков реализуется программным модулем  $MGSF$  или  $MGSF_k$  генерации потоков, адаптированным для генерации потоков данных, таких как потоки битов, например.

Как показано на фиг. 2А, пятый этап  $C_5$  кодирования состоит в построении глобального потока  $F$  на основе упомянутых выше  $L$  подпотоков  $F_1, F_2, \dots, F_m, \dots, F_L$ . Согласно одному из вариантов подпотоки  $F_1, F_2, \dots, F_m, \dots, F_L$  просто накладываются один на другой, так что необходим дополнительный сегмент информации, предназначенный для индикации местонахождения каждого подпотока  $F_m$  в глобальном потоке  $F$  для декодера. Этот глобальный поток затем передают по сети связи (не показана) удаленному терминалу. Этот терминал содержит декодер  $DO$ , представленный на фиг. 5А. Согласно другому варианту, который особенно предпочтителен, поскольку он не требует декодирования и потом повторного кодирования изображения, кодирующее устройство  $CO$ , перед передачей потока  $F$  декодеру  $DO$ , предварительно располагает  $L$  подпотоки  $F_1, F_2, \dots, F_m, \dots, F_L$  в заданном порядке, соответствующем порядку, в котором указанный декодер  $DO$  способен декодировать эти подпотоки.

Таким образом, как это будет подробно рассмотрено далее в настоящем описании, декодер согласно настоящему изобретению способен выделить подпотоки F1, F2, ..., Fm, ..., FL из состава глобального потока F и назначить эти подпотоки одному или нескольким декодирующим модулям, из которых состоит декодер. Следует отметить, что такое разложение глобального потока на подпотоки не зависит от выбора использования одного кодирующего модуля или нескольких кодирующих модулей работающих параллельно, и что при таком подходе можно иметь только кодирующее устройство или только декодер, содержащий модули, работающие параллельно.

Такая структура глобального потока F реализована в модуле CF построения потоков, таком как представлен на фиг. 3A и на фиг. 3C.

Различные специфичные подэтапы настоящего изобретения, такие как реализуемые в ходе указанного выше этапа C3 кодирования в кодирующем модуле UC или UCk, будут теперь описаны со ссылками на фиг. 2B.

В процессе выполнения этапа C31 кодирующий модуль UC или UCk выбирает в качестве текущего блока первый блок, подлежащий кодированию, в текущей строке SEk, представленной на фиг. 4A или 4B, такой как, например, первая строка SE1.

В процессе выполнения этапа C32 модуль UC или UCk проверяет, является ли текущий блок первым блоком (расположенным сверху и слева) изображения IE, которое было разбито на блоки в ходе указанного выше этапа C1.

Если текущий блок - первый, тогда на этапе C33 модуль MCE или MCEk энтропийного кодирования осуществляет инициализацию своих переменных состояния. Согласно представленному примеру, используемому арифметическое кодирование, описанное ранее, этот этап содержит инициализацию интервала, представляющего вероятность появления одного из символов из совокупности, составляющей заданное множество символов. Известным способом этот интервал инициализируют с двумя границами L и H, соответственно нижней и верхней. Значение нижней границы L фиксируют на 0, а значение верхней границы фиксируют на 1, что соответствует вероятности появления первого символа из совокупности всех символов, составляющих заданное множество символов. Размер R этого интервала определен, поэтому, здесь формулой  $R = H - L = 1$ . Инициализированный интервал обычно разбивают дополнительно на несколько заданных подинтервалов, представляющих соответственно вероятности появления символов из совокупности символов, составляющей заданное множество символов.

В качестве варианта, если используемый способ энтропийного кодирования представляет собой способ кодирования LZW, инициализируется кодовая таблица строк символов, так что эта таблица содержит все возможные символы по одному и только по одному разу.

Если после указанного выше этапа C32 оказалось, что текущий блок не является первым блоком изображения IE, определяют на этапе C40, который будет рассмотрен в настоящем описании позднее, наличие необходимых ранее кодированных и декодированных блоков.

В процессе выполнения этапа C34 кодируют первый текущий блок MB1 первой строки SE1, представленной на фиг. 4A или 4B. Такой этап C34 содержит несколько подэтапов с C341 по C348, которые будут описаны ниже.

В ходе выполнения первого подэтапа C341, представленного на фиг. 2B, осуществляют кодирование с прогнозированием применительно к текущему блоку MB1 посредством известного способа внутрикадрового и/или межкадрового прогнозирования, согласно которому блок MB1 прогнозируют относительно по меньшей мере одного ранее кодированного и декодированного блока.



Само собой разумеется, что возможны также другие режимы внутрикадрового прогнозирования, такие как режимы, предлагаемые в стандарте H.264.

Текущий блок  $MB_1$  может быть также подвергнут кодированию с прогнозированием в режиме, в ходе которого блок  $MB_1$  прогнозируют относительно блока, полученного на основе ранее кодированного и декодированного изображения. Безусловно, можно представить и другие типы прогнозирования. Из совокупности возможных способов прогнозирования для текущего блока оптимальный способ прогнозирования выбирают согласно критерию искажения скорости передачи данных, хорошо известному специалистам в рассматриваемой области.

Упомянутый выше этап кодирования с прогнозированием делает возможным построение прогнозированного блока  $MVp_1$ , представляющего собой аппроксимацию текущего блока  $MB_1$ . Информация, относящаяся к кодированию с прогнозированием, будет в последующем записана в потоке  $F$ , передаваемом декодеру  $DO$ . Такая информация содержит в частности указание типа прогнозирования (межкадровое или внутрикадровое) и, если это подходит, режима внутрикадрового прогнозирования, типа разбиения блока или макроблока, если последний был разбит, индекса опорного изображения и вектора смещения, используемого в режиме межкадрового прогнозирования. Кодирующее устройство  $CO$  сжимает эту информацию.

В процессе выполнения следующего подэтапа  $S342$  вычитают прогнозируемый блок  $MVp_1$  из текущего блока  $MB_1$  для получения остаточного блока  $MVr_1$ .

В процессе выполнения следующего подэтапа  $S343$  осуществляют преобразование прогнозируемого блока  $MVp_1$  согласно обычной процедуре прямого преобразования, такой как, например, дискретное косинусное преобразование типа  $DCT$ , для получения преобразованного блока  $MVt_1$ .

В процессе выполнения следующего подэтапа  $S344$  выполняют квантование преобразованного блока  $MVt_1$  согласно обычной процедуре квантования, такой как, например, скалярное квантование. В результате получен блок  $MVq_1$  квантованных коэффициентов.

В процессе выполнения следующего подэтапа  $S345$  осуществляют энтропийное кодирование блока квантованных коэффициентов  $MVq_1$ . В предпочтительном варианте здесь выполняется энтропийное кодирование по стандарту  $CAVAC$ . Такой этап содержит:

- а) считывание символа или символов из состава заданного множества символов, ассоциированных с текущим блоком;
- б) ассоциирование цифровой информации, такой как биты, с прочитанным(и) символом(ами).

В указанном выше варианте, в котором в качестве алгоритма кодирования использован алгоритм кодирования  $LZW$ , с символом, подлежащим кодированию, ассоциируют цифровой сегмент информации, соответствующий коду рассматриваемого символа в текущей кодовой таблице, и затем обновляют эту кодовую таблицу согласно известной процедуре.

В процессе выполнения следующего подэтапа  $S346$  выполняют деквантование блока  $MVq_1$  согласно обычной процедуре деквантования, которая представляет собой операцию, обратную процедуре квантования, выполненной на этапе  $S344$ . В результате получен блок  $MVdq_1$  деквантованных коэффициентов.

В процессе выполнения следующего подэтапа  $S347$  выполняют обратное

преобразование блока  $MBDq_1$  деквантованных коэффициентов, представляющее собой операцию, обратную процедуре прямого преобразования, выполненной на этапе С343 выше. В результате получен декодированный остаточный блок  $MBDr_1$ .

5 В процессе выполнения следующего подэтапа С348 формируют декодированный блок  $MBD_1$  путем суммирования прогнозируемого блока  $MBr_1$  с декодированным остаточным блоком  $MBDr_1$ . Следует отметить, что последний блок является таким же, как декодированный блок, полученный после завершения выполнения способа декодирования изображения ИЕ, который был рассмотрен выше в настоящем описании.  
10 Декодированный блок  $MBD_1$  делают, таким образом, доступным для использования кодирующим модулем  $UCk$  или каким-либо другим кодирующим модулем, составляющим часть заданного числа  $R$  кодирующих модулей.

По завершении упомянутого выше этапа С34 кодирования модуль МСЕ или МСЕК энтропийного кодирования, такой как показан на фиг. 3В, содержит все вероятности, такие, как вероятности, постоянно обновляемые совместно в процессе кодирования  
15 первого блока. Эти вероятности соответствуют разнообразным элементам возможных синтаксисов и различным ассоциированным контекстам кодирования.

После выполнения указанного выше этапа С34 кодирования проверяют в ходе этапа С35, является ли текущий блок  $j$ -м блоком той же самой строки, где  $j$  - представляет собой заданную величину, известную кодирующему устройству СО и равную по меньшей  
20 мере 1.

Если это так, то в ходе выполнения этапа С36, представленного на фиг. 2В, множество вероятностей, вычисленных для  $j$ -блока, сохраняют в буферной памяти МТ в составе кодирующего устройства СО, такого как показано на фиг. 3А или 3В и на фиг. 4А или  
25 4В, и размер указанной памяти адаптирован для сохранения вычисленного числа вероятностей.

В процессе выполнения этапа С37, представленного на фиг. 2В, кодирующий модуль УС или  $UCk$  проверяет, является ли текущий блок строки  $SEk$ , который был только что закодирован, последним блоком изображения ИЕ. Такой этап выполняется также, если  
30 в процессе осуществления этапа С35 оказалось, что текущий блок не является  $j$ -м блоком строки  $SE1$ .

Если оказалось, что текущий блок представляет собой последний блок изображения ИЕ, на этапе С38 выполнение способа кодирования завершается.

Если текущий блок - не последний, на этапе С39 выбирают следующий блок  $MV_j$ ,  
35 подлежащий кодированию, в соответствии с порядком просмотра изображения, обозначенным стрелкой PS на фиг. 4А или 4В.

В процессе выполнения этапа С40, представленного на фиг. 2В, определяют доступность предварительно кодированных и декодированных блоков, необходимых для кодирования текущего блока  $MV_j$ .  
40

Если происходит обработка первой строки  $SE1$ , этот этап состоит в проверке доступности по меньшей мере одного блока, расположенного слева от текущего блока  $MV_j$ , подлежащего кодированию. Однако с учетом порядка PS просмотра изображения, выбранного в варианте, представленном на фиг. 4А и 4В, блоки в рассматриваемой  
45 строке  $SEk$  кодируют один за другим. Следовательно, всегда (за исключением обработки первого блока строки) имеется и доступен расположенный слева кодированный и декодированный блок. В примере, представленном на фиг. 4А или 4В, это блок, расположенный непосредственно слева рядом с текущим блоком, подлежащим

кодированию.

Если рассматривается строка SE<sub>k</sub>, отличная от первой строки, указанный этап проверки дополнительно содержит проверку, имеется ли заданное число N' блоков, расположенных в предыдущей строке SE<sub>k-1</sub>, например два блока, находящихся  
5 соответственно сверху и сверху справа от текущего блока, и доступных для кодирования текущего блока, иными словами, если они уже были кодированы и затем декодированы посредством кодирующего модуля UC или UC<sub>k-1</sub>.

Поскольку этот этап проверки замедляет работу способа кодирования, в альтернативном варианте настоящего изобретения, в котором кодирование строк  
10 осуществляется по параллельному типу, тактовый сигнал CLK, показанный на фиг. 3С, приспособлен для опережающей синхронизации кодирования блоков таким образом, чтобы гарантировать доступность двух блоков, расположенных соответственно сверху и сверху справа от текущего блока, не требуя обязательной проверки доступности этих двух блоков. Таким образом, кодирующий модуль UC<sub>k</sub> всегда начинает кодировать  
15 первый блок со сдвигом на заданное число N' (при, например, N' = 2) кодированных и декодированных блоков предыдущей строки SE<sub>k-1</sub>, используемых для кодирования текущего блока. С программной точки зрения реализация такого тактового сигнала делает возможным значительно ускорить и сократить затраты времени на обработку  
20 блоков изображения IE в кодирующем устройстве CO.

В процессе выполнения этапа С41, представленного на фиг. 2В, проверяют, является ли текущий блок первым блоком рассматриваемой строки SE<sub>k</sub>.

Если текущий блок - первый, в процессе выполнения этапа С42 считывают из буферной памяти МТ исключительно вероятности появления символов, вычисленные  
25 во время кодирования j-го блока предыдущей строки SE<sub>k-1</sub>.

Согласно первому варианту, представленному на фиг. 4А, j-й блок является первым блоком предыдущей строки SE<sub>k-1</sub> (j = 1). Такое считывание состоит в замене вероятностей в кодирующем устройстве по стандарту САВАС вероятностями,  
30 записанными в буферной памяти МТ. Процесс чтения, захватывающий соответствующие первые блоки второй, третьей и четвертой строк SE<sub>2</sub>, SE<sub>3</sub> и SE<sub>4</sub>, показан на фиг. 4А тонкими стрелками.

Согласно второму варианту, упомянутого выше этапа С43, представленного на фиг. 4В, j-й блок является вторым блоком предыдущей строки SE<sub>k-1</sub> (j = 2). Такое считывание  
35 состоит в замене вероятностей в кодирующем устройстве по стандарту САВАС вероятностями, записанными в буферной памяти МТ. Процесс чтения, захватывающий соответствующие первые блоки второй, третьей и четвертой строк SE<sub>2</sub>, SE<sub>3</sub> и SE<sub>4</sub>, показан на фиг. 4В тонкими штриховыми стрелками.

После этапа С42 текущий блок кодируют и затем декодируют путем повторения этапов с С34 по С38, описанных выше.

Если после указанного выше этапа С41 оказалось, что текущий блок не является  
40 первым блоком рассматриваемой строки SE<sub>k</sub>, предпочтительно не считывать вероятности для ранее кодированного и декодированного блока, расположенного в той же самой строке SE<sub>k</sub>, иными словами кодированного и декодированного блока, расположенного (в представленном примере) непосредственно рядом слева от текущего блока. Действительно, при последовательном порядке PS просмотра блоков  
45 изображения для считывания блоков, расположенных в той же самой строке, как показано на фиг. 4А или 4В, вероятности появления символов, присутствующие в кодирующем устройстве, использующем алгоритм САВАС, в момент начала кодирования текущего блока являются точно теми, какие присутствуют после

декодирования/декодирования предыдущего блока в той же самой строке.

Затем, в процессе выполнения этапа С43, представленного на фиг. 2В, производят обучение вероятностям появления символов для энтропийного кодирования текущего блока, так что эти вероятности соответствуют исключительно тем, которые были  
5 вычислены для указанного предыдущего блока в той же самой строке, как показано двойными сплошными стрелками на фиг. 4А или 4В.

После этапа С43 текущий блок кодируют и затем декодируют путем повторения этапов с С34 по С38, описанных выше.

После этого в процессе выполнения этапа С44 проверяют, является ли текущий блок  
10 последним блоком рассматриваемой строки SEk.

Если текущий блок - не последний в этой строке, после этапа С44 снова выполняют этап С39 выбора следующего блока MB<sub>j</sub>, подлежащего кодированию.

Если текущий блок является последним блоком в рассматриваемой строке SEk, в процессе выполнения этапа С45 кодирующее устройство СО, представленное на фиг. 3А или 3С, осуществляет опустошение, как рассмотрено выше в настоящем описании.  
15 С этой целью кодирующий модуль UСk передает соответствующему модулю MGSFk генерации подпотока полностью все биты, которые были ассоциированы с символом (ами), считываемым(и) в процессе кодирования каждого блока рассматриваемой строки SEk, таким образом, что этот модуль MGSFk записывает в подпоток F<sub>m</sub> данных,  
20 содержащий двоичную последовательность, представляющую закодированные блоки рассматриваемой строки SEk, все эти биты полностью. Такое опустошение символически обозначено на фиг. 4А и 4В треугольником в конце каждой строки SEk.

В процессе выполнения этапа С46, представленного на фиг. 2В, кодирующий модуль UС или UСk выполняет процедуру, идентичную указанному выше этапу С33, иными  
25 словами, снова инициализирует интервальное представление вероятности появления символа, входящего в состав заданного множества символов. Такая повторная инициализация обозначена на фиг. 4А и 4В черной точкой в начале каждой строки SEk.

Преимущество выполнения этапов С45 и С46 на этом уровне кодирования состоит в том, что в процессе кодирования следующего блока, обрабатываемого кодирующим  
30 модулем UС или кодирующим модулем UСk, кодирующее устройство СО находится в инициализированном состоянии. Таким образом, как это будет рассмотрено далее в настоящем описании, для декодирующего модуля становится возможным работать параллельно, чтобы непосредственно декодировать сжатый поток F от этой точки, поскольку модулю достаточно для этого быть в инициализированном состоянии.

### 35 **Подробное описание одного из вариантов декодирующей части**

Теперь будет описан один из вариантов способа декодирования согласно настоящему изобретению, в котором этот способ декодирования реализован посредством  
программного обеспечения или аппаратным способом путем модификации декодера, первоначально совместимого со стандартом H.264/MPEG-4 AVC.

40 Способ декодирования согласно настоящему изобретению представлен в форме алгоритма, содержащего этапы с D1 по D4 и показанного на фиг. 5А.

Способ декодирования согласно рассматриваемому варианту настоящего изобретению реализован в декодирующем устройстве DО, представленном па фиг. 6А.

Как показано на фиг. 5А, первый этап D1 декодирования представляет собой  
45 идентификацию в потоке F совокупности из L подпотоков F1, F2, ..., F<sub>m</sub>, ..., FL, содержащих соответственно Р подмножеств SE1, SE2, ..., SEk, ..., SEP предварительно закодированных блоков или макроблоков MB, как это изображено на фиг. 4А и 4В. С этой целью каждый подпоток F<sub>m</sub> в составе потока F ассоциируют с индикатором,

позволяющим декодеру DO определить местонахождение каждого подпотока  $F_m$  в потоке  $F$ . В качестве варианта, по завершении указанного выше этапа С3 кодирования, кодирующее устройство СО упорядочивает подпотоки  $F_1, F_2, \dots, F_m, \dots, F_L$  в потоке  $F$  в соответствии с порядком, ожидаемым декодером DO, что позволяет избежать вставки индикаторов подпотоков в поток  $F$ . Такое предложение делает возможным уменьшение затрат с точки зрения скорости передачи данных в потоке  $F$  данных.

В примере, представленном на фиг. 4А или 4В, указанные блоки MB обладают квадратной формой, и все имеют одинаковый размер. В зависимости от размера изображения, который необязательно кратен размеру блоков, последние блоки слева и последние блоки внизу изображения могут не быть квадратными. В альтернативном варианте блоки, например, могут быть прямоугольной формы и/или могут быть не совмещены один с другим.

Каждый блок или макроблок может далее сам быть разбит на субблоки, которые сами могут быть разбиты на части.

Такая идентификация осуществляется посредством модуля EXDO выделения потоков, такого, как представлен на фиг. 6А.

В примере, показанном на фиг. 4А или 4В, заданное число  $P$  равно 6, но для большей ясности чертежа изображены только четыре подмножества  $SE_1, SE_2, SE_3$  и  $SE_4$ .

Как показано на фиг. 5А, второй этап D2 декодирования состоит в декодировании каждого из подмножеств блоков  $SE_1, SE_2, SE_3$  и  $SE_4$ , так что блоки из состава любого рассматриваемого подмножества декодируют в соответствии с заданным последовательным порядком PS прохождения через эти блоки. В примере, представленном на фиг. 4А или 4В, блоки текущего подмножества  $SE_k$  ( $1 \leq k \leq P$ ) декодируют один за другим слева направо, как показано стрелкой PS. По завершении этапа D2 получают декодированные подмножества блоков  $SED_1, SED_2, SED_3, \dots, SED_k, \dots, SEDP$ .

Такое декодирование может быть последовательного типа и, следовательно, может быть произведено посредством единственного декодирующего модуля.

Однако для создания возможности использовать преимущества архитектуры декодирования, содержащей несколько платформ, декодирование подмножеств блоков представляет собой декодирование параллельного типа и реализуется посредством нескольких  $R$  декодирующих модулей  $UD_k$  ( $1 \leq k \leq R$ ), например, при  $R=4$ , как это представлено на фиг. 6А. Это предложение, таким образом, позволяет значительно ускорить выполнение способа декодирования. Согласно известным технологиям декодирующее устройство DO содержит буферную память MT, адаптированную для хранения вероятностей появления символов, постоянно обновляемых совместно в процессе декодирования текущего блока.

Как показано более подробно на фиг. 6В, каждый из декодирующих модулей  $UD_k$  содержит:

- модуль для энтропийного декодирования текущего блока посредством обучения по меньшей мере одной вероятности появления символа, вычисленной по меньшей мере для одного ранее декодированного блока, обозначенный MDEk;
- модуль для декодирования с прогнозированием применительно к текущему блоку с учетом указанного ранее декодированного блока, обозначенный MDPk.

Модуль SUDPk кодирования с прогнозированием способен осуществлять декодирование с прогнозированием применительно к текущему блоку в соответствии с обычными способами прогнозирования, такого как, например, режим внутрикадрового и/или межкадрового прогнозирования.

Указанный модуль MDEк энтропийного декодирования представляет собой модуль стандарта САВАС, но модифицированный согласно настоящему изобретению, как это будет рассмотрено далее в этом описании.

В качестве варианта, указанный модуль MDEк энтропийного декодирования может представлять собой известное декодирующее устройство для кода Хаффмена.

В примере, представленном на фиг. 4А или 4В, первый модуль UD1 декодирует блоки первой строки SE1 слева направо. После достижения последнего блока в первой строке SE1 этот модуль переходит к первому блоку (n+1)-й строки, здесь это 5-я строка, и т.д. Второй модуль UD2 декодирует блоки второй строки SE2 слева направо. После достижения последнего блока во второй строке SE2 этот модуль переходит к первому блоку (n+2)-й строки, здесь это 6-я строка, и т.д. Такой просмотр повторяется вплоть до модуля UD4, который декодирует блоки четвертой строки SE2 слева направо. После достижения последнего блока в первой строке модуль переходит к первому блоку (n+4)-й строки, здесь это 8-я строка, и т.д. и т.п., пока не будет декодирован последний блок последнего идентифицированного потока.

Безусловно, возможны и другие варианты просмотра изображения, отличные от того варианта, который был только что описан выше. Например, каждый декодирующий модуль может обрабатывать не вложенные строки, как было рассмотрено выше, а вложенные столбцы. Можно также просматривать строки или столбцы в любом направлении.

Как показано на фиг. 5А, третий этап D3 кодирования состоит в реконструкции декодированного изображения ID на основе каждого декодированного подмножества SED1, SED2, ..., SEDk, ..., SEDP, полученного на этапе D2 декодирования. Более точно, декодированные блоки каждого декодированного подмножества SED1, SED2, ..., SEDk, ..., SEDP передают в модуль URI реконструкции изображения, как представлено на фиг. 6А. В процессе выполнения этапа D3 модуль URI записывает декодированные блоки в состав декодированного изображения по мере того, как эти блоки становятся доступными.

В процессе выполнения четвертого этапа D4 декодирования, представленного на фиг. 5А, модуль URI показанный на фиг. 6А, передает на выход полностью декодированное изображение ID.

Различные специфичные подэтапы настоящего изобретения, такие как реализуемые в ходе указанного выше этапа D2 параллельного декодирования в декодирующем модуле UDk, будут теперь описаны со ссылками на фиг. 5В.

В процессе выполнения этапа D21 декодирующий модуль UDk выбирает в качестве текущего блока первый блок, подлежащий декодированию, в текущей строке SEk, представленной на фиг. 4А или 4В.

В процессе выполнения этапа D22 декодирующий модуль UDk проверяет, является ли текущий блок первым блоком декодированного изображения, в этом примере первым блоком подпотока F1.

Если это так, в процессе выполнения этапа D23 модуль MDE или MDEк энтропийного декодирования осуществляет инициализацию своих переменных состояния. Согласно представленному примеру этот этап содержит инициализацию интервала, представляющего вероятность появления одного из символов из совокупности, составляющей заданное множество символов.

В качестве варианта, если используемый способ энтропийного декодирования представляет собой способ декодирования LZW, инициализируется кодовая таблица строк символов, так что эта таблица содержит все возможные символы по одному и

только по одному разу. Этап D23 идентичен указанному выше этапу C33 кодирования и потому далее описан не будет.

Если после указанного выше этапа D22 оказалось, что текущий блок не является первым блоком декодированного изображения ID, определяют на этапе D30, который  
5 будет рассмотрен в настоящем описании позднее, наличие и доступность необходимых ранее декодированных блоков.

В процессе выполнения этапа D24 декодируют первый текущий блок MB1 первой строки SE1, представленной на фиг. 4A или 4B. Такой этап D24 содержит несколько подэтапов с D241 по D246, которые будут описаны ниже.

10 В процессе выполнения первого подэтапа D241 осуществляют энтропийное декодирование синтаксических элементов, относящихся к текущему блоку. Такой этап содержит главным образом:

- а) считывание битов из подпотока, ассоциированного с первой строкой SE1;
- б) реконструкцию символов на основе прочитанных битов.

15 В указанном выше варианте, в котором в качестве алгоритма декодирования использован алгоритм декодирования LZW, считывают цифровой сегмент информации, соответствующий коду символа, подлежащего кодированию, из текущей кодовой таблицы, реконструируют символ на основе прочитанного кода и затем обновляют эту кодовую таблицу согласно известной процедуре.

20 Более точно, синтаксические элементы, связанные с текущим блоком, декодируют посредством использующего алгоритм CABAC модуля MDE1 энтропийного декодирования, такого, как представлено на фиг. 6B. Последний декодирует подпоток F1 битов из состава сжатого файла для получения синтаксических элементов и в то же самое время заново обновляет вероятности таким способом, чтобы в момент, когда  
25 модуль декодирует какой-либо символ, вероятности появления этого символа были идентичны вероятностям, получаемым в процессе кодирования этого же самого символа на указанном выше этапе C345 энтропийного кодирования.

В ходе выполнения подэтапа D242 осуществляют декодирование с прогнозированием применительно к текущему блоку MB1 посредством известного способа  
30 внутрикадрового и/или межкадрового прогнозирования, согласно которому блок MB1 прогнозируют относительно по меньшей мере одного ранее декодированного блока.

Само собой разумеется, что возможны также другие режимы внутрикадрового прогнозирования, такие как режимы, предлагаемые в стандарте H.264.

В процессе выполнения этого этапа декодирование с прогнозированием  
35 осуществляется с помощью синтаксических элементов, декодированных на предыдущем этапе и содержащих, в частности, указание типа прогнозирования (межкадровое или внутрикадровое) и, если это подходит, режима внутрикадрового прогнозирования, типа разбиения блока или макроблока, если последний был разбит, индекса опорного изображения и вектора смещения, используемого в режиме межкадрового  
40 прогнозирования.

Упомянутый выше этап декодирования с прогнозированием делает возможным построение прогнозированного блока MBp1.

В процессе выполнения следующего подэтапа D243 формируют квантованный остаточный блок MBq1 с помощью ранее декодированных синтаксических элементов.  
45

В процессе выполнения следующего подэтапа D244 осуществляют деквантование квантованного остаточного блока MBq1 согласно обычной процедуре деквантования, представляющей собой операцию, обратную процедуре квантования, выполненной на

этапе С344, для получения декодированного деквантованного блока  $MBDt_1$ .

В процессе выполнения следующего подэтапа D245 осуществляют обратное преобразование деквантованного блока  $MBDt_1$ , представляющее собой операцию, обратную процедуре прямого преобразования, выполненной на этапе С343 выше. В результате получают декодированный остаточный блок  $MBDr_1$ .

В процессе выполнения следующего подэтапа D246 формируют декодированный блок  $MBD_1$  путем суммирования прогнозируемого блока  $MBr_1$  с декодированным остаточным блоком  $MBDr_1$ . Декодированный блок  $MBD_1$  делают, таким образом, доступным для использования декодирующим модулем UD1 или каким-либо другим декодирующим модулем, составляющим часть заданного числа N декодирующих модулей.

По завершении упомянутого выше этапа D246 декодирования модуль MDE1 энтропийного декодирования, такой как показан на фиг. 6B, содержит все вероятности, такие как постоянно обновляемые совместно в процессе декодирования первого блока. Эти вероятности соответствуют разнообразным элементам возможных синтаксисов и различным ассоциированным контекстам декодирования.

После выполнения указанного выше этапа D24 декодирования проверяют в ходе этапа D25, является ли текущий блок j-м блоком той же самой строки, где j представляет собой заданную величину, известную декодеру DO и равную по меньшей мере 1.

Если это так, то в ходе выполнения этапа D26, множество вероятностей, вычисленных для j-блока, сохраняют в буферной памяти MT в составе декодера DO, такого как показано на фиг. 6A и на фиг. 4A или 4B, при этом размер указанной памяти адаптирован для сохранения вычисленного числа вероятностей.

В процессе выполнения этапа D27 декодирующий модуль UDk проверяет, является ли текущий блок, только что декодированный, последним блоком последнего подпотока.

Если это так, то в ходе этапа D28 выполнение способа декодирования завершается.

Если текущий блок - не последний, на этапе D29 выбирают следующий блок  $MV_j$ , подлежащий декодированию, в соответствии с порядком просмотра изображения, обозначенным стрелкой PS на фиг. 4A или 4B.

Если в ходе указанного выше этапа D25 оказалось, что текущий блок не является j-м блоком рассматриваемой строки SEDk, выполняют рассмотренный выше этап D27.

В процессе выполнения этапа D30, следующего за рассмотренным выше этапом D29, определяют доступность предварительно декодированных блоков, необходимых для декодирования текущего блока  $MV_j$ . Если учесть тот факт, что эта процедура использует параллельное декодирование блоков разными декодирующими модулями UDk, может случиться, что эти блоки еще не были декодированы посредством декодирующего модуля, назначенного для декодирования этих блоков, и поэтому эти блоки пока еще недоступны. Указанный этап проверки состоит в проверке, имеется ли заданное число N' блоков, расположенных в предыдущей строке SEk-1, например два блока, находящихся соответственно сверху и сверху справа от текущего блока, и доступных для декодирования текущего блока, иными словами, если они уже были декодированы посредством декодирующего модуля UDk-1, назначенного для их декодирования. Этот этап проверки состоит также в проверке доступности по меньшей мере одного блока, расположенного слева от текущего блока  $MV_j$ , подлежащего декодированию. Однако с учетом порядка PS просмотра изображения, выбранного в варианте, представленном на фиг. 4A или 4B, блоки в рассматриваемой строке SEk декодируют один за другим.



Следовательно, всегда (за исключением обработки первого блока строки) имеется и доступен расположенный слева декодированный блок. В примере, представленном на фиг. 4А или 4В, это блок, расположенный слева непосредственно рядом с текущим блоком, подлежащим декодированию. С этой целью проверяют доступность только  
 5 этих двух блоков, расположенных соответственно сверху и сверху справа от текущего блока.

Поскольку этот этап проверки замедляет работу способа декодирования, в альтернативном варианте настоящего изобретения тактовый сигнал CLK, показанный на фиг. 6А, приспособлен для опережающей синхронизации декодирования блоков  
 10 таким образом, чтобы гарантировать доступность двух блоков, расположенных соответственно сверху и сверху справа от текущего блока, не требуя обязательной проверки доступности этих двух блоков. Таким образом, как показано на фиг. 4А или 4В, декодирующий модуль UDK всегда начинает декодировать первый блок со сдвигом на заданное число  $N'$  (здесь,  $N' = 2$ ) декодированных блоков предыдущей строки SEk-  
 15 1, используемых для декодирования текущего блока. С программной точки зрения реализация такого тактового сигнала делает возможным значительно ускорить и сократить затраты времени на обработку блоков каждого подмножества SEk в декодере DO.

В процессе выполнения этапа D31 проверяют, является ли текущий блок первым  
 20 блоком рассматриваемой строки SEk.

Если текущий блок - первый, в процессе выполнения этапа D32 считывают из буферной памяти MT исключительно вероятности появления символов, вычисленные во время декодирования j-го блока предыдущей строки SEk-1.

Согласно первому варианту, представленному на фиг. 4А, j-й блок является первым  
 25 блоком предыдущей строки SEk-1 ( $j = 1$ ). Такое считывание состоит в замене вероятностей в декодере по стандарту САВАС вероятностями, записанными в буферной памяти MT. Процесс чтения, захватывающий соответствующие первые блоки второй, третьей и четвертой строк SE2, SE3 и SE4, показан на фиг. 4А тонкими стрелками.

Согласно второму варианту упомянутого выше этапа D32, представленного на фиг.  
 30 4В, j-й блок является вторым блоком предыдущей строки SEk-1 ( $j = 2$ ). Такое считывание состоит в замене вероятностей в декодере по стандарту САВАС вероятностями, записанными в буферной памяти MT. Процесс чтения, захватывающий соответствующие первые блоки второй, третьей и четвертой строк SE2, SE3 и SE4, показан на фиг. 4В тонкими штриховыми стрелками.

35 После выполнения этапа D32 текущий блок декодируют путем повторения этапов с D24 по D28, описанных выше.

Если после указанного выше этапа D31 оказалось, что текущий блок не является первым блоком рассматриваемой строки SEk, предпочтительно не считывать вероятности для ранее декодированного блока, расположенного в той же самой строке  
 40 SEk, иными словами декодированного блока, расположенного (в представленном примере) слева непосредственно рядом от текущего блока. Действительно, при последовательном порядке PS просмотра для считывания блоков, расположенных в той же самой строке, как показано на фиг. 4А или 4В, вероятности появления символов, присутствующие в декодере, использующем алгоритм САВАС, в момент начала  
 45 декодирования текущего блока являются точно теми же, какие присутствуют после декодирования предыдущего блока в той же самой строке.

Затем, в процессе выполнения этапа D33 производят обучение вероятностям появления символов для энтропийного декодирования текущего блока, так что эти

вероятности соответствуют исключительно тем, которые были вычислены для указанного предыдущего блока в той же самой строке, как показано двойными сплошными стрелками на фиг. 4А или 4В.

5 После этапа D33 текущий блок декодируют путем повторения этапов с D24 по D28, описанных выше.

В процессе выполнения этапа D34 проверяют, является ли текущий блок последним блоком рассматриваемой строки SE<sub>k</sub>.

Если текущий блок - не последний в этой строке, после этапа D34 снова выполняют этап D29 выбора следующего блока MB<sub>j</sub>, подлежащего декодированию.

10 Если текущий блок является последним блоком рассматриваемой строки SE<sub>k</sub>, в процессе выполнения этапа D35 декодирующий модуль UD<sub>k</sub> выполняет процедуру, идентичную указанному выше этапу D23, иными словами снова инициализирует интервальное представление вероятности появления символа, входящего в состав заданного множества символов. Такая повторная инициализация обозначена на фиг. 15 4А и 4В черной точкой в начале каждой строки SE<sub>k</sub>.

Таким образом, в начале каждой строки декодер DO находится в инициализированном состоянии, что обеспечивает более высокую степень гибкости с точки зрения выбора уровня параллелизма при декодировании и оптимизации затрат времени на обработку данных при декодировании.

20 В примере схемы кодирования/декодирования, представленном на фиг. 7А, кодирующее устройство СО содержит единственный кодирующий модуль UC, как показано на фиг. 3А, тогда как декодер DO содержит шесть декодирующих модулей.

Кодирующий модуль UC последовательно кодирует строки SE1, SE2, SE3, SE4, SE5 и SE6. В представленном примере строки с SE1 по SE4 закодированы полностью, строка 25 SE5 находится в стадии кодирования, а строка SE6 еще не была кодирована. С учетом последовательного характера кодирования кодирующий модуль UC приспособлен для передачи потока F, содержащего подпотоки F1, F2, F3, F4, упорядоченные один за другим в том же порядке, в котором происходило кодирование строк SE1, SE2, SE3 и SE4. С этой целью подпотоки F1, F2, F3 и F4 символически обозначены с такой же 30 штриховкой, как и соответствующие закодированные строки SE1, SE2, SE3 и SE4.

Благодаря выполнению этапов опустошения в конце кодирования каждой закодированной строки и повторной инициализации интервала вероятностей в начале кодирования или декодирования следующей строки, подлежащей кодированию/декодированию, декодер DO каждый раз, когда он считывает подпоток, чтобы декодировать его, находится в 35 инициализированном состоянии и может поэтому оптимальным образом декодировать параллельно четыре подпотока F1, F2, F3, F4 с использованием декодирующих модулей UD1, UD2, UD3 и UD4, которые могут быть, например, установлены на четырех разных платформах.

40 В примере схемы кодирования/декодирования, представленном на фиг. 7В, кодирующее устройство СО содержит два кодирующих модуля UC1 и UC2, как показано на фиг. 3С, тогда как декодер DO содержит шесть декодирующих модулей.

Кодирующий модуль UC1 последовательно кодирует строки с нечетными номерами SE1, SE3 и SE5, тогда как кодирующий модуль UC2 последовательно кодирует строки с четными номерами SE2, SE4 и SE6. Для этого строки SE1, SE3 и SE5 имеют белый 45 фон, а строки SE2, SE4 и SE6 имеют фон, заполненный точками. В представленном примере строки с SE1 по SE4 закодированы полностью, строка SE5 находится в стадии кодирования, а строка SE6 еще не была кодирована. Учитывая тот факт, что выполняемое кодирование является кодированием параллельного типа уровня 2,

кодирующий модуль UC1 адаптирован для передачи подпотока  $F_{2n+1}$ , разложенного на две части F1 и F3, полученные после кодирования соответственно строк SE1 и SE3, тогда как кодирующий модуль UC2 адаптирован для передачи подпотока  $F_{2n}$ ,

5 разложенного на две части F2 и F4, полученные после кодирования соответственно строк SE2 и SE4. Кодирующее устройство CO поэтому предназначено для передачи декодеру DO потока F, содержащего результат наложения двух подпотоков  $F_{2n+1}$  и  $F_{2n}$ , вследствие чего порядок следования подпотоков F1, F3, F2, F4 отличается от того, что показано на фиг. 7А. С этой целью подпотоки F1, F2, F3 и F4 символически обозначены с такой же штриховкой, как и соответствующие закодированные строки SE1, SE2, SE3 и SE4, подпотоки F1 и F3 имеют белый фон (кодирование строк с нечетными номерами), а подпотоки F2 и F4 имеют точечный фон (кодирование строк с четными номерами).

10 Что касается преимуществ, отмеченных в связи с фиг. 7А, такая схема кодирования/декодирования дополнительно предоставляет преимущества способности использовать декодер, уровень параллелизма при декодировании в котором полностью независим от уровня параллелизма при кодировании, что делает возможным в еще большей степени оптимизировать работу кодирующего устройства/декодера.

#### (57) Формула изобретения

20 1. Способ кодирования по меньшей мере одного изображения, причем способ содержит:

сегментирование изображения на множество блоков;

группировку блоков в predetermined число подмножеств блоков;

кодирование, с использованием модуля энтропийного кодирования, текущего блока из упомянутых подмножеств блоков, при этом кодирование содержит:

25 когда текущий блок является первым блоком в порядке кодирования подмножества, которое не является первым подмножеством изображения в порядке кодирования,

определение вероятностей возникновения символа для текущего блока, причем вероятности являются теми, которые были определены посредством кодирования некоторого predetermined блока из по меньшей мере одного другого

30 подмножества, при этом упомянутый predetermined блок является вторым блоком в порядке кодирования в упомянутом другом подмножестве,

инициализацию переменных состояния модуля энтропийного кодирования, и кодирование текущего блока; и

35 генерирование по меньшей мере одного подпотока данных для упомянутого по меньшей мере одного изображения.

2. Способ по п. 1, в котором подмножества блоков кодируются последовательно или параллельно.

3. Способ по п. 2, в котором, когда по меньшей мере два подмножества блоков кодируются параллельно с по меньшей мере одним другим подмножеством блоков, упомянутые по меньшей мере два кодируемых подмножества блоков содержатся в одном подпотоке данных.

4. Способ по п. 1, в котором, когда кодируемые подмножества блоков предназначены, чтобы декодироваться параллельно в некотором predetermined порядке, подпотоки данных сначала упорядочивают согласно упомянутому predetermined порядку перед передачей их декодеру.

45 5. Устройство кодирования по меньшей мере одного изображения, содержащее: модуль разделения, сконфигурированный с возможностью сегментирования изображения на множество блоков;

модуль вычисления, сконфигурированный с возможностью группировки блоков в некоторое predetermined число подмножеств блоков;

5 блок кодирования, сконфигурированный с возможностью кодирования, с использованием модуля энтропийного кодирования, текущего блока из упомянутых подмножеств блоков, при этом кодирование содержит:

когда текущий блок является первым блоком в порядке кодирования подмножества, которое не является первым подмножеством изображения в порядке кодирования,

10 определение вероятностей возникновения символа для текущего блока, причем вероятности являются теми, которые были определены посредством кодирования некоторого predetermined блока из по меньшей мере одного другого подмножества, при этом упомянутый predetermined блок является вторым блоком в порядке кодирования в упомянутом другом подмножестве,

инициализацию переменных состояния модуля энтропийного кодирования, и кодирование текущего блока; и

15 модуль генерирования потока, сконфигурированный с возможностью генерирования по меньшей мере одного подпотока данных для упомянутого по меньшей мере одного изображения.

6. Долговременный считываемый компьютером носитель, хранящий компьютерную программу, содержащую инструкции, которые, при исполнении одним или более процессорами, побуждают упомянутый один или более процессоров выполнять операции, содержащие:

сегментирование изображения на множество блоков;

группировку блоков в predetermined число подмножеств блоков;

25 кодирование, с использованием модуля энтропийного кодирования, текущего блока из упомянутых подмножеств блоков, при этом кодирование содержит:

когда текущий блок является первым блоком в порядке кодирования подмножества, которое не является первым подмножеством изображения в порядке кодирования,

30 определение вероятностей возникновения символа для текущего блока, причем вероятности являются теми, которые были определены посредством кодирования некоторого predetermined блока из по меньшей мере одного другого подмножества, при этом упомянутый predetermined блок является вторым блоком в порядке кодирования в упомянутом другом подмножестве,

инициализацию переменных состояния модуля энтропийного кодирования, и кодирование текущего блока; и

35 генерирование по меньшей мере одного подпотока данных для упомянутого по меньшей мере одного изображения.

7. Способ декодирования потока, представляющего по меньшей мере одно закодированное изображение, причем способ содержит:

40 идентификацию в потоке некоторого predetermined числа подпотоков данных, соответствующих соответственно по меньшей мере одному подмножеству блоков, которые должны быть декодированы;

декодирование текущего блока в подмножествах блоков, соответствующих упомянутому predetermined числу подпотоков данных, посредством использования модуля энтропийного декодирования, при этом декодирование содержит:

45 когда текущий блок является первым подлежащим декодированию блоком из подмножества, которое не является первым подмножеством закодированного изображения в порядке декодирования,

считывание вероятностей возникновения символа для текущего блока, причем

вероятности являются теми, которые были определены посредством декодирования некоторого предопределенного блока из по меньшей мере одного другого подмножества, при этом упомянутый предопределенный блок является вторым подлежащим декодированию блоком в упомянутом другом подмножестве в порядке

5 декодирования,

инициализацию переменных состояния модуля энтропийного декодирования, и декодирование текущего блока.

8. Способ по п. 7, в котором подмножества блоков декодируются последовательно или параллельно.

10 9. Способ по п. 8, в котором, когда по меньшей мере два подмножества блоков декодируются параллельно с по меньшей мере одним другим подмножеством блоков, один из идентифицированных подпотоков данных представляет упомянутые по меньшей мере два подмножества блоков.

15 10. Способ по п. 8, в котором, когда кодированные подмножества блоков декодируются параллельно в некотором предопределенном порядке, подпотоки данных, соответствующие кодированным подмножествам блоков, упорядочены в упомянутый предопределенный порядок в потоке, который должен быть декодирован.

11. Устройство декодирования потока, представляющего по меньшей мере одно кодированное изображение, содержащее:

20 модуль извлечения потока, сконфигурированный с возможностью идентификации в потоке некоторого предопределенного числа подпотоков данных, соответствующих соответственно по меньшей мере одному подмножеству блоков, которые должны быть декодированы;

25 блок декодирования, сконфигурированный с возможностью декодирования текущего блока в подмножествах блоков, соответствующих идентифицированным подпотокам данных, с использованием модуля энтропийного декодирования, при этом декодирование содержит:

30 когда текущий блок является первым подлежащим декодированию блоком из подмножества, которое не является первым подмножеством кодированного изображения в порядке декодирования,

35 считывание вероятностей возникновения символа для текущего блока, причем вероятности являются теми, которые были определены посредством декодирования некоторого предопределенного блока из по меньшей мере одного другого подмножества, при этом упомянутый предопределенный блок является вторым подлежащим декодированию блоком в упомянутом другом подмножестве в порядке декодирования,

инициализацию переменных состояния модуля энтропийного декодирования, и декодирование текущего блока.

40 12. Долговременный считываемый компьютером носитель, хранящий компьютерную программу, содержащую инструкции, которые, при исполнении одним или более процессорами, побуждают упомянутый один или более процессоров выполнять операции, содержащие:

45 идентификацию в потоке некоторого предопределенного числа подпотоков данных, соответствующих соответственно по меньшей мере одному подмножеству блоков, которые должны быть декодированы;

декодирование текущего блока в подмножествах блоков, соответствующих идентифицированным подпотокам данных, посредством использования модуля энтропийного декодирования, при этом декодирование содержит:

когда текущий блок является первым подлежащим декодированию блоком из подмножества, которое не является первым подмножеством кодированного изображения в порядке декодирования,

5 считывание вероятностей возникновения символа для текущего блока, причем вероятности являются теми, которые были определены посредством декодирования некоторого predetermined блока из по меньшей мере одного другого подмножества, при этом упомянутый predetermined блок является вторым подлежащим декодированию блоком в упомянутом другом подмножестве в порядке декодирования,

10 инициализацию переменных состояния модуля энтропийного декодирования, и декодирование текущего блока.

15

20

25

30

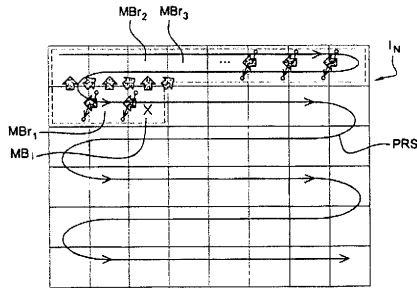
35

40

45

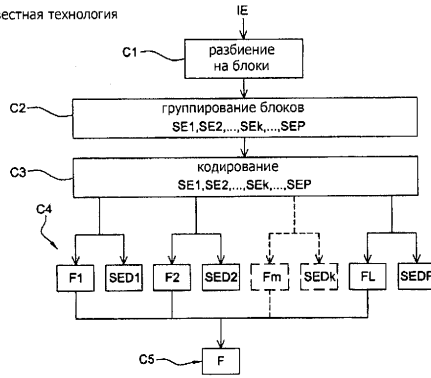
1

1/9



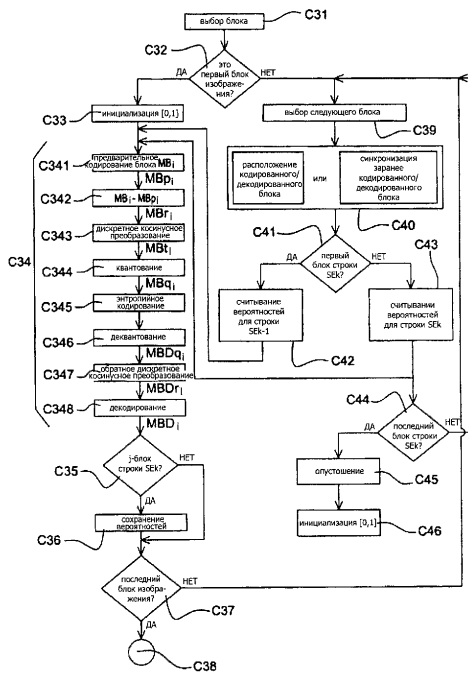
Фиг. 1

известная технология



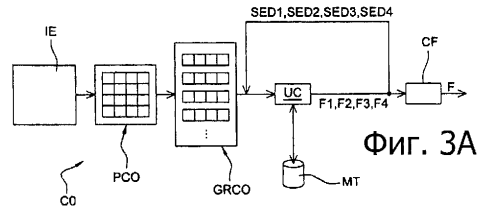
Фиг. 2А

2

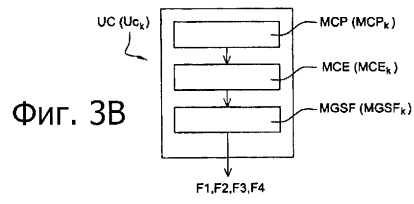


Фиг. 2В

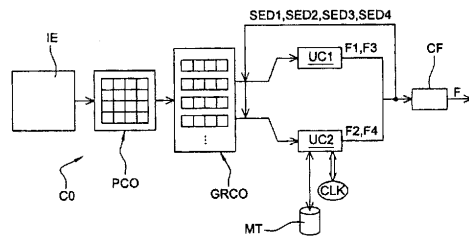




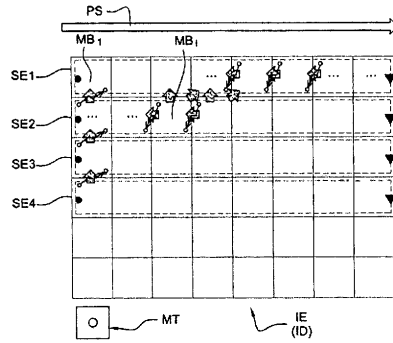
Фиг. 3А



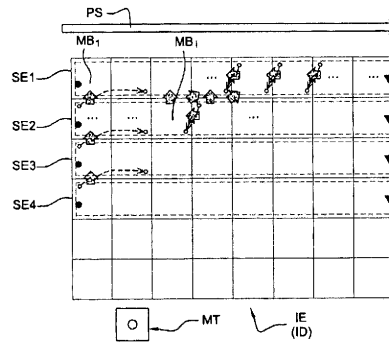
Фиг. 3В



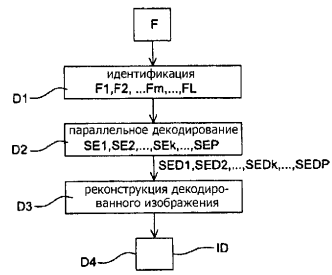
Фиг. 3С



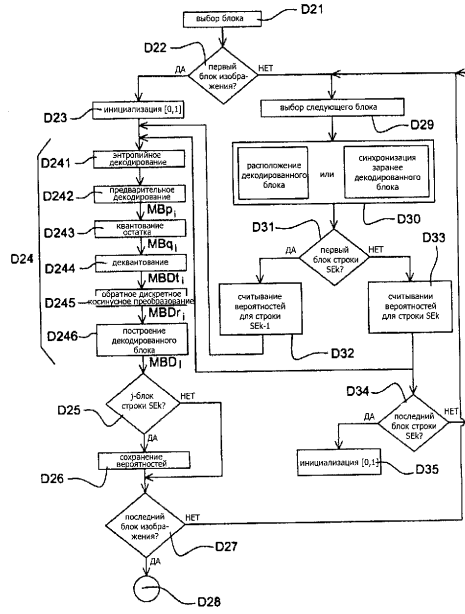
Фиг. 4А



Фиг. 4В

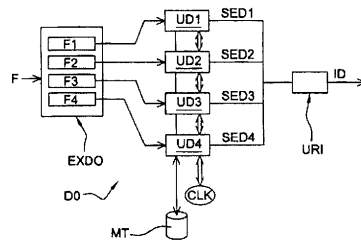


Фиг. 5А

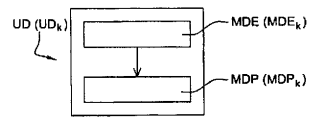


Фиг. 5B

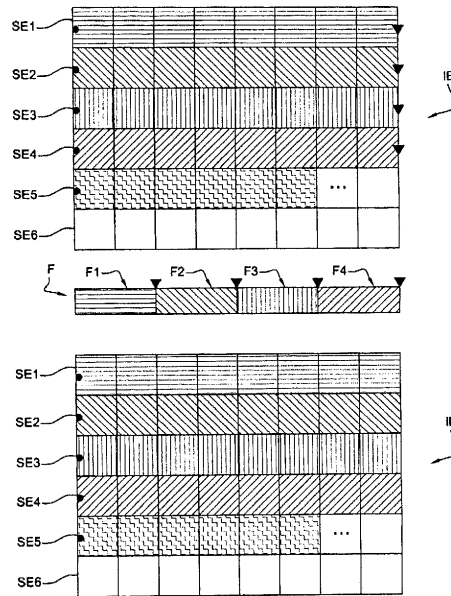
7/9



Фиг. 6А

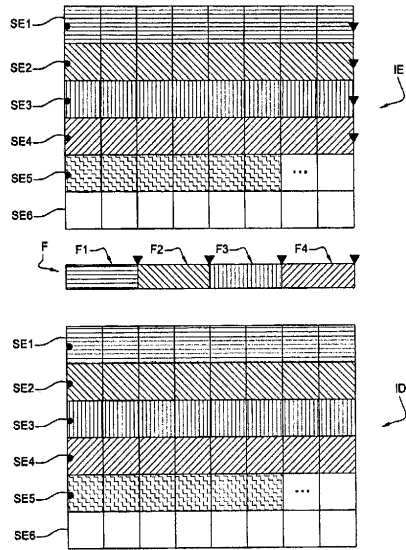


Фиг. 6В



Фиг. 7А

9/9



Фиг. 7В