

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: 2010124854/08, 16.06.2010

(24) Дата начала отсчета срока действия патента:
16.06.2010

Приоритет(ы):

(30) Конвенционный приоритет:
17.06.2009 FI 20095686

(43) Дата публикации заявки: 27.12.2011 Бюл. № 36

(45) Опубликовано: 20.09.2013 Бюл. № 26

(56) Список документов, цитированных в отчете о поиске: GB 2362055, 07.11.2001. Swilem A. et al. "A Fast Search Algorithm for Mean Pyramids Vector Quantization Using", Faculty of computers & Information-Cairo University, 27.03.2008, найдено по адресу: <http://infos2008.fci.cu.edu.eg/infos/MM-03-P013-019.pdf>. SONG B.C. et al. "A Fast Search Algorithm for Vector Quaqntization Using L2-Norm (см. прод.)

Адрес для переписки:

109012, Москва, ул. Ильинка, 5/2, ООО
"Союзпатент"

(72) Автор(ы):

КЯРККЯЙНЕН Туомас (FI)

(73) Патентообладатель(и):

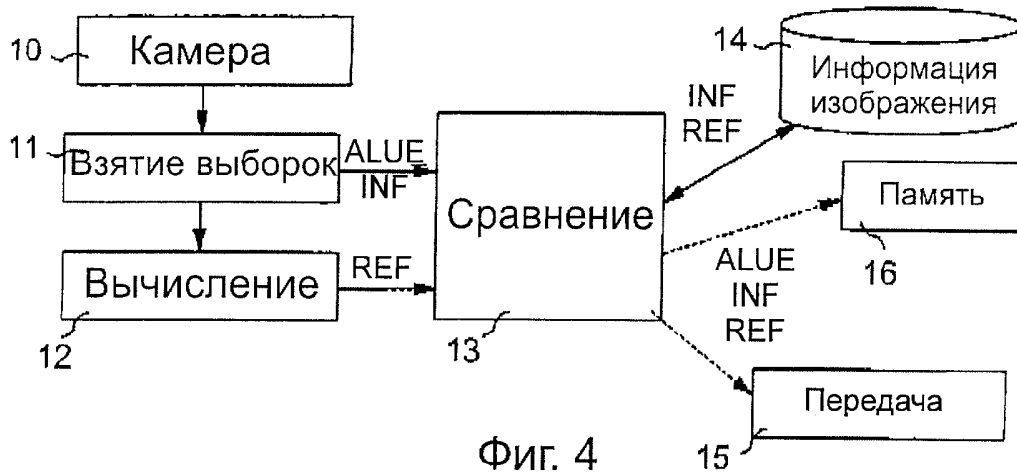
ГУРУЛОДЖИК МАЙКРОУСИСТЕМС
ОЙ (FI)

(54) ПРОЦЕССОР ИЗОБРАЖЕНИЙ, ГЕНЕРАТОР ИЗОБРАЖЕНИЙ И КОМПЬЮТЕРНАЯ ПРОГРАММА

(57) Реферат:

Изобретение относится к обработке и генерированию изображений. Техническим результатом является обеспечение быстрого уплотнения изображения или видеоданных в формат, занимающий как можно меньшую емкость памяти. Процессор изображений выполнен с возможностью: принимать изображение; получать специфичные для области выборки (11) информации изображения области и вычислять (12) опорное значение (REF) на основе этих выборок; сохранять в память (16) или передавать (15) специфичные для области опорные значения (REF), вычисленные для области, и

идентификатор (ALUE) области; проверять (13), запомнено ли уже в памяти (16) или передано (15) ли опорное значение (REF), полученное в результате вычисления, и соответствующая ему информация (INF) изображения; и сохранять в память (16) или передавать (15) вычисленное опорное значение (REF) и соответствующую информацию (INF) изображения, для которой вычислено это опорное значение, если опорное значение (REF) и соответствующая информация (INF) изображения ранее не были сохранены в память или не переданы. 4 н. и 6 з.п. ф-лы, 10 ил.



Фиг. 4

(56) (продолжение):

Pyramid of Codewords", IEEE Transactions on Image Processing. Vol. 11, № 1, январь 2002, найдено по адресу: <http://www-isl.Kaist.ac.kr/papers/IJ/49-Jan2002-IEEE-IP-bcsong.pdf>. RU 2003130965 A, 27.04.2005.

RU 2493600 C2

RU 2493600 C2



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(51) Int. Cl.
G06T 1/00 (2006.01)
H04N 7/26 (2006.01)

(12) **ABSTRACT OF INVENTION**

(21)(22) Application: 2010124854/08, 16.06.2010

(24) Effective date for property rights:
16.06.2010

Priority:

(30) Convention priority:
17.06.2009 FI 20095686

(43) Application published: 27.12.2011 Bull. 36

(45) Date of publication: 20.09.2013 Bull. 26

Mail address:

109012, Moskva, ul. Il'inka, 5/2, OOO
"Sojuzpatent"

(72) Inventor(s):

KJaRKKJaJNEN Tuomas (FI)

(73) Proprietor(s):

GURULODZhIK MAJKROUSYSTEMS OJ (FI)

(54) **IMAGE PROCESSOR, IMAGE GENERATOR AND COMPUTER PROGRAMME**

(57) Abstract:

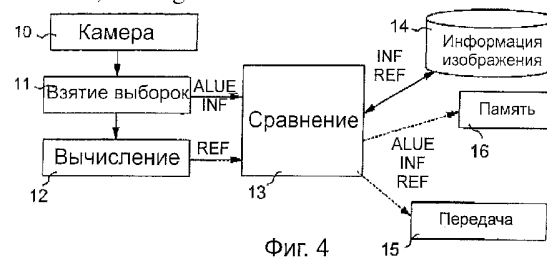
FIELD: information technology.

SUBSTANCE: image processor configured to: receive an image; take area-specific samples (11) of the image information of the area and calculate (12) a reference value (REF) based on the samples; store in memory (16) or transmit (15) the area-specific reference values (REF) calculated for an area and an area identifier (ALUE); check (13) whether the reference value (REF) obtained as a result of the calculation and the corresponding image information (INF) have already been stored in the memory (16) or transmitted (15); and store in the memory (16) or transmit (15) the calculated reference value (REF) and the corresponding image information (INF), for

which said reference value was calculated, if the reference value (REF) and the corresponding image information (INF) have not been stored in the memory or transmitted previously.

EFFECT: providing fast compression of images or video data in a format which occupies minimal space.

10 cl, 15 dwg



Фиг. 4

RU 2 493 600 C2

RU 2 493 600 C2

Область техники, к которой относится изобретение

Изобретение относится к обработке и генерированию изображений и, в частности, к решению, которое позволяет уплотнять изображение или видеоданные в формат, занимающий как можно меньшую емкость памяти или пропускную способность и позволяющий восстанавливать изображение или видеоданные после такого уплотнения.

Уровень техники

Ключевой проблемой при уплотнении изображений и видеоданных (далее - «изображение») является объем данных в уплотненном изображении, т.к. эти данные затем сохраняются в памяти для последующего использования или передачи по сети передачи данных получателю для использования. Этот объем данных должен быть как можно меньше.

Далее, другой важный вопрос при уплотнении состоит в генерировании уплотненного изображения так, чтобы изображение, генерируемое из сохраненных или переданных данных, соответствовало исходному изображению как можно точнее.

Третий критичный для уплотнения вопрос - это требующиеся ресурсы, т.е. быстрое уплотнение и разуплотнение изображения должно требовать от процессора в используемом устройстве как можно меньшую пропускную способность.

Известные решения для уплотнения изображения и для восстановления его после уплотнения не могут в должной мере удовлетворить вышеуказанным критериям.

Раскрытие изобретения

Цель данного изобретения состоит в решении вышеуказанной проблемы и в обеспечении более эффективного решения для обработки изображения и выдачи обработанного изображения при восстановлении. Это достигается с помощью процессора изображений согласно независимому п.1 формулы изобретения, процессору изображений согласно независимому п.6 формулы изобретения, компьютерной программе согласно независимым пп.10 и 11 формулы изобретения и машиночитаемому носителю данных согласно независимому п.12 формулы изобретения.

Изобретение использует возможность разделения изображения на области, для которых вычисляется специфичное для области опорное значение. Если несколько областей имеют одно и то же опорное значение, эти области изображения с достаточной точностью соответствуют друг другу в отношении своего внешнего вида и, следовательно, информацию изображения всех этих областей не нужно сохранять или передавать по отдельности, но достаточно посылать информацию изображения для этих областей однократно. Когда генерируется изображение для областей с одними и теми же опорными значениями, используется одна и та же информация изображения, сохраненная или переданная ранее, посредством включения информации в области изображения, подлежащего генерированию с одним и тем же опорным значением. Это позволяет избежать повторной передачи или хранения одной и той же информации. Лишь когда информация изображения не сохранена или не передана ранее вместе с соответствующим опорным значением, эту информацию изображения необходимо передавать далее или сохранять.

Предпочтительные варианты осуществления процессора изображений и генератора изображений по изобретению раскрываются в зависимых пунктах.

Краткое описание чертежей

Изобретение будет описываться ниже более подробно со ссылкой на сопровождающие чертежи, на которых:

фиг.1 иллюстрирует разделение изображения на области;
 фиг.2а-2е иллюстрируют выборку из области изображения;
 фиг.3а-3б иллюстрируют таблицы данных, подлежащих передаче или хранению;
 5 фиг.4 и 5 иллюстрируют первый вариант осуществления процессора изображений;
 фиг.6 и 7 иллюстрируют первый вариант осуществления генератора изображений;
 фиг.8 и 9 иллюстрируют второй вариант осуществления генератора изображений;
 фиг.10 иллюстрирует обработку видеоизображения.

Осуществление изобретения

10 Фиг.1 иллюстрирует разделение изображения 1 на области A1-A_n. Число и размер этих областей можно в каждом случае выбирать по отдельности. Альтернативой является разделение изображения на области, содержащие каждая 8×8 пикселей. Яркость цвета этих пикселей предпочтительно составляет по меньшей мере 24 бита (RGB, красный-зеленый-синий), но наилучший результат получается за счет
 15 использования как можно более высокой яркости цвета, такой как яркость в 64 бита. Более крупная область перемещения требует более точной яркости цвета для получения однозначно определяемой области.

Фиг.2а-2е иллюстрируют выборку из области изображения. Можно предположить,
 20 например, что фиг.2а-2е показывают выборку из области A1 на фиг.1, что означает, что некоторые пиксели из информации изображения в этой области выбраны в качестве выборок.

Выборки можно выбирать с помощью известного самого по себе метода избыточной выборки, что означает, что в случае фиг.2а применен сеточный алгоритм
 25 для взятия девяти выборок из местоположения, показанного точками на фиг.2. В этом случае каждая выборка представляет численное значение пикселя в рассматриваемом местоположении, т.е. значение, практически определяющее цвет этого пикселя. Альтернатива состоит в осуществлении выборки по случайному алгоритму (фиг.2б), по алгоритму диска Пуассона (фиг.2с), по алгоритму Джиттера (фиг.2д) или по
 30 алгоритму повернутой сетки (фиг.2е).

Когда необходимые выборки взяты из обрабатываемой области, для рассматриваемой области изображения вычисляется опорное значение на основе этих
 35 выборок. Опорное значение можно вычислять так, чтобы оно соответствовало среднему от численных значений выборок из области, т.е. пикселей. Практические проверки показали, что хороший результат получается, например, за счет использования области изображения размером 16×16, метода избыточной выборки и цветовой яркости пикселей, которая составляет 24 бита, а размер исходного
 40 изображения равен 1920×1080 пикселей. Избыточную выборку можно осуществлять, например, следующим образом:

1) Исходное изображение разделяется на прямоугольники равного размера, причем каждый прямоугольник соответствует одной области изображения.

2) Вычисляется взвешенная сумма всех пикселей внутри прямоугольника или
 45 пересекаемых прямоугольником. Значение пикселя для пикселя, расположенного внутри прямоугольника, принимает весовое значение 1. Если же пиксел пересекается с прямоугольником и прямоугольник имеет область пересечения $a < 1$, значение такого пикселя принимает взвешенное значение a .

3) Для вычисления опорного значения взвешенная сумма делится на площадь
 50 поверхности прямоугольника (выраженную в пикселях, в этом случае площадь поверхности равна числу пикселей по горизонтали в прямоугольнике, умноженному на число пикселей по вертикали в прямоугольнике).

Как раскрыто выше, в качестве примера, все пиксели в области учитываются для вычисления опорного значения, и области конкретно являются прямоугольными. Однако это не обязательно; опорное значение можно также вычислять, принимая выборки только от некоторых пикселей в области и делением изображения на области какой-нибудь иной формы, нежели прямоугольник, например, треугольной.

Фиг.3а и 3б представляют собой таблицы данных, подлежащих передаче или хранению. Когда выборки областей A1-A_n изображения 1 на фиг.1 берутся так, как раскрыто в связи с фиг.2а-2е, и на основе этих значений вычисляются опорные значения, данные, приведенные в таблицах 3а и 3б, совместно сохраняются, либо эти данные, альтернативно, немедленно передаются далее в генератор изображения для восстановления изображения на приемном конце.

Когда выборки взяты, например, из области A1 на фиг.1 и на основе этого получено опорное значение 1 для рассматриваемой области, опорное значение 1 сохраняется в таблице на фиг.3б в связи с идентификатором области для области A1. Затем проверяется, сохранена ли уже информация изображения, соответствующая опорному значению 1 в связи с опорным значением 1 в таблице на фиг.3а. Предполагая, что это не сделано, информация DATA_A изображения области A1 на фиг.1 сохраняется в таблице 3а в качестве информации изображения, соответствующей опорному значению 1, тем самым информация изображения практически выделяет все значения пикселей в этой области. Если число пикселей равно 8×8, информация изображения содержит 64 пиксела, которые таким образом представляют цвета этих пикселей.

Процесс продолжается путем сохранения опорного значения 253 для области A2 в таблице на фиг.3б и информации изображения области A2 в качестве информации изображения, соответствующей опорному значению 253 (не показано).

При переходе к области A5 снова получается опорное значение 1. В этом случае опорное значение 1 сохраняется в таблице на фиг.3б для идентификатора области A5. Затем проверяется, содержит ли уже таблица 3а информацию изображения, сохраненную для опорного значения 1. Проверка подтверждает это, потому что информация DATA_A изображения области A1 сохранена вместе с областью A1 для опорного значения 1 в таблице на фиг.3бА, следовательно, эту информацию изображения не нужно сохранять вновь для области A5.

Процесс протекает, как описано выше, до тех пор, пока все области A1-A_n изображения 1 не будут проверены. Фиг.3б показывает в каждом идентификаторе области A1-A_n опорное значение этой области. Таким образом, при восстановлении изображения возможно генерировать исходное изображение путем поиска в таблице на фиг.3а информации DATA_A-DATA_G изображения, соответствующей опорному значению области, а извлеченную информацию изображения можно показать в изображении в области, указанной идентификатором области A1-A_n. Информацию изображения в таблице на фиг.3а нужно сохранять только один раз для каждого опорного значения, и изображение для всех областей с одним и тем же опорным значением можно генерировать из одной и той же информации изображения. Это позволяет минимизировать объем подлежащей хранению информации изображения, что, в свою очередь, экономит память и полосу пропускания при передаче данных в зависимости от приложения.

Фиг.4 и 5 иллюстрируют первый вариант осуществления процессора изображений. Этот пример предполагает, что на этапе А процессор изображений принимает от камеры 10 сигнал изображения, содержащий изображение. На практике сигнал

изображения можно принимать от какого-либо иного средства, нежели камера, например, от жесткого диска, диска или путем считывания из памяти. На этапе В блок 11 выборки берет специфичные для области выборки, а блок 12 вычисления осуществляет специфичное для области вычисление опорного значения, как обсуждено
5 выше со ссылкой на фиг.1 и 2а-2е, например.

На этапе С блок 13 сравнения использует базу 14 данных, содержащую информацию изображения, чтобы найти, послана ли или сохранена ли ранее информация INF изображения, соответствующая вычисленному опорному
10 значению REF. В этом случае база 14 данных может содержать сохраненные в ней данные таблицы на фиг.3а. Блок сравнения принимает из блока 12 вычисления опорное значение REF, вычисленное для обрабатываемой области изображения, и на основе этого выполняет поиск в базе 14 данных, чтобы найти, сохранена ли в этой
15 базе 14 данных предыдущая информация изображения, связанная с опорным значением REF.

Если база 14 данных содержит информацию изображения для вычисленного опорного значения REF, это означает, что информация изображения уже сохранена ранее в памяти или передана дальше, т.е. ее не нужно посылать или сохранять вновь.
20 Блок 13 сравнения тогда передает на этапе D идентификатор ALUE области, полученный из блока 11 выборки, и опорное значение, полученное из блока 12 вычисления, в блок 15 передачи, например, для немедленной дальнейшей передачи по линии связи к приемному генератору изображений, или альтернативно, блок 13
25 сравнения сохраняет идентификатор ALUE области и опорное значение REF в память 16 для использования позже.

Однако если блок 13 сравнения обнаруживает на этапе С, что база 14 данных не содержит никакой информации изображения для вычисленного опорного значения REF, это означает, что информация изображения является новой. В этом
30 случае блок 13 сравнения передает на этапе E идентификатор ALUE области, полученный из блока 11 выборки, опорное значение REF, полученное из блока 12 вычисления, а также информацию INF изображения, полученную из блока 11 выборки, в блок 15 передачи, например, для немедленной дальнейшей передачи по линии связи к
35 приемному генератору изображений, или альтернативно, блок 13 сравнения сохраняет идентификатор ALUE области, опорное значение REF и информацию INF изображения в память 16 для использования позже. В то же время блок 13 сравнения передает также опорное значение REF и информацию INF изображения к базе 14 данных для
40 сохранения в базе 14 данных.

Блок 15 передачи и память 16 не обязательно должны быть в одно и то же время в одном и том же устройстве. Если данные, генерированные процессором изображений, подлежат передаче непосредственно к приемному генератору изображений, например,
45 по линии связи, достаточно одного блока 15 передачи. И опять-таки, если данные не предназначены для непосредственной передачи, достаточно одной памяти, и сохраненные в памяти 16 данные можно передавать позже для использования генератором изображений либо по линии связи, либо посредством их переноса из
50 одного генератора изображений в другой генератор изображений на запоминающем средстве, например, таком как карта памяти, диск или жесткий диск.

На этапе F проверяется, содержит ли еще изображение такие области, которые не обработаны. Если да, процесс возвращается к этапу В для обработки следующей области, т.е. блок 11 выборки берет выборки, из которых блок 12 вычисления
55 вычисляет опорное значение, и на этой основе блок 13 сравнения действует, как

описано выше. Только после того, как все области изображения обработаны, процесс переходит к этапу G.

Если нужно обработать единственное изображение, то больше изображений нет, и процессор изображений прекращает свою работу после этапа G. Однако в случае
5 видео, например, содержащего множество следующих одно за другим изображений, процесс возвращается из этапа G к этапу A до тех пор, пока не будут обработаны все изображения в этом видео, как описано выше.

Что касается видеоизображения, один вариант осуществления позволяет также
10 блоку 11 выборки формировать индекс для средства 13 сравнения, чтобы показать, какое изображение обрабатывается, т.е. какое изображение в видео является связанным с последним опорным значением и идентификатором области. Средство сравнения передает этот индекс далее посредством блока 15 или сохраняет его в
15 память 16 вместе с идентификатором ALUE области и опорным значением REF. Использование этого индекса не является необходимым во всех вариантах осуществления. Блок-схема на фиг.4 предназначена лишь, чтобы иллюстрировать работу видеопроцессора без ограничения его физической структуры. На практике, блок выборки, блок вычисления, блок сравнения и блок передачи могут быть
20 воплощены посредством схемного решения или как комбинация компьютерной программы и схемных решений, и в этом случае процессор, например, исполняет конкретную программу для обеспечения работы блоков. Память 16 и база 14 данных могут быть воплощены единственным запоминающим средством, таким как карта памяти или жесткий диск. Альтернатива состоит в том, что процессор изображений на
25 практике составлен из компьютера, IP камеры, мобильного телефона или портативного компьютера с камерой, присоединенной к нему или встроенной в него, и подготовленный для исполнения компьютерной программы, которая управляет рассматриваемым устройством, как описано выше.

Фиг.6 и 7 иллюстрируют первый вариант осуществления генератора изображений. Генератор изображений на фиг.6, который может функционировать согласно блок-схеме алгоритма на фиг.7, может быть, например, мобильным телефоном, IP камерой, компьютером или портативным компьютером, принимающим данные по линии связи
30 от процессора изображений, раскрытого со ссылкой на фиг.4 и 5, и непосредственно генерирует изображение для отображающего устройства на основе этих принятых данных. Это может быть связано с видеоконференцией, например, или, в общем случае, с оборудованием, пригодным для передачи видео, посредством чего видео
35 может уплотняться в компактный формат для передачи по линии связи и восстановления генератором изображений, чтобы соответствовать исходному изображению с чрезвычайно высокой точностью.

На этапе H блок 20 приема принимает данные по линии связи. Подлежащие приему данные содержат по меньшей мере идентификатор ALUE области, который в
40 индивидуальном порядке различает область в изображении, к которому относится содержащееся в этих данных опорное значение REF. Помимо этого, данные могут также содержать индекс, указывающий, какое изображение обрабатывается. В случае видео, например, этот индекс может показывать, какое изображение в видео является тем, к которому относятся принятое опорное значение REF и принятый
45 идентификатор области. Далее, принятые данные могут также включать в себя информацию INF изображения вместе с идентификатором ALUE области и опорным значением REF. На этапе I блок 20 приема проверяет, содержится ли информация INF изображения.

Если эта проверка показывает, что информация изображения включена в данные, блок 20 приема сохраняет на этапе J эту информацию INF изображения и опорное значение REF в базу 21 данных. Это позволяет бит за битом собирать данные, соответствующие данным в таблице на фиг.3а, в базу данных во время работы генератора изображения. Когда информация изображения сохранена, блок 20 приема на этапе L передает принятый идентификатор ALUE области и принятое опорное значение REF в генерирующий блок 22, который на основе информации INF изображения генерирует изображение в область, соответствующую идентификатору ALUE области подлежащего генерированию изображения.

Затем снова, если проверка показывает, что никакая информация изображения не включена в данные, т.е. принятые данные имеют идентификатор ALUE области и опорное значение REF, но не имеют информации изображения, блок 20 приема осуществляет на этапе K поиск в базе 21 данных на соответствие информации INF изображения на основе опорного значения REF. Это означает, что пара, составленная из информации изображения и опорного значения, уже принята и сохранена в базу 21 данных. Блок 20 приема затем передает принятый идентификатор ALUE области и информацию INF изображения, извлеченную из базы данных на этапе L, в блок 22 генерирования изображения, который генерирует изображение на основе информации INF изображения в область, соответствующую идентификатору ALUE области того изображения, которое подлежит генерированию.

На этапе M проверяется, закончен ли прием данных. Если нет, процесс возвращается к этапу H для приема новых данных. Этапы блок-схемы алгоритма на фиг.7 повторяются, таким образом, до тех пор, пока не будут приняты данные для всех необходимых областей изображения и пока изображение не будет сгенерировано на этой основе для правильной области изображения с помощью принятой или сохраненной информации изображения. Однако не всегда нужно вырабатывать изображение для всех областей изображения на основе информации изображения, потому что, например, в связи с воспроизведением видео возможно генерировать изображение с помощью предыдущего видеоизображения в качестве основы и изменять только те части, которые изменены. Следовательно, данные принимаются только для тех частей изображения, которые изменены по отношению к ранее выработанному изображению.

Блок-схема на фиг.6 предназначена только для иллюстрации работы генератора изображений без ограничения его структуры. На практике блок приема и блок генерирования изображения могут быть выполнены как схемное решение или как комбинация компьютерной программы и схемного решения, например, процессора, выполняющего конкретную программу для осуществления работы блоков. База 21 данных может быть реализована одним отдельным запоминающим средством, таким как запоминающая схема или жесткий диск.

Фиг.8 и 9 иллюстрируют второй вариант осуществления генератора изображения. Генератор изображения на фиг.8, который может работать, как показано на блок-схеме алгоритма на фиг.9, может быть мобильным телефоном, IP камерой, компьютером или портативным компьютером, например, с информацией таблиц 3а и 3б, сохраненным в его памяти, который генерирует изображение или видео на их основе. Альтернатива состоит в том, что необходимые данные переносятся к генератору изображений на машиночитаемом носителе, таком как карта памяти, диск или жесткий диск.

Ниже предполагается, в качестве примера, что память 30 имеет сохраненную в ней

информацию таблицы на фиг.3b, т.е. идентификаторы ALUE области для областей изображения, подлежащих генерированию в правильном порядке, и соответствующие опорные значения REF, а база 1 данных имеет информацию таблицы на фиг.3а, т.е. сохраненные в ней опорные значения REF и соответствующую информацию INF изображения. В дополнение в вышеуказанным данным в памяти может сохраняться индекс, чтобы указывать, к какому из изображений (в числовом порядке) относятся конкретный идентификатор области и соответствующее опорное значение, так что когда генерированию подлежит, например, видео, информацию изображения конкретной области можно генерировать в правильное изображение.

На этапе N идентификатор ALUE области первой области, подлежащей обработке, и соответствующее опорное значение REF считываются из памяти 31 и передаются для использования в блок 32 генерирования изображения. На основе принятого опорного значения REF блок 32 генерирования затем на этапе O проводит поиск информации INF изображения, соответствующей этому опорному значению. Затем блок 32 генерирования изображения на основе информации INF изображения генерирует на этапе P изображение в область, соответствующую идентификатору области изображения. На этапе Q проверяется, завершена ли обработка последней области, и если нет, процесс возвращается к этапу N.

Блок-схема на фиг.8 предназначена только для иллюстрации работы генератора изображений без ограничения его структуры. На практике блок 32 генерирования изображения может быть выполнен как схемное решение или как комбинация компьютерной программы и схемного решения, например, процессора, выполняющего конкретную программу для осуществления работы блока. База 31 данных и память 30 могут быть реализованы одним отдельным запоминающим средством, таким как запоминающая схема или жесткий диск. Альтернатива состоит в том, что генератор изображения на фиг.8 представляет собой компьютер, мобильный телефон, IP камеру, портативный компьютер или тому подобное, которые на основе данных, сохраненных в его памяти, или данных, считанных из запоминающего средства, генерирует изображение или видео для отображающего устройства.

В отличие от вариантов осуществления на этих чертежах возможен также вариант, в котором память генератора изображений содержит базу 21 или 31 данных, заранее сохраненную в ней и снабженную информацией, показанной в таблице на фиг.3а. Альтернативно, эти данные можно считывать из запоминающего средства, такого как карта памяти или диск. Однако информация, показанная в таблице на фиг.3b, может передаваться к генератору изображений по линии связи без необходимости ее сохранения на каком-либо этапе в память генератора изображений.

Фиг.10 иллюстрирует обработку видеоизображения в соответствии с одним вариантом осуществления. В этом варианте осуществления процессор изображений осуществляет поиск изображения, следующего в строке, для обработки областей изображений, которые изменились, и только данные, связанные с этими изображениями, сохраняются в память или передаются для использования в генератор изображений.

В этом варианте осуществления процессор изображений поддерживает в памяти данные по опорным значениям областей изображения, обработанного последним, которое иллюстрируется позицией P1. Память таким образом содержит опорное значение REF каждой области на P1, сохраненное в памяти, во взаимосвязи с идентификатором области. В начале, т.е. перед обработкой первого видеоизображения, предполагается, что опорное значение, например, равно 0,

сохранено для каждой области.

Когда первое видеоизображение 100 поступает на обработку, осуществляются выборка и вычисление опорных значений, как раскрыто выше, для каждой области в изображении 100. Затем области изображения 100 одна за другой и вычисленные для них опорные значения сравниваются с опорными значениями соответствующих опорных значений на P1 в памяти. В связи с обработкой первого изображения 100 определяется, что опорные значения всех областей в изображении 100 отклоняются от опорных значений соответствующих областей на P1, поддерживаемой в памяти. Измененные значения изображения 100 указаны крестиком на фиг.10. Стрелка используется для иллюстрации того, что опорные значения всех областей изображения 100 сохранены в соответствующих областях P1, поддерживаемой в памяти. Опорные значения этих областей также сохраняются в памяти или передаются для использования в генератор изображений. Когда обрабатывается первое изображение, информация изображения, соответствующая опорным значениям измененных областей, еще не передана ранее и не сохранена в памяти, вот почему это делается на данной стадии.

Следующим подлежащим обработке изображением является изображение 101. После выборки и вычисления опорных значений следует сравнение, в котором опорные значения областей изображения 101 сравниваются одно за другим с опорными значениями соответствующих областей на P1, сохраненной в памяти. В данном примере видно, что только опорные значения областей изображения 101, помеченные крестиком на фиг.10, отклоняются от опорных значений соответствующих областей на P1. Иными словами, изменены только области, помеченные крестиком.

В этом случае только опорные значения измененных областей, помеченных крестиком, сохраняются в памяти или передаются вместе с индексом, указывающим, что рассматриваются области изображения 101. В то же время опорные значения измененных областей сохраняются в соответствующие области P1 для замены предыдущих опорных значений, поддерживаемых для этих областей.

Если проверка показывает, что информация изображения, соответствующая опорным значениям измененных областей, ранее не передавалась и не сохранялась в память, это осуществляется аналогично описанному в связи с приведенными выше вариантами осуществления.

Следующим на очереди является видеоизображение 102. Вновь осуществляются выборка и вычисление опорных значений для каждой области по отдельности. За этим следует сравнение, чтобы найти, какие области изображения изменились. Вновь это сравнение осуществляется с помощью P1, поддерживаемой в памяти и обновленной на этой стадии ранее обработанным изображением 101. Это сравнение показывает, что опорные значения на P1 и изображении 102 отклоняются друг от друга в отношении областей, помеченных крестиком на изображении фиг.10.

Это означает, что только опорные значения и идентификаторы областей измененных областей, помеченных крестиком, сохраняются в памяти или передаются вместе с индексом, показывающим, что рассматриваемыми областями являются области изображения 102. А в то же время опорные значения измененных областей сохраняются в соответствующие области на P1, поддерживаемой в памяти, чтобы заменить опорные значения, поддерживаемые ранее для этих областей.

Та же самая процедура повторяется для всех изображений в этом видео, причем следующим является изображение 103. Следовательно, только информация об

измененных областях будет сохраняться или направляться, что снижает емкость памяти, необходимую для хранения уплотненного видео, и, соответственно, полоса пропускания для передачи данных, необходимая для передачи.

5 Генератор изображения, который принимает данные, полученные, как показано на фиг.10, генерирует изображение путем поддержания в памяти видеоизображения, отображенного последним, и обновляет в это изображение только те области, для которых он принимает новые данные. В начале память поддерживает изображение 100. Когда он принимает данные, согласно которым индекс равен 101 и
10 область A1=REF 235, область A8=REF 5, он обновляет область A1, сохраненную в памяти, путем выработки для нее информации изображения, соответствующей опорному значению REF 235, и, аналогично, информации изображения, соответствующей опорному значению REF 5 для области A8. Информация изображения прочих областей остается неизменной.

15 Следует понимать, что вышеприведенное описание и сопровождающие чертежи предназначены только для иллюстрации изобретения. Специалист найдет очевидным, что изобретение можно изменять и модифицировать также и иными путями без отхода от объема охраны этого изобретения.

20

Формула изобретения

1. Процессор изображений, характеризующийся тем, что он выполнен с возможностью:

- 25 принимать (A) изображение;
разделять информацию принятого изображения на области (A1-An), имеющие заданный размер и уникальный идентификатор (ALUE) области;
получать (B) специфичные для области выборки (11) из информации изображения каждой области и вычислять (12) опорное значение (REF) для каждой области на
30 основе этих выборок;
сохранять (D, E) в память (16) или передавать (15) специфичные для области (A1-An) опорное значение (REF), вычисленное для области, и идентификатор (ALUE) области;
35 проверять (C, 13), запомнено ли уже в памяти (16) или передано (15) ли опорное значение (REF), полученное в результате вычисления, и соответствующая ему информация (INF) изображения; и
сохранять (E) в память (16) или передавать (15) вычисленное опорное значение (REF) и во взаимосвязи с ним соответствующую информацию (INF)
40 изображения, для которой вычислено это опорное значение, если на основе проверки опорное значение (REF) и соответствующая информация (INF) изображения не сохранены в память и не переданы ранее.

2. Процессор изображений по п.1, характеризующийся тем, что он является процессором изображений для видеоизображений и выполнен с возможностью
45 осуществлять специфичное для области сохранение в память (16) или передачу (15) вычисленного опорного значения (REF) и идентификатора (ALUE) области для упомянутой области (A1-An) путем сохранения также в память или путем передачи индекса, показывающего, к какому изображению относятся это опорное значение и этот идентификатор области.

50 3. Процессор изображений по п.1 или 2, характеризующийся тем, что он является процессором изображений для видеоизображений и выполнен с возможностью:
проверять, изменено ли опорное значение (REF), вычисленное для области, по

сравнению с опорным значением (REF) соответствующей области в предыдущем обработанном изображении; и

5 осуществлять сохранение в память или передачу упомянутого опорного значения (REF) и соответствующего идентификатора (ALUE) области и, аналогично, сохранение в память или передачу упомянутой информации (INF) изображения, только
если проверка показывает, что опорное значение (REF) области изменено по сравнению с предыдущим обработанным изображением.

4. Процессор изображений по п.1, характеризующийся тем, что процессор
10 изображений выполнен с возможностью вычислять (12) опорное значение (REF) путем вычисления среднего по выборкам, состоящим из значений отдельных пикселей области (A1-An).

5. Процессор изображений по п.1, характеризующийся тем, что процессор
15 изображений выполнен с возможностью получать выборки (11) и вычислять (12) опорное значение (REF) методом избыточной выборки.

6. Процессор изображений для генерирования изображений, выполненный с
возможностью:

20 поддерживать (H, I, J, K) в памяти (21) опорные значения (REF) и информацию (INF) изображения, соответствующую опорным значениям;

генерировать (22) на основе информации (INF) изображения часть изображения для области (A1-An) изображения, отличающийся тем, что выполнен с возможностью
принимать (H, 20) данные, включающие по меньшей мере идентификатор (ALUE)
области и опорное значение (REF);

25 проверять (I), включают ли принятые данные также информацию (INF) изображения;

если проверка показывает, что информация (INF) изображения включена,
сохранять (J) в память (21) принятое опорное значение (REF) и во взаимосвязи с ним
30 информацию (INF) изображения, и когда сохранение выполнено, выполнять
указанную генерацию на основе принятой информации (INF) изображения для
области (A1-An), указанной принятым идентификатором (ALUE) области; а

если проверка показывает, что информация изображения не включена,
извлекать (K) из памяти (21) на основе принятого опорного значения (REF)
35 информацию (INF) изображения, соответствующую этому опорному значению (REF); и
выполнять указанную генерацию (22) на основе информации (INF) изображения,
извлеченной из памяти (21), для области (A1-An) изображения, указанной принятым
идентификатором (ALUE) области.

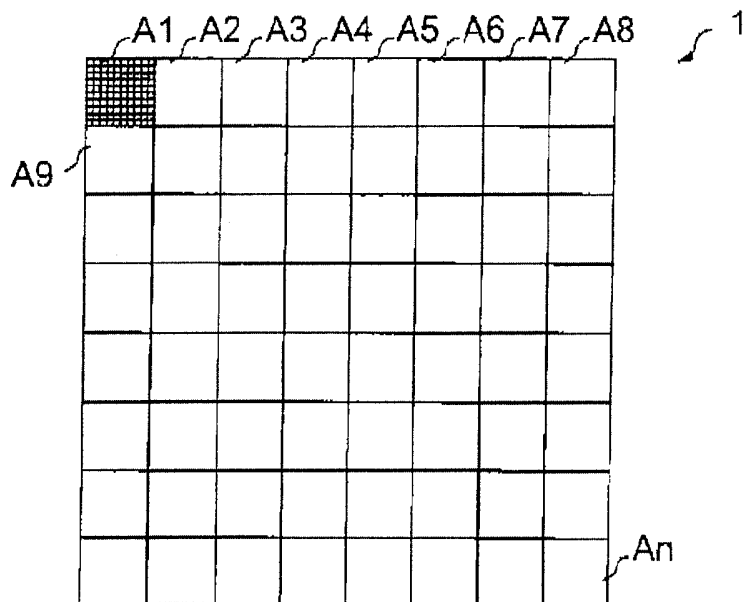
40 7. Процессор изображений по п.6, отличающийся тем, что генератор изображений
является генератором для видеоизображения и выполнен с возможностью принимать
также индекс во взаимосвязи с приемом (H) данных, причем указанный индекс
показывает, к какому изображению относятся указанное опорное значение (REF) и
указанный идентификатор (ALUE) области.

45 8. Процессор изображений по п.6, отличающийся тем, что генератор изображений
выполнен с возможностью генерировать видеоизображение путем сохранения в
памяти предыдущего сформированного изображения этого видео и генерировать
следующее изображение на основе изображения, сохраняемого в памяти, лишь на
50 основе изменения тех областей (A1-An) в предыдущем изображении, для которых
генератор изображений принимает идентификатор (ALUE) области и соответствующее
опорное значение (REF).

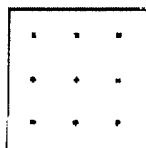
9. Машиночитаемый носитель данных, характеризующийся тем, что содержит

хранящуюся на нем компьютерную программу для обработки изображения, которая при выполнении обеспечивает управление компьютером для осуществления этапов, на которых:

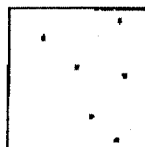
- 5 принимают (A) изображение;
- разделяют информацию этого изображения на области (A1-An), имеющие заданный размер и уникальный идентификатор (ALUE) области;
- получают (B) специфичные для области выборки (11) информации изображения каждой области и вычисляют (12) опорное значение (REF) для каждой области на
- 10 основе выборки;
- сохраняют (D, E) в память (16) или передают (15) специфичные для области (A1-An) опорное значение (REF), вычисленное для указанной области, и идентификатор (ALUE) области;
- 15 проверяют (C, 13), сохранено ли ранее в памяти (16) или передано (15) ли опорное значение (REF), полученное в результате вычисления, и соответствующая информация (INF) изображения; и
- сохраняют (E) в памяти (16) или передают (15) вычисленное опорное значение (REF) и во взаимосвязи с ним соответствующую информацию (INF) изображения, для
- 20 которой вычислено опорное значение, если согласно проверке опорное значение (REF), полученное на основе вычисления, и соответствующая информация (INF) изображения не сохранены ранее в памяти или не переданы.
10. Машиночитаемый носитель данных, характеризующийся тем, что содержит
- 25 хранящуюся на нем компьютерную программу для генерирования изображения, которая при выполнении обеспечивает управление компьютером для осуществления этапов, на которых:
- сохраняют (H, I, J, K) в памяти (21) опорные значения (REF) и информацию (INF) изображения, соответствующую опорным значениям;
- 30 принимают (H) данные, включающие, по меньшей мере, идентификаторы (ALUE) области и опорное значение (REF);
- проверяют (I), включают ли принятые данные также информацию (INF) изображения;
- если проверка показывает, что информация (INF) изображения включена,
- 35 сохраняют (J) в память (21) принятое опорное значение (REF) и во взаимосвязи с ним информацию (INF) изображения, и когда сохранение выполнено, генерируют часть изображения на основе принятой информации (INF) изображения для области (A1-An), указанной принятым идентификатором (ALUE) области; а
- 40 если проверка показывает, что информация изображения не включена,
- извлекают (K) из памяти (21) на основе опорного значения (REF) информацию (INF) изображения, соответствующую опорному значению (REF); и
- генерируют (22) для области (A1-An) изображения, указанной принятым идентификатором (ALUE) области, часть изображения на основе информации (INF)
- 45 изображения, извлеченной из памяти (21).



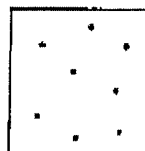
Фиг. 1



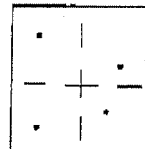
Фиг. 2а



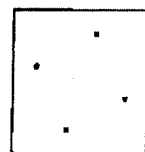
Фиг. 2b



Фиг. 2с



Фиг. 2d



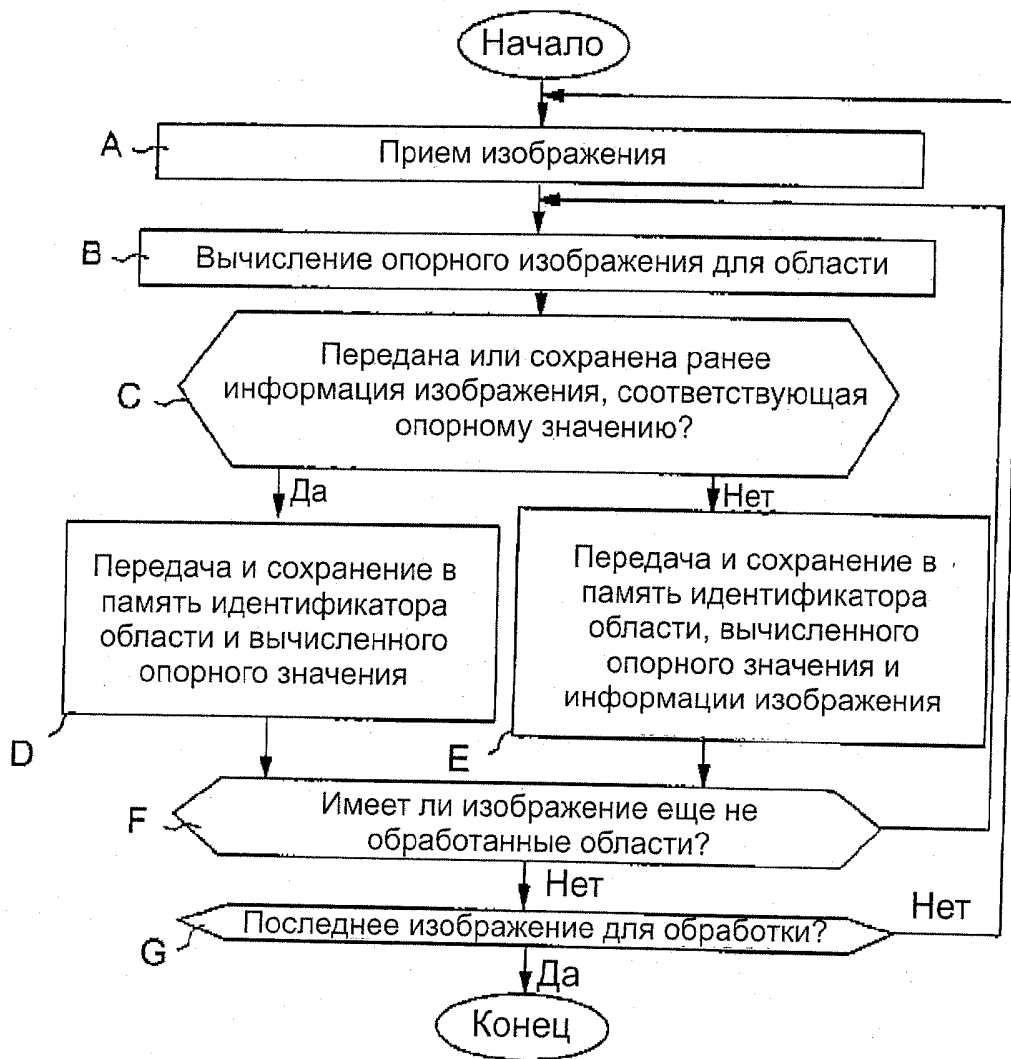
Фиг. 2е

Опорное значение	Информация изображения
1	DATA_A
2	DATA_B
3	DATA_C
4	DATA_D
5	DATA_E
6	DATA_F
n	DATA_G

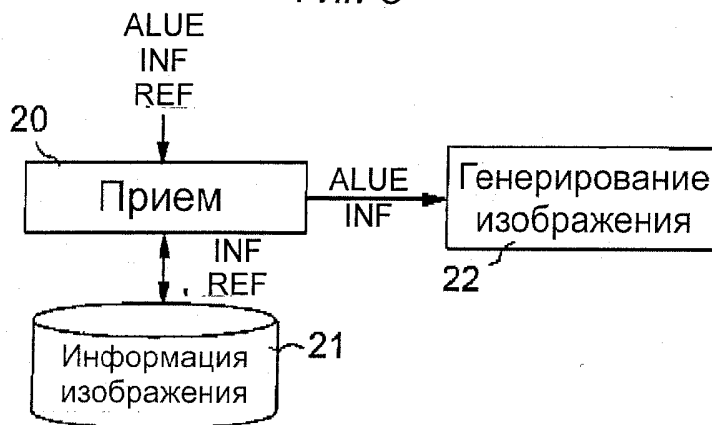
Фиг. 3а

ALUE	Опорное значение
A1	1
A2	253
A3	99
A4	253
A5	1
A6	14
An	253

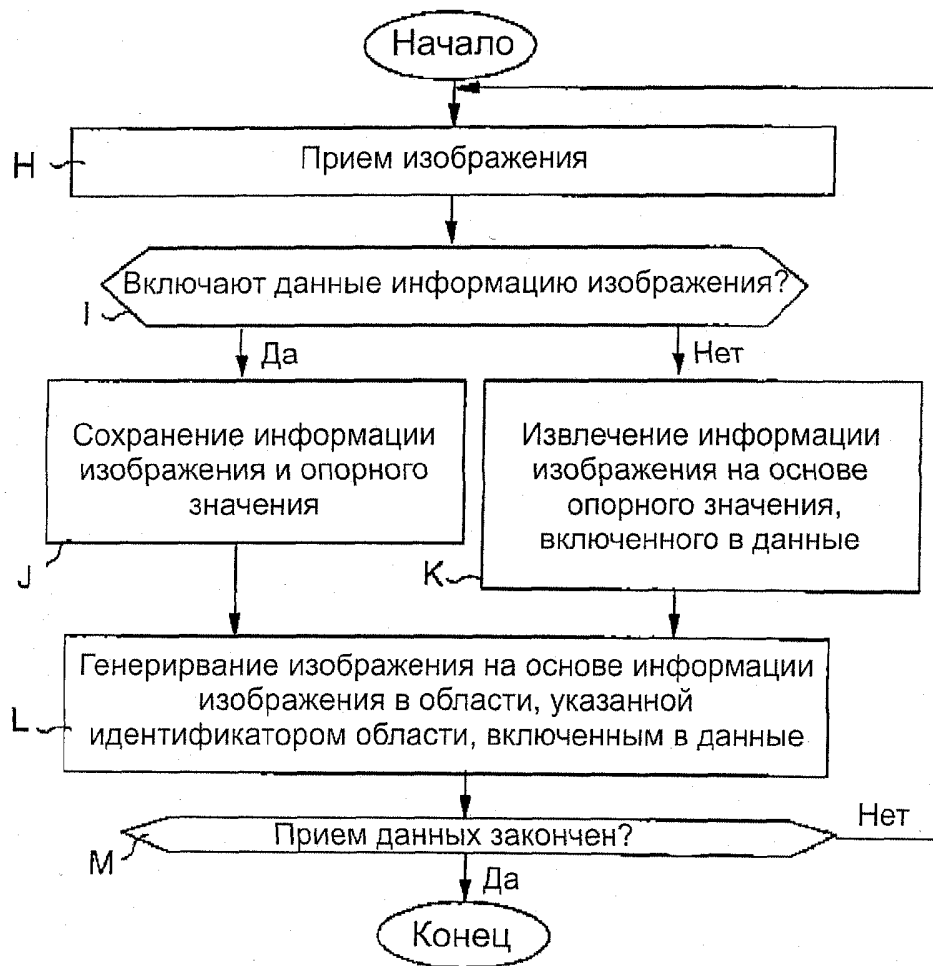
Фиг. 3б



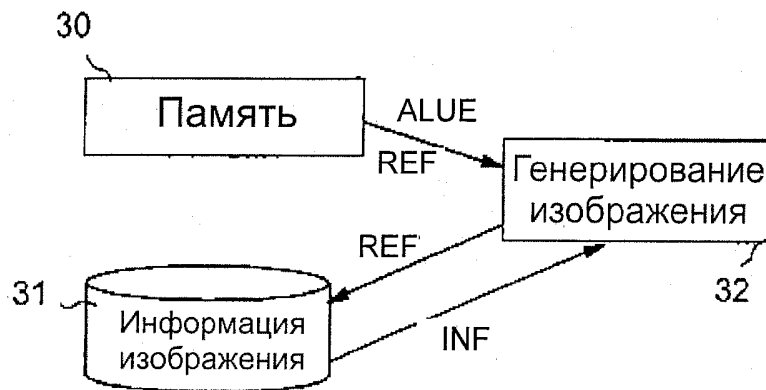
Фиг. 5



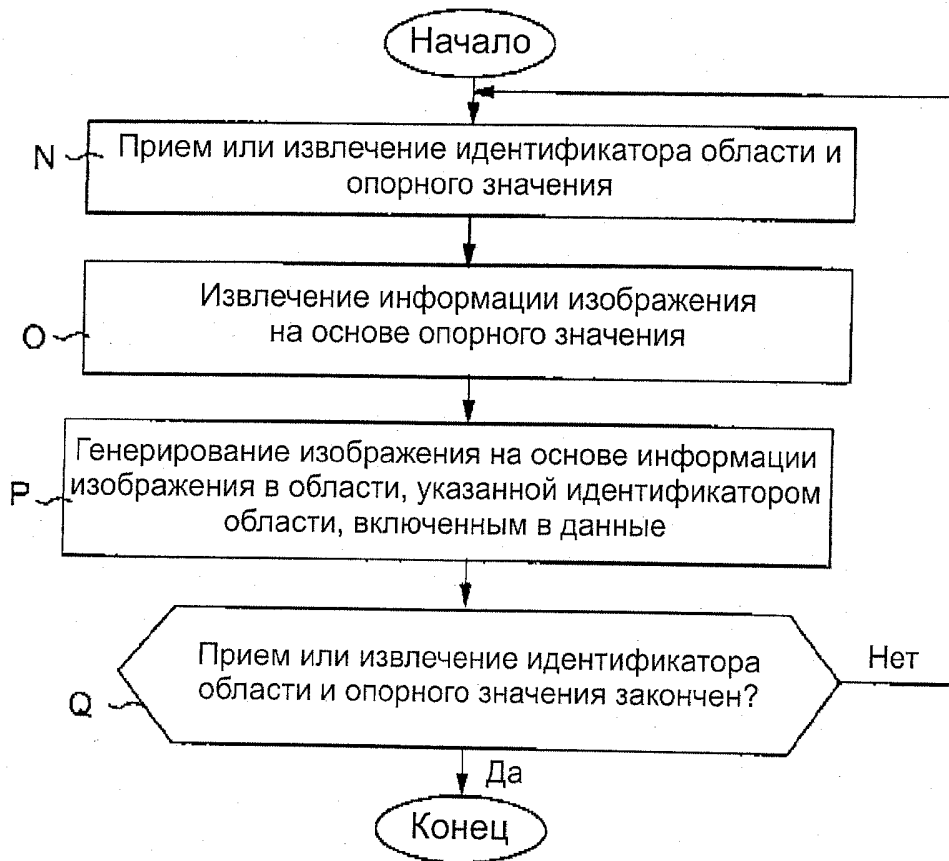
Фиг. 6



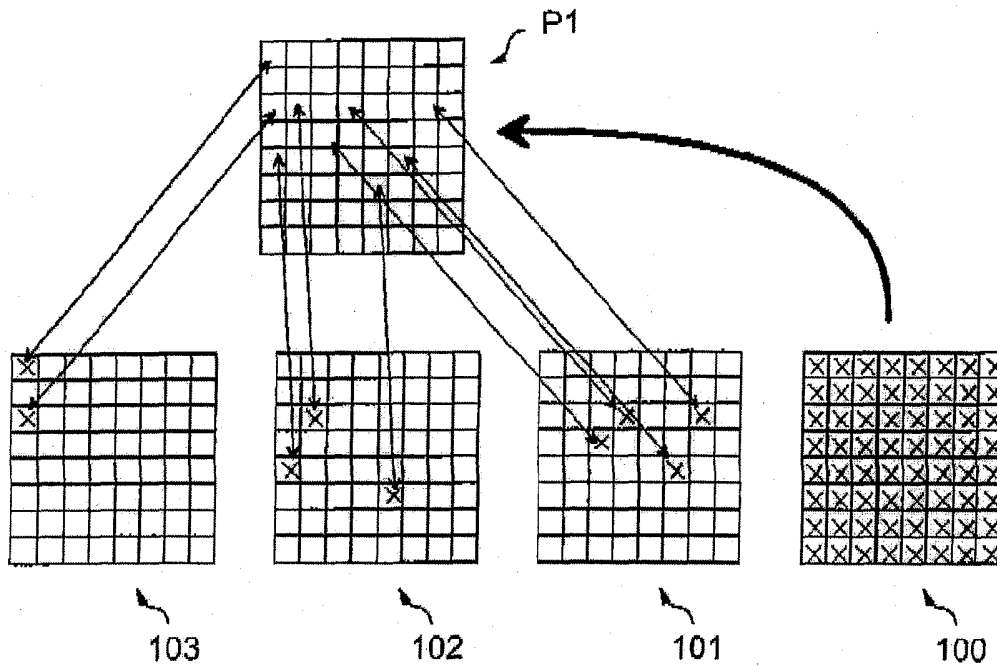
Фиг. 7



Фиг. 8



Фиг. 9



Фиг. 10