



УКРАЇНА

(19) **UA** (11) **115142** (13) **C2**  
(51) МПК

**H04N 19/13** (2014.01)  
**H04N 19/70** (2014.01)  
**H04N 19/463** (2014.01)  
**H04N 19/103** (2014.01)  
**H03M 7/30** (2006.01)

МІНІСТЕРСТВО  
ЕКОНОМІЧНОГО  
РОЗВИТКУ І ТОРГІВЛІ  
УКРАЇНИ

**(12) ОПИС ДО ПАТЕНТУ НА ВИНАХІД**

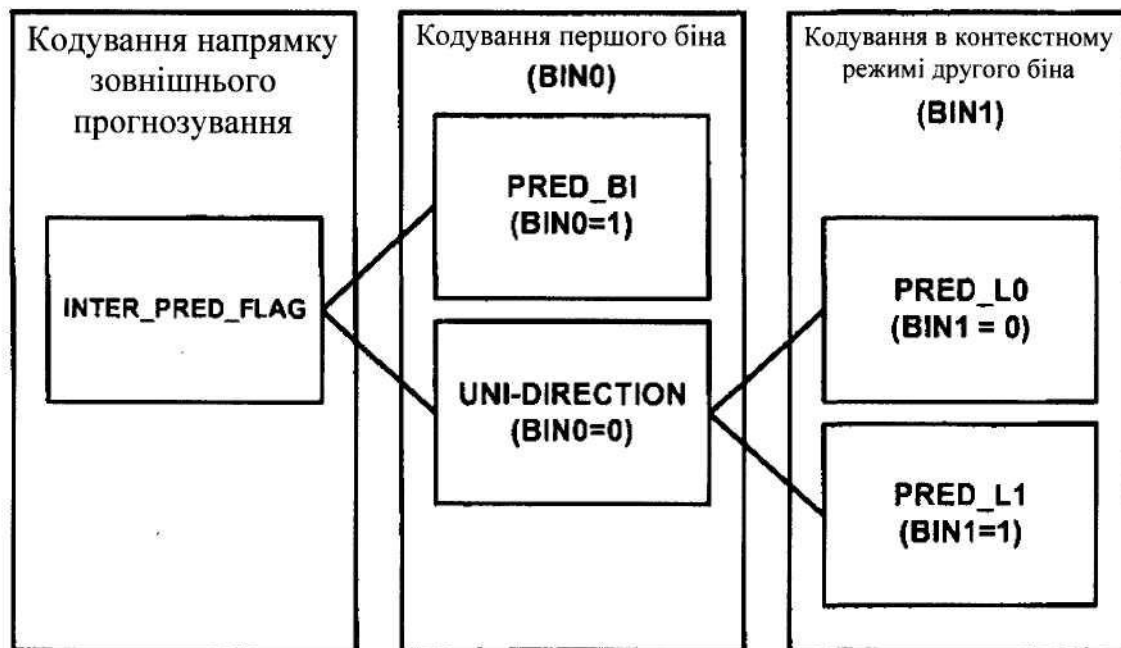
<b>(21)</b> Номер заявки:	<b>а 2014 12158</b>	<b>(72)</b> Винахідник(и):	<b>Карчєвіч Марта (US), Сєрьогін Вадім (US), Ван Сянлінь (US), Кобан Мухаммед Зейд (US)</b>
<b>(22)</b> Дата подання заявки:	<b>02.04.2013</b>	<b>(73)</b> Власник(и):	<b>КВЕЛКОММ ІНКОРПОРЕЙТЕД, Attn: International IP Administration, 5775 Morehouse Drive, San Diego, California 92121-1714, United States of America (US)</b>
<b>(24)</b> Дата, з якої є чинними права на винахід:	<b>25.09.2017</b>	<b>(74)</b> Представник:	<b>Мошинська Ніна Миколаївна, реєстр. №115</b>
<b>(31)</b> Номер попередньої заявки відповідно до Паризької конвенції:	<b>61/623,043, 61/637,218, 61/640,568, 61/647,422, 61/665,151, 13/828,173</b>	<b>(56)</b> Перелік документів, взятих до уваги експертизою:	<b>MARPE D ET AL, "Contetxt-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression Standard", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE SERVICE CENTER, PISCATAWAY, NJ, US, (20030701), vol. 13, no. 7, doi:10.1109/TCSVT.2003.815173, ISSN 1051- 8215, pages 620 - 636, XP002509017 [I] 1-91 * section II. The CABAC framework;; figure 1; table II * DOI: <a href="http://dx.doi.org/10.1109/TCSVT.2003.815173">http://dx.doi.org/10.1109/TCSVT.2003.815173</a></b>
<b>(32)</b> Дата подання попередньої заявки відповідно до Паризької конвенції:	<b>11.04.2012, 23.04.2012, 30.04.2012, 15.05.2012, 27.06.2012, 14.03.2013</b>		
<b>(33)</b> Код держави-учасниці Паризької конвенції, до якої подано попередню заявку:	<b>US, US, US, US, US, US</b>		
<b>(41)</b> Публікація відомостей про заявку:	<b>25.02.2015, Бюл.№ 4</b>		
<b>(46)</b> Публікація відомостей про видачу патенту:	<b>25.09.2017, Бюл.№ 18</b>		
<b>(86)</b> Номер та дата подання міжнародної заявки, поданої відповідно до Договору РСТ	<b>PCT/US2013/034968, 02.04.2013</b>		

**(54) ОБХІДНІ БІНИ ДЛЯ КОДУВАННЯ ОПОРНИХ ІНДЕКСІВ ПРИ КОДУВАННІ ВІДЕО**

**(57) Реферат:**

В одному з прикладів, аспекти даного розкриття винаходу стосуються способу для декодування синтаксичного елемента опорного індексу в процесі декодування відео, який включає в себе етап, на якому декодують щонайменше один бін значення опорного індексу в режимі контекстного кодування процесу контекстно-адаптивного двійкового арифметичного кодування (CABAC). Спосіб також включає в себе етапи, на яких декодують, якщо значення опорного індексу містить більше бінів, ніж щонайменше один бін, кодований у режимі контекстного кодування, щонайменше інший бін значення опорного індексу в режимі кодування з обходом процесу CABAC і бінаризують значення опорного індексу.

UA 115142 C2



Фіг. 8А

Дана заявка вимагає пріоритет за попередньою заявкою на патент США номер 61/623,043, поданою 11 квітня 2012, попередньою заявкою на патент США номер 61/637,218, поданою 23 квітня 2012, попередньою заявкою на патент США номер 61/640,568, поданою 30 квітня 2012, попередньою заявкою на патент США номер 61/647,422, поданою 15 травня 2012, і  
 5 попередньою заявкою на патент США номер 61/665,151, поданою 27 червня 2012, зміст кожної з яких включено в даний документ шляхом посилання, у всій їх повноті.

#### Галузь техніки

Дане розкриття винаходу стосується кодування відео, а конкретніше методів для кодування синтаксичних елементів у ході процесу кодування відео.

#### Рівень техніки

Можливості цифрового відеопредставлення можуть впроваджуватися в широкий різноманітності пристроїв, що включають у себе цифрові телевізори, цифрові системи безпосереднього мовлення, бездротові широкомовні системи, кишенькові персональні комп'ютери (КПК), переносні або настільні комп'ютери, планшетні комп'ютери, цифрові камери, цифрові пристрої запису, цифрові мультимедійні програвачі, відеоігрові пристрої, відеоігрові приставки, стільникові або супутникові радіотелефони, пристрої для відеоконференцзв'язку і тому подібні. Пристрої цифрового відеопредставлення реалізують такі методи стиснення відео, як описані в стандартах, визначуваних MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Частина 10, Удосконалене Кодування Відео (AVC-Advanced Video Coding), стандарті  
 20 Високоєфективне Відеокодування (HEVC-High Efficiency Video coding), що знаходиться на даний час у стадії розробки, і розширеннях таких стандартів, щоб більш ефективно передавати, приймати і зберігати цифрове відео.

Методи стиснення відео включають в себе просторове прогнозування і/або часове прогнозування, щоб зменшити або усунути надмірність, властиву відеопослідовностям. Для блокового кодування відео, відеокадр або слайс може бути розділений на блоки. Відеокадр, як альтернатива, може згадуватися як зображення. Кожен блок може бути додатково розділений. Блоки в зображенні або слайсі з внутрішнім кодуванням (I) кодуються з використанням просторового прогнозування відносно опорних вибірок у сусідніх блоках у одному і тому ж зображенні або слайсі. Блоки в зображенні або слайсі з зовнішнім кодуванням (P або B) можуть  
 30 використовувати просторове прогнозування відносно опорних вибірок у сусідніх блоках у одному і тому ж зображенні або слайсі або часове прогнозування відносно опорних вибірок в інших опорних зображеннях. Просторове або часове прогнозування дає в результаті прогнозуючий блок для блока, що підлягає кодуванню. Залишкові дані представляють різниці елементів зображення між вихідним блоком, що підлягає кодуванню, і прогнозуючим блоком.

Блок із зовнішнім кодуванням кодується відповідно до вектора руху, який вказує на блок опорних вибірок, що формують прогнозуючий блок, і залишкових даних, які показують різницю між кодованим блоком і прогнозуючим блоком. Блок із внутрішнім кодуванням кодується відповідно до режиму внутрішнього кодування і залишкових даних. Для додаткового стиснення, залишкові дані можуть бути перетворені з області елементів зображення в область  
 40 перетворення, що дає залишкові коефіцієнти перетворення, які потім можуть квантуватися. Квантовані коефіцієнти перетворення, спочатку розміщені в двовимірному масиві, можуть розгортатися в конкретному порядку, щоб виробити одновимірний вектор коефіцієнтів перетворення для ентропійного кодування. Ентропійне кодування може також застосовуватися до різних інших синтаксичних елементів, використовуваних у процесі кодування відео.

#### Розкриття винаходу

Методи в даному розкритті винаходу загалом стосуються ентропійного кодування відеоданих. Наприклад, при виконанні контекстно-адаптивного кодування, відеокодер може кодувати кожен біт або "бін" даних, використовуючи імовірнісні оцінки, які можуть показувати імовірність біна, що має дане двійкове значення. Імовірнісні оцінки можуть входити до складу імовірнісної моделі, також відомої як "контекстна модель". Відеокодер може вибирати контекстну модель, визначаючи контекст для біна. Контекст для біна може включати в себе значення пов'язаних бінів раніше закодованих синтаксичних елементів. Після кодування біна, відеокодер може оновлювати контекстну модель на основі значення біна, щоб відобразити найбільш останні імовірнісні оцінки. На відміну від застосування режиму контекстного  
 50 кодування, відеокодер може застосовувати режим кодування з обходом. Наприклад, відеокодер може використовувати режим з обходом, щоб обійти, або пропустити, нормальний процес арифметичного кодування. У таких випадках, відеокодер може використовувати фіксовану імовірнісну модель (яка не оновлюється в ході кодування), щоб кодувати з обходом біни.

Методи в даному розкритті винаходу стосуються ефективного контекстного кодування синтаксичних елементів, пов'язаних з відеоданими з зовнішнім кодуванням. Наприклад, аспекти

даного розкриття винаходу стосуються ефективного кодування значень опорних індексів, предикторів векторів руху, значень різниць векторів руху і тому подібного. У деяких випадках, відеокодер може виконувати контекстне кодування для деяких бінів синтаксичного елемента і кодування з обходом для інших бінів синтаксичного елемента. Наприклад, відеокодер може

5 контекстно кодувати один або більше бінів значення опорного індексу і кодувати з обходом один або більше інших бінів значення опорного індексу.

В одному з прикладів, аспекти даного розкриття винаходу стосуються способу для кодування синтаксичного елемента опорного індексу в процесі кодування відео, який включає в себе етапи, на яких бінаризують значення опорного індексу, кодують щонайменше один бін

10 бінаризованого значення опорного індексу в режимі контекстного кодування процесу контекстно-адаптивного двійкового арифметичного кодування (CABAC-Context-adaptive binary arithmetic coding) і кодують, якщо бінаризоване значення опорного індексу містить більше бінів, ніж щонайменше один бін, кодований у режимі контекстного кодування, щонайменше інший бін бінаризованого значення опорного індексу в режимі кодування з обходом процесу CABAC.

В іншому прикладі, аспекти даного розкриття винаходу стосуються пристрою для кодування синтаксичного елемента опорного індексу в процесі кодування відео, який містить один або більше процесорів, щоб бінаризувати значення опорного індексу, кодувати щонайменше один бін бінаризованого значення опорного індексу в режимі контекстного кодування процесу

20 контекстно-адаптивного двійкового арифметичного кодування (CABAC) і кодувати, якщо бінаризоване значення опорного індексу містить більше бінів, ніж щонайменше один бін, кодований у режимі контекстного кодування, щонайменше інший бін бінаризованого значення опорного індексу в режимі кодування з обходом процесу CABAC.

В іншому прикладі, аспекти даного розкриття винаходу стосуються пристрою для кодування синтаксичного елемента опорного індексу в процесі кодування відео, який містить засіб для бінаризації значення опорного індексу, засіб для кодування щонайменше одного біна бінаризованого значення опорного індексу в режимі контекстного кодування процесу

25 контекстно-адаптивного двійкового арифметичного кодування (CABAC) і засіб для кодування, якщо бінаризоване значення опорного індексу містить більше бінів, ніж щонайменше один бін, кодований у режимі контекстного кодування, щонайменше іншого біна бінаризованого значення опорного індексу в режимі кодування з обходом процесу CABAC.

В іншому прикладі, аспекти даного розкриття винаходу стосуються способу для декодування синтаксичного елемента опорного індексу в процесі декодування відео, який включає в себе етапи, на яких декодують щонайменше один бін значення опорного індексу в режимі

35 контекстного кодування процесу контекстно-адаптивного двійкового арифметичного кодування (CABAC), декодують, якщо значення опорного індексу містить більше бінів, ніж щонайменше один бін, кодований у режимі контекстного кодування, щонайменше інший бін значення опорного індексу в режимі кодування з обходом процесу CABAC і бінаризують значення опорного індексу.

В іншому прикладі, аспекти даного розкриття винаходу стосуються пристрою для декодування синтаксичного елемента опорного індексу в процесі декодування відео, який містить один або більше процесорів, виконаних з можливістю декодування щонайменше одного біна значення опорного індексу в режимі контекстного кодування процесу контекстно-

40 адаптивного двійкового арифметичного кодування (CABAC), декодування, якщо значення опорного індексу містить більше бінів, ніж щонайменше один бін, кодований у режимі контекстного кодування, щонайменше іншого біна значення опорного індексу в режимі кодування з обходом процесу CABAC і бінаризації значення опорного індексу.

В іншому прикладі, аспекти даного розкриття винаходу стосуються постійного зчитуваного комп'ютером носія, що зберігає на собі інструкції, які при виконанні змушують один або більше процесорів декодувати щонайменше один бін значення опорного індексу в режимі контекстного

50 кодування процесу контекстно-адаптивного двійкового арифметичного кодування (CABAC), декодувати, якщо значення опорного індексу містить більше бінів, ніж щонайменше один бін, кодований у режимі контекстного кодування, щонайменше інший бін значення опорного індексу в режимі кодування з обходом процесу CABAC і бінаризувати значення опорного індексу.

Деталі одного або більше прикладів наводяться на прикладених кресленнях і в нижченаведеному описі. Інші ознаки, задачі і переваги будуть очевидні з опису і креслень, а також з формули винаходу.

Короткий опис креслень

Фіг. 1 є структурною схемою, яка ілюструє типову систему кодування і декодування відео.

Фіг. 2 є структурною схемою, яка ілюструє типовий кодер відео.

60 Фіг. 3 є структурною схемою, яка ілюструє типовий декодер відео.

Фіг. 4 є структурною схемою, яка ілюструє типовий процес арифметичного кодування.

Фіг. 5A є структурною схемою, яка ілюструє типовий рядок даних прогнозування.

Фіг. 5B є структурною схемою, яка ілюструє інший типовий рядок даних прогнозування.

Фіг. 6 є структурною схемою, яка ілюструє інший типовий рядок даних прогнозування.

5 Фіг. 7 є структурною схемою, яка ілюструє інший типовий рядок даних прогнозування.

Фіг. 8A є структурною схемою, яка ілюструє контекстне кодування синтаксичного елемента напрямку зовнішнього прогнозування з трьома можливими значеннями.

Фіг. 8B є структурною схемою, яка ілюструє кодування з обходом синтаксичного елемента напрямку зовнішнього прогнозування, відповідно до аспектів даного розкриття винаходу.

10 Фіг. 9 є блок-схемою послідовності операцій способу, яка ілюструє приклад ентропійного кодування значення опорного індексу, відповідно до аспектів даного розкриття винаходу.

Фіг. 10 є блок-схемою послідовності операцій способу, яка ілюструє приклад ентропійного декодування значення опорного індексу, відповідно до аспектів даного розкриття винаходу.

15 Фіг. 11 є блок-схемою послідовності операцій способу, яка ілюструє приклад ентропійного кодування даних прогнозування, відповідно до аспектів даного розкриття винаходу.

Фіг. 12 є блок-схемою послідовності операцій способу, яка ілюструє приклад ентропійного декодування даних прогнозування, відповідно до аспектів даного розкриття винаходу.

#### Докладний опис

20 Пристрій кодування відео може стискати відеодані з використанням просторової і часової надмірності. Наприклад, кодер відео може використовувати просторову надмірність, кодуючи блок відносно сусідніх, раніше кодованих блоків. Аналогічно, кодер відео може використовувати часову надмірність, кодуючи блок відносно даних раніше кодованих зображень. Зокрема, кодер відео може прогнозувати поточний блок, виходячи з даних просторово сусідніх елементів (згадується як внутрішнє кодування) або з даних одного або більше інших зображень (згадується як зовнішнє кодування). Потім кодер відео може обчислити залишкову похибку для

25 блока як різницю між фактичними значеннями елементів зображення для блока і прогнозованими значеннями елементів зображення для блока. Відповідно, залишкова похибка для блока може включати в себе значення різниць по кожному елементу зображення в області елементів зображення (або просторовій області).

30 Відеокодер може виконати оцінку руху і компенсацію руху при зовнішньому прогнозуванні блока відеоданих. Наприклад, оцінка руху виконується на кодері відео і включає в себе обчислення одного або більше векторів руху. Вектор руху може показувати відхилення блока відеоданих у поточному зображенні відносно опорних вибірок опорного зображення. Опорна вибірка може бути блоком, для якого виявляється близька відповідність блоку, що підлягає

35 кодуванню, якщо говорити про різницю елементів зображення, що може бути визначено по сумі абсолютної різниці (SAD-sum of absolute difference), сумі квадрата різниці (SSD-sum of squared difference) або інших показниках різниці. Опорна вибірка може виявитися де завгодно в межах опорного зображення або опорного слайсу, і не обов'язково на границі блока опорного зображення або слайсу. У деяких прикладах, опорна вибірка може виявитися в позиції

40 фрагментованого елемента зображення.

Дані, що задають вектор руху, можуть описувати, наприклад, горизонтальну складову вектора руху, вертикальну складову вектора руху, розрізнення для вектора руху (наприклад, точність до однієї чверті елемента зображення або точність до однієї восьмої елемента зображення), опорне зображення, на яке вказує вектор руху, і/або список опорних зображень

45 (наприклад, список 0 (L0), список 1 (L1) або об'єднаний список (LC)) для вектора руху, наприклад, як показано напрямком прогнозування. Опорний індекс (ref\_idx) може ідентифікувати конкретне зображення в списку опорних зображень, на яке вказує вектор руху. Таким чином, синтаксичний елемент ref\_idx служить індексом у списку опорних зображень, наприклад L0, L1 або LC.

50 Після ідентифікації опорного блока, визначається різниця між вихідним блоком відеоданих і опорним блоком. Ця різниця може згадуватися як залишкові дані прогнозування і показує різницю елементів зображення між значеннями елементів зображення в блоці для кодування і значеннями елементів зображення в опорному блоці, вибраному для представлення кодованого блока. Для досягнення кращого стиснення, залишкові дані прогнозування можуть

55 бути перетворені, наприклад, з використанням дискретного косинусного перетворення (ДКП), цілочислового перетворення, перетворення Карунена-Лоева (K-L-Karhunen-Loeve) або іншого перетворення. Для додаткового стиснення, коефіцієнти перетворення можуть квантуватися.

Потім ентропійний кодер здійснює ентропійне кодування символів або синтаксичних елементів, пов'язаних з блоком відеоданих, і квантованих коефіцієнтів перетворення. Приклади

60 схем ентропійного кодування включають в себе контекстно-адаптивне кодування зі змінною

довжиною кодового слова (CAVLC-Context Adaptive Variable Length Coding), контекстно-адаптивне двійкове арифметичне кодування (CABAC), ентропійне кодування з розбивкою інтервалу імовірності (PIPE-Probability Interval Partitioning Entropy Coding) або тому подібне. Перед контекстним кодуванням, кодер відео може переводити абсолютне значення кожного значення, що підлягає кодуванню, у бінаризовану форму. Таким чином, кожне ненульове кодоване значення може бути "бінаризоване", наприклад, використовуючи унарну кодову таблицю або іншу схему кодування, яка переводить значення в кодове слово, що має один або більше бітів або "бінів".

Що стосується CABAC, як приклад, відеокoder може вибрати імовірнісну модель (також згадувану як контекстна модель) для кодових символів, пов'язаних із блоком відеоданих. Наприклад, на кодері, цільовий символ може бути закодований за допомогою імовірнісної моделі. На декодері, цільовий символ може піддаватися синтаксичному аналізу за допомогою імовірнісної моделі. У деяких випадках, біни можуть бути закодовані з використанням поєднання контекстно-адаптивного кодування і контекстно-незалежного адаптивного кодування. Наприклад, відеокoder може використовувати режим з обходом, щоб обійти, або пропустити, нормальний процес арифметичного кодування для одного або більше бінів, при використанні контекстно-адаптивного кодування для інших бінів. У таких прикладах, відеокoder може використовувати фіксовану імовірнісну модель, щоб кодувати з обходом біни. Тобто, кодовані з обходом біни не включають в себе оновлення контексту або імовірності. Взагалі, як більш докладно описано нижче з посиланням на Фіг. 4, контекстне кодування бінів може згадуватися як кодування бінів з використанням режиму контекстного кодування. Точно так само, кодування з обходом бінів може згадуватися як кодування бінів з використанням режиму кодування з обходом.

Контекстна модель для кодування біна синтаксичного елемента може бути основана на значеннях пов'язаних бінів раніше кодованих сусідніх синтаксичних елементів. Як один приклад, контекстна модель для кодування біна поточного синтаксичного елемента може бути основана на значеннях пов'язаних бінів раніше кодованих сусідніх синтаксичних елементів, наприклад вище і лівіше від поточного синтаксичного елемента. Позиції, з яких виводиться контекст, можуть згадуватися як окіл контекстної підтримки (також згадується як "контекстна підтримка" або просто "підтримка"). Наприклад, відносно кодування бінів карти значущості (наприклад, що показує місцеположення ненульових коефіцієнтів перетворення в блоці відеоданих), для задавання контекстної моделі може використовуватися п'ятиточкова підтримка.

У деяких прикладах, контекстна модель (Ctx) може бути індексом або зсувом, що застосовується для вибору одного з множини різних контекстів, кожний з яких може відповідати конкретній імовірнісній моделі. Отже, у будь-якому випадку, як правило, своя імовірнісна модель задається для кожного контексту. Після кодування біна, імовірнісна модель додатково оновлюється на основі значення біна, щоб відобразити найбільш останні імовірнісні оцінки для біна. Наприклад, імовірнісна модель може обслуговуватися як стан у кінцевому автоматі. Кожен конкретний стан може відповідати визначеному значенню імовірності. Наступний стан, що відповідає оновленню імовірнісної моделі, може залежати від значення поточного біна (наприклад, біна, кодованого в даний момент). Відповідно, на вибір імовірнісної моделі можуть впливати значення раніше кодованих бінів, оскільки ці значення показують, щонайменше частково, імовірність біна, що має дане значення. Процес контекстного кодування, описаний вище, може в широкому розумінні згадуватися як режим контекстно-адаптивного кодування.

Процес оновлення імовірності, описаний вище, може уповільнювати процес кодування. Наприклад, передбачимо, що два біни використовують одну і ту ж контекстну модель (наприклад, ctx(0)) для цілей контекстно-адаптивного кодування. У цьому прикладі, перший бін може використовувати ctx(0), щоб визначити імовірнісну модель для кодування. Значення першого біна впливає на імовірнісну модель, пов'язану з ctx(0). Відповідно, оновлення імовірності повинно бути виконане до кодування другого біна з використанням ctx(0). Таким чином, оновлення імовірності може привносити затримку в цикл кодування.

Що стосується кодування відео, як інший приклад, відеокoder може контекстно-адаптивно кодувати послідовність бінів (наприклад, bin(0), bin(1), ..., bin(n)) опорного індексу (ref\_idx). Як зазначалося вище, опорний індекс (ref\_idx) може ідентифікувати конкретне зображення в списку опорних зображень, на яке вказує вектор руху. Один опорний індекс (ref\_idx) може включати в себе, наприклад, до 15 бінів. Для пояснення передбачимо, що відеокoder виводить три контексти для кодування бінів і застосовує ці контексти, виходячи з номера кодованого біна (наприклад, позначеного за допомогою індексів ctx(0), ctx(1) і ctx(2) контекстів). Тобто, у цьому прикладі, відеокoder може використовувати ctx(0), щоб кодувати bin(0), ctx(1), щоб кодувати bin(1), і ctx(2), щоб кодувати інші біни (наприклад, bin(2)-bin(n)).

В описаному вище прикладі, третій контекст (ctx(2)) спільно використовується деякою кількістю бінів (наприклад, до 13 бінів). Використання однієї і тієї ж імовірнісної моделі, щоб кодувати bin(2)-bin(n), таким чином, може створювати затримку між послідовними циклами кодування. Наприклад, як зазначалося вище, неодноразовий виклик того самого контексту і очікування оновлення моделі після кожного біна може являти собою критичний параметр для продуктивності кодера.

Більше того, кореляція між bin(2) і bin(n) може бути недостатньою, щоб виправдувати часові й обчислювальні ресурси, пов'язані з оновленням імовірнісної моделі. Тобто, однією потенційною перевагою контекстно-адаптивного кодування є здатність адаптувати імовірнісну модель, основуючись на раніше кодованих бінах (якщо наявний один і той же контекст). Якщо значення першого біна, при цьому, має невелику залежність або вплив на значення наступного біна, може мати місце невелике підвищення ефективності, пов'язане з оновленням імовірності. Відповідно, біни, що демонструють низьку кореляцію, не можуть з такою ж користю застосовувати контекстно-адаптивне кодування, як біни з відносно більш високими кореляціями.

Аспекти даного розкриття винаходу стосуються ефективного контекстного кодування синтаксичних елементів, пов'язаних з відеоданими з зовнішнім кодуванням. Наприклад, аспекти даного розкриття винаходу стосуються ефективного кодування значень опорних індексів, предикторів векторів руху, значень різниць векторів руху і тому подібного. У деяких випадках, відеокодер може виконувати контекстне кодування для деяких бінів синтаксичного елемента і кодування з обходом для інших бінів синтаксичного елемента.

Звертаючись до конкретного прикладу кодування опорних індексів, описаного вище, відповідно до аспектів даного розкриття винаходу, відеокодер може застосовувати ctx(0) для bin(0), ctx(1) для bin(1), ctx(2) для bin(2) і може кодувати з обходом інші біни значення опорного індексу, не вимагаючи яких-небудь контекстів. Іншими словами, відеокодер може використовувати ctx(2) як контекст для САВАС-кодування bin(2) з бінаризованого значення опорного індексу, але може кодувати з обходом будь-які біни, що ідуть за bin(2).

Враховуючи, що значення опорних індексів можуть складати 15 бінів або більше у довжину, обмеження кількості бінів, які контекстно кодуються, таким чином, може привести до економії обчислювальних і/або часових ресурсів у порівнянні з контекстним кодуванням усіх бінів опорного індексу. Більше того, як зазначалося вище, кореляція між бітами значення опорного індексу може бути невисокою (наприклад, значення bin(3) зі значення опорного індексу може не забезпечувати корисне показання стосовно імовірності bin(4) зі значенням "1" або "0"), що зменшує вигоду від контекстного кодування. Відповідно, обсяг часових і обчислювальних ресурсів, зекономлених завдяки контекстному кодуванню меншої кількості бінів значення опорного індексу, можуть переважити підвищення ефективності кодування, пов'язане з контекстним кодуванням усіх бінів значення опорного індексу.

Інші аспекти даного розкриття винаходу загалом стосуються групування контекстно кодованих бінів і контекстно-незалежно кодованих бінів під час кодування. Наприклад, як зазначалося вище, деякі синтаксичні елементи можуть кодуватися з використанням поєднання контекстного кодування і кодування з обходом. Тобто, деякі синтаксичні елементи можуть мати один або більше бінів, що контекстно кодуються, і один або більше інших бінів, що кодуються з обходом.

Для прикладу передбачимо, що є два синтаксичних елементи, кожний з яких має контекстно кодовану частину (що включає в себе один або більше контекстно кодованих бінів) і кодовану з обходом частину (що включає в себе один або більше кодованих з обходом бінів). У цьому прикладі, відеокодер може кодувати контекстно кодовану частину першого синтаксичного елемента, потім кодовану з обходом частину першого синтаксичного елемента, потім контекстно кодовану частину другого синтаксичного елемента, потім кодовану з обходом частину другого синтаксичного елемента.

В описаному вище прикладі, відеокодер може три рази перемикатися між режимом контекстного кодування і режимом кодування з обходом, щоб кодувати два синтаксичних елементи. Наприклад, відеокодер перемикається між контекстним кодуванням і кодуванням з обходом після контекстно кодованих бінів першого синтаксичного елемента, після кодованих з обходом бінів першого синтаксичного елемента і після контекстно кодованих бінів другого синтаксичного елемента. Перемикання між контекстним кодуванням і кодуванням з обходом, таким чином, може бути неефективним в обчислювальному відношенні. Наприклад, перемикання між контекстним кодуванням і кодуванням з обходом може витрачати один або більше тактових циклів. Відповідно, перемикання між контекстним кодуванням і кодуванням з обходом для кожного елемента може привнести запізнювання, внаслідок переходів між контекстним кодуванням і кодуванням з обходом.

Аспекти даного розкриття винаходу включають в себе групування контекстно кодованих бінів і контекстно-незалежно кодованих бінів (наприклад, обхідних бінів) під час кодування. Наприклад, відносно описаного вище прикладу, відповідно до аспектів даного розкриття винаходу, відеокодер може кодувати контекстно кодовані біни першого синтаксичного

5 елемента, потім контекстно кодовані біни другого синтаксичного елемента, потім кодовані з обходом біни першого синтаксичного елемента, потім кодовані з обходом біни другого синтаксичного елемента. Відповідно, відеокодер тільки один раз переходить між режимом контекстного кодування і режимом кодування з обходом, наприклад між контекстно кодованими бінами і контекстно-незалежно кодованими бінами.

10 Групування бінів, таким чином, може зменшити частоту, з якою відеокодер перемикається між режимом контекстного кодування і режимом кодування з обходом. Відповідно, аспекти даного розкриття винаходу можуть зменшити запізнювання при кодуванні синтаксичних елементів, що включають в себе поєднання контекстно кодованих бінів і кодованих з обходом бінів. У деяких прикладах, як описано відносно Фіг. 5-8 нижче, біни, пов'язані з даними

15 прогнозування, можуть групуватися відповідно до методів даного розкриття винаходу. Наприклад, як описується в даному документі, дані прогнозування можуть, у загальному випадку, включати в себе дані, пов'язані з зовнішнім прогнозуванням. Наприклад, дані прогнозування можуть включати в себе дані, що показують значення опорних індексів, вектори руху, предиктори векторів руху, значення різниць векторів руху тощо.

20 Фіг. 1 є структурною схемою, яка ілюструє типову систему 10 кодування і декодування відео, що може бути виконана з можливістю кодування даних прогнозування відповідно до прикладів даного розкриття винаходу. Як показано на Фіг. 1, система 10 включає в себе пристрій-джерело 12, що передає закодоване відео на пристрій-адресат 14 через канал 16 зв'язку. Закодоване відео також може зберігатися на інформаційному носії 34 або обслуговуючому вузлі 36 для

25 збереження файлів і може бути доступне зі сторони пристрою-адресата 14 за необхідності. При збереженні на інформаційному носії або обслуговуючому вузлі для збереження файлів, кодер 20 відео може надавати кодовані відеодані на інший пристрій, такий як мережний інтерфейс, пристрій з можливістю запису або штампування компакт-диска (CD), диска Blu-ray або цифрового відеодиска (DVD) або інші пристрої, для збереження кодованого відео на

30 інформаційному носії. Точно так само, окремий від декодера 30 відео пристрій, такий як мережний інтерфейс, пристрій читання CD або DVD або тому подібне, може витягати кодовані відеодані з інформаційного носія і надавати витягнуті дані декодеру 30 відео. Пристрій-джерело 12 і пристрій-адресат 14 можуть містити будь-який із широкої різноманітності пристроїв, що включають у себе настільні комп'ютери, портативні (тобто переносні) комп'ютери, планшетні комп'ютери, телевізійні приставки, мікротелефонні трубки, наприклад так називані смартфони, телевізори, камери, пристрої відображення, цифрові мультимедійні програвачі, відеоігрові приставки або тому подібне. У багатьох випадках, такі пристрої можуть бути обладнані для бездротового зв'язку. Отже, канал 16 зв'язку може містити

40 бездротовий канал, дротовий канал або комбінацію бездротових і дротових каналів, придатних для передачі закодованих відеоданих. Аналогічно, обслуговуючий вузол 36 для збереження файлів може бути доступний зі сторони пристрою-адресата 14 через будь-яке стандартне інформаційне з'єднання, у тому числі з'єднання через мережу Інтернет. Це може включати в себе бездротовий канал (наприклад, Wi-Fi-з'єднання), дротове з'єднання (наприклад, DSL, кабельний модем і т. д.) або комбінацію обох, що є придатним для одержання доступу до

45 закодованих відеоданих, які зберігаються на обслуговуючому вузлі для збереження файлів. Методи для кодування даних прогнозування, відповідно до прикладів згідно з даним розкриттям винаходу, можуть застосовуватися до кодування відео з задачею підтримки будь-якого з множини мультимедійних додатків, таких як ефірні телепередачі, передачі кабельного телебачення, передачі супутникового телебачення, потокові передачі відео, наприклад через

50 мережу Інтернет, кодування цифрового відео для збереження на носії для збереження даних, декодування цифрового відео, збереженого на носії для збереження даних, або інші додатки. У деяких прикладах, система 10 може бути виконана з можливістю підтримки односторонньої або двосторонньої передачі відео, щоб підтримувати такі додатки, як відтворення потокового відео, відтворення записаного відео, мовлення відеоданих і/або відеотелефонія.

55 У прикладі на Фіг. 1, пристрій-джерело 12 включає в себе джерело 18 відео, кодер 20 відео, модулюючий/демодулюючий пристрій 22 і передавач 24. В пристрої-джерелі 12, джерело 18 відео може включати в себе таке джерело, як пристрій захоплення відео, такий як відеокамера, архів відео, що містить раніше захоплене відео, інтерфейс зовнішнього джерела відео, щоб приймати відео від постачальника інформаційної відеопродукції, і/або система комп'ютерної

60 графіки для генерування даних комп'ютерної графіки як вихідного відео або комбінація таких



джерел. Як один приклад, якщо джерелом 18 відео є відеокамера, пристрій-джерело 12 і пристрій-адресат 14 можуть утворювати так називані камерофони або відеотелефони. Проте, методи, описувані в даному розкритті винаходу, можуть бути застосовні до кодування відео загалом і можуть застосовуватися для бездротових і/або дротових додатків або додатка, у якому закодовані відеодані зберігаються на локальному диску.

Захоплене, попередньо захоплене або згенероване комп'ютером відео може бути закодоване кодером 20 відео. Закодоване відео може модулюватися модемом 22 згідно зі стандартом зв'язку, таким як протокол бездротового зв'язку, і передаватися на пристрій-адресат 14 за допомогою передавача 24. Модем 22 може включати в себе різні змішувачі, фільтри, підсилювачі або інші компоненти, призначені для модуляції сигналу. Передавач 24 може включати в себе схеми, призначені для передачі даних, у тому числі підсилювачі, фільтри, і одну або більше антен.

Захоплене, попередньо захоплене або згенероване комп'ютером відео, що закодоване кодером 20 відео, також може зберігатися на інформаційному носії 34 або на обслуговуючому вузлі 36 для збереження файлів для наступного використання. Інформаційний носій 34 може включати в себе диски Blu-ray, DVD, CD-ROM, пам'ять з груповим перезаписом або будь-які інші придатні цифрові інформаційні носії для збереження закодованого відео. Надалі до закодованого відео, збереженого на інформаційному носії 34, може одержати доступ пристрій-адресат 14 для декодування і відтворення.

Обслуговуючий вузол 36 для збереження файлів може бути обслуговуючим вузлом будь-якого типу, здатним зберігати закодоване відео і передавати це закодоване відео на пристрій-адресат 14. Типові обслуговуючі вузли для збереження файлів включають в себе мережний обслуговуючий вузол (наприклад, для інформаційного центра у всесвітній мережі), обслуговуючий вузол з доступом по протоколу стандарту FTP, пристрій збереження даних, що підключається до мережі (NAS-network attached storage), локальний дисковий накопичувач або пристрій будь-якого іншого типу, здатний зберігати закодовані відеодані і передавати їх на пристрій-адресат. Передача закодованих відеоданих від обслуговуючого вузла 36 для збереження файлів може бути потоковою передачею, передачею завантаження або їх комбінацією. Обслуговуючий вузол 36 для збереження файлів може бути доступний для пристрою-адресата 14 через будь-яке стандартне інформаційне з'єднання, у тому числі з'єднання через мережу Інтернет. Це може включати в себе бездротовий канал (наприклад, Wi-Fi-з'єднання), дротове з'єднання (наприклад, DSL, кабельний модем, Ethernet, USB і т. д.) або комбінацію обох, що є придатним для одержання доступу до закодованих відеоданих, які зберігаються на обслуговуючому вузлі для збереження файлів.

Пристрій-адресат 14, у прикладі на Фіг. 1, включає в себе приймач 26, модем 28, декодер 30 відео і пристрій 32 відображення. Приймач 26 в пристрої-адресаті 14 приймає інформацію по каналу 16, а модем 28 демодулює інформацію, щоб одержати демодульований бітовий потік для декодера 30 відео. Інформація, повідомлена по каналу 16, може включати в себе різну синтаксичну інформацію, згенеровану кодером 20 відео, для використання декодером 30 відео при декодуванні відеоданих. Така синтаксична структура також може бути віднесена до закодованих відеоданих, що зберігаються на інформаційному носії 34 або обслуговуючому вузлі 36 для збереження файлів. І кодер 20 відео, і декодер 30 відео можуть утворювати частину відповідного кодера-декодера (КОДЕК), який здатний кодувати або декодувати відеодані.

Пристрій 32 відображення може бути об'єднаний з пристроєм-адресатом 14 або бути зовнішнім відносно нього. У деяких прикладах, пристрій-адресат 14 може включати в себе вбудований пристрій відображення, а також може бути виконаний з можливістю взаємодії з зовнішнім пристроєм відображення. В інших прикладах, пристрій-адресат 14 може бути пристроєм відображення. У загальному випадку, пристрій 32 відображення відображає декодоване відео користувачу і може містити будь-який з різноманітних пристроїв відображення, таких як рідкокристалічний (РК) пристрій відображення, плазмовий пристрій відображення, пристрій відображення на органічних світловипромінюючих діодах (ОСВД) або пристрій відображення іншого типу.

У прикладі на Фіг. 1, канал 16 зв'язку може містити будь-яке бездротове або дротове середовище зв'язку, таке як радіочастотний (РЧ) спектр або одна або більше фізичних ліній передачі, або будь-яка комбінація бездротових і дротових передавальних середовищ. Канал 16 зв'язку може бути частиною пакетної мережі, такої як локальна обчислювальна мережа, глобальна обчислювальна мережа або всесвітня мережа, така як мережа Інтернет. Канал 16 зв'язку звичайно являє собою будь-яке придатне середовище зв'язку або набір різних середовищ зв'язку для передачі відеоданих від пристрою-джерела 12 на пристрій-адресат 14, у тому числі будь-яку відповідну комбінацію дротових або бездротових передавальних

середовищ. Канал 16 зв'язку може включати в себе маршрутизатори, перемикачі, базові станції або будь-яке інше обладнання, яке може використовуватися для полегшення зв'язку від пристрою-джерела 12 до пристрою-адресата 14.

Кодер 20 відео і декодер 30 відео можуть працювати згідно зі стандартом стиснення відео, таким як стандарт Високоєфективного Відеокодування (HEVC), що знаходиться на даний час у стадії розробки, і може узгоджуватися з тестовою моделлю HEVC (HM-HEVC Test Model). Як альтернатива, кодер 20 відео і декодер 30 відео можуть працювати відповідно до інших корпоративних або промислових стандартів, таких як стандарт ITU-T H.264, інакше іменований MPEG-4, Частина 10, Удосконалене Кодування Відео (AVC), або розширеннями таких стандартів. Методи згідно з даним розкриттям винаходу, однак, не обмежуються якимось конкретним стандартом кодування. Інші приклади включають в себе MPEG-2 і ITU-T H.263.

Хоч це і не показано на Фіг. 1, у деяких аспектах, кодер 20 відео і декодер 30 відео, кожний, можуть бути об'єднані з кодером і декодером аудіо і можуть включати в себе належні модулі мультимплексування-демультиплексування або інше апаратне і програмне забезпечення, щоб керувати кодуванням як аудіо, так і відео у загальному потоці даних або в окремих потоках даних. За необхідності, у деяких прикладах, модулі мультимплексування-демультиплексування можуть узгоджуватися з протоколом пристрою мультимплексування стандарту ITU H.223 або іншими протоколами, такими як протокол користувацьких дейтаграм (UDP-user datagram protocol).

І кодер 20 відео, і декодер 30 відео можуть бути реалізовані у вигляді будь-якої з множини придатних схем кодера або декодера, залежно від конкретного випадку, що включає в себе процесор, такий як один або більше мікропроцесорів, цифрових сигнальних процесорів (ЦСП), спеціалізованих процесорів або схем обробки даних, спеціалізованих інтегральних схем (СІС), програмованих користувачем вентильних матриць (ПКВМ), незмінних логічних схем, дискретних логічних схем, програмне забезпечення, апаратне забезпечення, програмно-апаратне забезпечення або будь-які їх комбінації. Відповідно, різні модулі в складі кодера 20 відео і декодера 30 відео аналогічним чином можуть бути реалізовані за допомогою будь-якого з множини таких структурних елементів або їх комбінацій. Якщо методи реалізуються частково в програмному забезпеченні, пристрій може зберігати інструкції для програмного забезпечення на придатному постійному зчитуваному комп'ютером носії і виконувати інструкції в апаратному забезпеченні, використовуючи один або більше процесорів, щоб виконувати методи згідно з даним розкриттям винаходу. І кодер 20 відео, і декодер 30 відео можуть входити до складу одного або більше кодерів або декодерів, кожний з яких може складати частину комбінованого кодера/декодера (КОДЕК) у відповідному пристрої.

Дане розкриття винаходу може, у загальному випадку, стосуватися кодера 20 відео, "сигналізуючого" визначену інформацію іншому пристрою, такому як декодер 30 відео. Потрібно розуміти, однак, що кодер 20 відео може сигналізувати інформацію, зв'язуючи деякі синтаксичні елементи з різними закодованими частинами відеоданих. Тобто, кодер 20 відео може "сигналізувати" дані, зберігаючи визначені синтаксичні елементи в заголовках різних закодованих частин відеоданих. У деяких випадках, такі синтаксичні елементи можуть бути закодовані і збережені (наприклад, збережені на пристрій 32 збереження) до того, як будуть прийняті і декодовані декодером 30 відео. Таким чином, термін "сигналізація" взагалі може стосуватися повідомлення синтаксичної структури або інших даних для декодування стиснутих відеоданих, чи відбувається такий зв'язок у реальному або майже реальному часі або протягом деякого проміжку часу, що могло б відбуватися при збереженні синтаксичних елементів на носій під час кодування, які потім можуть бути витягнуті декодуючим пристроєм у будь-який момент після збереження на цьому носії.

Як зазначалося вище, JCT-VC працює над розробкою стандарту HEVC. Заходи по стандартизації HEVC ґрунтуються на моделі, що розвивається, пристрою кодування відео, яка згадується як тестова модель HEVC (HM). HM передбачає декілька додаткових можливостей для пристрою кодування відео в порівнянні з існуючими пристроями, наприклад, що відповідають вимогам, ITU-T H.264/AVC. У даному розкритті винаходу звичайно використовується термін "відеоблок" для позначення кодового вузла в CU. У деяких особливих випадках, у даному розкритті винаходу ще може використовуватися термін "відеоблок" для позначення блока дерева, тобто LCU, або CU, що включає в себе кодовий вузол, а також PU і TU.

Відеопослідовність звичайно включає в себе ряд відеокадрів або зображень. Група зображень (GOP-group of pictures) у загальному випадку містить ряд з одного або більше відеозображень. GOP може включати в себе синтаксичні дані в заголовку GOP, заголовку одного або більше зображень або в іншій ділянці, які описують множину зображень, включених

до складу GOP. Кожен слайс зображення може включати в себе синтаксичні дані слайсу, що описують режим кодування для відповідного слайсу. Кодер 20 відео звичайно працює з відеоблоками в межах окремих відеослайсів для того, щоб кодувати відеодані. Відеоблок може відповідати кодовому вузлу в межах CU. Відеоблоки можуть мати постійні або змінні розміри і

5 можуть відрізнятися за розміром відповідно до визначеного стандарту кодування.

Як приклад, і як зазначалося вище, НМ підтримує прогнозування для PU різних розмірів (також згадуються як типи PU). Передбачаючи, що розмір конкретної CU складає  $2N \times 2N$ , НМ підтримує внутрішнє прогнозування в PU з розмірами  $2N \times 2N$  або  $N \times N$  і зовнішнє прогнозування в симетричних PU з розмірами  $2N \times 2N$ ,  $2N \times N$ ,  $N \times 2N$  або  $N \times N$ . НМ також підтримує асиметричне

10 розділення для зовнішнього прогнозування в PU з розмірами  $2N \times nU$ ,  $2N \times nD$ ,  $nL \times 2N$  і  $nR \times 2N$ . При асиметричному розділенні, один напрямок CU не розділяється, тоді як інший напрямок розділяється на 25 % і 75 %. Частина CU, що відповідає 25 %-ій частині, позначається "n", за чим іде позначення "Up" (вверх), "Down" (вниз), "Left" (вліво) або "Right" (вправо). Таким чином, наприклад, " $2N \times nU$ " стосується CU  $2N \times 2N$ , що розділяється по горизонталі з PU  $2N \times 0,5N$

15 вверху і PU  $2N \times 1,5N$  вниз. Інші типи розділення теж можливі.

У даному розкритті винаходу, " $N \times N$ " і " $N$  на  $N$ " можуть використовуватися взаємозамінно для позначення розмірностей в елементах зображення відеоблока в значеннях вертикальної і горизонтальної розмірностей, наприклад  $16 \times 16$  елементів зображення або 16 на 16 елементів зображення. У загальному випадку, блок  $16 \times 16$  буде мати 16 елементів зображення у

20 вертикальному напрямку ( $y=16$ ) і 16 елементів зображення в горизонтальному напрямку ( $x=16$ ). Точно так само, блок  $N \times N$  у загальному значенні має  $N$  елементів зображення у вертикальному напрямку і  $N$  елементів зображення в горизонтальному напрямку, де  $N$  являє собою ненегативне ціле число. Елементи зображення в блоці можуть розташовуватися в рядках і стовпцях. Більше того, блоки не обов'язково повинні мати таку ж кількість елементів

25 зображення в горизонтальному напрямку, як і у вертикальному напрямку. Наприклад, блоки можуть містити  $N \times M$  елементів зображення, де  $M$  не обов'язково дорівнює  $N$ .

Слідом за кодуванням із внутрішнім прогнозуванням або зовнішнім прогнозуванням, з використанням PU з CU, кодер 20 відео може обчислити залишкові дані для TU з CU. PU можуть містити дані елементів зображення в просторовій області (також називаній областю

30 елементів зображення), а TU можуть містити коефіцієнти в області перетворення з наступним застосуванням перетворення, наприклад дискретного косинусного перетворення (ДКП), цілочислового перетворення, вейвлет-перетворення або концептуально подібного перетворення, до залишкових відеоданих. Залишкові дані можуть відповідати різницям елементів зображення між елементами зображення незакодованого, вихідного зображення і

35 значеннями прогнозування, що відповідають PU. Кодер 20 відео може формувати TU, що включають у себе залишкові дані для CU, а потім перетворювати TU, щоб одержати коефіцієнти перетворення для CU.

У деяких прикладах, як зазначалося вище, TU можуть задаватися згідно з RQT. Наприклад, RQT може представляти спосіб, яким перетворення (наприклад, ДКП, целочислове

40 перетворення, вейвлет-перетворення або одне або більше інших перетворень) застосовуються до вибірок залишкової яскравості і вибірок залишкової кольоровості, пов'язаних із блоком відеоданих. Тобто, як зазначалося вище, залишкові вибірки, що відповідають CU, можуть розбиватися на менші модулі з використанням RQT. У загальному випадку, RQT є рекурсивним представленням розділення CU на TU.

Слідом за застосуванням тих або інших перетворень до залишкових даних для одержання коефіцієнтів перетворення, кодер 20 відео може виконати квантування коефіцієнтів

45 перетворення. Під квантуванням, у загальному випадку, мається на увазі процес, у якому коефіцієнти перетворення квантуються, щоб можна було зменшити обсяг даних, використовуваних для представлення коефіцієнтів, забезпечуючи додаткове стиснення. Процес квантування може зменшити бітову глибину, пов'язану з деякими або всіма коефіцієнтами.

50 Наприклад,  $n$ -бітове значення може округлятися в меншу сторону до  $m$ -бітового значення в ході квантування, де  $n$  більше  $m$ .

У деяких прикладах, кодер 20 відео може задіяти попередньо заданий порядок розгорнення для розгорнення квантованих коефіцієнтів перетворення, щоб одержати упорядкований вектор,

55 що може бути ентропійно закодований. В інших прикладах, кодер 20 відео може виконувати адаптивне розгорнення. Після розгорнення квантованих коефіцієнтів перетворення для формування одновимірної вектора, кодер 20 відео може ентропійно кодувати одновимірний вектор, наприклад, відповідно до контекстно-адаптивного кодування зі змінною довжиною кодового слова (CAVLC), контекстно-адаптивного двійкового арифметичного кодування

60 (CABAC), оснований на синтаксисі контекстно-адаптивного двійкового арифметичного

кодуванням (SBAC-syntax-based context-adaptive binary arithmetic coding), ентропійного кодування з розбивкою інтервалу імовірності (PIPE) або іншої методики ентропійного кодування. Кодер 20 відео також може ентропійно кодувати синтаксичні елементи, пов'язані з закодованими відеоданими, для використання декодером 30 відео при декодуванні відеоданих.

5 Поточна версія HEVC розрахована на використання CABAC для ентропійного кодування.

У деяких прикладах, кодер 20 відео може кодувати синтаксичні елементи, використовуючи поєднання контекстно-адаптивного кодування і контекстно-незалежного адаптивного кодування. Наприклад, кодер 20 відео може контекстно кодувати біни, вибираючи імовірнісну модель або "контекстну модель", що працює з контекстом, щоб закодувати біни. І навпаки, кодер 20 відео

10 може кодувати з обходом біни, обходячи або пропускаючи нормальний процес арифметичного кодування при кодуванні бінів. У таких прикладах, кодер 20 відео може використовувати фіксовану імовірнісну модель, щоб кодувати з обходом біни.

Як зазначалося вище, процес оновлення імовірнісної моделі, пов'язаний з контекстним кодуванням, може привносити затримку в процес кодування. Наприклад, кодер 20 відео може

15 контекстно кодувати послідовність бінів (наприклад,  $\text{bin}(0)$ ,  $\text{bin}(1)$ , ...,  $\text{bin}(n)$ ) опорного індексу ( $\text{ref\_idx}$ ). Один опорний індекс ( $\text{ref\_idx}$ ) може включати в себе, наприклад, до 15 бінів. Для пояснення передбачимо, що кодер 20 відео виводить три контексти для кодування бінів і застосовує контексти, виходячи з номера кодованого біна (наприклад, позначеного за допомогою індексів  $\text{ctx}(0)$ ,  $\text{ctx}(1)$  і  $\text{ctx}(2)$  контекстів). Тобто, у цьому прикладі, кодер 20 відео

20 може використовувати  $\text{ctx}(0)$ , щоб кодувати  $\text{bin}(0)$ ,  $\text{ctx}(1)$ , щоб кодувати  $\text{bin}(1)$ , і  $\text{ctx}(2)$ , щоб кодувати інші біни (наприклад,  $\text{bin}(2)$ - $\text{bin}(n)$ ).

В описаному вище прикладі, третій контекст ( $\text{ctx}(2)$ ) спільно використовується деякою кількістю бінів (наприклад, до 13 бінів). Використання однієї і тієї ж імовірнісної моделі, щоб кодувати  $\text{bin}(2)$ - $\text{bin}(n)$ , таким чином, може створювати затримку між послідовними циклами

25 кодування. Наприклад, неодноразовий виклик одного і того ж контексту і очікування оновлення моделі після кожного біна може являти собою критичний параметр для продуктивності кодера.

Більше того, кореляція між  $\text{bin}(2)$  і  $\text{bin}(n)$  може бути недостатньою, щоб виправдувати часові й обчислювальні ресурси, пов'язані з оновленням імовірнісної моделі. Тобто, однією потенційною перевагою контекстно-адаптивного кодування є здатність адаптувати імовірнісну

30 модель, основуючись на раніше кодованих бінах (якщо наявний один і той же контекст). Якщо значення першого біна, при цьому, має невелику залежність або вплив на значення наступного біна, може мати місце невелике підвищення ефективності, пов'язане з оновленням імовірності. Відповідно, біни, що демонструють низьку кореляцію, не можуть з такою ж користю застосовувати контекстно-адаптивне кодування, як біни з відносно більш високими кореляціями.

Відповідно до аспектів даного розкриття винаходу, кодер 20 відео може закодувати синтаксичний елемент опорного індексу, кодуючи щонайменше один бін бінаризованого значення опорного індексу в процесі контекстно-адаптивного двійкового арифметичного кодування (CABAC) і кодуючи щонайменше інший бін бінаризованого значення опорного індексу в режимі кодування з обходом процесу контекстно-адаптивного двійкового арифметичного

40 кодування (CABAC).

В одному з ілюстративних прикладів, кодер 20 відео може застосовувати  $\text{ctx}(0)$  для  $\text{bin}(0)$ ,  $\text{ctx}(1)$  для  $\text{bin}(1)$ ,  $\text{ctx}(2)$  для  $\text{bin}(2)$  і може кодувати з обходом інші біни значення опорного індексу, не вимагаючи яких-небудь контекстів. Іншими словами, відеокодер може використовувати  $\text{ctx}(2)$  як контекст для CABAC-кодування  $\text{bin}(2)$  з бінаризованого значення опорного індексу, але може

45 кодувати з обходом будь-які біни, що ідуть за  $\text{bin}(2)$ .

Враховуючи, що значення опорних індексів можуть складати 15 бінів або більше у довжину, обмеження кількості бінів, що контекстно кодуються, таким чином, може привести до економії обчислювальних і/або часових ресурсів у порівнянні з контекстним кодуванням усіх бінів опорного індексу. Більше того, як зазначалося вище, кореляція між бітами значення опорного індексу може бути невисокою (наприклад, значення  $\text{bin}(3)$  зі значення опорного індексу може не забезпечувати корисне показання стосовно імовірності  $\text{bin}(4)$  зі значенням "1" або "0"), що зменшує вигоду від контекстного кодування. Відповідно, обсяг часових і обчислювальних ресурсів, зекономлених завдяки контекстному кодуванню меншої кількості бінів значення опорного індексу, можуть переважити підвищення ефективності кодування, пов'язане з

55 контекстним кодуванням усіх бінів значення опорного індексу.

Відповідно до інших аспектів даного розкриття винаходу, кодер 20 відео може групувати контекстно кодовані біни і контекстно-незалежно кодовані біни під час кодування. Наприклад, як зазначалося вище, деякі синтаксичні елементи можуть кодуватися з використанням поєднання контекстного кодування і кодування з обходом. Тобто, деякі синтаксичні елементи можуть мати

один або більше бінів, що контекстно кодуються, і один або більше інших бінів, що кодуються з обходом.

У деяких прикладах, кодер 20 відео може перемикатися між контекстним кодуванням і кодуванням з обходом, щоб кодувати послідовність синтаксичних елементів. Однак перемикання між контекстним кодуванням і кодуванням з обходом може витратити один або більше тактових циклів. Відповідно, перемикання між контекстним кодуванням і кодуванням з обходом для кожного елемента може привнести запізнювання, внаслідок переходів між контекстним кодуванням і кодуванням з обходом.

Відповідно до аспектів даного розкриття винаходу, кодер 20 відео може групувати контекстно кодовані біни і контекстно-незалежно кодовані біни (наприклад, обхідні біни) під час кодування. Наприклад, кодер 20 відео може контекстно кодувати біни, пов'язані більше ніж з одним синтаксичним елементом. Потім кодер 20 відео може кодувати з обходом біни, пов'язані більше ніж з одним синтаксичним елементом. В інших прикладах, кодер 20 відео може виконувати кодування з обходом до контекстного кодування. У будь-якому випадку, ці методи дозволяють кодеру 20 відео звести до мінімуму переходи між контекстним кодуванням і кодуванням з обходом. Відповідно, аспекти даного розкриття винаходу можуть зменшити запізнювання при кодуванні синтаксичних елементів, що включають в себе поєднання контекстно кодованих бінів і кодованих з обходом бінів.

Декодер 30 відео, після прийому кодованих відеоданих, може виконати прохід декодування, загалом взаємозворотний проходу кодування, описаному відносно кодера 20 відео. Наприклад, декодер 30 відео може приймати закодований бітовий потік і декодувати цей бітовий потік. Відповідно до аспектів даного розкриття винаходу, наприклад, декодер 30 відео може декодувати синтаксичний елемент опорного індексу, кодуючи щонайменше один бін бінаризованого значення опорного індексу за допомогою процесу контекстно-адаптивного двійкового арифметичного кодування (CABAC) і кодуючи щонайменше інший бін бінаризованого значення опорного індексу в режимі кодування з обходом процесу контекстно-адаптивного двійкового арифметичного кодування (CABAC).

Відповідно до інших аспектів даного розкриття винаходу, декодер 30 відео може декодувати бітовий потік, що має згруповані контекстно кодовані біни і контекстно-незалежно кодовані біни (наприклад, обхідні біни). Наприклад, декодер 30 відео може декодувати контекстно кодовані біни, пов'язані більше ніж з одним синтаксичним елементом. Потім декодер 30 відео може декодувати кодовані з обходом біни, пов'язані більше ніж з одним синтаксичним елементом. В інших прикладах, декодер 30 відео може виконувати кодування з обходом до контекстного кодування (залежно від розташування бінів у бітовому потоці, що підлягає декодуванню). У будь-якому випадку, ці методи дозволяють декодеру 30 відео звести до мінімуму переходи між контекстним кодуванням і кодуванням з обходом. Відповідно, аспекти даного розкриття винаходу можуть зменшити запізнювання при кодуванні синтаксичних елементів, що включають в себе поєднання контекстно кодованих бінів і кодованих з обходом бінів.

Фіг. 2 є структурною схемою, яка ілюструє приклад кодера 20 відео, що може використовувати методи для кодування даних прогнозування відповідно до прикладів даного розкриття винаходу. Незважаючи на те, що аспекти кодера 20 відео описуються відносно ілюстративного HEVC-кодування, методи даного розкриття винаходу не обмежуються яким-небудь конкретним стандартом або способом кодування, який може передбачати кодування даних прогнозування.

Кодер 20 відео може виконувати внутрішнє і зовнішнє кодування CU у межах відеозображень. Внутрішнє кодування ґрунтується на просторовому прогнозуванні, щоб зменшити або усунути просторову надмірність у відеоданих у межах даного зображення. Зовнішнє кодування ґрунтується на часовому прогнозуванні, щоб зменшити або усунути часову надмірність між поточним зображенням і раніше кодованими зображеннями відеопослідовності. Під внутрішнім режимом (I-режим) може матися на увазі будь-який з декількох просторових режимів стиснення відео. Під зовнішніми режимами, такими як односпрямоване прогнозування (P-режим) або двоспрямоване прогнозування (B-режим), може матися на увазі будь-який з декількох часових режимів стиснення відео.

Як показано на Фіг. 2, кодер 20 відео приймає поточний відеоблок у межах зображення, що підлягає кодуванню. У прикладі на Фіг. 2, кодер 20 відео включає в себе модуль 44 компенсації руху, модуль 42 оцінки руху, модуль 46 внутрішнього прогнозування, пам'ять 64 опорних зображень, підсумовуючий пристрій 50, модуль 52 обробки перетворення, модуль 54 квантування і модуль 56 ентропійного кодування. Модуль 52 обробки перетворення, проілюстрований на Фіг. 2, являє собою модуль, що застосовує діюче перетворення або комбінації перетворення до блока залишкових даних, і його не слід плутати з блоком

коефіцієнтів перетворення, які теж можуть згадуватися як одиниця перетворення (TU-transform unit) у CU. Для відновлення відеоблока, кодер 20 відео також включає в себе модуль 58 зворотного квантування, модуль 60 обробки зворотного перетворення і підсумовуючий пристрій 62. Антиблоковий фільтр (не показаний на Фіг. 2) також може входити до складу для фільтрації

5 границь блоків, щоб усунути артефакти блоковості з відновлюваного відео. За необхідності, антиблоковий фільтр, найчастіше, буде фільтрувати вихідні дані підсумовуючого пристрою 62.

У ході процесу кодування, кодер 20 відео приймає зображення або слайс, що підлягає кодуванню. Зображення або слайс можуть бути розділені на декілька відеоблоків, наприклад найбільші одиниці кодування (LCU-largest coding unit). Модуль 42 оцінки руху і модуль 44

10 компенсації руху виконують кодування з зовнішнім прогнозуванням прийнятого відеоблока відносно одного або більше блоків в одному або більше опорних зображеннях, щоб забезпечити часове стиснення. Модуль 46 внутрішнього прогнозування може виконувати кодування з внутрішнім прогнозуванням прийнятого відеоблока відносно одного або більше сусідніх блоків у тому ж самому зображенні або слайсі, що і блок, що підлягає кодуванню, щоб

15 забезпечити просторове стиснення.

Модуль 40 вибору режиму може вибрати один з режимів кодування, внутрішній або зовнішній, наприклад на основі похибки (тобто спотворення) результатів для кожного режиму, і надає результуючий блок з внутрішнім або зовнішнім прогнозуванням (наприклад, одиницю прогнозування (PU-prediction unit)) на підсумовуючий пристрій 50, щоб генерувати залишкові

20 дані блока, і на підсумовуючий пристрій 62, щоб відновлювати закодований блок для використання в опорному зображенні. Підсумовуючий пристрій 62 об'єднує прогнозований блок зі зворотно квантованими, зворотно перетвореними даними від модуля 60 обробки зворотного перетворення для блока, щоб відновлювати закодований блок, що більш докладно описується

нижче. Деякі зображення можуть позначатися як I-кадри, причому всі блоки в I-кадрі кодуються

25 в режимі внутрішнього прогнозування. У деяких випадках, модуль 46 внутрішнього прогнозування може виконувати кодування з внутрішнім прогнозуванням блока в зображенні з прямим прогнозуванням (Р-кадр) або зображенні з подвійним прогнозуванням (В-кадр), наприклад, коли пошук руху, виконуваний модулем 42 оцінки руху, не дає достатнього прогнозування блока.

30 Модуль 42 оцінки руху і модуль 44 компенсації руху можуть бути високоінтегрованими, але проілюстровані окремо для концептуальних цілей. Оцінка руху (або пошук руху) являє собою процес генерування векторів руху, що оцінюють рух для відеоблоків. Вектор руху, наприклад, може показувати відхилення одиниці прогнозування в поточному зображенні відносно опорної вибірки опорного зображення. Модуль 42 оцінки руху обчислює вектор руху для одиниці

35 прогнозування зображення з зовнішнім кодуванням, порівнюючи одиницю прогнозування з опорними вибірками опорного зображення, збереженого в пам'яті 64 опорних зображень.

Прогнозуючий блок (також згадується як опорна вибірка) являє собою блок, для якого виявляється близька відповідність блоку, що підлягає кодуванню, якщо говорити про різницю елементів зображення, що може бути визначено по сумі абсолютної різниці (SAD), сумі

40 квадрата різниці (SSD) або інших показниках різниці. У деяких прикладах, кодер 20 відео може обчислювати значення для неповних цілочислових позицій елементів зображення опорних зображень, збережених у пам'яті 64 опорних зображень, що також може згадуватися як буфер опорного зображення. Наприклад, кодер 20 відео може інтерполювати значення однієї

45 четвертої позицій елементів зображень, однієї восьмої позицій елементів зображень або інших дробових позицій елементів зображень опорного зображення. Отже, модуль 42 оцінки руху може виконувати пошук руху відносно повних позицій елементів зображень і дробових позицій елементів зображень і виводити вектор руху з точністю до дробового елемента зображення.

Модуль 42 оцінки руху обчислює вектор руху для PU відеоблока в слайсі з зовнішнім кодуванням, порівнюючи позицію PU з позицією прогнозуючого блока опорного зображення.

50 Відповідно, у загальному випадку, дані для вектора руху можуть включати в себе список опорних зображень, індекс у списку опорних зображень (ref\_idx), горизонтальну складову і вертикальну складову. Опорне зображення може бути вибране з першого списку опорних зображень (List 0), другого списку опорних зображень (List 1) або об'єданого списку опорних зображень (List C), кожний з яких ідентифікує одне або більше опорних зображень, збережених

55 у пам'яті 64 опорних зображень.

Модуль 42 оцінки руху може генерувати і відправляти вектор руху, що ідентифікує прогнозуючий блок опорного зображення, на модуль 56 ентропійного кодування і модуль 44 компенсації руху. Тобто, модуль 42 оцінки руху може генерувати і відправляти дані вектора руху, що ідентифікують список опорних зображень, який містить прогнозуючий блок, індекс у списку опорних зображень, що ідентифікує зображення прогнозуючого блока, і горизонтальну і

вертикальну складові, щоб визначити місцеположення прогнозуючого блока в межах ідентифікованого зображення.

У деяких прикладах, замість відправлення дійсного вектора руху для поточного PU, модуль 42 оцінки руху може прогнозувати вектор руху, щоб додатково зменшити обсяг даних, необхідних для повідомлення вектора руху. У цьому випадку, замість кодування і повідомлення самого вектора руху, модуль 42 оцінки руху може генерувати різницю векторів руху (MVD-motion vector difference) відносно відомого (або одержуваного) вектора руху. MVD може включати в себе горизонтальну складову і вертикальну складову, відповідні горизонтальній складовій і вертикальній складовій відомого вектора руху. Відомий вектор руху, що може використовуватися з MVD, щоб задавати поточний вектор руху, може задаватися так називаним предиктором вектора руху (MVP-motion vector predictor). У загальному випадку, щоб бути придатним MVP, вектор руху, використовуваний для прогнозування, повинен указувати на те ж саме опорне зображення, що і вектор руху, кодований у даний час.

Коли наявна множина кандидатів-предикторів вектора руху (від множини блоків-кандидатів), модуль 42 оцінки руху може визначати предиктор вектора руху для поточного блока відповідно до попередньо заданих критеріїв вибору. Наприклад, модуль 42 оцінки руху може вибрати найбільш точний предиктор з набору кандидатів, основуючись на аналізі швидкості і спотворення кодування (наприклад, використовуючи аналіз витрат залежно від швидкості і спотворення або інший аналіз ефективності кодування). В інших прикладах, модуль 42 оцінки руху може генерувати середнє з кандидатів-предикторів вектора руху. Також можливі й інші способи вибору предиктора вектора руху.

Після вибору предиктора вектора руху, модуль 42 оцінки руху може визначити індекс (mvp\_flag) предиктора вектора руху, який може використовуватися, щоб інформувати декодер відео (наприклад, такий як декодер 30 відео) про те, де знайти MVP у списку опорних зображень, що містить блоки-кандидати MVP. Модуль 42 оцінки руху також може визначити MVD (горизонтальну складову і вертикальну складову) між поточним блоком і вибраним MVP. Індеси MVP і MVD можуть використовуватися для відновлення вектора руху.

У деяких прикладах, модуль 42 оцінки руху може замість цього реалізовувати так називаний "режим злиття", у якому модуль 42 оцінки руху може виконувати "злиття" інформації руху (такої як вектори руху, індекси опорних зображень, напрямки прогнозування або інша інформація) прогнозуючого відеоблока з поточним відеоблоком. Відповідно, відносно режиму злиття, поточний відеоблок успадковує інформацію руху від іншого відомого (або одержуваного) відеоблока. Модуль 42 оцінки руху може побудувати список кандидатів для режиму злиття, що включає в себе декілька сусідніх блоків у просторовому і/або часовому напрямку як кандидати для режиму злиття. Модуль 42 оцінки руху може визначити значення індексу (наприклад, merge\_idx), який може використовуватися, щоб інформувати декодер відео (наприклад, такий як декодер 30 відео) про те, де знайти відеоблок для злиття в списку опорних зображень, що містить блоки-кандидати для злиття.

Модуль 46 внутрішнього прогнозування може виконувати внутрішнє прогнозування прийнятого блока, як альтернативу для зовнішнього прогнозування, виконуваного модулем 42 оцінки руху і модулем 44 компенсації руху. Модуль 46 внутрішнього прогнозування може прогнозувати прийнятий блок відносно сусідніх, раніше кодованих блоків, наприклад блоків вище, вище і правіше, вище і лівіше або лівіше від поточного блока, передбачаючи порядок кодування для блоків зліва направо, зверху вниз. Модуль 46 внутрішнього прогнозування може бути виконаний з можливістю множини різних режимів внутрішнього прогнозування. Наприклад, модуль 46 внутрішнього прогнозування може бути виконаний з можливістю визначеної кількості спрямованих режимів прогнозування, наприклад тридцять чотири спрямованих режими прогнозування, на основі розміру CU, що підлягає кодуванню.

Модуль 46 внутрішнього прогнозування може вибрати режим внутрішнього прогнозування, наприклад, обчислюючи значення похибки для різних режимів внутрішнього прогнозування і вибираючи режим, що дає найменше значення похибки. Спрямовані режими прогнозування можуть включати в себе функції для об'єднання значень просторово сусідніх елементів зображення і застосування об'єднаних значень до однієї або більше позицій елементів зображення в PU. Після обчислення значень для всіх позицій елементів зображення в PU, модуль 46 внутрішнього прогнозування може обчислити значення похибки для режиму прогнозування, основуючись на різницях елементів зображення між PU і прийнятим блоком, що підлягає кодуванню. Модуль 46 внутрішнього прогнозування може продовжити тестування режимів прогнозування, поки не буде виявлений режим прогнозування, що дає прийнятне значення похибки. Тоді модуль 46 внутрішнього прогнозування може відправити PU на підсумовуючий пристрій 50.

Кодер 20 відео формує залишковий блок, віднімаючи дані прогнозування, обчислені модулем 44 компенсації руху або модулем 46 внутрішнього прогнозування, з вихідного відеоблока, що підлягає кодуванню. Підсумовуючий пристрій 50 являє собою компонент або компоненти, що виконують цю операцію віднімання. Залишковий блок може відповідати двовимірній матриці значень різниць елементів зображення, причому кількість значень у залишковому блоці та ж, що і кількість елементів зображення в PU, що відповідають залишковому блоку. Значення в залишковому блоці можуть відповідати різницям, тобто похибки, між значеннями поєднаних елементів зображення в PU і у вихідному блоці, що підлягає кодуванню. Різниці можуть бути різницями кольоровості або яскравості, залежно від типу блока, який кодується.

Модуль 52 обробки перетворення може формувати одну або більше одиниць перетворення (TU) із залишкового блока. Модуль 52 обробки перетворення вибирає перетворення з множини перетворень. Перетворення може вибиратися на основі однієї або більше характеристик кодування, таких як розмір блока, режим кодування або тому подібне. Потім модуль 52 обробки перетворення застосовує вибране перетворення до TU, одержуючи відеоблок, що містить двовимірний масив коефіцієнтів перетворення.

Модуль 52 обробки перетворення може відправити одержані в результаті коефіцієнти перетворення на модуль 54 квантування. Потім модуль 54 квантування може квантувати коефіцієнти перетворення. Потім модуль 56 ентропійного кодування може виконати розгорнення квантованих коефіцієнтів перетворення в матриці відповідно до режиму розгорнення. Дане розкриття винаходу описує модуль 56 ентропійного кодування як такий, що виконує розгорнення. Проте, варто розуміти, що, в інших прикладах, розгорнення можуть виконувати інші модулі обробки, такі як модуль 54 квантування.

Після того, як коефіцієнти перетворення були розгорнуті в одновимірний масив, модуль 56 ентропійного кодування може застосувати до коефіцієнтів ентропійне кодування, таке як CAVLC, CABAC, основане на синтаксисі контекстно-адаптивне двійкове арифметичне кодування (SBAC), ентропійне кодування з розбивкою інтервалу імовірності (PIPE) або інша методика ентропійного кодування.

Для виконання CABAC, модуль 56 ентропійного кодування може вибрати контекстну модель, щоб застосувати визначений контекст для кодування символів, що повинні передаватися. Контекст може мати відношення до того, наприклад, чи є сусідні значення ненульовими або ні. Модуль 56 ентропійного кодування також може ентропійно кодувати синтаксичні елементи, такі як сигнал, що характеризує вибране перетворення.

Модуль 56 ентропійного кодування може ентропійно кодувати дані прогнозування. При зовнішньому прогнозуванні відеоданих, наприклад, дані прогнозування можуть включати в себе дані, що показують значення опорних індексів, вектори руху, предиктори векторів руху, значення різниці векторів руху тощо. Тобто, як зазначалося вище, оцінка руху (модулем 42 оцінки руху) визначає один або більше індексів для опорних зображень (ref\_idx) і напрямок прогнозування (pred\_dir: пряме, зворотне або двоспрямоване). Модуль 56 ентропійного кодування може ентропійно кодувати синтаксичні елементи, що представляють вектори руху (наприклад, горизонтальну складову і вертикальну складову векторів руху), індекси опорних зображень і напрямок прогнозування. Модуль 56 ентропійного кодування може включати закодовані синтаксичні елементи в закодований бітовий потік відео, який потім може бути декодований декодером відео (таким як декодер 30 відео, описаний нижче) для використання в процесі декодування відео. Тобто, ці синтаксичні елементи можуть бути передбачені для PU із зовнішнім кодуванням, щоб дозволити декодеру 30 відео декодувати і відтворювати відеодані, що задаються за допомогою PU.

У деяких прикладах, як докладніше описано відносно Фіг. 4 нижче, модуль 56 ентропійного кодування (або інший модуль кодування в кодері 20 відео) може бінаризувати синтаксичні елементи до ентропійного кодування синтаксичних елементів. Наприклад, модуль 56 ентропійного кодування може перевести абсолютне значення кожного синтаксичного елемента, що підлягає кодуванню, в двійкову форму. Модуль 56 ентропійного кодування може використовувати процес кодування унарним кодом, зрізаним унарним кодом або інший, щоб бінаризувати синтаксичні елементи. Що стосується значень опорних індексів, наприклад, якщо максимальна кількість опорних зображень у списку опорних зображень дорівнює чотирьом, тобто опорний індекс (ref\_idx) має значення в діапазоні від 0 до 3, то може застосовуватися наступна бінаризація з Таблиці 1.



Таблиця 1

Опорний індекс	Бінаризація
0	0
1	10
2	110
3	111

Як показано в Таблиці 1, бінаризоване значення змінюється від одного біта до трьох бітів, залежно від значення опорного індексу.

У деяких прикладах, модуль 56 ентропійного кодування може ентропійно кодувати значення опорних індексів, використовуючи три різних контексти (наприклад, ctx0, ctx1 і ctx2). Наприклад, модуль 56 ентропійного кодування може ентропійно кодувати перший бін (bin0) і другий бін (bin1), використовуючи ctx0 і ctx1, відповідно, а третій бін (bin2) і інші біни кодуються за допомогою контексту ctx2. У цьому прикладі, ctx2 спільно використовується всіма бінами, починаючи з bin2 і рахуючи його, тобто bin2 і бінами після bin2, наприклад bin3, bin4 і так далі. У деяких прикладах, можуть передбачатися додаткові біни, крім bin2, наприклад, якщо максимальна кількість опорних зображень більше чотирьох.

Як зазначалося вище, спільне використання контексту ctx2 бінами може бути неефективним, через оновлення імовірності, пов'язані з контекстним кодуванням. Відповідно до аспектів даного розкриття винаходу, модуль 56 ентропійного кодування може виконувати CABAC-кодування значення опорного індексу, виділяючи ctx2 для кодування bin2 і кодуючи всі біни після bin2 з використанням режиму кодування з обходом. Знову ж, кодування з обходом, у загальному випадку, включає в себе кодування бінів з використанням фіксованої імовірності (контексти не вимагаються). Наприклад, модуль 56 ентропійного кодування може кодувати з обходом біни після bin2 значення опорного індексу, використовуючи кодування кодом Голомба, кодування експоненціальним кодом Голомба, кодування кодом Голомба-Райса або інші процеси кодування, які обходять механізм CABAC-кодування.

В іншому прикладі, модуль 56 ентропійного кодування може контекстно кодувати меншу кількість бінів значення опорного індексу, видаляючи ctx2. Тобто, відповідно до аспектів даного розкриття винаходу, модуль 56 ентропійного кодування може закодувати bin2 і всі наступні біни, використовуючи режим з обходом CABAC. У цьому прикладі, модуль 56 ентропійного кодування може виконувати CABAC-кодування bin0 з використанням контексту ctx0, а bin1 - з використанням контексту ctx1, і може кодувати з обходом bin2 і інші біни, що ідуть за bin2, з використанням режиму з обходом CABAC. Видалення контексту, таким чином, може знизити загальну складність, пов'язану з кодуванням значень опорних індексів.

Ще в одному прикладі, модуль 56 ентропійного кодування може кодувати меншу кількість бінів значення опорного індексу, видаляючи і ctx1, і ctx2. Тобто, відповідно до аспектів даного розкриття винаходу, модуль 56 ентропійного кодування може кодувати bin1 і всі наступні біни з використанням режиму з обходом CABAC, тим самим додатково знижуючи складність, пов'язану з кодуванням значень опорних індексів. У цьому прикладі, модуль 56 ентропійного кодування може виконувати CABAC-кодування bin0 з використанням контексту ctx0 і може кодувати з обходом bin1, bin2 і інші біни, що ідуть за bin2, з використанням режиму з обходом CABAC.

Інші аспекти даного розкриття винаходу загалом стосуються того, яким чином модуль 56 ентропійного кодування бінаризує значення опорних індексів. Наприклад, як зазначалося вище, модуль 56 ентропійного кодування може бінаризувати значення опорних індексів, використовуючи процес кодування унарним кодом, зрізаним унарним кодом або інший. В іншому прикладі, модуль 56 ентропійного кодування може використовувати процес кодування експоненціальним кодом Голомба, щоб бінаризувати значення опорного індексу.

У деяких прикладах, відповідно до аспектів даного розкриття винаходу, модуль 56 ентропійного кодування може реалізовувати комбінацію процесів бінаризації. Наприклад, як докладніше описано відносно Фіг. 4 нижче, модуль 56 ентропійного кодування може комбінувати процес кодування унарним кодом (або зрізаним унарним кодом) з процесом кодування експоненціальним кодом Голомба, щоб бінаризувати значення опорних індексів. В ілюстративному прикладі, модуль 56 ентропійного кодування може комбінувати зрізаний унарний код довжини (4) з експоненціальним кодом Голомба (наприклад, експоненціальний код Голомба 0-го порядку). У такому прикладі, модуль 56 ентропійного кодування може бінаризувати першу кількість бінів (наприклад, два, три, чотири або тому подібне) значення

опорного індексу з використанням унарного коду і може бінаризувати інші біни опорного індексу, використовуючи експоненціальний код Голомба.

У будь-якому випадку, модуль 56 ентропійного кодування може реалізовувати методи для контекстного кодування одного або більше бінів значення опорного індексу і кодування з обходом одного або більше інших бінів значення опорного індексу за допомогою будь-якої схеми бінаризації. Наприклад, як зазначалося вище, модуль 56 ентропійного кодування може контекстно кодувати (наприклад, виконувати CABAC-кодування) першу кількість бінів бінаризованого синтаксичного елемента і кодувати з обходом інші біни. В описаному вище прикладі, у якому зрізаний унарний код довжини (4) комбінується з експоненціальним кодом Голомба 0-го порядку, модуль 56 ентропійного кодування може контекстно кодувати перші два біни (або будь-яку іншу кількість бінів) зрізаного унарного коду, а потім кодувати з обходом другу частину унарного коду і всього експоненціального коду Голомба. В інших прикладах, модуль 56 ентропійного кодування може використовувати інші схеми бінаризації. Наприклад, модуль 56 ентропійного кодування може використовувати двійковий код фіксованої довжини замість експоненціального коду Голомба, описаного у вищенаведених прикладах.

У деяких прикладах, модуль 56 ентропійного кодування може зрізати або видаляти біни з бінаризованого значення опорного індексу перед кодуванням цього значення. Додатково або як альтернатива, модуль 56 ентропійного кодування може групувати біни, кодовані з використанням контексту, і біни, кодовані з використанням режиму з обходом. Наприклад, модуль 56 ентропійного кодування може кодувати опорні індекси В-зображення шляхом контекстного кодування одного або більше бінів значення першого опорного індексу, контекстно кодувати один або більше бінів значення другого індексу, кодувати з обходом один або більше інших бінів значення першого опорного індексу і кодувати з обходом один або більше інших бінів значення другого опорного індексу (у порядку, представленому вище). Відповідно, модуль 56 ентропійного кодування переходить між режимом контекстного кодування і режимом кодування з обходом тільки один раз, наприклад між контекстно кодованими бінами і контекстно-незалежно кодованими бінами.

Слідом за ентропійним кодуванням модулем 56 ентропійного кодування, одержане в результаті кодоване відео може бути передане на інший пристрій, такий як декодер 30 відео, або заархівоване для наступної передачі або пошуку. Модуль 58 зворотного квантування і модуль 60 обробки зворотного перетворення застосовують зворотне квантування і зворотне перетворення, відповідно, щоб відновлювати залишковий блок в області елементів зображення, наприклад, для наступного використання як опорного блока.

Модуль 44 компенсації руху може обчислити опорний блок, додаючи залишковий блок до прогнозуючого блока одного з зображень у пам'яті 64 опорних зображень. Модуль 44 компенсації руху також може застосувати один або більше інтерполюючих фільтрів до відновленого залишкового блока, щоб обчислити неповні цілочислові значення елементів зображення для використання при оцінці руху.

Підсумовуючий пристрій 62 додає відновлений залишковий блок до блока прогнозування з компенсацією руху, одержаного модулем 44 компенсації руху, щоб одержати відновлений відеоблок для збереження в пам'яті 64 опорних зображень. Відновлений відеоблок може використовуватися модулем 42 оцінки руху і модулем 44 компенсації руху як опорний блок для зовнішнього кодування блока в наступному зображенні.

Фіг. 3 є структурною схемою, яка ілюструє приклад декодера 30 відео, що декодує закодовану відеопослідовність. У прикладі на Фіг. 3, декодер 30 відео включає в себе модуль 70 ентропійного декодування, модуль 72 компенсації руху, модуль 74 внутрішнього прогнозування, модуль 76 зворотного квантування, модуль 78 зворотного перетворення, пам'ять 82 опорних зображень і підсумовуючий пристрій 80.

Як передумова, декодер 30 відео може приймати стиснені відеодані, що були стиснені для передачі через мережу в так називані "одиниці рівня мережної абстракції" або одиниці NAL (network abstraction layer unit). Кожна одиниця NAL може включати в себе заголовок, який ідентифікує тип даних, що зберігаються в одиниці NAL. Є два типи даних, що звичайно зберігаються в одиницях NAL. Перший тип даних, що зберігаються в одиниці NAL, є даними рівня кодування відео (VCL-video coding layer), що включають в себе стиснені відеодані. Другий тип даних, що зберігаються в одиниці NAL, згадуються як дані не-VCL, які включають в себе додаткову інформацію, таку як набори параметрів, що задають дані заголовка, загальні для великої кількості одиниць NAL, і доповнююча розширена інформація (SEI-supplemental enhancement information).

Наприклад, набори параметрів можуть містити інформацію заголовка рівня послідовності (наприклад, у наборах параметрів послідовності (SPS-sequence parameter set)) і інформацію, що

рідко змінюється, заголовка рівня зображень (наприклад, у наборах параметрів зображення (PPS-picture parameter set)). Інформація, що рідко змінюється, яка міститься в наборах параметрів, не повинна повторюватися для кожної послідовності або зображення, за рахунок чого підвищується ефективність кодування. Крім цього, використання наборів параметрів дає

5 можливість позасмугової передачі інформації заголовка, що дозволяє уникнути необхідності надлишкових передач для забезпечення стійкості до помилок.

У ході процесу декодування, декодер 30 відео приймає закодований бітовий потік відео, що представляє відеоблоки закодованого відеослайсу і пов'язані синтаксичні елементи. У загальному випадку, модуль 70 ентропійного декодування ентропійно декодує бітовий потік,

10 щоб згенерувати квантовані коефіцієнти, вектори руху і інші синтаксичні елементи. Декодер 30 відео може приймати синтаксичні елементи на рівні відеослайсів і/або на рівні відеоблоків.

Наприклад, якщо відеослайс кодується як слайс із внутрішнім кодуванням (I), модуль 74 внутрішнього прогнозування може генерувати дані прогнозування для відеоблока поточного відеослайсу на основі повідомлюваного режиму внутрішнього прогнозування і даних з раніше

15 декодованих блоків поточного зображення. Якщо зображення кодується як слайс із зовнішнім кодуванням (тобто B, P або GPB), модуль 72 компенсації руху одержує прогноуючі блоки (також іменовані як опорні вибірки) для відеоблока поточного відеослайсу на основі векторів руху й інших синтаксичних елементів, що приймаються від модуля 70 ентропійного декодування. Прогноуючі блоки можуть одержуватися з одного з опорних зображень у межах

20 одного зі списків опорних зображень. Декодер 30 відео може скласти списки опорних зображень, List 0 і List 1, використовуючи задані за замовчуванням методи складання на основі опорних зображень, що зберігаються в пам'яті 82 опорних зображень.

Модуль 70 ентропійного декодування може декодувати бітовий потік, використовуючи такий же процес, як і реалізований у кодері 20 відео (наприклад, CABAC, CAVLC і т. д.). Процес ентропійного кодування, використовуваний кодером, може бути повідомлений у закодованому бітовому потоці або може бути попередньо заданим процесом. Наприклад, модуль 70 ентропійного декодування може приймати закодовані бінаризовані синтаксичні елементи. Модуль 70 ентропійного декодування може декодувати бітовий потік (наприклад, використовуючи контекстно-адаптивний режим або режим з обходом) і бінаризувати декодовані

25 значення, щоб одержати декодовані синтаксичні елементи.

У деяких випадках, модуль 70 ентропійного декодування може ентропійно декодувати дані прогнозування. Як зазначалося вище відносно кодера 20 відео, дані прогнозування можуть включати в себе дані, що показують значення опорних індексів, вектори руху, предиктори векторів руху, значення різниць векторів руху тощо. Тобто, модуль 70 ентропійного декодування

30 може ентропійно декодувати синтаксичні елементи, що представляють вектори руху (наприклад, горизонтальну складову і вертикальну складову векторів руху), індекси опорних зображень і напрямки прогнозування. Ці синтаксичні елементи можуть бути передбачені для PU із зовнішнім кодуванням, щоб дозволити декодеру 30 відео декодувати і відтворювати відеодані, що задаються за допомогою PU.

У деяких прикладах, як зазначалося вище, модуль 70 ентропійного декодування може ентропійно декодувати значення опорних індексів, використовуючи три різних контексти (наприклад, ctx0, ctx1 і ctx2). Наприклад, модуль 56 ентропійного декодування може ентропійно декодувати перший бін (bin0) і другий бін (bin1) з використанням ctx0 і ctx1, відповідно, і декодувати третій бін (bin2) і інші біни за допомогою контексту ctx2. Спільне використання

35 контексту ctx2 бінами може бути неефективним, через оновлення імовірності, пов'язані з контекстним кодуванням.

Відповідно до аспектів даного розкриття винаходу, модуль 70 ентропійного декодування може виконувати CABAC-кодування значення опорного індексу, виділяючи ctx2 для кодування bin2 і кодуючи всі біни після bin2 з використанням режиму кодування з обходом. В іншому

40 прикладі, модуль 70 ентропійного декодування може контекстно кодувати меншу кількість бінів значення опорного індексу, видаляючи ctx2. Тобто, відповідно до аспектів даного розкриття винаходу, модуль 70 ентропійного декодування може декодувати bin2 і всі наступні біни, використовуючи режим з обходом CABAC. Ще в одному прикладі, модуль 70 ентропійного декодування може кодувати меншу кількість бінів значення опорного індексу, видаляючи і ctx1 і

45 ctx2. Тобто, відповідно до аспектів даного розкриття винаходу, модуль 70 ентропійного декодування може декодувати bin1 і всі наступні біни, використовуючи режим з обходом CABAC, тим самим додатково знижуючи складність, пов'язану з кодуванням значень опорних індексів.

Інші аспекти даного розкриття винаходу загалом стосуються того, яким чином модуль 70 ентропійного декодування бінаризує значення опорних індексів. У деяких прикладах, модуль 70

50

ентропійного декодування може бінаризувати значення опорних індексів, використовуючи процес кодування унарним кодом, зрізаним унарним кодом або інший. В іншому прикладі, модуль 56 ентропійного кодування може використовувати процес кодування експоненціальним кодом Голомба, щоб бінаризувати значення опорного індексу.

5 У деяких прикладах, відповідно до аспектів даного розкриття винаходу, модуль 70 ентропійного декодування може реалізовувати комбінацію процесів бінаризації. Наприклад, як докладніше описано відносно Фіг. 4 нижче, модуль 70 ентропійного декодування може комбінувати процес кодування унарним кодом (або зрізаним унарним кодом) з процесом кодування експоненціальним кодом Голомба, щоб бінаризувати значення опорних індексів. В ілюстративному прикладі, модуль 70 ентропійного декодування може комбінувати зрізаний унарний код довжини (4) з експоненціальним кодом Голомба (наприклад, експоненціальний код Голомба 0-го порядку). У такому прикладі, модуль 70 ентропійного декодування може бінаризувати першу кількість бінів (наприклад, два, три, чотири або тому подібне) значення опорного індексу з використанням унарного коду і може бінаризувати інші біни опорного індексу, використовуючи експоненціальний код Голомба.

15 У будь-якому випадку, модуль 70 ентропійного декодування може реалізовувати методи для контекстного кодування одного або більше бінів значення опорного індексу і кодування з обходом одного або більше інших бінів значення опорного індексу за допомогою будь-якої схеми бінаризації. Наприклад, як зазначалося вище, модуль 70 ентропійного декодування може контекстно кодувати (наприклад, виконувати CABAC-кодування) першу кількість бінів бінаризованого синтаксичного елемента і кодувати з обходом інші біни. В описаному вище прикладі, у якому зрізаний унарний код довжини (4) комбінується з експоненціальним кодом Голомба 0-го порядку, модуль 70 ентропійного декодування може контекстно кодувати перші два біни (або будь-яку іншу кількість бінів) зрізаного унарного коду, а потім кодувати з обходом 25 другу частину унарного коду і весь експоненціальний код Голомба. В інших прикладах, модуль 70 ентропійного декодування може використовувати інші схеми бінаризації. Наприклад, модуль 70 ентропійного декодування може використовувати двійковий код фіксованої довжини замість експоненціального коду Голомба, описаного у вищенаведених прикладах.

У деяких прикладах, модуль 70 ентропійного декодування може зрізати або видаляти біни з бінаризованого значення опорного індексу перед кодуванням цього значення. Додатково або як альтернатива, модуль 70 ентропійного декодування може групувати біни, кодовані з використанням контексту, і біни, кодовані з використанням режиму з обходом. Наприклад, модуль 70 ентропійного декодування може кодувати опорні індекси В-зображення шляхом контекстного кодування одного або більше бінів значення першого опорного індексу, контекстно кодувати один або більше бінів значення другого індексу, кодувати з обходом один або більше 35 інших бінів значення першого опорного індексу і кодувати з обходом один або більше інших бінів значення другого опорного індексу (у порядку, представленому вище). Відповідно, модуль 70 ентропійного декодування переходить між режимом контекстного кодування і режимом кодування з обходом тільки один раз, наприклад між контекстно кодованими бінами і контекстно-незалежно кодованими бінами.

Після ентропійного декодування синтаксичних елементів і коефіцієнтів перетворення, у деяких прикладах, модуль 70 ентропійного декодування (або модуль 76 зворотного квантування) може розгортати прийняті значення коефіцієнтів перетворення, використовуючи розгорнення, дзеркальне режиму розгорнення, використовуваному модулем 56 ентропійного кодування (або модулем 54 квантування) кодера 20 відео. Хоча і показані у вигляді окремих функціональних модулів для простоти ілюстрації, структура і функціональні можливості модуля 45 70 ентропійного декодування, модуль 76 зворотного квантування і інші модулі декодера 30 відео можуть бути у високому ступені інтегровані один з одним.

Модуль 76 зворотного квантування виконує зворотне квантування, тобто деквантування, 50 квантованих коефіцієнтів перетворення, наданих у бітовому потоці і декодованих модулем 70 ентропійного декодування. Процес зворотного квантування може включати в себе традиційний процес, наприклад, аналогічний процесам, які пропонувані для HEVC або задаються стандартом декодування H.264. Процес зворотного квантування може включати в себе використання параметра квантування QP, що обчислюється кодером 20 відео для CU, щоб визначити ступінь квантування і, точно так само, ступінь зворотного квантування, які повинні 55 бути застосовані. Модуль 76 зворотного квантування може виконувати зворотне квантування коефіцієнтів перетворення або до, або після того, як коефіцієнти перетворення з одновимірною масиву в двовимірний масив.

Модуль 74 внутрішнього прогнозування може генерувати дані прогнозування для поточного 60 блока поточного зображення на основі сигналізованого режиму внутрішнього прогнозування і

даних з раніше декодованих блоків поточного зображення. Модуль 72 компенсації руху може витягати вектор руху, напрямок прогнозування руху й опорний індекс із закодованого бітового потоку. Напрямок опорного прогнозування показує, чи є режим зовнішнього прогнозування односпрямованим (наприклад, Р-кадр) або двоспрямованим (В-кадр). Опорний індекс показує

5 опорне зображення, на яке спрямований вектор руху. На основі витягнутого напрямку прогнозування руху, індексу опорного зображення і вектора руху, модуль компенсації руху одержує блок з компенсацією руху для поточної частини. Ці блоки з компенсацією руху використовуються для відтворення прогнозуючого блока, використововуваного для одержання залишкових даних.

10 Модуль 72 компенсації руху може одержувати блоки з компенсацією руху, можливо, виконуючи інтерполяцію на основі інтерполюючих фільтрів. Ідентифікатори для інтерполюючих фільтрів, які будуть використовуватися для оцінки руху з точністю до неповного елемента зображення, можуть бути включені до складу синтаксичних елементів. Модуль 72 компенсації руху може використовувати інтерполюючі фільтри, які використовуються кодером 20 відео під

15 час кодування відеоблока, щоб обчислити інтерпольовані значення для неповних цілих елементів зображення опорного блока. Модуль 72 компенсації руху може визначати інтерполюючі фільтри, використововувані кодером 20 інформації, виходячи з синтаксичної інформації, що приймається, і використовувати інтерполюючі фільтри для одержання прогнозуючих блоків.

20 Модуль 72 компенсації руху може приймати дані прогнозування, які показують, де витягати інформацію руху для поточного блока. Наприклад, модуль 72 компенсації руху може приймати інформацію прогнозування вектора руху, таку як індекс MVP (*mvp\_flag*), MVD, прапор злиття (*merge\_flag*) і/або індекс злиття (*merge\_idx*), і використовувати таку інформацію для ідентифікації інформації руху, використововуваної для прогнозування поточного блока.

25 Наприклад, модуль 72 компенсації руху може генерувати список кандидатів для MVP або злиття. Потім модуль 72 компенсації руху може використовувати індекс MVP або злиття для ідентифікації інформації руху, використововуваної для прогнозування вектора руху поточного блока. Тобто, модуль 72 компенсації руху може ідентифікувати MVP зі списку опорних зображень, використовуючи індекс MVP (*mvp\_flag*). Модуль 72 компенсації руху може

30 об'єднувати ідентифікований MVP з прийнятою MVD, щоб визначити вектор руху для поточного блока. В інших прикладах, модуль 72 компенсації руху може ідентифікувати кандидата для злиття зі списку опорних зображень, використовуючи індекс злиття (*merge\_idx*), щоб визначити інформацію руху для поточного блока. У будь-якому випадку, після визначення інформації руху для поточного блока, модуль 72 компенсації руху може згенерувати прогнозуючий блок для

35 поточного блока.

Додатково, модуль 72 компенсації руху і модуль 74 внутрішнього прогнозування, у прикладі для HEVC, можуть використовувати деяку синтаксичну інформацію (наприклад, надану деревом квадрантів), щоб визначити розміри LCU, використовуваних для кодування зображення (зображень) закодованої відеопослідовності. Модуль 72 компенсації руху і модуль 74

40 внутрішнього прогнозування також можуть використовувати синтаксичну інформацію, щоб визначити інформацію розбивки, що описує як розбивається кожна CU зображення закодованої відеопослідовності (а крім того, як розбиваються неповні CU). Синтаксична інформація також може включати в себе режими, що показують, як кодується кожна розбивка (наприклад, із внутрішнім або зовнішнім прогнозуванням, і для режиму кодування з внутрішнім

45 прогнозуванням і зовнішнім прогнозуванням), одне або більше опорних зображень (і/або опорних списків, що містять ідентифікатори для опорних зображень) для кожної PU із зовнішнім кодуванням, і іншу інформацію, щоб декодувати закодовану відеопослідовність.

Підсумовуючий пристрій 80 об'єднує залишкові блоки з відповідними блоками прогнозування, згенерованими модулем 72 компенсації руху або модулем 74 внутрішнього

50 прогнозування, щоб сформувати декодовані блоки. За необхідності, може також застосовуватися антиблоковий фільтр для фільтрації декодованих блоків, щоб усунути артефакти блоковості. Потім декодовані відеоблоки зберігаються в пам'яті 82 опорних зображень, що забезпечує опорні блоки для наступної компенсації руху, а також одержується декодоване відео для представлення на пристрої відображення (такому як пристрій 32

55 відображення на Фіг. 1).

Фіг. 4 є структурною схемою, що ілюструє типовий процес арифметичного кодування. Типовий процес арифметичного кодування, зображений на Фіг. 4, загалом, описується як виконуваний кодером 20 відео. Проте, варто розуміти, що методи, описувані відносно Фіг. 4, можуть виконуватися різними іншими відеокодерами, у тому числі декодером 30 відео.

Наприклад, як зазначалося вище відносно Фіг. 3, декодер 30 відео може виконувати процес декодування, що є взаємозворотним для процесу, виконуваного кодером 20 відео.

Приклад на Фіг. 4 містить пристрій 100 бінаризації, засіб 102 контекстного моделювання, механізм 104 кодування і обхідний кодер 106. Пристрій 100 бінаризації відповідає за бінаризацію прийнятого синтаксичного елемента. Наприклад, пристрій 100 бінаризації може зіставляти синтаксичний елемент з декількома так називаними бінами, причому кожен бін являє собою двійкове значення. Як приклад для ілюстрації, пристрій 100 бінаризації може зіставляти синтаксичний елемент з бінами, використовуючи зрізаний унарний (TU-truncated unary) код. У загальному випадку, кодування унарним кодом може передбачати генерування рядка бінів довжини N+1, де перші N бінів дорівнюють 1, а останній бін дорівнює 0. Кодування зрізаним унарним кодом може мати на один менше бінів, ніж кодування унарним кодом, за рахунок установлення максимуму для найбільшого можливого значення синтаксичного елемента (сMax). Приклад кодування зрізаним унарним кодом показаний у Таблиці 2 при сMax=10.

Таблиця 2

Значення	Рядок бінів
0	0
1	10
2	110
3	1110
4	11110
5	111110
6	1111110
7	11111110
8	111111110
9	1111111110
10	1111111111

При виконанні на декодері 30 відео, декодер 30 відео може шукати 0, щоб визначити, коли завершується синтаксичний елемент, кодований у даний час. Як докладніше описано нижче, кодування зрізаним унарним кодом являє собою лише приклад, і пристрій 100 бінаризації може виконувати різноманітні інші процеси бінаризації (а також і комбінації процесів бінаризації), щоб бінаризувати синтаксичні елементи.

Засіб 102 контекстного моделювання може відповідати за визначення контекстної моделі (також згадуваної як імовірнісна модель) для даного біна. Наприклад, засіб 102 контекстного моделювання може вибирати імовірнісну модель, що працює з контекстом для кодових символів, пов'язаних з блоком відеоданих. У загальному випадку, імовірнісна модель зберігає імовірність для кожного біна того, дорівнює він "1" або "0".

Засіб 102 контекстного моделювання може вибирати імовірнісну модель з ряду наявних імовірнісних моделей. У деяких прикладах, як докладніше описано нижче, контекст, використовуваний засобом 102 контекстного моделювання, може визначатися на основі номера біна, що підлягає кодуванню. Тобто, контекст може залежати від позиції біна в рядку бінів, згенерованому пристроєм 100 бінаризації. У будь-якому випадку, на кодері 20 відео, цільовий символ може кодуватися з використанням вибраної імовірнісної моделі. На кодері 20 відео, цільовий символ може піддаватися синтаксичному аналізу з використанням вибраної імовірнісної моделі.

Механізм 104 кодування кодує бін, використовуючи визначену імовірнісну модель (від засобу 102 контекстного моделювання). Після того, як механізм 104 кодування закодував бін, механізм 104 кодування може оновити імовірнісну модель, пов'язану з контекстом, використовуваним для кодування біна. Тобто, вибрана імовірнісна модель оновлюється на основі фактичного кодованого значення (наприклад, якщо значення біна було "1", лічильник частоти появи "1" збільшується). Кодування бінів з використанням засобу 102 контекстного моделювання і механізму 104 кодування може згадуватися як кодування бінів з використанням режиму контекстного кодування.

Обхідний кодер 106 кодує біни з використанням фіксованої імовірності. На відміну від контекстного кодування (за допомогою засобу 102 контекстного моделювання і механізму 104 кодування), обхідний кодер 106 не оновлює процес кодування з обходом, основується на фактичних значеннях кодованих бінів. Відповідно, у загальному випадку, обхідний кодер 106 може кодувати з обходом біни швидше, ніж при контекстному кодуванні. Кодування бінів з

використанням обхідного кодера 106 може згадуватися як кодування бінів з використанням режиму кодування з обходом. Приклади режимів кодування з обходом включають в себе кодування кодом Голомба, кодування експоненціальним кодом Голомба, кодування кодом Голомба-Райса або будь-який інший придатний процес кодування, що обходить засіб 102 контекстного моделювання і механізм 104 кодування.

Кодовані біни (від механізму 104 кодування й обхідного кодера 106) об'єднуються, щоб сформувати кодований бітовий потік. Щоб декодувати закодований бітовий потік, декодер відео (такий як декодер 30 відео) може виконати процес, дзеркальний відносно показаного на Фіг. 4. Тобто, декодер 30 відео може виконувати контекстне кодування (з використанням засобу 102 контекстного моделювання і механізму 104 кодування) або кодування з обходом (з використанням обхідного кодера 106) на закодованому бітовому потоці, щоб згенерувати декодований рядок бінів. Потім декодер 30 відео може бінаризувати рядок бінів (за допомогою пристрою 100 бінаризації), щоб згенерувати синтаксичні значення.

Процес арифметичного кодування, показаний на Фіг. 3, може використовуватися для кодування відеоданих. Наприклад, процес кодування, показаний на Фіг. 3, може використовуватися для кодування даних прогнозування, у тому числі значень опорних індексів, векторів руху, предикторів векторів руху, значень різниць векторів руху і тому подібного.

В ілюстративному прикладі, пристрій 100 бінаризації може бінаризувати опорний індекс (`ref_idx`). У деяких прикладах, одержаний в результаті рядок бінів для опорного індексу може бути до 15 бінів у довжину, залежно від кількості опорних зображень, що доступні як зразки. Як зазначалося вище, у деяких прикладах, усі біни значення опорного індексу можуть контекстно кодуватися з використанням засобу 102 контекстного моделювання і механізму 104 кодування. Більше того, один або більше з бінів можуть спільно використовувати контекст. Проте, контекстне кодування всіх бінів і спільне використання контексту більше ніж один біном може бути неефективним внаслідок запізнювань, пов'язаних з контекстним кодуванням.

Відповідно до аспектів даного розкриття винаходу, як показано на Фіг. 4, кодер 20 відео може кодувати рядок бінів для опорного індексу, основуючись на номері біна для біна, що підлягає кодуванню. Наприклад, кодер 20 відео може кодувати конкретний бін у рядку бінів для опорного індексу відповідно до відносної позиції конкретного біна в рядку бінів. В одному з прикладів, кодер 20 відео може контекстно кодувати перший, другий бін і третій бін опорного індексу і може кодувати з обходом інші біни опорного індексу. Тобто, кодер 20 відео може кодувати перший бін (`bin0`) з використанням засобу 102 контекстного моделювання і механізму 104 кодування з першим контекстом `ctx0`, другий бін (`bin1`) з використанням засобу 102 контекстного моделювання і механізму 104 кодування з другим контекстом `ctx1` і третій бін (`bin2`) з використанням засобу 102 контекстного моделювання і механізму 104 кодування з третім контекстом `ctx2`. Однак, кодер 20 відео може кодувати четвертий бін (`bin3`) і будь-які інші наступні біни, використовуючи обхідний кодер 106.

В іншому прикладі, кодер 20 відео може знизити кількість бінів, які контекстно кодуються. Наприклад, кодер 20 відео може кодувати перший бін (`bin0`) з використанням засобу 102 контекстного моделювання і механізму 104 кодування з першим контекстом `ctx0` і другий бін (`bin1`) з використанням засобу 102 контекстного моделювання і механізму 104 кодування з другим контекстом `ctx1`. У цьому прикладі, однак, кодер 20 відео може кодувати з обходом третій бін (`bin2`) і будь-які інші наступні біни, використовуючи обхідний кодер 106.

У ще одному прикладі, кодер 20 відео може додатково знизити кількість бінів, які контекстно кодуються. Наприклад, кодер 20 відео може кодувати перший бін (`bin0`) з використанням засобу 102 контекстного моделювання і механізму 104 кодування з першим контекстом `ctx0`. Однак, кодер 20 відео може кодувати другий бін (`bin1`) і будь-які інші наступні біни, використовуючи обхідний кодер 106.

Аспекти даного розкриття винаходу також стосуються того, як пристрій 100 бінаризації виконує бінаризацію для відеоданих. Наприклад, відповідно до аспектів даного розкриття винаходу, пристрій 100 бінаризації може розділити синтаксичний елемент більше ніж на одну частину. Тобто, пристрій 100 бінаризації може використовувати кодування зрізаним унарним кодом, щоб кодувати префікс (при відносно невеликому `sMax`, що описано вище), і може використовувати інший спосіб кодування, щоб кодувати суфікс. В одному з прикладів, пристрій 100 бінаризації може використовувати експоненціальний код Голомба  $k$ -го порядку, щоб кодувати суфікс.

У деяких прикладах, можуть контекстно кодуватися тільки біни префікса, тоді як біни суфікса можуть кодуватися з обходом. Таблиця 3 показує приклад зрізаного унарного коду, скомбінованого з експоненціальним кодом Голомба, при `sMax=4` для префікса, і експоненціальний код Голомба 0-го порядку для суфікса. Ці методи також можуть

застосовуватися для значень опорних індексів, а також і інших синтаксичних елементів, у тому числі значень різниць векторів руху або інших синтаксичних елементів, використовуваних при кодуванні за допомогою удосконаленого прогнозування вектора руху (AMVP-advanced motion vector prediction).

5

Таблиця 3

Значення	Рядок бінів (зрізаний унарний)	Рядок бінів (експ. Голомба 0-го порядку)
0	0	
1	10	
2	110	
3	1110	
4	1111	0
5	1111	100
6	1111	101
7	1111	11000
8	1111	11001
9	1111	11010
10	1111	11011

У прикладі, показаному в Таблиці 3, біни зі зрізаним унарним кодом можуть контекстно кодуватися, тоді як біни з експоненціальним кодом Голомба можуть кодуватися з обходом.

10 Методи, згідно з даним розкриттям винаходу, можуть включати в себе, наприклад, застосування контекстного кодування до бінів після деякої кількості бінів для бінаризації експоненціальним кодом Голомба в префіксній частині. Методи, згідно з даним розкриттям винаходу, також включають в себе, наприклад, застосування кодування на основі контексту до деякого числа бінів (наприклад, попередньо заданого числа бінів префікса) і застосування

15 кодування з обходом до інших бінів. Наприклад, замість того, щоб кодувати всі біни в префіксній частині, використовуючи контекст, bin2 і наступні біни в префіксній частині можуть кодуватися в режимі з обходом. В іншому прикладі, режим з обходом може застосовуватися до всіх бінів після i/або включаючи bin1. У ще одному прикладі, режим з обходом може застосовуватися до всіх бінів префіксної частини. Аналогічний підхід, з використанням кодування в режимі з

20 обходом після деякого числа контекстно кодованих бінів, може використовуватися для будь-якого способу бінаризації. Тобто, хоча дане розкриття винаходу й описує використання схем кодування експоненціальним кодом Голомба і кодування зрізаним унарним кодом, можуть використовуватися й інші способи бінаризації.

25 Ще в одному прикладі, вищенаведені методи згідно з даним розкриттям винаходу можуть бути реалізовані разом з іншими процесами бінаризації, у тому числі з комбінаціями процесів бінаризації. Тобто, в одному прикладі, процес кодування унарним кодом може використовуватися для бінаризації значення опорного індексу. В іншому прикладі, процес кодування зрізаним унарним кодом може використовуватися для бінаризації значення опорного індексу. Ще в одному прикладі, процес кодування експоненціальним кодом Голомба може

30 використовуватися для бінаризації значення опорного індексу. Інші процеси бінаризації і комбінації процесів бінаризації також можливі. Тобто, наприклад, процес кодування унарним кодом (або зрізаним унарним кодом) може комбінуватися з процесом кодування експоненціальним кодом Голомба для бінаризації значення опорного індексу. В одному з ілюстративних прикладів, зрізаний унарний код довжини (4) може комбінуватися з

35 експоненціальним кодом Голомба (наприклад, експоненціальним кодом Голомба 0-го порядку). У такому прикладі, перша кількість бінів (наприклад, два, три, чотири або тому подібне) значення опорного індексу може кодуватися з унарним кодом, тоді як інші біни опорного індексу можуть кодуватися з експоненціальним кодом Голомба.

40 У будь-якому випадку, методи, викладені вище відносно CABAC-кодування і кодування з обходом значення опорного індексу, можуть бути застосовані до будь-якого бінаризованого значення опорного індексу. Тобто, відповідно до аспектів даного розкриття винаходу, перша кількість бінів бінаризованого значення опорного індексу може контекстно кодуватися (наприклад, кодуватися за допомогою механізму CABAC), тоді як інші біни можуть кодуватися з обходом. В описаному вище прикладі, у якому зрізаний унарний код довжини (4) комбінується з

45 експоненціальним кодом Голомба 0-го порядку, перші два біни (або будь-яка інша кількості



бінів) зрізаного унарного коду можуть контекстно кодуватися, а друга частина унарного коду і весь експоненціальний код Голомба можуть кодуватися з обходом.

Варто розуміти, що зрізаний унарний код довжини (4) і експоненціальний код Голомба 0-го порядку надані лише для прикладу, і ті ж або аналогічні методи можуть застосовуватися для зрізаного унарного коду іншої довжини, як і для експоненціального коду Голомба іншого порядку. Більше того, описані вище процеси бінаризації надані лише для прикладу і можуть використовуватися інші бінаризуючі коди. Наприклад, двійковий код фіксованої довжини може використовуватися замість експоненціального коду Голомба, описаного в наведених вище прикладах. Крім цього, приклад двох контекстно кодованих бінів для частини бінаризації зрізаним унарним кодом наданий для ілюстрації і може використовуватися інша кількість контекстно кодованих і кодованих з обходом бінів.

У будь-якому випадку, аспекти даного розкриття винаходу також стосуються зрізання частини бінаризованого значення. Наприклад, оскільки кількість опорних індексів відома попередньо, відповідно до аспектів даного розкриття винаходу, експоненціальний код Голомба або код фіксованої довжини може бути зрізаний. Тобто, експоненціальний код Голомба k-го порядку може використовуватися пристроєм 100 бінаризації. Як приклад, експоненціальний код Голомба 0-го порядку може застосовуватися при стисненні відео. Ця бінаризація складається з експоненціального префікса, кодованого унарним кодом, і суфікса фіксованої довжини довжиною (prefix-1), приклад чого показаний у Таблиці 4 нижче.

Таблиця 4

Елемент	Експоненціальний код Голомба 0-го порядку	Унарний код	Двійкове число
0	1	1	000
1	01 0	01	001
2	01 1	001	010
3	001 00	0001	011
4	001 01	00001	100
5	001 10	000001	101
6	001 11	0000001	110
7	0001 000	00000001	111
8	0001 001	000000001	...
9	0001 010	...	
10	0001 011		
11	0001 100		
12	0001 101		
13	0001 110		
14	0001 111		
15	00001 0000		
...	...		

Наприклад, значення 10 (наприклад, що відповідає елементу зі значенням 10 у першому стовпці Таблиці 4) представляється бінаризованим кодовим словом 0001011, де 0001 є префіксом, а 011 є суфіксом. Елемент може бути вхідними даними, що кодуються з використанням кодових слів з Таблиці 4 або з використанням таблиць, показаних і описаних нижче. Наприклад, кодер 20 відео може приймати елементи і перетворювати елементи в кодові слова відповідно до таблиць, показаних й описаних нижче. Точно так само, декодер 30 відео може приймати кодові слова і перетворювати кодові слова в синтаксичні елементи (наприклад, вхідні дані) відповідно до таблиць, показаних й описаних нижче.

Відновлене значення може бути одержане відповідно до рівняння (1), показаного нижче:  

$$\text{value} = 2^{(\text{prefix}-1)} + \text{suffix} - 1 \quad (1)$$

У цьому прикладі, префікс представлений унарним кодом, де 0001 відповідає 4, а суфікс є значенням, представленим у двійковій системі обчислення, де 011 відповідає 3, як показано у вищенаведеній Таблиці 4. Відповідно, у цьому прикладі, застосування рівняння (1) дає в результаті наступні значення:  $2^{(4-1)} + 3 - 1 = 10$ .

Цей код взагалі може представляти нескінченні числа; однак, у деяких сценаріях, число елементів може бути відоме. У цьому випадку, кодове слово може бути скорочене, з урахуванням максимального числа можливих елементів.

Наприклад, якщо максимальне число елементів дорівнює двом (наприклад, елементи 0 і 1), нормальним кодовим словом експоненціального коду Голомба для 1 є 010. Однак не існує елемента більше, ніж два. Відповідно, нормальний код 010 може бути скорочений до 1. Цей тип бінаризації може згадуватися як зрізаний експоненціальний код Голомба і може

використовуватися в таких стандартах кодування відео, як H.264/AVC. Однак, відносно стандарту H.264, зрізаний експоненціальний код Голомба використовується тільки тоді, коли максимальне число елементів дорівнює 1. Для інших випадків, використовується нормальна бінаризація експоненціальним кодом Голомба.

Відповідно до аспектів даного розкриття винаходу, нормальне кодування експоненціальним кодом Голомба може бути додатково зрізане, наприклад, як у наведеному вище прикладі, у якому максимальне число елементів дорівнює 1. Взагалі, якщо максимальне число елементів відоме попередньо, суфікс кодового слова бінаризації експоненціальним кодом Голомба може бути зрізаний за допомогою видалення надлишкових бітів. Наприклад, якщо максимальне число елементів дорівнює 9, то два біти, відмічені напівжирним курсивом і підкресленням у показаній нижче Таблиці 5, можуть бути видалені з кодових слів.

Таблиця 5

Елемент	Експоненціальний код Голомба 0-го порядку	Зрізаний експоненціальний код Голомба 0-го порядку
0	1	1
1	01 0	01 0
2	01 1	01 1
3	001 00	001 00
4	001 01	001 01
5	001 10	001 10
6	001 11	001 11
7	0001 000	0001 0
8	0001 001	01

Тобто, для елемента 7, показаного в Таблиці 5, перші два 00 суфікса можуть бути видалені. Крім цього, для елемента 8, показаного в Таблиці 5, перші два 00 суфікса можуть бути видалені. Відповідно, зрізаний експоненціальний код Голомба 0-го порядку показує 0001 0 для елемента 7 і 00001 1 для елемента 8.

Описані вище методи можуть бути реалізовані, наприклад, шляхом порівняння суфікса фіксованої довжини для останнього префікса (0001 у наведеному вище прикладі) з нормальним експоненціальним кодом Голомба. Наприклад, якщо число елементів в останній групі менше, ніж у нормальному експоненціальному коді Голомба, надлишкові біти можуть бути видалені. Іншими словами, пристрій 100 бінаризації може генерувати результуючий зрізаний експоненціальний код Голомба 0-го порядку, порівнюючи суфікс фіксованої довжини для останнього префікса, і, якщо число елементів у цій останній групі менше, ніж у нормальному експоненціальному коді Голомба, надлишкові біти можуть бути видалені.

Наприклад, у цьому прикладі, пристрій 100 бінаризації може визначити число елементів, префікс яких такий же, як префікс останнього елемента, коли є максимальне число елементів для кодування, яке відоме попередньо. Наприклад, у Таблиці 5, префікс для останнього елемента 0001 і є два елементи (наприклад, елемент 7 і елемент 8), префікс яких такий же, як префікс останнього елемента, коли є максимальне число елементів (наприклад, 9 у цьому прикладі).

Тоді пристрій бінаризації може порівняти число елементів, префікс яких такий же, як префікс останнього елемента, з числом елементів у нормальному експоненціальному коді Голомба з таким же префіксом. Наприклад, у наведеній вище Таблиці 4, є вісім елементів (тобто елементи від 7 до 14), префікс яких 0001 (тобто такий же, як префікс попереднього елемента). У цьому прикладі, пристрій 100 бінаризації може визначити, що число елементів, префікс яких такий же, як префікс останнього елемента, менше, ніж число елементів у нормальному експоненціальному коді Голомба з таким же префіксом.

Якщо це так, пристрій 100 бінаризації може зрізати біти з кодових слів, префікс яких такий же, як останній префікс, щоб генерувати зрізані кодові слова. У деяких прикладах, біти зрізаються із суфікса; однак, аспекти даного розкриття винаходу цим не обмежуються. Пристрій 100 бінаризації може визначати кількість бітів для зрізання, основуючись на кількості елементів, префікс яких такий же, як останній префікс.

Наприклад, у наведеній вище Таблиці 5, є два елементи з таким же префіксом, як останній префікс (наприклад, елементи 7 і 8). Пристрій 100 бінаризації може зрізати біни з кодових слів елементів 7 і 8, щоб генерувати зрізані кодові слова, що проілюстровано в останньому стовпці Таблиці 5. У цьому прикладі, оскільки є два елементи з таким же префіксом, як останній префікс, пристрій 100 бінаризації може визначити, що тільки один бін необхідний у суфіксі для представлення цих двох елементів. Наприклад, 0 у суфіксі може представляти один елемент (наприклад, елемент 7), а 1 у суфіксі може представляти інший елемент (наприклад, елемент 8). Відповідно, для елемента 7 у наведеній вище Таблиці 5, пристрій 100 бінаризації може зрізати перші два біни суфікса, залишаючи тільки 0 як суфікс для зрізаного кодового слова. Крім того, для елемента 8 у наведеній вище Таблиці 5, пристрій 100 бінаризації може зрізати перші два біни суфікса, залишаючи тільки 1 як суфікс для зрізаного кодового слова.

Описані вище методи можуть бути реалізовані для кодування аудіовізуальної інформації (наприклад, кодування і/або декодування відеоданих). Наприклад, відповідно до аспектів даного розкриття винаходу, декодер відео, такий як декодер 30 відео, може приймати одне або більше кодових слів, що характеризують аудіовізуальні дані, і може бути максимальне число елементів, які можуть використовуватися для кодування. Декодер 30 відео може переводити кодові слова в елементи відповідно до таблиці кодування. Таблиця кодування може бути побудована таким чином, що щонайменше деякі з кодових слів, які мають однаковий префікс, зрізаються, якщо цей же префікс є останнім префіксом у таблиці кодування, і кількість кодових слів, які мають однаковий префікс, менше, ніж максимальне число унікальних кодових слів, що могли б мати такий же префікс. Наприклад, для префікса 0001, Таблиця 4 ілюструє унікальні можливості для кодових слів, а Таблиці 5 і 6 (продемонстровані вище і нижче) демонструють приклади кодових слів, що спільно використовують один і той же префікс і зрізаються відповідно до методів даного розкриття винаходу.

Методи також можуть бути виконані кодером 20 відео. Наприклад, кодер 20 відео може приймати один або більше елементів, що характеризують аудіовізуальні дані. Кодер 20 відео може переводити елементи в одне або більше кодових слів відповідно до таблиці кодування, і може бути максимальне число елементів, які можуть використовуватися для кодування. Таблиця кодування може бути побудована таким чином, що щонайменше деякі з кодових слів, які мають однаковий префікс, зрізаються, якщо цей же префікс є останнім префіксом у таблиці кодування, і кількість кодових слів, які мають однаковий префікс, менше, ніж максимальне число унікальних кодових слів, що могли б мати такий же префікс. Знову ж, наприклад, для префікса 0001, Таблиця 4 ілюструє унікальні можливості для кодових слів, а Таблиці 5 і 6 (продемонстровані вище і нижче) демонструють приклади кодових слів, що спільно використовують один і той же префікс і зрізаються відповідно до методів даного розкриття винаходу.

Таким чином, методи можуть зменшити кількість бінів, необхідних для кодування відеоданих, коли відоме максимальне число кодованих елементів. Зменшення бінів приводить до меншої кількості бітів, які необхідно сигналізувати або прийняти, що дає ефективність використання смуги пропускання.

Як ще один приклад, при максимальному числі елементів, що дорівнює 11, у наведеній нижче Таблиці 6 продемонстровані кодові слова для зрізаного експоненціального коду Голомба. Біни, відмічені напівжирним курсивом і підкресленням у Таблиці 6, можуть бути видалені з кодових слів.

Таблиця 6

Елемент	Експоненціальний код Голомба 0-го порядку	Зрізаний експоненціальний код Голомба 0-го порядку
0	1	1
1	01 0	01 0
2	01 1	01 1
3	001 00	001 00
4	001 01	001 01
5	001 10	001 10
6	001 11	001 11
7	0001 000	0001 00
8	0001 001	0001 01
9	0001 010	0001 10
10	0001 011	0001 11

Як показано в Таблиці 6, перші біни в суфіксі для елементів 7, 8, 9 і 10 можуть бути зрізані (позначені напівжирним курсивом і підкресленням). У цьому прикладі, тільки один бін із суфікса може бути зрізаний, тому що чотири елементи представлені кодовим словом. З цієї причини, в зрізаних кодових словах, суфікс починається з 00 і закінчується на 11, щоб охопити чотири елементи, кожний з яких має один і той же префікс.

Приклади, продемонстровані в наведених вище Таблицях 5 і 6, надані лише як приклади, і такий же процес може бути застосований для будь-якої кількості максимальних елементів. Так, у деяких прикладах, аспекти даного розкриття винаходу стосуються прийому зрізаних кодових слів. Зрізані кодові слова можуть бути згенеровані шляхом визначення першої кількості елементів. Перша кількість елементів може показувати, скільки кодових слів у першій таблиці кодування має префікс, який є таким же, як префікс кодового слова, що відповідає останньому елементу в першій таблиці кодування, коли є максимальне число елементів, які можуть використовуватися для кодування. У цьому прикладі, першою таблицею кодування може бути Таблиця 5 або Таблиця 6. Аспекти стосуються прийому зрізаних кодових слів, які попередньо відсортовані або обчислюються на ходу під час роботи.

Аспекти даного розкриття винаходу також стосуються визначення другої кількості елементів, яка показує, скільки кодових слів у другій таблиці кодування має префікс, що є таким же, як префікс кодових слів, які відповідають останньому елементу в першій таблиці кодування. У цьому прикладі, другою таблицею кодування може бути наведена вище Таблиця 4. У деяких прикладах, перша таблиця кодування може бути підмножиною другої таблиці кодування, що оснований на максимальному числі елементів, які можуть використовуватися для кодування.

У деяких прикладах, якщо перша кількість елементів менше, ніж друга кількість елементів, аспекти даного розкриття винаходу стосуються зрізання бінів з кодових слів у першій таблиці кодування, префікс яких такий же, як префікс кодового слова, що відповідає останньому елементу в першій таблиці кодування, щоб генерувати зрізані кодові слова, і кодування відеоданих з використанням зрізаних кодових слів. У деяких прикладах, зрізання кодових слів включає в себе зрізання бінів з суфіксів або префіксів кодових слів або поєднання цього. У деяких прикладах, зрізання бінів ґрунтується на першій кількості елементів, причому перша кількість елементів показує, скільки кодових слів у першій таблиці кодування має префікс, який є таким же, як префікс кодового слова, що відповідає останньому елементу в першій таблиці кодування. У деяких прикладах, кодування є кодуванням кодом Голомба.

Як альтернатива або додатково, префікс теж може бути зрізаний з використанням зрізаного унарного коду. Наприклад, якщо максимальне число елементів дорівнює 4, то префікс і суфікс можуть бути зрізані, як показано в Таблиці 7 нижче.

Таблиця 7

Елемент	Експоненціальний код Голомба 0-го порядку	Зрізаний експоненціальний код Голомба 0-го порядку
0	1	1
1	01 0	01 0
2	01 1	01 1
3	001 00	00

Зрізані біни в Таблиці 7 представлені напівжирним курсивом і підкресленням. У прикладі, продемонстрованому в Таблиці 7, кодове слово для елемента 3 коротше, ніж кодові слова для елементів 1 або 2. Додаткове переупорядкування або зіставлення для бінаризації зрізаним експоненціальним кодом Голомба може бути застосоване, наприклад, шляхом призначення більш короткого кодового слова 00 елементу 1, що частіше зустрічається, а 010 елементу 3. Таке переупорядкування або зіставлення може бути виконане з використанням таблиць зіставлення.

У деяких прикладах, переупорядкування також може ґрунтуватися на частоті появи конкретних елементів. Наприклад, більш короткі ключові слова можуть призначатися елементам, що найбільш часто зустрічаються. Це зіставлення кодових слів може бути особливо ефективним у тих випадках, коли елементи упорядковані по частоті появи.

Тоді як визначені наведені вище приклади були описані відносно кодування експоненціальним кодом Голомба 0-го порядку, варто розуміти, що в більш загальному розумінні методи застосовні до кодування експоненціальним кодом Голомба k-го порядку. Більше того, методи не обмежуються відеостандартом HEVC, а можуть застосовуватися до

будь-якого стандарту стиснення відео або, у більш широкому розумінні, для будь-якого додатка, у якому виконується бінаризація.

Що стосується нового стандарту HEVC (а також розширень до стандарту HEVC, таких як масштабоване кодування відео (SVC-scalable video coding) або багатовидове кодування відео (MVC-multi-view video coding)), описані вище методи бінаризації зрізаним експоненціальним кодом Голомба можуть застосовуватися до бінаризації різноманітних синтаксичних елементів. Приклади включають в себе значення опорних індексів, внутрішній режим, індекс злиття, параметри квантування (або дельта параметрів квантування) або будь-який інший синтаксичний елемент, для яких попередньо відома кількість елементів.

Тоді як наведений вище приклад описує зрізання експоненціального коду Голомба, описувані методи зрізання також можуть застосовуватися до коду фіксованої довжини. Тобто, у прикладах, у яких синтаксичний елемент (наприклад, опорний індекс) бінаризується з використанням більше ніж одного процесу бінаризації (наприклад, зрізаним унарним кодом і експоненціальним кодом Голомба), попередньо задане число бінів може кодуватися за допомогою CABAC, тоді як інші біни можуть бути зрізані і кодуватися з обходом.

У деяких прикладах, може застосовуватися алгоритм для визначення кількості бінів, що можуть бути зрізані (наприклад, зрізані з частини бінаризації синтаксичного елемента за допомогою експоненціального коду Голомба або коду фіксованої довжини). В одному з прикладів, передбачимо, що попередньо задана кількість бінів залишається для кодування з обходом. У цьому прикладі, відеокoder (такий як coder 20 відео або декодер 30 відео) може визначати кількість бінів, що залишилися, які можуть бути зрізані, обчислюючи округлений у більшу сторону  $\log_2$  для бінів, що залишилися.

Фіг. 5A і 5B є структурними схемами, що ілюструють типові рядки бінів, пов'язані з даними прогнозування. Наприклад, Фіг. 5A загалом ілюструє опорний індекс ( $\text{ref\_idx}$ ), різницю ( $\text{mvd}$ ) векторів руху і індекс ( $\text{mvr\_idx}$ ) предиктора вектора руху для зображення, що прогнозується, виходячи з одного опорного зображення.

Фіг. 5B загалом ілюструє перший опорний індекс ( $\text{ref\_idx\_L0}$ ), першу різницю ( $\text{mvd\_L0}$ ) векторів руху (що представляє горизонтальну складову і вертикальну складову) і перший індекс ( $\text{mvr\_idx\_L0}$ ) предиктора вектора руху, а також другий опорний індекс ( $\text{ref\_idx\_L1}$ ), різницю ( $\text{mvd\_L1}$ ) векторів руху (що представляє горизонтальну складову і вертикальну складову) і другий індекс ( $\text{mvr\_idx\_L1}$ ) предиктора вектора руху для зображення, що прогнозується, виходячи з двох опорних зображень (B-зображення). Тобто, для PU з подвійним прогнозуванням, можуть кодуватися два опорних індекси, з одним опорним індексом для кожного списку зі списків L0 і L1. Відповідно, до двох опорних індексів може кодуватися для кожної PU і до восьми індексів може кодуватися для кожної CU.

Рядки 120 (Фіг. 5A) і 124 (Фіг. 5B) бінів включають в себе дані прогнозування, пов'язані з методом удосконаленого прогнозування вектора руху (AMVP). При AMVP, вектор руху для кодованого в даний момент блока може кодуватися як різницеве значення (тобто дельта) відносно іншого вектора руху, такого як вектор руху, пов'язаний із сусіднім у просторі або часі блоком. Наприклад, coder 20 відео може формувати список кандидатів-предикторів вектора руху, що включає в себе вектори руху, пов'язані з одним або більше сусідніми блоками в просторовому і часовому напрямках. Coder 20 відео може вибрати найбільш точний предиктор (MVP) вектора руху зі списку кандидатів на основі, наприклад, аналізу витрат залежно від швидкості і спотворення.

Coder 20 відео може позначити опорне зображення для дійсного вектора руху, використовуючи опорний індекс ( $\text{ref\_idx}$ ). Крім цього, coder 20 відео може позначити вибраний MVP, використовуючи індекс ( $\text{mvr\_idx}$ ) предиктора вектора руху, що ідентифікує MVP у списку кандидатів. Coder 20 відео також може позначити різницю між вектором руху поточного блока (дійсним вектором руху) і MVP, використовуючи різницю ( $\text{mvd}$ ) векторів руху. Як зазначалося вище,  $\text{mvd}$  може включати в себе горизонтальну складову і вертикальну складову, відповідну горизонтальній складовій і вертикальній складовій  $\text{mvr}$ .

Декодер 30 відео таким же чином може формувати список кандидатів для MVP. Потім декодер 30 відео може використовувати прийнятий індекс ( $\text{mvr\_idx}$ ) предиктора вектора руху, щоб визначити, де знайти MVP у списку кандидатів. Декодер 30 відео може об'єднати різницю ( $\text{mvd}$ ) векторів руху з предиктором вектора руху (визначеним з використанням індексу ( $\text{mvr\_idx}$ ) предиктора вектора руху), щоб відновлювати вектор руху.

Прогнозування векторів руху таким чином (наприклад, за допомогою різницевого значення) може вимагати меншої кількості бітів, що будуть включені в бітовий потік, у порівнянні з кодуванням дійсного вектора(ів) руху. Що стосується Фіг. 5B, зображення з подвійним прогнозуванням можуть включати в себе дані прогнозування, пов'язані з зображеннями з двох

різних списків, наприклад списку 0 і списку 1. Як показано в прикладі на Фіг. 5B, дані прогнозування, пов'язані зі списком 0, можуть передувати даним прогнозування, що пов'язані зі списком 1. Тобто, рядок 124 бітів включає в себе перший опорний індекс (`ref_idx_L0`), першу різницю (`mvd_L0`) векторів руху (наприклад, що представляє і горизонтальну, і вертикальну складові) і перший індекс (`mrp_idx_L0`) предиктора вектора руху, за якими ідуть другий опорний індекс (`ref_idx_L1`), друга різниця (`mvd_L1`) векторів руху (наприклад, що представляє і горизонтальну, і вертикальну складові) і другий індекс (`mrp_idx_L1`) предиктора вектора руху.

У деяких прикладах, синтаксична структура, пов'язана з AMVP, може кодуватися з використанням поєднання контекстного кодування і кодування з обходом. Наприклад, як показано в прикладах на Фіг. 5A і 5B, деякі з бітів даних прогнозування контекстно кодуються, у той час як інші біти кодуються з обходом. Тобто, один або більше бітів значень різниць векторів руху (а також і інші значення, такі як значення опорних індексів, що описано нижче з посиланням на Фіг. 7) можуть контекстно кодуватися, у той час як один або більше інших бітів значень різниць векторів руху можуть кодуватися з обходом.

Що стосується прикладу на Фіг. 5A, опорний індекс (`ref_idx`) і перша частина різниці (`mvd`) векторів руху можуть бути контекстно кодованими, про що свідчать контекстно кодовані біти 128. Друга частина різниці (`mvd`) векторів руху може бути кодовою з обходом, про що свідчать кодовані з обходом біти 130. Крім цього, індекс (`mrp_idx`) предиктора вектора руху може бути контекстно кодованим, про що свідчать контекстно кодовані біти 132.

Що стосується прикладу на Фіг. 5B, перший опорний індекс (`ref_idx_L0`) і перша частина першої різниці (`mvd_L0`) векторів руху (наприклад, що представляє і горизонтальну, і вертикальну складові) можуть бути контекстно кодованими, про що свідчать контекстно кодовані біти 136. Друга частина першої різниці (`mvd_L0`) векторів руху (наприклад, що представляє і горизонтальну, і вертикальну складові) може бути кодовою з обходом, про що свідчать кодовані з обходом біти 138. Крім цього, перший індекс (`mrp_idx_L0`) предиктора вектора руху, другий опорний індекс (`ref_idx_L1`) і перша частина другої різниці (`mvd_L1`) векторів руху (наприклад, що представляє і горизонтальну, і вертикальну складові) можуть бути контекстно кодованими, про що свідчать контекстно кодовані біти 140. Друга частина другої різниці (`mvd_L1`) векторів руху (наприклад, що представляє і горизонтальну, і вертикальну складові) теж може бути кодовою з обходом, про що свідчать кодовані з обходом біти 142. Нарешті, другий індекс (`mrp_idx_L1`) предиктора вектора руху може бути контекстно кодованим, про що свідчать контекстно кодовані біти 144.

Відповідно, приклад на Фіг. 5B ілюструє синтаксичну структуру на основі PU з внутрішнім режимом для подвійного прогнозування, для якого відеокодеру, можливо, буде потрібно перемикатися чотири рази між контекстним кодуванням і кодуванням з обходом, щоб обробити біти. Перемикання між контекстним кодуванням і кодуванням з обходом для кодування рядків 120 і 124 бітів може бути неефективним. Наприклад, перемикання між контекстним кодуванням і кодуванням з обходом може витратити один або більше тактових циклів. Відповідно, перемикання між контекстним кодуванням і кодуванням з обходом для кожного елемента може привнести запізнювання, внаслідок переходів між контекстним кодуванням і кодуванням з обходом.

Відповідно до аспектів даного розкриття винаходу, як докладніше описано нижче відносно Фіг. 6 і 7, контекстні біти й обхідні біти можуть бути згруповані, щоб зменшити переходи між контекстним кодуванням і кодуванням з обходом. Наприклад, відносно Фіг. 5A, аспекти даного розкриття винаходу стосуються групування контекстно кодованих бітів 128 і 132 разом так, щоб ці біти не розділялися кодованими з обходом бітами 130. Таким чином, може бути зроблений єдиний перехід між контекстним кодуванням і кодуванням з обходом у ході кодування рядка 120 бітів.

Точно так само, відносно Фіг. 5B, аспекти даного розкриття винаходу стосуються групування контекстно кодованих бітів 136, 140 і 144 так, щоб ці біти не розділялися кодованими з обходом бітами 138 і 142. Знову ж, групування контекстно кодованих бітів і кодованих з обходом бітів таким чином може дозволити відеокодеру (такому як кодер 20 відео або декодер 30 відео) робити єдиний перехід між контекстним кодуванням і кодуванням з обходом. Усунення множинних переходів між контекстним кодуванням і кодуванням з обходом може підвищити ефективність за рахунок обмеження запізнювання, пов'язаного з переходами.

Фіг. 6 є структурною схемою, яка ілюструє інший типовий рядок 140 бітів даних прогнозування. Рядок 140 бітів включає в себе перший опорний індекс (`ref_idx_L0`), першу різницю (`mvd_L0`) векторів руху (наприклад, що представляє і горизонтальну, і вертикальну складові) і перший індекс (`mrp_idx_L0`) предиктора вектора руху, за якими ідуть другий опорний індекс (`ref_idx_L1`), друга різниця (`mvd_L1`) векторів руху (наприклад, що представляє і

горизонтальну, і вертикальну складові) і другий індекс (*mvp\_idx\_L1*) предиктора вектора руху. Рядок 140 бітів включає в себе контекстно кодовані біти 144 і кодовані з обходом біти 148. Наприклад, контекстно кодовані біти 144 можуть кодуватися з використанням режиму контекстного кодування процесу контекстно-адаптивного кодування (наприклад, САВАС), у той час як кодовані з обходом біти 148 можуть кодуватися з використанням фіксованої імовірності (наприклад, режим кодування з обходом САВАС).

Відповідно до аспектів даного розкриття винаходу, контекстно кодовані біти 144 групуються для кодування до кодованих з обходом бітів 148. Тобто, у прикладі, показаному на Фіг. 6, контекстно кодовані біти 144 включають в себе контекстно кодовані біти першого опорного індексу (*ref\_idx\_L0*), контекстно кодовані біти другого опорного індексу (*ref\_idx\_L1*), контекстно кодовані біти першого індексу (*mvp\_idx\_L0*) предиктора вектора руху, контекстно кодовані біти другого індексу (*mvp\_idx\_L1*) предиктора вектора руху, контекстно кодовані біти першої різниці (*mvd\_L0*) векторів руху і контекстно кодовані біти другої різниці (*mvd\_L1*) векторів руху. Крім цього, кодовані з обходом біти 148 включають в себе кодовані з обходом біти першої різниці (*mvd\_L0*) векторів руху і кодовані з обходом біти другої різниці (*mvd\_L1*) векторів руху.

У деяких прикладах, біти можуть бути згруповані на основі номерів бітів синтаксичних елементів, що підлягають кодуванню. В одному з ілюстративних прикладів, відеокодер (такий як кодер 20 відео або декодер 30 відео) може кодувати перші два біти значень різниць векторів руху, використовуючи контекстне кодування, і кодувати інші біти, використовуючи кодування з обходом. Відповідно, у цьому прикладі, контекстно кодовані біти 144 можуть включати в себе перші два біти і значення першої різниці (*mvd\_L0*) векторів руху і значення (*mvd\_L1*) другої різниці векторів руху, у той час як кодовані з обходом біти 148 можуть включати в себе інші біти значень різниць векторів руху.

Упорядкування синтаксичних елементів, показане на Фіг. 6, надане лише для прикладу. В інших прикладах, синтаксичні елементи можуть бути упорядковані по-іншому, наприклад по значеннях опорних індексів, предикторах вектора руху і значеннях різниць векторів руху в альтернативному порядку (або з чергуванням). Тобто, в іншому прикладі, відеокодер може кодувати PU наступним чином: контекстно кодовані біти опорного індексу *L0*, контекстно кодовані біти опорного індексу *L1*, кодовані в режимі з обходом біти опорного індексу *L0*, кодовані в режимі з обходом біти опорного індексу *L1*. У додаткових прикладах, обхідні біти 148 можуть кодуватися до контекстно кодованих бітів 144.

У будь-якому випадку, Фіг. 6 демонструє дані прогнозування, що мають групу з одного або більше контекстно кодованих бітів 144 і групу з одного або більше кодованих з обходом бітів 148. Групування контекстно кодованих бітів і кодованих з обходом бітів таким чином, як зазначалося вище, може зменшити запізнювання, пов'язане з кодуванням бітів. Наприклад, відеокодер може кодувати всі контекстно кодовані біти 144, зробити єдиний перехід від контекстного кодування до кодування з обходом і кодувати всі кодовані з обходом біти. Усунення множинних переходів між контекстним кодуванням і кодуванням з обходом може підвищити ефективність за рахунок обмеження запізнювання, пов'язаного з переходами між контекстним кодуванням і кодуванням з обходом.

У деяких прикладах, як показано на Фіг. 6, контекстно кодовані біти і кодовані з обходом біти можуть бути згруповані для блока відеоданих (наприклад, по кожній PU). В інших прикладах, контекстно кодовані біти і кодовані з обходом біти можуть бути згруповані для CU (наприклад, однієї або більше CU з LCU), для повної LCU або для повного слайсу відеоданих. У таких прикладах, контекстно кодовані біти можуть групуватися і кодуватися для CU/LCU/слайсу до кодованих з обходом бітів з CU/LCU/слайсу, або навпаки.

Фіг. 7 є структурною схемою, яка ілюструє інший типовий рядок 160 бітів даних прогнозування. У прикладі на Фіг. 7, рядок 160 бітів включає в себе перший опорний індекс (*ref\_idx\_L0*), першу різницю (*mvd\_L0*) векторів руху (наприклад, що представляє і горизонтальну, і вертикальну складові) і перший індекс (*mvp\_idx\_L0*) предиктора вектора руху, за якими ідуть другий опорний індекс (*ref\_idx\_L1*), друга різниця (*mvd\_L1*) векторів руху (наприклад, що представляє і горизонтальну, і вертикальну складові) і другий індекс (*mvp\_idx\_L1*) предиктора вектора руху. Рядок 160 бітів включає в себе контекстно кодовані біти 164 і кодовані з обходом біти 168. Наприклад, контекстно кодовані біти 164 можуть кодуватися з використанням режиму контекстного кодування процесу контекстно-адаптивного кодування (наприклад, САВАС), у той час як кодовані з обходом біти 168 можуть кодуватися з використанням фіксованої імовірності (наприклад, режим кодування з обходом САВАС).

Відповідно до аспектів даного розкриття винаходу, аналогічно прикладу, показаному на Фіг. 6, контекстно кодовані біти 164 групуються для кодування до обхідних бітів 168. Однак, у прикладі, показаному на Фіг. 7, опорні індекси (*ref\_idx\_L0* і *ref\_idx\_L1*) включають в себе

поєднання контекстно кодованих бінів поряд з кодованими з обходом бінами. Тобто, опорні індекси можуть кодуватися відповідно до прикладів, описаних вище відносно Фіг. 4, з одним або більше бінами, кодованими з використанням контекстно-адаптивного режиму, і одним або більше іншими бінами, кодованими з використанням режиму з обходом.

Відповідно, у прикладі на Фіг. 7, контекстно кодовані біни 164 включають в себе контекстно кодовані біни першого опорного індексу (`ref_idx_L0`), контекстно кодовані біни другого опорного індексу (`ref_idx_L1`), контекстно кодовані біни першого індексу (`mvp_idx_L0`) предиктора вектора руху, контекстно кодовані біни другого індексу (`mvp_idx_L1`) предиктора вектора руху, контекстно кодовані біни першої різниці (`mvd_L0`) векторів руху (наприклад, що представляє і горизонтальну, і вертикальну складові) і контекстно кодовані біни другої різниці (`mvd_L1`) векторів руху (наприклад, що представляє і горизонтальну, і вертикальну складові). Крім цього, кодовані з обходом біни 168 включають в себе кодовані з обходом біни першого опорного індексу (`ref_idx_L0`), кодовані з обходом біни другого опорного індексу (`ref_idx_L1`), кодовані з обходом біни першої різниці (`mvd_L0`) векторів руху і кодовані з обходом біни другої різниці (`mvd_L1`) векторів руху.

Як описано вище відносно Фіг. 6, біни можуть бути згруповані на основі номерів бінів синтаксичних елементів, що підлягають кодуванню. В одному з ілюстративних прикладів, відеокодер (такий як кодер 20 відео або декодер 30 відео) може кодувати перші два біни значення різниць векторів руху, використовуючи контекстне кодування, і кодувати інші біни, використовуючи кодування з обходом. Крім цього, відеокодер може кодувати перші два біни опорних індексів, використовуючи контекстне кодування, і кодувати інші біни, використовуючи кодування з обходом. Відповідно, у цьому прикладі, контекстно кодовані біни 144 можуть включати в себе перші два біни і значення першої різниці (`mvd_L0`) векторів руху і значення другої різниці (`mvd_L1`) векторів руху, а також перші два біни і значення першого опорного індексу (`ref_idx_L0`) і значення другого опорного індексу (`ref_idx_L1`). Кодовані з обходом біни 148 можуть включати в себе інші біни значень різниць векторів руху й опорних індексів.

Групування контекстно кодованих бінів і кодованих з обходом бінів може зменшити запізнювання, пов'язане з кодуванням бінів. Наприклад, відеокодер може кодувати всі контекстно кодовані біни 144, зробити єдиний перехід від контекстного кодування до кодування з обходом і кодувати всі кодовані з обходом біни. Усунення множинних переходів між контекстним кодуванням і кодуванням з обходом може підвищити ефективність за рахунок обмеження запізнювання, пов'язаного з переходами між контекстним кодуванням і кодуванням з обходом.

Варто розуміти, що методи, описані відносно Фіг. 6 і 7, можуть бути виконані з використанням будь-якої схеми бінаризації. Більше того, як зазначалося вище, упорядкування синтаксичних елементів надане лише для прикладу. Крім цього, контекстно кодовані біни і кодовані з обходом біни можуть бути згруповані для PU, для однієї або більше CU, для повної LCU або для повного слайсу відеоданих. У таких прикладах, контекстно кодовані біни можуть групуватися і кодуватися для PU/CU/LCU/слайсу до кодованих з обходом бінів з PU/CU/LCU/слайсу, або навпаки.

Фіг. 8A і 8B загалом ілюструють кодування синтаксичного елемента в напрямку зовнішнього прогнозування. Наприклад, як зазначалося вище, на доповнення до обчислення вектора руху, оцінка руху до того ж визначає індекс опорного кадру (`ref_idx`) і напрямок прогнозування для B-слайсів (`inter_pred_idc`: вперед від L0, назад від L1, або двоспрямоване, або `inter_pred_flag`: в одному напрямку від LC або в двох напрямках від L0 і L1). Вектор руху (наприклад, горизонтальна складова і вертикальна складова вектора руху), індекс опорного кадру і напрямок прогнозування звичайно ентропійно кодуються кодером як синтаксичні елементи і вміщуються в закодований бітовий потік відео, що буде декодуватися декодером відео для використання в процесі декодування відео. Ці синтаксичні елементи можуть бути передбачені для PU із зовнішнім кодуванням, щоб дозволити декодеру відео декодувати і відтворювати відеодані, що задаються за допомогою PU.

У деяких випадках, опорні індекси списку L0 і списку L1 можуть використовуватися, щоб сигналізувати опорні індекси для режиму двоспрямованого прогнозування (`Pred_BI`), а опорний індекс об'єднаного списку (`CL`) використовується, щоб сигналізувати опорний індекс для режиму (`Pred_LC`) односпрямованого прогнозування. Опорний індекс LC є опорним індексом об'єднаного списку опорних зображень, що включає в себе об'єднання опорних зображень зі списків L0 і L1 з дубльованими опорними зображеннями, видаленими відповідно до попереднього заданого правила (або явної сигналізації). Опорний індекс LC, отже, зіставляється з опорним індексом для одного зі списків L0 або L1.



- У таких випадках, синтаксичний елемент напрямку зовнішнього прогнозування (`inter_pred_flag`) має тільки два можливих значення (у двох напрямках або в одному напрямку від LC). Коли синтаксичний елемент напрямку зовнішнього прогнозування бінаризується, може знадобитися кодування тільки одного біна, щоб позначити напрямок зовнішнього прогнозування як або в двох напрямках, або в одному напрямку. Таблиця 8, показана нижче, ілюструє синтаксичний елемент напрямку зовнішнього прогнозування.

Таблиця 8

Тип слайсу	<code>inter_pred_flag</code>	Ім'я <code>inter_pred_flag</code>
P	<code>inferred</code>	<code>Pred_L0</code>
B	0	<code>Pred_LC</code>
	1	<code>Pred_BI</code>

- В інших прикладах, як пропонується в представленні Lee T. і Park J. "On Reference List Combination" ("Про об'єднання опорних списків"), JCTVC-I0125, Женева, квітень 2012, об'єднаний список (LC) може бути видалений. У таких прикладах, замість цього використовують синтаксичний елемент напрямку зовнішнього прогнозування (`inter_pred_flag` або `inter_pred_idc`) з трьома можливими значеннями (у двох напрямках, в одному напрямку від L0 або в одному напрямку від L1). Коли режим прогнозування є режимом прогнозування в одному напрямку, може знадобитися кодування додаткового біна, що позначає або `Pred_L0`, або `Pred_L1`. Таблиця 9 ілюструє зміну в кодуванні синтаксичного елемента напрямку зовнішнього прогнозування (відносно наведеної вище Таблиці 8).

Таблиця 9

Тип слайсу	<code>inter_pred_idc</code>	Ім'я <code>inter_pred_idc</code>
P	<code>inferred</code>	<code>Pred_L0</code>
B	0	<code>Pred_L0</code>
	1	<code>Pred_L1</code>
	2	<code>Pred_BI</code>

- Фіг. 8A ілюструє структуру кодування, описану вище відносно Таблиці 9. Як показано на Фіг. 8A, синтаксичний елемент напрямку зовнішнього прогнозування (`inter_pred_flag`) може кодуватися за допомогою процесу CABAC з використанням двох бінів. Перший бін (`bin(0)`) показує, є режим зовнішнього прогнозування режимом в одному напрямку (`bin(0)=0`) або в двох напрямках (`pred_BI`) (`bin(0)=1`). Другий бін умовно кодується, тільки якщо перший бін показує режим прогнозування в одному напрямку. Другий бін (`bin(1)`) показує, спрямований режим прогнозування в одному напрямку від List 0 (`pred_L0`) (`bin(1)=0`) або від List 1 (`pred_L1`) (`bin(1)=1`).

- Виведення індексу контекстної моделі для напрямку зовнішнього прогнозування тільки з двома можливими значеннями (у двох напрямках або в одному напрямку від LC) можуть визначатися на основі глибини CU, що може мати значення в діапазоні 0...3 для першого біна (`bin0`), як показано в наведеному нижче рівнянні:

$$ctxIdx = cuDepth.$$

- У прикладі на Фіг. 8A, другий бін (`bin(1)`) може кодуватися з додатковим контекстом або може кодуватися за допомогою повторного використання одного контексту першого біна (`bin(0)`). Однак введення додаткового контексту може збільшити складність, пов'язану з кодуванням синтаксичного елемента напрямку зовнішнього прогнозування. Більше того, повторне використання одного з контекстів першого біна може зменшити кількість контекстів, використовуваних для кодування `bin0`, і відеокодеру буде потрібно виконувати додаткову перевірку цієї умови.

- Фіг. 8B є структурною схемою, яка ілюструє кодування з обходом синтаксичного елемента напрямку зовнішнього прогнозування, відповідно до аспектів даного розкриття винаходу. Як показано на Фіг. 8B, перший бін (`bin(0)`) синтаксичного елемента напрямку зовнішнього прогнозування (`inter_pred_flag`) може бути контекстом, кодованим за допомогою процесу CABAC, а другий бін (`bin(1)`) може кодуватися в режимі з обходом процесу CABAC. Перший бін (`bin(0)`) показує, є режим зовнішнього прогнозування режимом в одному напрямку (`bin(0)=0`) або в двох напрямках (`pred_BI`) (`bin(0)=1`). У цьому прикладі, `bin(0)` може кодуватися з використанням одного з чотирьох можливих контекстів, `ctxIdx=0...3`. Другий бін (`bin(1)`) може

умовно кодуватися, тільки якщо перший біт показує режим прогнозування в одному напрямку. Другий біт ( $\text{bin}(1)$ ) показує, спрямований режим прогнозування в одному напрямку від List 0 ( $\text{pred\_L0}$ ) ( $\text{bin}(1)=0$ ) або від List 1 ( $\text{pred\_L1}$ ) ( $\text{bin}(1)=1$ ). Відповідно до аспектів даного розкриття винаходу,  $\text{bin}(1)$  може кодуватися, не використовуючи ніяких контекстів (наприклад, кодуватися з використанням режиму з обходом процесу CABAC).

Таким чином, відповідно до аспектів даного розкриття винаходу, відеокoder (наприклад, coder 20 відео або декодер 30 відео) може бінаризувати значення напрямку зовнішнього прогнозування і кодувати щонайменше один біт бінаризованого значення напрямку зовнішнього прогнозування в режимі з обходом. Більш точно, відеокoder може вибрати контекст, щоб контекстно кодувати перший біт ( $\text{bin}(0)$ ) для значення напрямку зовнішнього прогнозування за допомогою процесу CABAC і кодувати другий біт ( $\text{bin}(1)$ ) у режимі з обходом процесу CABAC. Як альтернатива, ці методи дозволяють відеокoderу кодувати перший біт ( $\text{bin}(0)$ ) для значення напрямку зовнішнього прогнозування в режимі з обходом і також кодувати другий біт ( $\text{bin}(1)$ ) у режимі з обходом. Відповідно, чотири контексти, доступні для кодування  $\text{bin}(0)$ ,  $\text{ctxIdx}=0\dots3$ , можуть бути збережені.

Методи надають можливість кодування синтаксичного елемента напрямку зовнішнього прогнозування з трьома можливими значеннями (у двох напрямках, в одному напрямку від L0 або в одному напрямку від L1), не вимагаючи ніякого додаткового контексту або повторно використовуючи контекст (наприклад, контекст  $\text{bin0}$ ). Більше того, методи використовують режим з обходом, який не потребує контекстів, що може бути менш складним в обчислювальному відношенні, ніж контекстне кодування.

Фіг. 9 є блок-схемою послідовності операцій способу, яка ілюструє приклад ентропійного кодування значення опорного індексу, відповідно до аспектів даного розкриття винаходу. Хоч для цілей пояснення воно й описане загалом як виконване компонентами кодера 20 відео (Фіг. 1 і 2), варто розуміти, що інші модулі кодування відео, процесори, модулі обробки, апаратні модулі кодування, такі як кодери/декодери (КОДЕКИ) і т. п., також можуть бути виконані з можливістю виконання процесу, зображеного на Фіг. 9.

У прикладі на Фіг. 9, кодер 20 відео бінаризує прийнятий синтаксичний елемент (180). Кодер 20 відео може бінаризувати синтаксичний елемент відповідно до будь-якого з процесів бінаризації, описаних у даному розкритті винаходу. Приклади процесів бінаризації включають в себе бінаризацію унарним кодом, зрізаним унарним кодом, експоненціальним кодом Голомба або тому подібне.

Кодер 20 відео визначає, чи є бінаризований синтаксичний елемент значенням опорного індексу (182). У свою чергу, значення опорного індексу звичайно ідентифікує опорне зображення в списку опорних зображень для цілей зовнішнього прогнозування. Якщо бінаризований синтаксичний елемент є значенням опорного індексу (гілка "Так" етапу 182), кодер 20 відео може кодувати щонайменше один біт бінаризованого значення опорного індексу з використанням контекстно-адаптивного кодування, такого як CABAC (184). Крім цього, кодер 20 відео може кодувати щонайменше інший біт бінаризованого значення опорного індексу (у випадках, коли є додаткові біти, що підлягають кодуванню) з використанням кодування з обходом, яке обходить механізм контекстно-адаптивного кодування (186).

Як описувалося вище відносно Фіг. 4, кодер 20 відео, у деяких прикладах, може кодувати один, два або три біти, використовуючи контекстне кодування. Для контекстно кодованих бітів, кодер 20 відео може вибрати контекст на основі відносної позиції біта в рядку бітів. Наприклад, кодер 20 відео може вибрати контекст для першого біта, що відрізняється від контексту для другого біта.

У будь-якому випадку, кодер 20 відео може об'єднувати контекстно кодовані біти і кодовані з обходом біти для формування закодованого бітового потоку 188. У деяких прикладах, якщо кодований синтаксичний елемент не є значенням опорного індексу (гілка "Ні" етапу 182), кодер 20 відео може вибрати конкретний режим кодування (наприклад, з обходом або контекстно-адаптивний), щоб кодувати синтаксичний елемент (190). Кодер 20 відео може закодувати синтаксичний елемент з використанням вибраного режиму (192) і сформувати закодований бітовий потік (188).

Крім того, варто розуміти, що етапи, показані й описані з посиланням на Фіг. 9, надані просто як один приклад. Тобто, етапи способу, зображеного на Фіг. 9, не обов'язково повинні виконуватися в порядку, показаному на Фіг. 9, і може виконуватися менше етапів, додаткові або альтернативні етапи. Наприклад, у деяких випадках, кодер 20 відео може визначати, чи є синтаксичний елемент опорним індексом (крок 182), перед бінаризацією синтаксичного елемента (крок 180).

Фіг. 10 є блок-схемою послідовності операцій способу, яка ілюструє приклад ентропійного декодування значення опорного індексу, відповідно до аспектів даного розкриття винаходу. Хоч для цілей пояснення воно й описано загалом як виконване компонентами декодера 30 відео (Фіг. 1 і 3), варто розуміти, що інші модулі кодування відео, процесори, модулі обробки, апаратні модулі кодування, такі як кодери/декодери (КОДЕКИ) і т. п., також можуть бути виконані з можливістю виконання процесу, зображеного на Фіг. 10.

Декодер 30 відео спочатку може проводити синтаксичний аналіз кодованого синтаксичного елемента з закодованого бітового потоку (200). Наприклад, декодер 30 відео може зчитувати і сегментувати кодований синтаксичний елемент із закодованого бітового потоку відповідно до конкретного процесу синтаксичного аналізу (наприклад, хвильового синтаксичного аналізу). Кодований синтаксичний елемент може включати в себе множину кодованих бінів, тобто двійкових значень.

Декодер 30 відео також може визначити, чи є частина бітового потоку, декодована в даний момент, значенням опорного індексу (202). Якщо декодер 30 відео декодує значення опорного індексу (гілка "Так" етапу 202), декодер 30 відео може декодувати щонайменше один бін з використанням контекстно-адаптивного кодування (204). Крім цього, декодер 30 відео може декодувати щонайменше інший бін (у прикладах, коли є додаткові біни, що підлягають кодуванню) з використанням кодування з обходом (206). Як зазначалося вище відносно Фіг. 9, у деяких прикладах, декодер 30 відео може кодувати один, два або три біни, використовуючи контекстне кодування. Для контекстно кодованих бінів, декодер 30 відео може вибрати контекст на основі відносної позиції біна в рядку бінів. Наприклад, декодер 30 відео може вибрати контекст для першого біна, що відрізняється від контексту для другого біна.

Після декодування бінів для одержання декодованих двійкових значень, декодер 30 відео може бінаризувати декодований рядок бінів, щоб одержати декодований синтаксичний елемент (208). Наприклад, декодер 30 відео може зіставити декодований рядок бінів із синтаксичним елементом, використовуючи попередньо заданий процес. Тобто, у деяких випадках, декодер 30 відео може приймати ознаку того, що конкретний бін є кінцевим біном для синтаксичного елемента. Тоді, після завершення синтаксичного елемента, декодер 30 відео може зіставити рядок бінів з синтаксичним елементом, використовуючи таблицю бінаризації.

У деяких прикладах, якщо синтаксичний елемент, що підлягає кодуванню, не є значенням опорного індексу (гілка "Ні" етапу 202), декодер 30 відео може вибрати конкретний режим кодування (наприклад, з обходом або контекстно-адаптивний), щоб кодувати синтаксичний елемент (210). Декодер 30 відео може декодувати синтаксичний елемент, використовуючи вибраний режим (212), і бінаризувати декодований рядок бінів (208).

Крім того, варто розуміти, що етапи, показані й описані з посиланням на Фіг. 10, надані просто як один приклад. Тобто, етапи способу, зображеного на Фіг. 10, не обов'язково повинні виконуватися в порядку, показаному на Фіг. 10, і може виконуватися менше етапів, додаткові або альтернативні етапи.

Фіг. 11 є блок-схемою послідовності операцій способу, яка ілюструє приклад ентропійного кодування даних прогнозування, відповідно до аспектів даного розкриття винаходу. Хоч для цілей пояснення воно й описано загалом як виконване компонентами кодера 20 відео (Фіг. 1 і 2), варто розуміти, що інші модулі кодування відео, процесори, модулі обробки, апаратні модулі кодування, такі як кодери/декодери (КОДЕКИ) і т. п., також можуть бути виконані з можливістю виконання процесу, зображеного на Фіг. 11.

У прикладі на Фіг. 11, кодер 20 відео може бінаризувати один або більше синтаксичних елементів, кодованих у даний момент (220). Наприклад, кодер 20 відео може бінаризувати дані прогнозування, що включають у себе один або більше опорних індексів, вектори руху, предиктори вектора руху, індекси предикторів вектора руху, значення різниць векторів руху і т. п.

У будь-якому випадку, кодер 20 відео може визначати, чи включають кодовані синтаксичні елементи в себе біни для контекстного кодування і біни для кодування з обходом (222). Тобто, кодер 20 відео може визначати, чи кодуються біни синтаксичного елемента з використанням змішування контекстно-адаптивного кодування і кодування з обходом. Якщо є змішування як контекстного кодування, так і кодування з обходом (гілка "Так" етапу 222), кодер 20 відео може групувати контекстно кодовані біни і кодовані з обходом біни (224). Наприклад, кодер 20 відео може відділяти контекстно кодовані біни від кодованих з обходом бінів.

Потім кодер 20 відео може закодувати контекстно кодовані біни, використовуючи, наприклад, процес контекстно-адаптивного кодування (наприклад, такого як CABAC) (226). Крім цього, кодер 20 відео може закодувати кодовані з обходом біни, використовуючи режим з

обходом (226). Режим з обходом може обходити механізм контекстно-адаптивного кодування і використовувати фіксовану імовірність, щоб кодувати біни.

5 Якщо синтаксичний елемент, що підлягає кодуванню, не включає в себе одночасно і контекстно кодовані біни, і кодовані з обходом біни (гілка "Ні" етапу 222), кодер 20 відео може вибрати конкретний режим кодування (наприклад, з обходом або контекстно-адаптивний), щоб кодувати синтаксичний елемент (230). Потім кодер 20 відео може закодувати синтаксичний елемент, використовуючи вибраний режим (234).

10 У деяких випадках, групування бінів, описане відносно Фіг. 11, може виконуватися більше ніж для двох синтаксичних елементів. Наприклад, як описувалося вище відносно Фіг. 7, можуть групуватися всі контекстно кодовані біни, пов'язані з PU, так що контекстно кодовані біни з PU кодуються разом і кодовані з обходом біни з PU кодуються разом. Крім цього, групування може виконуватися на рівні CU, LCU або слайсу. Тобто, у деяких прикладах, усі контекстно кодовані біни для CU/LCU/слайсу можуть бути згруповані і кодуватися разом, тим самим дозволяючи кодеру 20 відео робити єдиний перехід між контекстним кодуванням і кодуванням з обходом.

15 Крім того, варто розуміти, що етапи, показані й описані з посиланням на Фіг. 11, надані просто як один приклад. Тобто, етапи способу, зображеного на Фіг. 11, не обов'язково повинні виконуватися в порядку, показаному на Фіг. 11, і може виконуватися менше етапів, додаткові або альтернативні етапи. Наприклад, у той час як Фіг. 11 демонструє кодер 20 відео, що кодує контекстно кодовані біни раніше кодованих з обходом бінів, в інших прикладах кодер 20 відео може кодувати кодовані з обходом біни раніше контекстно кодованих бінів.

20 Фіг. 12 є блок-схемою послідовності операцій способу, яка ілюструє приклад ентропійного декодування даних прогнозування, відповідно до аспектів даного розкриття винаходу. Хоч для цілей пояснення воно й описано загалом як виконване компонентами декодера 30 відео (Фіг. 1 і 3), варто розуміти, що інші модулі кодування відео, процесори, модулі обробки, апаратні модулі кодування, такі як кодери/декодери (КОДЕКИ) і т. п., також можуть бути виконані з можливістю виконання процесу, зображеного на Фіг. 12.

У прикладі на Фіг. 12, декодер 30 відео може визначати синтаксичний елемент (або елементи), що підлягає декодуванню (240). В одному з прикладів, декодер 30 відео може ідентифікувати синтаксичні елементи, пов'язані з даними прогнозування, для декодування. 30 Декодер 30 відео також може визначати, чи включають біни синтаксичного елемента(ів) у себе контекстно кодовані біни і кодовані з обходом біни (242). Якщо є змішування як контекстного кодування, так і кодування з обходом (гілка "Так" етапу 242), декодер 30 відео може декодувати контекстно кодовані біни (244). Декодер 30 відео також може декодувати, окремо від контекстно кодованих бінів, кодовані з обходом біни (246). Тобто, у тих випадках, коли контекстно кодовані біни згруповані окремо від кодованих з обходом бінів у бітовому потоці, що підлягає декодуванню, декодер 30 відео може декодувати усі контекстно кодовані біни окремо від декодування кодованих з обходом бінів. Після декодування бінів, декодер 30 відео може бінаризувати декодовані біни, щоб сформувати декодований синтаксичний елемент (248). Наприклад, декодер 30 відео може зіставити декодований рядок бінів з синтаксичним елементом, використовуючи таблицю бінаризації або інший процес бінаризації.

40 Якщо синтаксичний елемент, що підлягає кодуванню, не включає в себе одночасно і контекстно кодовані біни, і кодовані з обходом біни (гілка "Ні" етапу 242), декодер 30 відео може вибрати конкретний режим кодування (наприклад, з обходом або контекстно-адаптивний), щоб кодувати синтаксичний елемент (250). Потім декодер 30 відео може декодувати закодовані біни, використовуючи вибраний режим (252), і бінаризувати декодований рядок бінів, щоб сформувати декодований синтаксичний елемент (248).

45 Як описувалося вище відносно Фіг. 11, у деяких випадках, групування бінів може виконуватися більше ніж для двох синтаксичних елементів. Наприклад, групування може виконуватися на рівні PU, CU, LCU або слайсу. Тобто, у деяких прикладах, усі контекстно кодовані біни для PU/CU/LCU/слайсу можуть бути згруповані і кодуватися разом, тим самим дозволяючи декодеру 30 відео робити єдиний перехід між контекстним кодуванням і кодуванням з обходом.

50 Крім того, варто розуміти, що етапи, показані й описані з посиланням на Фіг. 12, надані просто як один приклад. Тобто, етапи способу, зображеного на Фіг. 12, не обов'язково повинні виконуватися в порядку, показаному на Фіг. 12, і може виконуватися менше етапів, додаткові або альтернативні етапи. Наприклад, у той час як Фіг. 12 демонструє декодер 30 відео, що декодує контекстно кодовані біни раніше кодованих з обходом бінів, в інших прикладах декодер 30 відео може кодувати кодовані з обходом біни раніше контекстно кодованих бінів.

60 Крім того, варто розуміти, що, залежно від прикладу, визначені дії або події будь-якого зі способів, описуваних у даному документі, можуть виконуватися в іншій послідовності, можуть

бути додані, поєднані або взагалі пропущені (наприклад, не всі описувані дії або події необхідні для практичної реалізації способу). Більше того, у деяких прикладах, дії або події можуть виконуватися одночасно, наприклад, завдяки використанню багатопотокової обробки, обробки переривань або декількох процесорів, а не послідовно. Крім цього, хоча визначені аспекти даного розкриття винаходу й описані для ясності як виконувані одним конструктивним елементом або модулем, варто розуміти, що методи даного розкриття винаходу можуть виконуватися комплексом конструктивних елементів або модулів, сполучених з відеокодером.

Кодер 20 відео може реалізовувати будь-який або всі з методів згідно з даним розкриттям винаходу для кодування опорного індексу й інших синтаксичних елементів у ході процесу кодування відео. Точно так само, декодер 30 відео може реалізовувати будь-який або всі з цих методів для кодування опорного індексу й інших синтаксичних елементів у ході процесу кодування відео. Відеокодер, як описується в даному розкритті винаходу, може належати до кодера відео або декодера відео. Аналогічно, модуль кодування відео може належати до кодера відео або декодера відео. Точно так само, кодування відео може належати до кодування відео або декодування відео.

В одному або більше прикладах, функції, описані в даному розкритті винаходу і приписувані кодеру 20 відео, декодеру 30 відео або будь-якому іншому модулю обробки, можуть бути реалізовані в апаратному забезпеченні, програмному забезпеченні, програмно-апаратному забезпеченні або будь-якій їх комбінації. При реалізації в програмному забезпеченні, функції можуть зберігатися на зчитуваному комп'ютером носії, або передаватися через нього, у вигляді однієї або більше інструкцій або коду і виконуватися апаратним модулем обробки. Зчитувані комп'ютером носії можуть включати в себе зчитувані комп'ютером інформаційні носії, що відповідають матеріальному носію, такому як носії для збереження даних, або середовища зв'язку, у тому числі будь-який носій, що полегшує перенесення комп'ютерної програми з одного місця в інше, наприклад, відповідно до протоколу зв'язку. Таким чином, зчитувані комп'ютером носії, у загальному значенні, можуть відповідати (1) матеріальним зчитуваним комп'ютером носіям для збереження даних, що є енергонезалежними, або (2) середовищу зв'язку, такому як сигнал або несуча хвиля. Носії для збереження даних можуть бути будь-якими доступними носіями, до яких можна одержати доступ за допомогою одного або більше комп'ютерів або одного або більше процесорів, щоб витягати інструкції, код і/або структури даних для реалізації методів, описаних у даному розкритті винаходу. Комп'ютерний програмний продукт може включати в себе зчитуваний комп'ютером носій.

Як приклад, але не обмеження, такі зчитувані комп'ютером інформаційні носії можуть містити ОЗП, ПЗП, ЕСППЗП, CD-ROM або інше сховище на оптичному диску, сховище на магнітних дисках або інші магнітні пристрої збереження, пам'ять із груповим перезаписом або будь-який інший носій, що може використовуватися для збереження необхідного програмного коду у формі інструкцій або структур даних і до якого можна одержати доступ за допомогою комп'ютера. Крім того, будь-яке з'єднання коректно називати зчитуваним комп'ютером носієм. Наприклад, якщо інструкції передаються від мережного вузла, обслуговуючого вузла або іншого віддаленого джерела з використанням коаксіального кабелю, волоконно-оптичного кабелю, витої пари, цифрової абонентської лінії (DSL-digital subscriber line) або бездротових технологій, таких як інфрачервоне випромінювання, радіозв'язок і НВЧ-випромінювання, то коаксіальний кабель, волоконно-оптичний кабель, вита пари, DSL або бездротові технології, такі як інфрачервоне випромінювання, радіозв'язок і НВЧ-випромінювання, охоплюються визначенням носія. Варто розуміти, однак, що зчитувані комп'ютером інформаційні носії і носії для збереження даних не включають в себе з'єднання, несучі хвилі, сигнали або інші тимчасові носії, а навпаки орієнтовані на енергонезалежні, матеріальні інформаційні носії. Магнітний диск і немагнітний диск, як використовується в даному документі, включають в себе компакт-диск (CD-compact disc), лазерний диск, оптичний диск, цифровий універсальний диск (DVD-digital versatile disc), гнучкий диск і диск Blu-ray, при цьому магнітні диски звичайно відтворюють дані магнітним способом, у той час як немагнітні диски відтворюють дані оптичним способом за допомогою лазерних джерел. Комбінації перерахованого вище теж повинні підпадати під визначення зчитуваних комп'ютером носіїв.

Інструкції можуть виконуватися одним або більше процесорами, такими як один або більше цифрових сигнальних процесорів (ЦСП), універсальних мікропроцесорів, спеціалізованих інтегральних схем (СІС), програмованих користувачем вентильних матриць (ПКВМ) або інших еквівалентних інтегральних або дискретних логічних схем або будь-яка їх комбінація. Кодер 20 відео або декодер 30 відео може включати в себе будь-який з множини таких одного або більше процесорів, виконаних з можливістю виконання функцій, описаних у даному розкритті винаходу. Відповідно, термін "процесор", як використовується в даному документі, може стосуватися будь-

якої з вищевказаних структур або будь-якої іншої структури, придатної для реалізації методів, описаних у даному документі. Крім цього, у деяких аспектах, функціональні можливості, описані в даному документі, можуть бути забезпечені в рамках виділених апаратних і/або програмних конструктивних елементів, виконаних з можливістю кодування і декодування, або вмонтованих в об'єднаний кодек. Крім того, ці методи могли б бути повною мірою реалізовані в одній або більше схемах або логічних елементах.

Методи згідно з даним розкриттям винаходу можуть бути реалізовані в широкій різноманітності пристроїв або інструментальних засобів, що включають у себе бездротовий мікротелефон, інтегральну схему (IC) або набір IC (наприклад, набір мікросхем). Різні компоненти, конструктивні елементи або модулі описані в даному розкритті винаходу, щоб підкреслити функціональні аспекти пристроїв, виконаних з можливістю виконання розкритих методів, але не обов'язково вимагають реалізації на різних апаратних модулях. Скоріше, як описано вище, різні модулі можуть бути об'єднані в апаратному модулі кодек або забезпечуються комплектом здатних взаємодіяти один з одним апаратних модулів, що включають у себе один або більше процесорів, як описано вище, у сполученні з придатним програмним забезпеченням і/або програмно-апаратним забезпеченням.

Були описані різні приклади. Ці й інші приклади входять в обсяг прикладеної формули винаходу.

## ФОРМУЛА ВИНАХОДУ

1. Спосіб для кодування синтаксичного елемента індексу опорного зображення в процесі кодування відео, причому спосіб включає етапи, на яких:  
виконують бінаризацію значення індексу опорного зображення;  
коднують щонайменше один бін бінаризованого значення індексу опорного зображення в режимі контекстного кодування процесу контекстно-адаптивного двійкового арифметичного кодування (CABAC);

визначають, що бінаризоване значення індексу опорного зображення містить більше бінів, ніж зазначений щонайменше один бін, який коднують в режимі контекстного кодування; і  
коднують, у відповідь на визначення, що бінаризоване значення індексу опорного зображення містить більше бінів, ніж зазначений щонайменше один бін, кодований у режимі контекстного кодування, щонайменше інший бін бінаризованого значення індексу опорного зображення в режимі кодування з обходом процесу CABAC.

2. Спосіб за п. 1,

у якому етап, на якому коднують щонайменше один бін бінаризованого значення індексу опорного зображення в режимі контекстного кодування, включає етапи, на яких:

коднують перший бін (bin0) бінаризованого значення індексу опорного зображення з першим контекстом (ctx0),

коднують другий бін (bin1) бінаризованого індексу опорного зображення з другим контекстом (ctx1), і

у якому етап, на якому коднують щонайменше інший бін бінаризованого значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому:

коднують третій бін (bin2) і всі інші біни після третього біна (bin2) у режимі кодування з обходом.

3. Спосіб за п. 2, у якому етап, на якому виконують бінаризацію значення індексу опорного зображення, включає етап, на якому виконують бінаризацію значення індексу опорного зображення, використовуючи комбіновані зрізаний унарний код і експоненціальний код Голомба.

4. Спосіб за п. 1, у якому етап, на якому коднують зазначений щонайменше інший бін бінаризованого значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому коднують зазначений щонайменше інший бін за допомогою щонайменше одного з наступних процесів кодування: унарним кодом, зрізаним унарним кодом, кодом Голомба, експоненціальним кодом Голомба або кодом Голомба-Райса.

5. Спосіб за п. 1, який додатково включає етапи, на яких:

виконують бінаризацію складових значення першої різниці векторів руху, пов'язаного з першим індексом опорного зображення, і складових значення другої різниці векторів руху, пов'язаного з другим індексом опорного зображення;

коднують першу частину складових значення першої різниці векторів руху і першу частину складових значення другої різниці векторів руху в режимі контекстного кодування; і

коднують другу частину складових значення першої різниці векторів руху і другу частину складових значення другої різниці векторів руху в режимі кодування з обходом.

6. Спосіб за п. 5, який додатково включає етапи, на яких:

групуєть першу частину складових значення першої різниці векторів руху і першу частину складових значення другої різниці векторів руху в першу групу для кодування в режимі контекстного кодування, і

5 групуєть другу частину складових значення першої різниці векторів руху і другу частину складових значення другої різниці векторів руху в другу групу для кодування в режимі кодування з обходом.

7. Спосіб за п. 1,

у якому етап, на якому кодують щонайменше один біт бінаризованого значення індексу опорного зображення в режимі контекстного кодування, включає етапи, на яких:

10 кодують перший біт (bin0) бінаризованого значення індексу опорного зображення з першим контекстом (ctx0),

кодують другий біт (bin1) бінаризованого індексу опорного зображення з другим контекстом (ctx1),

15 кодують третій біт (bin2) бінаризованого індексу опорного зображення з третім контекстом (ctx2), і

у якому етап, на якому кодують щонайменше інший біт бінаризованого значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому:

кодують всі інші біти після третього біта (bin2) у режимі кодування з обходом.

8. Спосіб за п. 1,

20 у якому етап, на якому кодують щонайменше один біт бінаризованого значення індексу опорного зображення в режимі контекстного кодування, включає етап, на якому:

кодують перший біт (bin0) бінаризованого значення індексу опорного зображення з першим контекстом (ctx0), і

25 у якому етап, на якому кодують щонайменше інший біт бінаризованого значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому:

кодують всі інші біти після першого біта (bin0) у режимі кодування з обходом.

9. Спосіб за п. 1,

у якому етап, на якому виконують бінаризацію значення індексу опорного зображення, включає етап, на якому кодують унарним кодом значення індексу опорного зображення;

30 у якому етап, на якому кодують щонайменше один біт бінаризованого значення індексу опорного зображення в режимі контекстного кодування, включає етап, на якому:

кодують щонайменше один біт кодованого унарним кодом значення індексу опорного зображення в режимі контекстного кодування; і

35 у якому етап, на якому кодують зазначений щонайменше інший біт бінаризованого значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому:

кодують щонайменше інший біт кодованого унарним кодом значення індексу опорного зображення в режимі кодування з обходом.

10. Спосіб за п. 9, у якому етап, на якому кодують унарним кодом значення індексу опорного зображення, включає етап, на якому кодують зрізаним унарним кодом значення індексу

40 опорного зображення.

11. Спосіб за п. 1,

у якому етап, на якому виконують бінаризацію значення індексу опорного зображення, включає етап, на якому кодують унарним кодом і кодують експоненціальним кодом Голомба значення індексу опорного зображення;

45 у якому етап, на якому кодують щонайменше один біт бінаризованого значення індексу опорного зображення в режимі контекстного кодування, включає етап, на якому:

кодують щонайменше один біт кодованої унарним кодом частини значення індексу опорного зображення в режимі контекстного кодування; і

50 у якому етап, на якому кодують щонайменше інший біт бінаризованого значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому:

кодують щонайменше інший біт кодованої унарним кодом частини значення індексу опорного зображення і кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення в режимі кодування з обходом.

55 12. Спосіб за п. 11, який додатково включає етап, на якому зрізають кодовану експоненціальним кодом Голомба частину значення індексу опорного зображення перед кодуванням кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення.

13. Спосіб за п. 1,

у якому етап, на якому виконують бінаризацію значення індексу опорного зображення, включає етап, на якому кодують унарним кодом і кодують кодом з фіксованою довжиною кодового слова

60 значення індексу опорного зображення;

у якому етап, на якому кодують щонайменше один біт бінаризованого значення індексу опорного зображення в режимі контекстного кодування, включає етап, на якому кодують щонайменше один біт кодованої унарним кодом частини значення індексу опорного зображення в режимі контекстного кодування; і

- 5 у якому етап, на якому кодують щонайменше інший біт бінаризованого значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому кодують щонайменше інший біт кодованої унарним кодом частини значення індексу опорного зображення і кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення в режимі кодування з обходом.
- 10 14. Спосіб за п. 13, який додатково включає етап, на якому зрізають кодовану кодом з фіксованою довжиною кодового слова частину значення індексу опорного зображення перед кодуванням кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення.  
15. Спосіб за п. 1,
- 15 у якому етап, на якому кодують у режимі контекстного кодування, включає етапи, на яких вибирають одну або більше імовірнісних моделей для кодування зазначеного щонайменше одного біта бінаризованого індексу опорного зображення і кодують зазначений щонайменше один біт бінаризованого індексу опорного зображення, використовуючи вибрані одну або більше імовірнісних моделей; і
- 20 у якому етап, на якому кодують у режимі кодування з обходом, включає етапи, на яких визначають фіксовану імовірність і кодують зазначений щонайменше інший біт бінаризованого значення індексу опорного зображення, використовуючи фіксовану імовірність.  
16. Пристрій для кодування синтаксичного елемента індексу опорного зображення в процесі кодування відео, причому пристрій містить:
- 25 буфер декодованих зображень, виконаний з можливістю збереження одного або більше опорних зображень; і  
один або більше процесорів, виконаних з можливістю:  
бінаризації значення індексу опорного зображення для опорного зображення із зазначених одного або більше опорних зображень;
- 30 кодування щонайменше одного біта бінаризованого значення індексу опорного зображення в режимі контекстного кодування процесу контекстно-адаптивного двійкового арифметичного кодування (CABAC); і  
визначення, що бінаризоване значення індексу опорного зображення містить більше бітів, ніж зазначений щонайменше один біт, який кодують в режимі контекстного кодування; і
- 35 кодування, у відповідь на визначення, що бінаризоване значення індексу опорного зображення містить більше бітів, ніж зазначений щонайменше один біт, кодований у режимі контекстного кодування, щонайменше іншого біта бінаризованого значення індексу опорного зображення в режимі кодування з обходом процесу CABAC.  
17. Пристрій за п. 16,
- 40 у якому, для кодування щонайменше одного біта бінаризованого значення індексу опорного зображення в режимі контекстного кодування, один або більше процесорів виконані з можливістю:  
кодування першого біта (bin0) бінаризованого значення індексу опорного зображення з першим контекстом (ctx0),
- 45 кодування другого біта (bin1) бінаризованого індексу опорного зображення з другим контекстом (ctx1), і  
у якому, для кодування щонайменше іншого біта бінаризованого значення індексу опорного зображення в режимі кодування з обходом, один або більше процесорів виконані з можливістю:  
кодування третього біта (bin2) і всіх інших бітів після третього біта (bin2) у режимі кодування з
- 50 обходом.  
18. Пристрій за п. 17, у якому, для бінаризації значення індексу опорного зображення, один або більше процесорів виконані з можливістю бінаризації значення індексу опорного зображення з використанням комбінованих зрізаного унарного коду й експоненціального коду Голомба.  
19. Пристрій за п. 16, у якому, для кодування зазначеного щонайменше іншого біта бінаризованого значення індексу опорного зображення в режимі кодування з обходом, один або
- 55 більше процесорів виконані з можливістю кодування зазначеного щонайменше іншого біта за допомогою щонайменше одного з наступних процесів кодування: унарним кодом, зрізаним унарним кодом, кодом Голомба, експоненціальним кодом Голомба або кодом Голомба-Райса.  
20. Пристрій за п. 16, у якому один або більше процесорів додатково виконані з можливістю:



- бінаризації складових значення першої різниці векторів руху, пов'язаного з першим індексом опорного зображення, і складових значення другої різниці векторів руху, пов'язаного з другим індексом опорного зображення;
- 5 кодування першої частини складових значення першої різниці векторів руху і першої частини складових значення другої різниці векторів руху в режимі контекстного кодування; і кодування другої частини складових значення першої різниці векторів руху і другої частини складових значення другої різниці векторів руху в режимі кодування з обходом.
21. Пристрій за п. 20, у якому один або більше процесорів додатково виконані з можливістю: групування першої частини складових значення першої різниці векторів руху і першої частини
- 10 складових значення другої різниці векторів руху в першу групу для кодування в режимі контекстного кодування, і групування другої частини складових значення першої різниці векторів руху і другої частини складових значення другої різниці векторів руху в другу групу для кодування в режимі кодування з обходом.
- 15 22. Пристрій за п. 16, у якому, для кодування щонайменше одного біна бінаризованого значення індексу опорного зображення в режимі контекстного кодування, один або більше процесорів виконані з можливістю: кодування першого біна (bin0) бінаризованого значення індексу опорного зображення з першим
- 20 контекстом (ctx0), кодування другого біна (bin1) бінаризованого індексу опорного зображення з другим контекстом (ctx1), кодування третього біна (bin2) бінаризованого індексу опорного зображення з третім контекстом (ctx2), і
- 25 у якому, для кодування щонайменше іншого біна бінаризованого значення індексу опорного зображення в режимі кодування з обходом, один або більше процесорів виконані з можливістю: кодування всіх інших бінів після третього біна (bin2) у режимі кодування з обходом.
23. Пристрій за п. 16, у якому, для кодування щонайменше одного біна бінаризованого значення індексу опорного
- 30 зображення в режимі контекстного кодування, один або більше процесорів виконані з можливістю: кодування першого біна (bin0) бінаризованого значення індексу опорного зображення з першим контекстом (ctx0), і
- у якому, для кодування щонайменше іншого біна бінаризованого значення індексу опорного
- 35 зображення в режимі кодування з обходом, один або більше процесорів виконані з можливістю: кодування всіх інших бінів після першого біна (bin0) у режимі кодування з обходом.
24. Пристрій за п. 16, у якому, для бінаризації значення індексу опорного зображення, один або більше процесорів виконані з можливістю кодування унарним кодом значення індексу опорного зображення;
- 40 у якому, для кодування щонайменше одного біна бінаризованого значення індексу опорного зображення в режимі контекстного кодування, один або більше процесорів виконані з можливістю: кодування щонайменше одного біна кодованого унарним кодом значення індексу опорного зображення в режимі контекстного кодування; і
- 45 у якому, для кодування зазначеного щонайменше іншого біна бінаризованого значення індексу опорного зображення в режимі кодування з обходом, один або більше процесорів виконані з можливістю: кодування щонайменше іншого біна кодованого унарним кодом значення індексу опорного зображення в режимі кодування з обходом.
- 50 25. Пристрій за п. 24, у якому, для унарного кодування значення індексу опорного зображення, один або більше процесорів виконані з можливістю зрізаного унарного кодування значення індексу опорного зображення.
26. Пристрій за п. 16, у якому, для бінаризації значення індексу опорного зображення, один або більше процесорів
- 55 виконані з можливістю кодування унарним кодом і кодування експоненціальним кодом Голомба значення індексу опорного зображення;
- у якому, для кодування щонайменше одного біна бінаризованого значення індексу опорного зображення в режимі контекстного кодування, один або більше процесорів виконані з можливістю:

кодування щонайменше одного біна кодованої унарним кодом частини значення індексу опорного зображення в режимі контекстного кодування; і

у якому, для кодування щонайменше іншого біна бінаризованого значення індексу опорного зображення в режимі кодування з обходом, один або більше процесорів виконані з можливістю:

5 кодування щонайменше іншого біна кодованої унарним кодом частини значення індексу опорного зображення і кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення в режимі кодування з обходом.

27. Пристрій за п. 26, у якому один або більше процесорів додатково виконані з можливістю зрізання кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення перед кодуванням кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення.

28. Пристрій за п. 16,

у якому, для бінаризації значення індексу опорного зображення, один або більше процесорів виконані з можливістю кодування унарним кодом і кодування кодом з фіксованою довжиною кодового слова значення індексу опорного зображення;

у якому, для кодування щонайменше одного біна бінаризованого значення індексу опорного зображення в режимі контекстного кодування, один або більше процесорів виконані з можливістю кодування щонайменше одного біна кодованої унарним кодом частини значення індексу опорного зображення в режимі контекстного кодування; і

у якому, для кодування зазначеного щонайменше іншого біна бінаризованого значення індексу опорного зображення в режимі кодування з обходом, один або більше процесорів виконані з можливістю кодування щонайменше іншого біна кодованої унарним кодом частини значення індексу опорного зображення і кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення в режимі кодування з обходом.

29. Пристрій за п. 28, у якому один або більше процесорів додатково виконані з можливістю зрізання кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення перед кодуванням кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення.

30. Пристрій за п. 16,

у якому, для кодування в режимі контекстного кодування, один або більше процесорів виконані з можливістю вибору однієї або більше імовірнісних моделей для кодування зазначеного щонайменше одного біна бінаризованого індексу опорного зображення і кодування зазначеного щонайменше одного біна бінаризованого індексу опорного зображення, використовуючи вибрані одну або більше імовірнісних моделей; і

у якому, для кодування в режимі кодування з обходом, один або більше процесорів виконані з можливістю визначення фіксованої імовірності і кодування зазначеного щонайменше іншого біна бінаризованого значення індексу опорного зображення, використовуючи фіксовану імовірність.

31. Пристрій для кодування синтаксичного елемента індексу опорного зображення в процесі кодування відео, причому пристрій містить:

засіб для бінаризації значення індексу опорного зображення;

засіб для кодування щонайменше одного біна бінаризованого значення індексу опорного зображення в режимі контекстного кодування процесу контекстно-адаптивного двійкового арифметичного кодування (CABAC);

засіб для визначення, що бінаризоване значення індексу опорного зображення містить більше бінів, ніж зазначений щонайменше один бін, який кодують в режимі контекстного кодування; і

засіб для кодування, у відповідь на визначення, що бінаризоване значення індексу опорного зображення містить більше бінів, ніж зазначений щонайменше один бін, кодований у режимі контекстного кодування, щонайменше іншого біна бінаризованого значення індексу опорного зображення в режимі кодування з обходом процесу CABAC.

32. Пристрій за п. 31,

у якому засіб для кодування щонайменше одного біна бінаризованого значення індексу опорного зображення в режимі контекстного кодування містить:

засіб для кодування першого біна (bin0) бінаризованого значення індексу опорного зображення з першим контекстом (ctx0),

засіб для кодування другого біна (bin1) бінаризованого індексу опорного зображення з другим контекстом (ctx1), і

у якому засіб для кодування щонайменше іншого біна бінаризованого значення індексу опорного зображення в режимі кодування з обходом містить:

засіб для кодування третього біна (bin2) і всіх інших бінів після третього біна (bin2) у режимі кодування з обходом.

33. Пристрій за п. 32, у якому засіб для бінаризації значення індексу опорного зображення містить засіб для бінаризації значення індексу опорного зображення з використанням комбінованих зрізаного унарного коду й експоненціального коду Голомба.

34. Пристрій за п. 31, у якому засіб для кодування зазначеного щонайменше іншого біна бінаризованого значення індексу опорного зображення в режимі кодування з обходом містить засіб для кодування зазначеного щонайменше іншого біна за допомогою щонайменше одного з наступних процесів кодування: унарним кодом, зрізаним унарним кодом, кодом Голомба, експоненціальним кодом Голомба або кодом Голомба-Райса.

35. Пристрій за п. 31, який додатково містить:

засіб для бінаризації складових значення першої різниці векторів руху, пов'язаного з першим індексом опорного зображення, і складових значення другої різниці векторів руху, пов'язаного з другим індексом опорного зображення;

засіб для кодування першої частини складових значення першої різниці векторів руху і першої частини складових значення другої різниці векторів руху в режимі контекстного кодування; і

засіб для кодування другої частини складових значення першої різниці векторів руху і другої частини складових значення другої різниці векторів руху в режимі кодування з обходом.

36. Пристрій за п. 35, який додатково містить:

засіб для групування першої частини складових значення першої різниці векторів руху і першої частини складових значення другої різниці векторів руху в першу групу для кодування в режимі контекстного кодування, і

засіб для групування другої частини складових значення першої різниці векторів руху і другої частини складових значення другої різниці векторів руху в другу групу для кодування в режимі кодування з обходом.

37. Пристрій за п. 31,

у якому засіб для кодування щонайменше одного біна бінаризованого значення індексу опорного зображення в режимі контекстного кодування містить:

засіб для кодування першого біна (bin0) бінаризованого значення індексу опорного зображення з першим контекстом (ctx0),

засіб для кодування другого біна (bin1) бінаризованого індексу опорного зображення з другим контекстом (ctx1), і

засіб для кодування третього біна (bin2) бінаризованого індексу опорного зображення з третім контекстом (ctx2), і

у якому засіб для кодування щонайменше іншого біна бінаризованого значення індексу опорного зображення в режимі кодування з обходом містить:

засіб для кодування всіх інших бінів після третього біна (bin2) у режимі кодування з обходом.

38. Пристрій за п. 31,

у якому засіб для кодування щонайменше одного біна бінаризованого значення індексу опорного зображення в режимі контекстного кодування містить:

засіб для кодування першого біна (bin0) бінаризованого значення індексу опорного зображення з першим контекстом (ctx0), і

у якому засіб для кодування щонайменше іншого біна бінаризованого значення індексу опорного зображення в режимі кодування з обходом містить:

засіб для кодування всіх інших бінів після першого біна (bin0) у режимі кодування з обходом.

39. Пристрій за п. 31,

у якому засіб для бінаризації значення індексу опорного зображення містить засіб для кодування унарним кодом значення індексу опорного зображення;

у якому засіб для кодування зазначеного щонайменше одного біна бінаризованого значення індексу опорного зображення в режимі контекстного кодування містить:

засіб для кодування щонайменше одного біна кодованого унарним кодом значення індексу опорного зображення в режимі контекстного кодування; і

у якому засіб для кодування зазначеного щонайменше іншого біна бінаризованого значення індексу опорного зображення в режимі кодування з обходом містить:

засіб для кодування щонайменше іншого біна кодованого унарним кодом значення індексу опорного зображення в режимі кодування з обходом.

40. Пристрій за п. 39, у якому засіб для кодування унарним кодом значення індексу опорного зображення містить засіб для кодування зрізаним унарним кодом значення індексу опорного зображення.

41. Пристрій за п. 31,

- у якому засіб для бінаризації значення індексу опорного зображення містить засіб для кодування унарним кодом і кодування експоненціальним кодом Голомба значення індексу опорного зображення;
- у якому засіб для кодування зазначеного щонайменше одного біна бінаризованого значення індексу опорного зображення в режимі контекстного кодування містить:
- 5 засіб для кодування щонайменше одного біна кодованої унарним кодом частини значення індексу опорного зображення в режимі контекстного кодування; і
- у якому засіб для кодування зазначеного щонайменше іншого біна бінаризованого значення індексу опорного зображення в режимі кодування з обходом містить:
- 10 засіб для кодування щонайменше іншого біна кодованої унарним кодом частини значення індексу опорного зображення і кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення в режимі кодування з обходом.
42. Пристрій за п. 41, який додатково містить засіб для зрізання кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення перед кодуванням кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення.
- 15 43. Пристрій за п. 31,
- у якому засіб для бінаризації значення індексу опорного зображення містить засіб для кодування унарним кодом і кодування кодом з фіксованою довжиною кодового слова значення індексу опорного зображення;
- 20 у якому засіб для кодування зазначеного щонайменше одного біна бінаризованого значення індексу опорного зображення в режимі контекстного кодування містить засіб для кодування щонайменше одного біна кодованої унарним кодом частини значення індексу опорного зображення в режимі контекстного кодування; і
- у якому засіб для кодування зазначеного щонайменше іншого біна бінаризованого значення індексу опорного зображення в режимі кодування з обходом містить засіб для кодування щонайменше іншого біна кодованої унарним кодом частини значення індексу опорного зображення і кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення в режимі кодування з обходом.
- 25 44. Пристрій за п. 43, який додатково містить засіб для зрізання кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення перед кодуванням кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення.
- 30 45. Пристрій за п. 31,
- у якому засіб для кодування в режимі контекстного кодування містить засіб для вибору однієї або більше імовірнісних моделей для кодування зазначеного щонайменше одного біна бінаризованого індексу опорного зображення і засіб для кодування зазначеного щонайменше одного біна бінаризованого індексу опорного зображення, використовуючи вибрані одну або більше імовірнісних моделей; і
- 35 у якому засіб для кодування в режимі кодування з обходом містить засіб для визначення фіксованої імовірності і засіб для кодування щонайменше іншого біна бінаризованого значення індексу опорного зображення, використовуючи фіксовану імовірність.
- 40 46. Спосіб для декодування синтаксичного елемента індексу опорного зображення в процесі декодування відео, причому спосіб включає етапи, на яких:
- 45 декодують щонайменше один бін значення індексу опорного зображення в режимі контекстного кодування процесу контекстно-адаптивного двійкового арифметичного кодування (CAVAC); визначають, що бінаризоване значення індексу опорного зображення містить більше бінів, ніж зазначений щонайменше один бін, який кодують в режимі контекстного кодування; і
- декодують, у відповідь на визначення, що значення індексу опорного зображення містить більше бінів, ніж зазначений щонайменше один бін, кодований у режимі контекстного кодування, щонайменше інший бін значення індексу опорного зображення в режимі кодування з обходом процесу CAVAC; і
- 50 виконують бінаризацію значення індексу опорного зображення.
47. Спосіб за п. 46,
- у якому етап, на якому декодують щонайменше один бін значення індексу опорного зображення в режимі контекстного кодування, включає етапи, на яких:
- 55 декодують перший бін (bin0) значення індексу опорного зображення з першим контекстом (ctx0), декодують другий бін (bin1) індексу опорного зображення з другим контекстом (ctx1), і
- у якому етап, на якому декодують щонайменше інший бін значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому:

декодує третій біт (bin2) і всі інші біти після третього біта (bin2) у режимі кодування з обходом.

48. Спосіб за п. 47, у якому етап, на якому виконують бінаризацію значення індексу опорного зображення, включає етап, на якому виконують бінаризацію значення індексу опорного зображення, використовуючи комбіновані зрізаний унарний код і експоненціальний код Голомба.

49. Спосіб за п. 46, у якому етап, на якому декодують зазначений щонайменше інший біт значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому декодують зазначений щонайменше інший біт за допомогою щонайменше одного з наступних процесів кодування: унарним кодом, зрізаним унарним кодом, кодом Голомба, експоненціальним кодом Голомба або кодом Голомба-Райса.

50. Спосіб за п. 46, який додатково включає етапи, на яких:

декодують першу частину складових значення першої різниці векторів руху і першу частину складових значення другої різниці векторів руху в режимі контекстного кодування;

декодують другу частину складових значення першої різниці векторів руху і другу частину складових значення другої різниці векторів руху в режимі кодування з обходом; і

виконують бінаризацію складових значення першої різниці векторів руху, пов'язаного з першим індексом опорного зображення, і складових значення другої різниці векторів руху, пов'язаного з другим індексом опорного зображення.

51. Спосіб за п. 50, який додатково включає етапи, на яких:

групують першу частину складових значення першої різниці векторів руху і першу частину складових значення другої різниці векторів руху в першу групу для кодування в режимі контекстного кодування, і

групують другу частину складових значення першої різниці векторів руху і другу частину складових значення другої різниці векторів руху в другу групу для кодування в режимі кодування з обходом.

52. Спосіб за п. 46,

у якому етап, на якому декодують щонайменше один біт значення індексу опорного зображення в режимі контекстного кодування, включає етапи, на яких:

декодують перший біт (bin0) значення індексу опорного зображення з першим контекстом (ctx0),

декодують другий біт (bin1) індексу опорного зображення з другим контекстом (ctx1),

декодують третій біт (bin2) індексу опорного зображення з третім контекстом (ctx2), і

у якому етап, на якому декодують щонайменше інший біт значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому:

декодують всі інші біти після третього біта (bin2) у режимі кодування з обходом.

53. Спосіб за п. 46,

у якому етап, на якому декодують щонайменше один біт значення індексу опорного зображення в режимі контекстного кодування, включає етап, на якому:

декодують перший біт (bin0) значення індексу опорного зображення з першим контекстом (ctx0), і

у якому етап, на якому декодують щонайменше інший біт значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому:

декодують всі інші біти після першого біта (bin0) у режимі кодування з обходом.

54. Спосіб за п. 46,

у якому етап, на якому декодують зазначений щонайменше один біт значення індексу опорного зображення в режимі контекстного кодування, включає етап, на якому:

декодують щонайменше один біт кодованого унарним кодом значення індексу опорного зображення в режимі контекстного кодування;

у якому етап, на якому декодують зазначений щонайменше інший біт значення індексу в режимі кодування з обходом, включає етап, на якому:

декодують щонайменше інший біт кодованого унарним кодом значення індексу опорного зображення в режимі кодування з обходом; і

у якому етап, на якому виконують бінаризацію значення індексу опорного зображення, включає етап, на якому кодують унарним кодом значення індексу опорного зображення.

55. Спосіб за п. 54, у якому етап, на якому кодують унарним кодом значення індексу опорного зображення, включає етап, на якому кодують зрізаним унарним кодом значення індексу опорного зображення.

56. Спосіб за п. 46,

у якому етап, на якому декодують зазначений щонайменше один біт значення індексу опорного зображення в режимі контекстного кодування, включає етап, на якому:

декодуєть щонайменше один біт кодованої унарним кодом частини значення індексу опорного зображення в режимі контекстного кодування;

у якому етап, на якому декодуєть зазначений щонайменше інший біт значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому:

5 декодуєть щонайменше інший біт кодованої унарним кодом частини значення індексу опорного зображення і кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення в режимі кодування з обходом; і

у якому етап, на якому виконують бінаризацію значення індексу опорного зображення, включає етап, на якому кодуєть унарним кодом і кодуєть експоненціальним кодом Голомба значення індексу опорного зображення.

10 57. Спосіб за п. 56, який включає додатковий етап, на якому зрізають кодовану експоненціальним кодом Голомба частину значення індексу опорного зображення перед декодуванням кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення.

15 58. Спосіб за п. 46,

у якому етап, на якому декодуєть зазначений щонайменше один біт значення індексу опорного зображення в режимі контекстного кодування, включає етап, на якому декодуєть щонайменше один біт кодованої унарним кодом частини значення індексу опорного зображення в режимі контекстного кодування;

20 у якому етап, на якому декодуєть зазначений щонайменше інший біт значення індексу опорного зображення в режимі кодування з обходом, включає етап, на якому декодуєть щонайменше інший біт кодованої унарним кодом частини значення індексу опорного зображення і кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення в режимі кодування з обходом; і

25 у якому етап, на якому виконують бінаризацію значення індексу опорного зображення, включає етап, на якому кодуєть унарним кодом і кодуєть кодом з фіксованою довжиною кодового слова значення індексу опорного зображення.

59. Спосіб за п. 58, який додатково включає етап, на якому зрізають кодовану кодом з фіксованою довжиною кодового слова частину значення індексу опорного зображення перед кодуванням кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення.

30 60. Спосіб за п. 46,

у якому етап, на якому декодуєть у режимі контекстного кодування, включає етапи, на яких вибирають одну або більше імовірнісних моделей для кодування зазначеного щонайменше одного біта бінаризованого індексу опорного зображення і декодуєть зазначений щонайменше один біт бінаризованого індексу опорного зображення, використовуючи вибрані одну або більше імовірнісних моделей; і

у якому етап, на якому декодуєть у режимі кодування з обходом, включає етапи, на яких визначають фіксовану імовірність і декодуєть зазначений щонайменше інший біт бінаризованого значення індексу опорного зображення, використовуючи фіксовану імовірність.

40 61. Пристрій для декодування синтаксичного елемента індексу опорного зображення в процесі декодування відео, причому пристрій містить:

буфер декодованих зображень, виконаний з можливістю зберігання одного або більше опорних зображень; і

45 один або більше процесорів, виконаних з можливістю:

декодування щонайменше одного біта значення індексу опорного зображення в режимі контекстного кодування процесу контекстно-адаптивного двійкового арифметичного кодування (CABAC), при цьому значення індексу опорного зображення пов'язане з опорним зображенням із зазначених одного або більше опорних зображень;

50 визначення, що бінаризоване значення індексу опорного зображення містить більше бітів, ніж зазначений щонайменше один біт, який кодуєть в режимі контекстного кодування;

декодування, у відповідь на визначення, що значення індексу опорного зображення містить більше бітів, ніж зазначений щонайменше один біт, кодований у режимі контекстного кодування, щонайменше іншого біта значення індексу опорного зображення в режимі кодування з обходом процесу CABAC; і

55 бінаризації значення індексу опорного зображення.

62. Пристрій за п. 61,

у якому, для декодування щонайменше одного біта значення індексу опорного зображення в режимі контекстного кодування, один або більше процесорів виконані з можливістю:

декодування першого біна (bin0) значення індексу опорного зображення з першим контекстом (ctx0),

декодування другого біна (bin1) індексу опорного зображення з другим контекстом (ctx1), і

у якому, для декодування щонайменше іншого біна значення індексу опорного зображення в режимі кодування з обходом, один або більше процесорів виконані з можливістю:

декодування третього біна (bin2) і всіх інших бінів після третього біна (bin2) у режимі кодування з обходом.

63. Пристрій за п. 62, у якому, для бінаризації значення індексу опорного зображення, один або більше процесорів виконані з можливістю бінаризації значення індексу опорного зображення, використовуючи комбінований зрізаний унарний код і експоненціальний код Голомба.

64. Пристрій за п. 61, у якому, для декодування зазначеного щонайменше іншого біна значення індексу опорного зображення в режимі кодування з обходом, один або більше процесорів виконані з можливістю декодування щонайменше іншого біна за допомогою щонайменше одного з наступних процесів кодування: унарним кодом, зрізаним унарним кодом, кодом Голомба, експоненціальним кодом Голомба або кодом Голомба-Райса.

65. Пристрій за п. 61, у якому один або більше процесорів додатково виконані з можливістю: декодування першої частини складових значення першої різниці векторів руху і першої частини складових значення другої різниці векторів руху в режимі контекстного кодування; декодування другої частини складових значення першої різниці векторів руху і другої частини складових значення другої різниці векторів руху в режимі кодування з обходом; і бінаризації складових значення першої різниці векторів руху, пов'язаного з першим індексом опорного зображення, і складових значення другої різниці векторів руху, пов'язаного з другим індексом опорного зображення.

66. Пристрій за п. 65, у якому один або більше процесорів додатково виконані з можливістю: групування першої частини складових значення першої різниці векторів руху і першої частини складових значення другої різниці векторів руху в першу групу для кодування в режимі контекстного кодування, і

групування другої частини складових значення першої різниці векторів руху і другої частини складових значення другої різниці векторів руху в другу групу для кодування в режимі кодування з обходом.

67. Пристрій за п. 61, у якому, для декодування щонайменше одного біна значення індексу опорного зображення в режимі контекстного кодування, один або більше процесорів виконані з можливістю:

декодування першого біна (bin0) значення індексу опорного зображення з першим контекстом (ctx0), і

декодування другого біна (bin1) індексу опорного зображення з другим контекстом (ctx1),

декодування третього біна (bin2) індексу опорного зображення з третім контекстом (ctx2), і

у якому, для декодування щонайменше іншого біна значення індексу опорного зображення в режимі кодування з обходом, один або більше процесорів виконані з можливістю:

декодування всіх інших бінів після третього біна (bin2) у режимі кодування з обходом.

68. Пристрій за п. 61,

у якому, для декодування щонайменше одного біна значення індексу опорного зображення в режимі контекстного кодування, один або більше процесорів виконані з можливістю:

декодування першого біна (bin0) значення індексу опорного зображення з першим контекстом (ctx0), і

у якому, для декодування щонайменше іншого біна значення індексу опорного зображення в режимі кодування з обходом, один або більше процесорів виконані з можливістю:

декодування всіх інших бінів після першого біна (bin0) у режимі кодування з обходом.

69. Пристрій за п. 61,

у якому, для декодування зазначеного щонайменше одного біна значення індексу опорного зображення в режимі контекстного кодування, один або більше процесорів виконані з можливістю:

декодування щонайменше одного біна кодованого унарним кодом значення індексу опорного зображення в режимі контекстного кодування;

у якому, для декодування зазначеного щонайменше іншого біна значення індексу в режимі кодування з обходом, один або більше процесорів виконані з можливістю:

декодування щонайменше іншого біна кодованого унарним кодом значення індексу опорного зображення в режимі кодування з обходом; і

у якому, для бінаризації значення індексу опорного зображення, один або більше процесорів виконані з можливістю кодування унарним кодом значення індексу опорного зображення.

70. Пристрій за п. 69, у якому, для кодування унарним кодом значення індексу опорного зображення, один або більше процесорів виконані з можливістю кодування зрізаним унарним кодом значення індексу опорного зображення.

71. Пристрій за п. 61,

- 5 у якому, для декодування зазначеного щонайменше одного біна значення індексу опорного зображення в режимі контекстного кодування, один або більше процесорів виконані з можливістю:

декодування щонайменше одного біна кодованої унарним кодом частини значення індексу опорного зображення в режимі контекстного кодування;

- 10 у якому, для декодування зазначеного щонайменше іншого біна значення індексу опорного зображення в режимі кодування з обходом, один або більше процесорів виконані з можливістю: декодування щонайменше іншого біна кодованої унарним кодом частини значення індексу опорного зображення і кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення в режимі кодування з обходом; і

- 15 у якому, для бінаризації значення індексу опорного зображення, один або більше процесорів виконані з можливістю кодування унарним кодом і кодування експоненціальним кодом Голомба значення індексу опорного зображення.

- 20 72. Пристрій за п. 71, у якому один або більше процесорів додатково виконані з можливістю зрізання кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення перед декодуванням кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення.

73. Пристрій за п. 61,

- 25 у якому, для декодування зазначеного щонайменше одного біна значення індексу опорного зображення в режимі контекстного кодування, один або більше процесорів виконані з можливістю декодування щонайменше одного біна кодованої унарним кодом частини значення індексу опорного зображення в режимі контекстного кодування;

- у якому, для декодування зазначеного щонайменше іншого біна значення індексу опорного зображення в режимі кодування з обходом, один або більше процесорів виконані з можливістю декодування щонайменше іншого біна кодованої унарним кодом частини значення індексу опорного зображення і кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення в режимі кодування з обходом; і

- 30 у якому, для бінаризації значення індексу опорного зображення, один або більше процесорів виконані з можливістю кодування унарним кодом і кодування кодом з фіксованою довжиною кодового слова значення індексу опорного зображення.

- 35 74. Пристрій за п. 73, у якому один або більше процесорів додатково виконані з можливістю зрізання кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення перед кодуванням кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення.

75. Пристрій за п. 61,

- 40 у якому, для декодування в режимі контекстного кодування, один або більше процесорів виконані з можливістю вибору однієї або більше імовірнісних моделей для кодування зазначеного щонайменше одного біна бінаризованого індексу опорного зображення і декодування зазначеного щонайменше одного біна бінаризованого індексу опорного зображення, використовуючи вибрані одну або більше імовірнісних моделей; і

- 45 у якому, для декодування в режимі кодування з обходом, один або більше процесорів виконані з можливістю визначення фіксованої імовірності і декодування зазначеного щонайменше іншого біна бінаризованого значення індексу опорного зображення, використовуючи фіксовану імовірність.

- 50 76. Зчитуваний комп'ютером носій, що зберігає інструкції, які при виконанні змушують один або більше процесорів:

декодувати щонайменше один бін значення індексу опорного зображення в режимі контекстного кодування процесу контекстно-адаптивного двійкового арифметичного кодування (CABAC);

визначати, що бінаризоване значення індексу опорного зображення містить більше бінів, ніж зазначений щонайменше один бін, який кодують в режимі контекстного кодування;

- 55 декодувати, у відповідь на визначення, що значення індексу опорного зображення містить більше бінів, ніж зазначений щонайменше один бін, кодований у режимі контекстного кодування, щонайменше інший бін значення індексу опорного зображення в режимі кодування з обходом процесу CABAC; і

виконувати бінаризацію значення індексу опорного зображення.

- 60 77. Зчитуваний комп'ютером носій за п. 76,



- у якому, для декодування щонайменше одного біна значення індексу опорного зображення в режимі контекстного кодування, інструкції змушують один або більше процесорів: декодувати перший бін (bin0) значення індексу опорного зображення з першим контекстом (ctx0),
- 5 декодувати другий бін (bin1) індексу опорного зображення з другим контекстом (ctx1), і у якому, для декодування щонайменше іншого біна значення індексу опорного зображення в режимі кодування з обходом, інструкції змушують один або більше процесорів: декодувати третій бін (bin2) і всі інші біни після третього біна (bin2) у режимі кодування з обходом.
- 10 78. Зчитуваний комп'ютером носій за п. 77, у якому, для бінаризації значення індексу опорного зображення, інструкції змушують один або більше процесорів виконувати бінаризацію значення індексу опорного зображення, використовуючи комбіновані зрізаний унарний код і експоненціальний код Голомба.
- 15 79. Зчитуваний комп'ютером носій за п. 76, у якому, для декодування зазначеного щонайменше іншого біна значення індексу опорного зображення в режимі кодування з обходом, інструкції змушують один або більше процесорів декодувати щонайменше інший бін за допомогою щонайменше одного з наступних процесів кодування: унарним кодом, зрізаним унарним кодом, кодом Голомба, експоненціальним кодом Голомба або кодом Голомба-Райса.
- 20 80. Зчитуваний комп'ютером носій за п. 76, який додатково містить інструкції, що змушують один або більше процесорів: декодувати першу частину складових значення першої різниці векторів руху і першу частину складових значення другої різниці векторів руху в режимі контекстного кодування; декодувати другу частину складових значення першої різниці векторів руху і другу частину складових значення другої різниці векторів руху в режимі кодування з обходом; і
- 25 виконувати бінаризацію складових значення першої різниці векторів руху, пов'язаного з першим індексом опорного зображення, і складових значення другої різниці векторів руху, пов'язаного з другим індексом опорного зображення.
81. Зчитуваний комп'ютером носій за п. 80, який додатково містить інструкції, що змушують один або більше процесорів:
- 30 групувати першу частину складових значення першої різниці векторів руху і першу частину складових значення другої різниці векторів руху в першу групу для кодування в режимі контекстного кодування, і групувати другу частину складових значення першої різниці векторів руху і другу частину складових значення другої різниці векторів руху в другу групу для кодування в режимі кодування з обходом.
- 35 82. Зчитуваний комп'ютером носій за п. 76, у якому, для декодування щонайменше одного біна значення індексу опорного зображення в режимі контекстного кодування, інструкції змушують один або більше процесорів: декодувати перший бін (bin0) значення індексу опорного зображення з першим контекстом (ctx0),
- 40 декодувати другий бін (bin1) індексу опорного зображення з другим контекстом (ctx1), декодувати третій бін (bin2) індексу опорного зображення з третім контекстом (ctx2), і у якому, для декодування щонайменше іншого біна значення індексу опорного зображення в режимі кодування з обходом, інструкції змушують один або більше процесорів: декодувати всі інші біни після третього біна (bin2) у режимі кодування з обходом.
- 45 83. Зчитуваний комп'ютером носій за п. 76, у якому, для декодування щонайменше одного біна значення індексу опорного зображення в режимі контекстного кодування, інструкції змушують один або більше процесорів: декодувати перший бін (bin0) значення індексу опорного зображення з першим контекстом (ctx0), і
- 50 у якому, для декодування щонайменше іншого біна значення індексу опорного зображення в режимі кодування з обходом, інструкції змушують один або більше процесорів: декодувати всі інші біни після першого біна (bin0) у режимі кодування з обходом.
84. Зчитуваний комп'ютером носій за п. 76,
- 55 у якому, для декодування зазначеного щонайменше одного біна значення індексу опорного зображення в режимі контекстного кодування, інструкції змушують один або більше процесорів: декодувати щонайменше один бін кодованого унарним кодом значення індексу опорного зображення в режимі контекстного кодування; у якому, для декодування щонайменше іншого біна значення індексу в режимі кодування з обходом, інструкції змушують один або більше процесорів:
- 60

декодувати щонайменше інший біт кодованого унарним кодом значення індексу опорного зображення в режимі кодування з обходом; і

у якому, для бінаризації значення індексу опорного зображення, інструкції змушують один або більше процесорів кодувати унарним кодом значення індексу опорного зображення.

5 85. Зчитуваний комп'ютером носій за п. 84, у якому, для кодування унарним кодом значення індексу опорного зображення, інструкції змушують один або більше процесорів кодувати зрізаним унарним кодом значення індексу опорного зображення.

86. Зчитуваний комп'ютером носій за п. 76,

10 у якому, для декодування зазначеного щонайменше одного біта значення індексу опорного зображення в режимі контекстного кодування, інструкції змушують один або більше процесорів: декодувати щонайменше один біт кодованої унарним кодом частини значення індексу опорного зображення в режимі контекстного кодування;

у якому, для декодування зазначеного щонайменше іншого біта значення індексу опорного зображення в режимі кодування з обходом, інструкції змушують один або більше процесорів:

15 декодувати щонайменше інший біт кодованої унарним кодом частини значення індексу опорного зображення і кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення в режимі кодування з обходом; і

у якому, для бінаризації значення індексу опорного зображення, інструкції змушують один або більше процесорів кодувати унарним кодом і кодувати експоненціальним кодом Голомба значення індексу опорного зображення.

20 87. Зчитуваний комп'ютером носій за п. 86, який додатково містить інструкції, що змушують один або більше процесорів зрізати кодовану експоненціальним кодом Голомба частину значення індексу опорного зображення перед декодуванням кодованої експоненціальним кодом Голомба частини значення індексу опорного зображення.

25 88. Зчитуваний комп'ютером носій за п. 76,

у якому, для декодування зазначеного щонайменше одного біта значення індексу опорного зображення в режимі контекстного кодування, інструкції змушують один або більше процесорів декодувати щонайменше один біт кодованої унарним кодом частини значення індексу опорного зображення в режимі контекстного кодування;

30 у якому, для декодування зазначеного щонайменше іншого біта значення індексу опорного зображення в режимі кодування з обходом, інструкції змушують один або більше процесорів декодувати щонайменше інший біт кодованої унарним кодом частини значення індексу опорного зображення і кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення в режимі кодування з обходом; і

35 у якому, для бінаризації значення індексу опорного зображення, інструкції змушують один або більше процесорів кодувати унарним кодом і кодувати кодом з фіксованою довжиною кодового слова значення індексу опорного зображення.

89. Зчитуваний комп'ютером носій за п. 88, який додатково містить інструкції, що змушують один або більше процесорів зрізати кодовану кодом з фіксованою довжиною кодового слова частину значення індексу опорного зображення перед кодуванням кодованої кодом з фіксованою довжиною кодового слова частини значення індексу опорного зображення.

90. Зчитуваний комп'ютером носій за п. 76,

у якому, для декодування в режимі контекстного кодування, інструкції змушують один або більше процесорів вибирати одну або більше імовірнісних моделей для кодування зазначеного щонайменше одного біта бінаризованого індексу опорного зображення і декодувати зазначений щонайменше один біт бінаризованого індексу опорного зображення, використовуючи вибрані одну або більше імовірнісних моделей; і

у якому, для декодування в режимі кодування з обходом, інструкції змушують один або більше процесорів визначати фіксовану імовірність і декодувати зазначений щонайменше інший біт бінаризованого значення індексу опорного зображення, використовуючи фіксовану імовірність.

50 91. Зчитуваний комп'ютером носій за п. 76, який додатково містить інструкції, що змушують один або більше процесорів:

декодувати перший біт синтаксичного елемента напрямку зовнішнього прогнозування в режимі контекстного кодування процесу CABAC; і

55 декодувати другий біт синтаксичного елемента напрямку зовнішнього прогнозування в режимі кодування з обходом процесу CABAC.

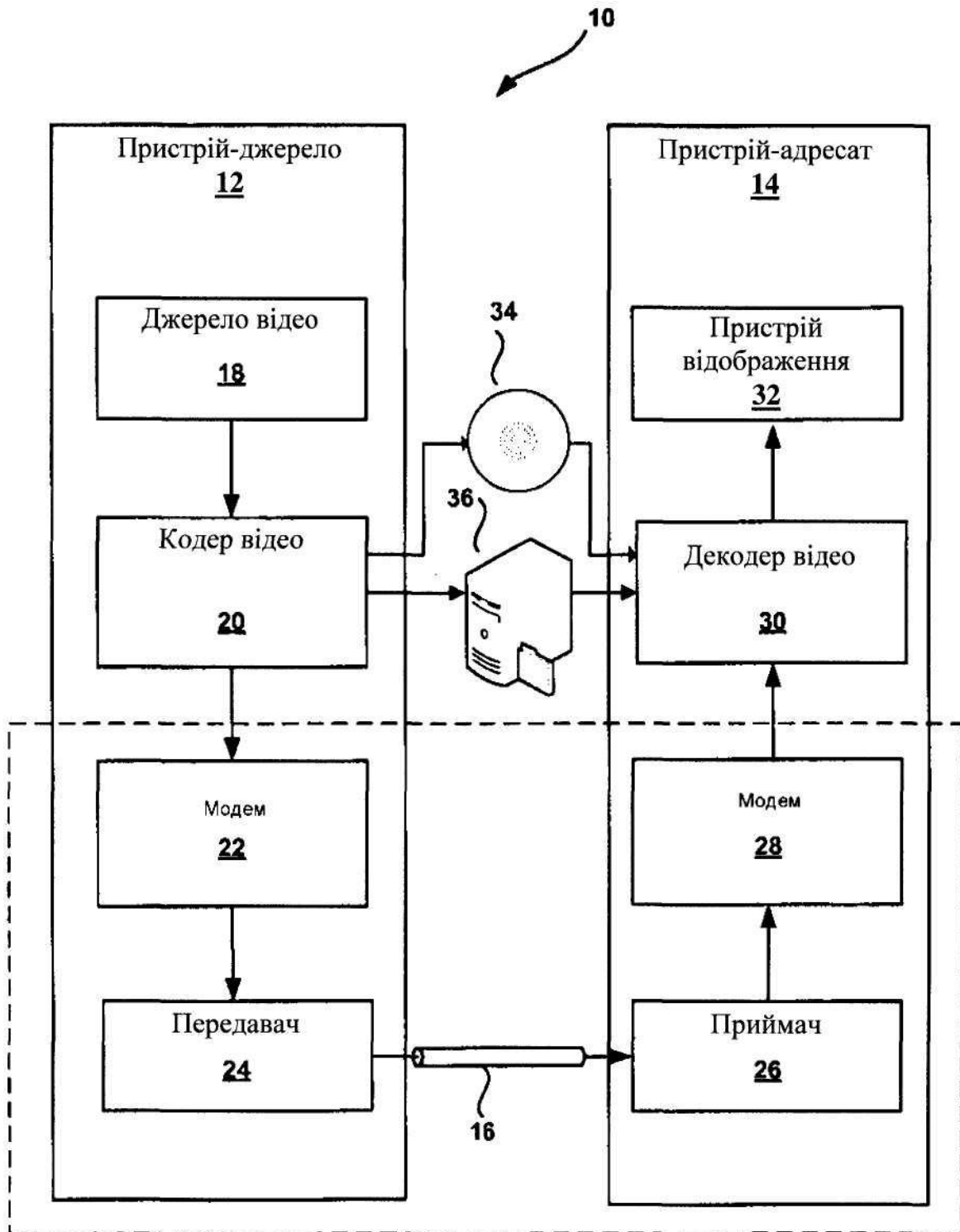
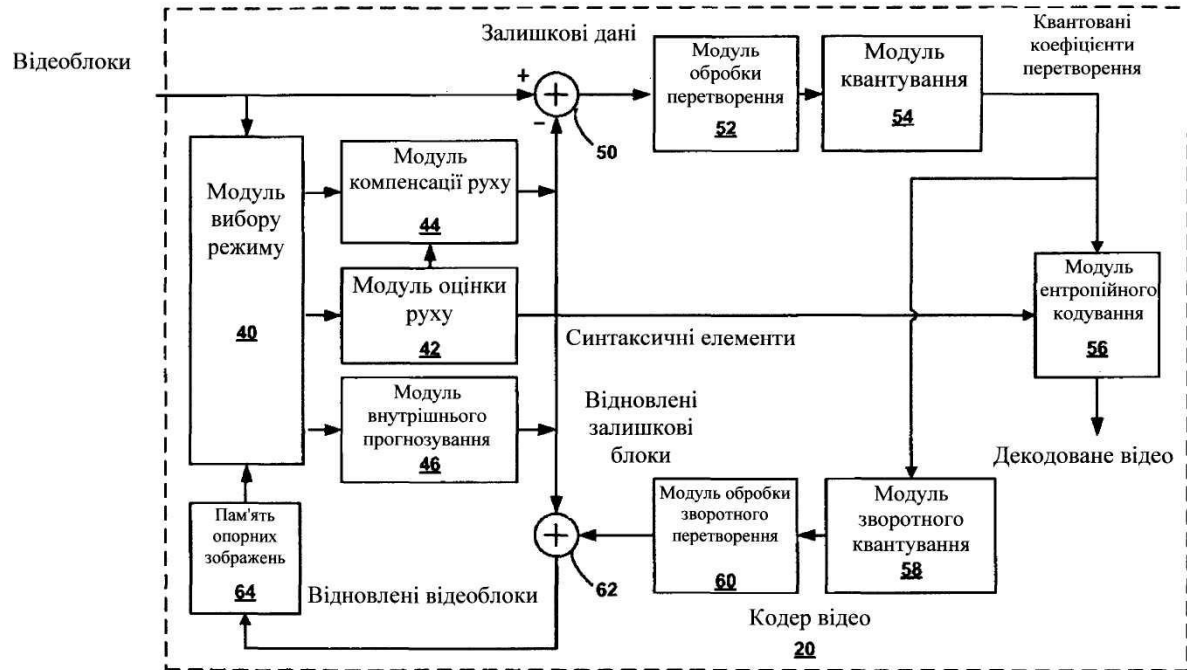
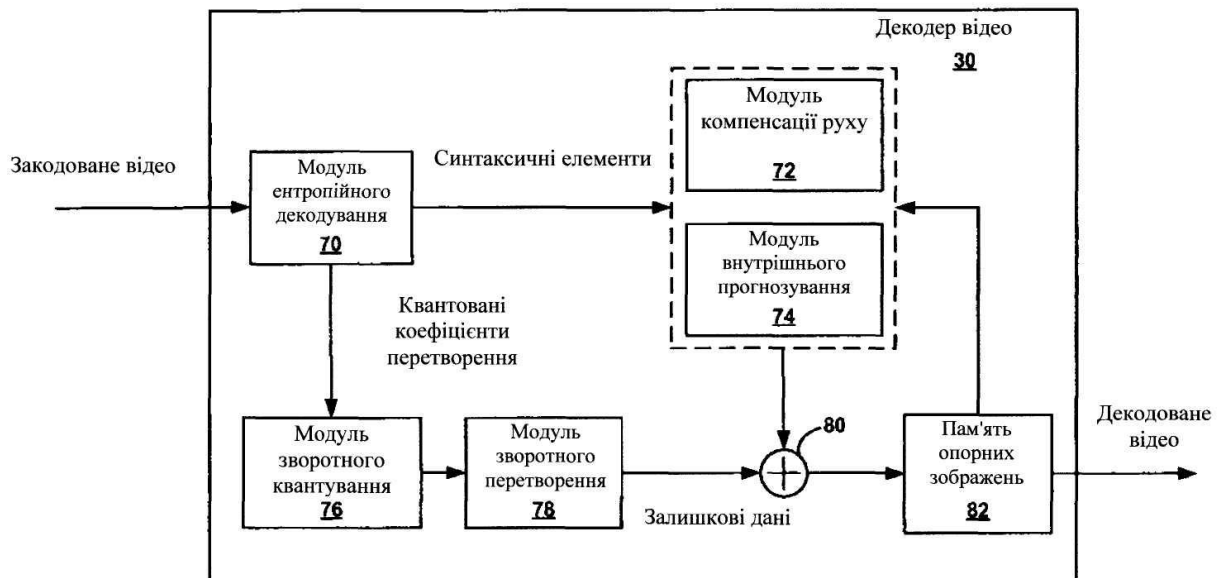


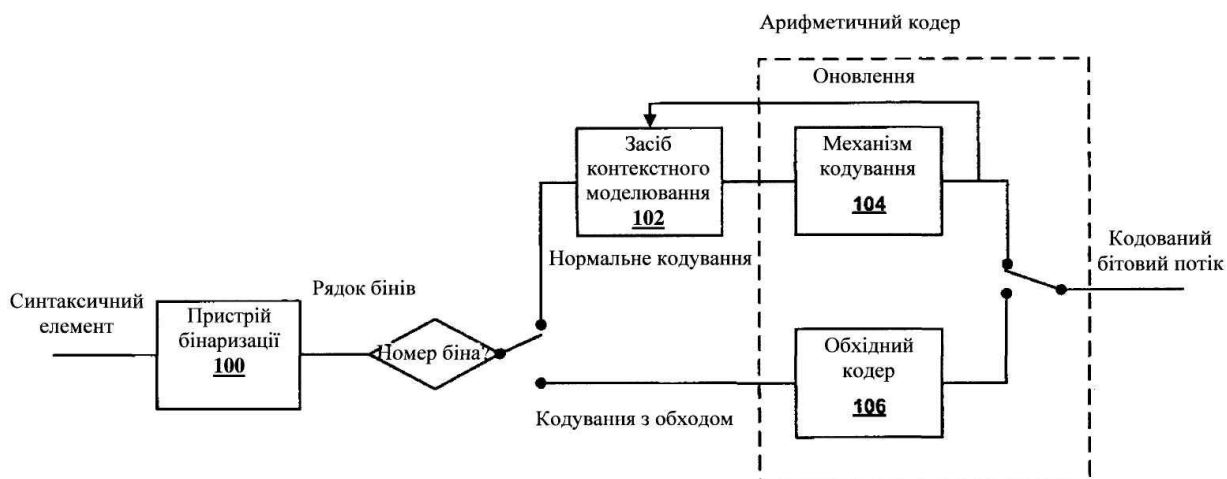
Fig. 1



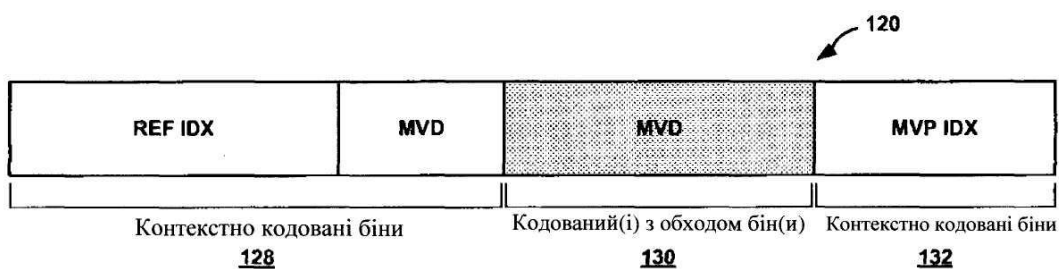
Фіг. 2



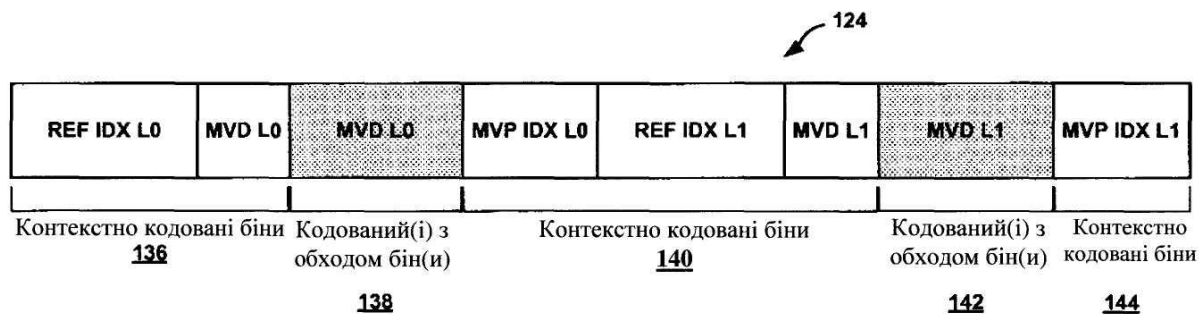
Фіг. 3



Фіг. 4



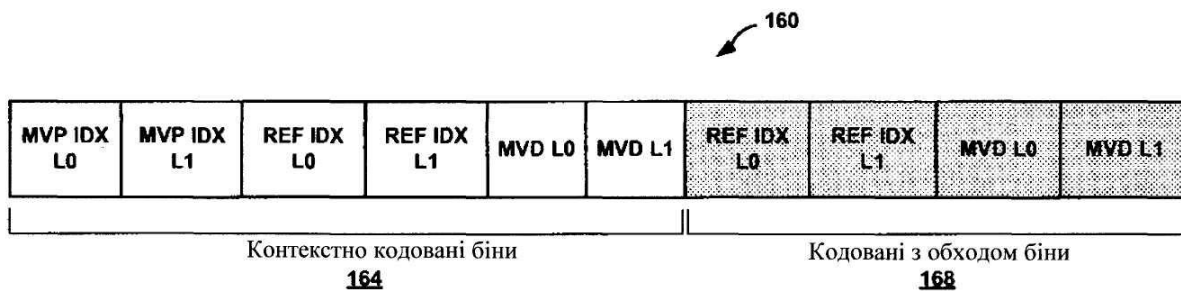
Фіг. 5A



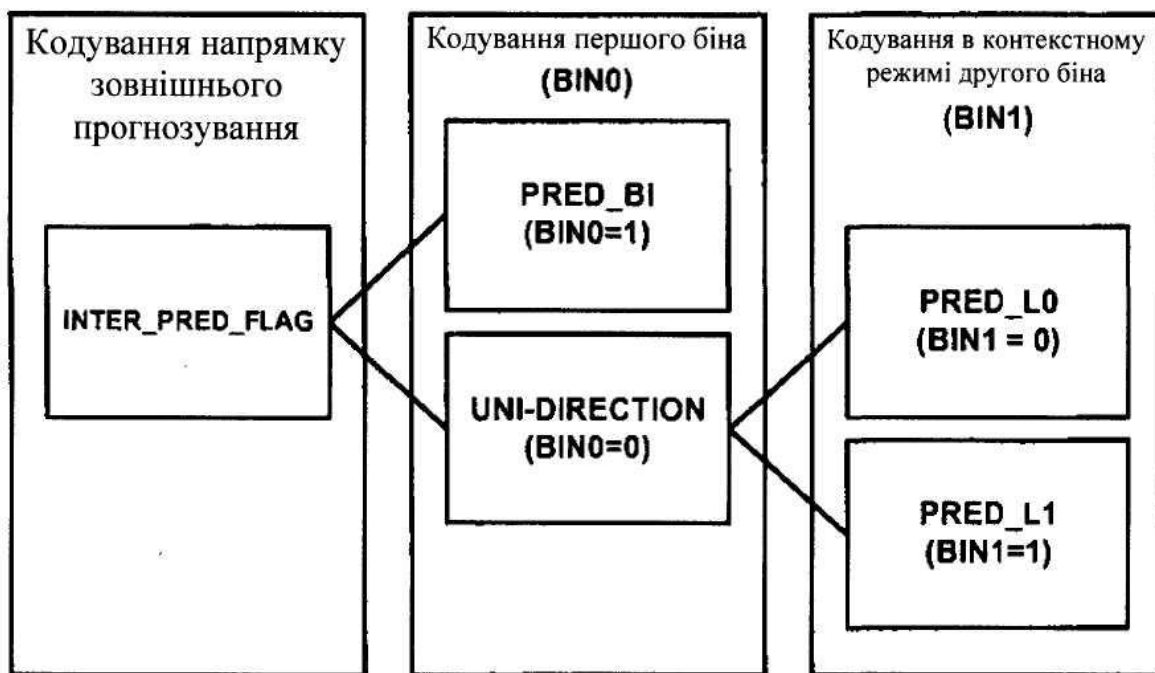
Фіг. 5B



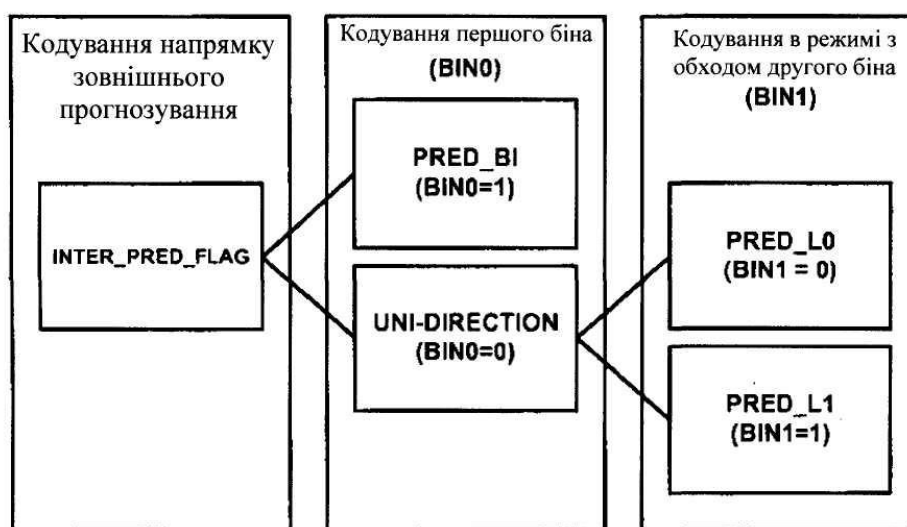
Фіг. 6



Фіг. 7



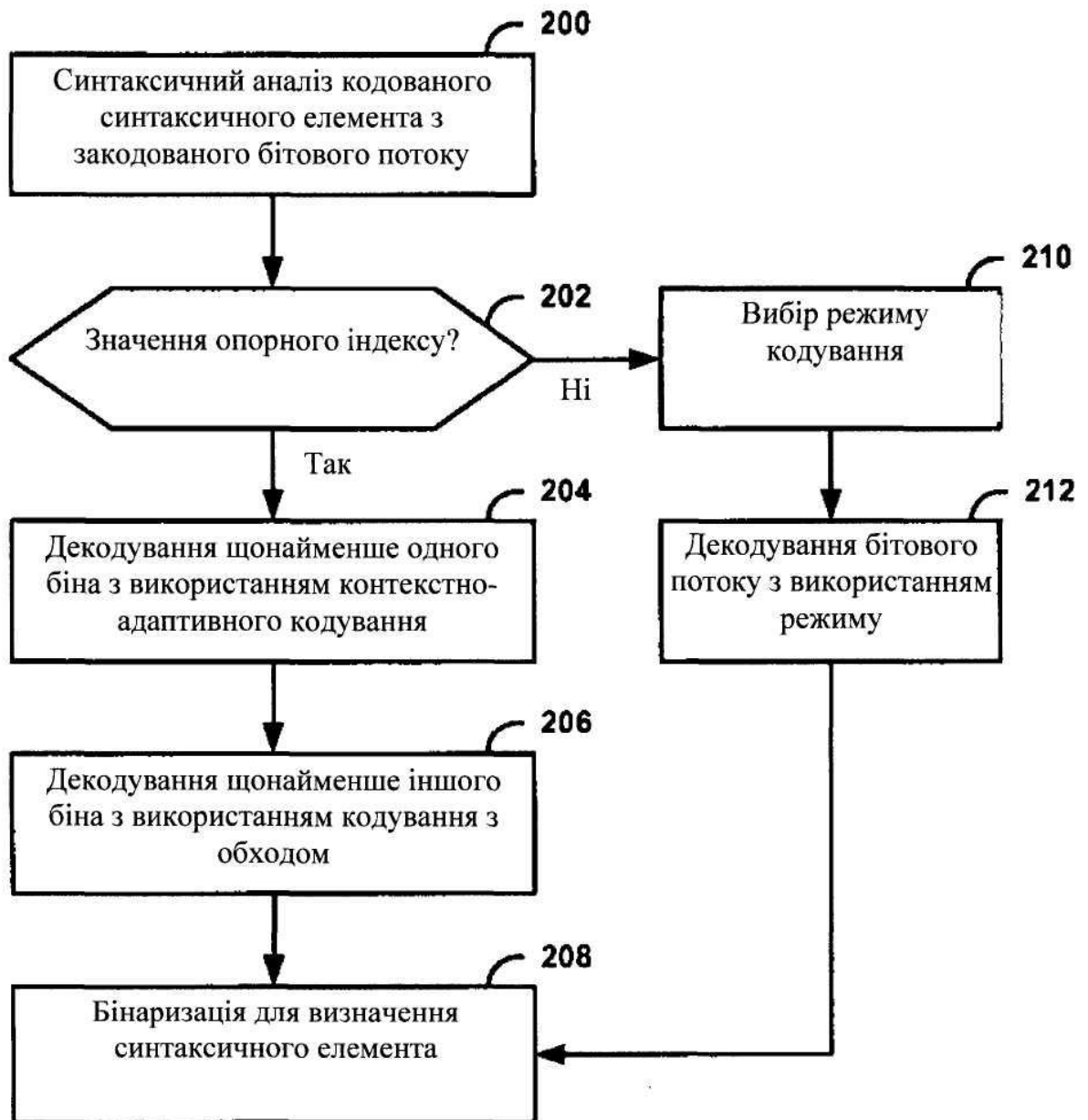
Фіг. 8А



Фіг. 8В



Фіг. 9

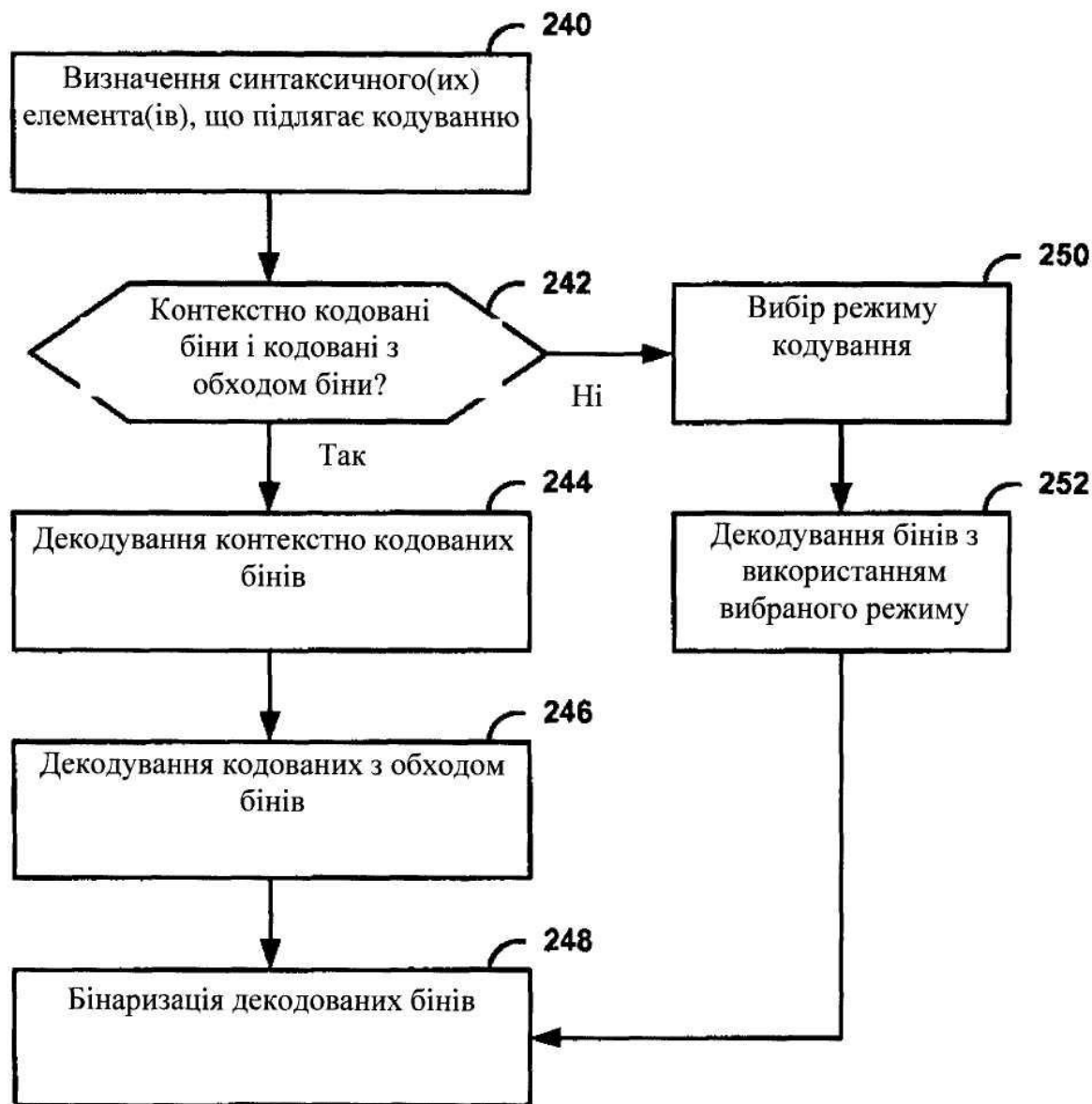


Фіг. 10





Фіг. 11



Фіг. 12