



УКРАЇНА

(19) **UA** (11) **119432** (13) **C2**
(51) МПК (2019.01)

H04N 7/00

H04N 21/234 (2011.01)

H04N 21/6336 (2011.01)

H04N 21/647 (2011.01)

МІНІСТЕРСТВО
ЕКОНОМІЧНОГО
РОЗВИТКУ І ТОРГІВЛІ
УКРАЇНИ

(12) ОПИС ДО ПАТЕНТУ НА ВИНАХІД

(21) Номер заявки: а 2015 00660	(72) Винахідник(и): Ван Є-Куй (US), Чень Ін (US)
(22) Дата подання заявки: 17.06.2013	(73) Власник(и): КВЕЛКОММ ІНКОРПОРЕЙТЕД, Attn: International IP Administration, 5775 Morehouse Drive, San Diego, California 92121-1714, United States of America (US)
(24) Дата, з якої є чинними права на винахід: 25.06.2019	(74) Представник: Мошинська Ніна Миколаївна, реєстр. №115
(31) Номер попередньої заявки відповідно до Паризької конвенції: 61/665,667, 13/773,060	(56) Перелік документів, взятих до уваги експертизою: US 2010/254456 A1, 07.10.2010 US 2005/123274 A1, 09.06.2005 Gary J. Sullivan. CRA pictures with broken links. 100. MPEG meeting; 30.04.2012 - 04.05.2012; Geneva; (Motion Picture Expert Group or ISO/IEC JTC1/SC29/WG11), 2012- 05-07, Nr: m24653. XP 030052996
(32) Дата подання попередньої заявки відповідно до Паризької конвенції: 28.06.2012, 21.02.2013	
(33) Код держави-учасниці Паризької конвенції, до якої подано попередню заявку: US, US	
(41) Публікація відомостей про заявку: 27.04.2015, Бюл.№ 8	
(46) Публікація відомостей про видачу патенту: 25.06.2019, Бюл.№ 12	
(86) Номер та дата подання міжнародної заявки, поданої відповідно до Договору РСТ РСТ/US2013/046152, 17.06.2013	

(54) АДАПТАЦІЯ ПОТОКОВОЇ ПЕРЕДАЧІ НА ОСНОВІ ЗОБРАЖЕНЬ ЧИСТОГО ДОВІЛЬНОГО ДОСТУПУ (CRA)

(57) Реферат:

Розкриті системи, способи та пристрої для обробки відеоданих. Деякі зразкові системи, способи та пристрої приймають зовнішню вказівку на відеодекодері. Зразкові системи, способи та пристрої обробляють зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі зовнішньої вказівки.

UA 119432 C2

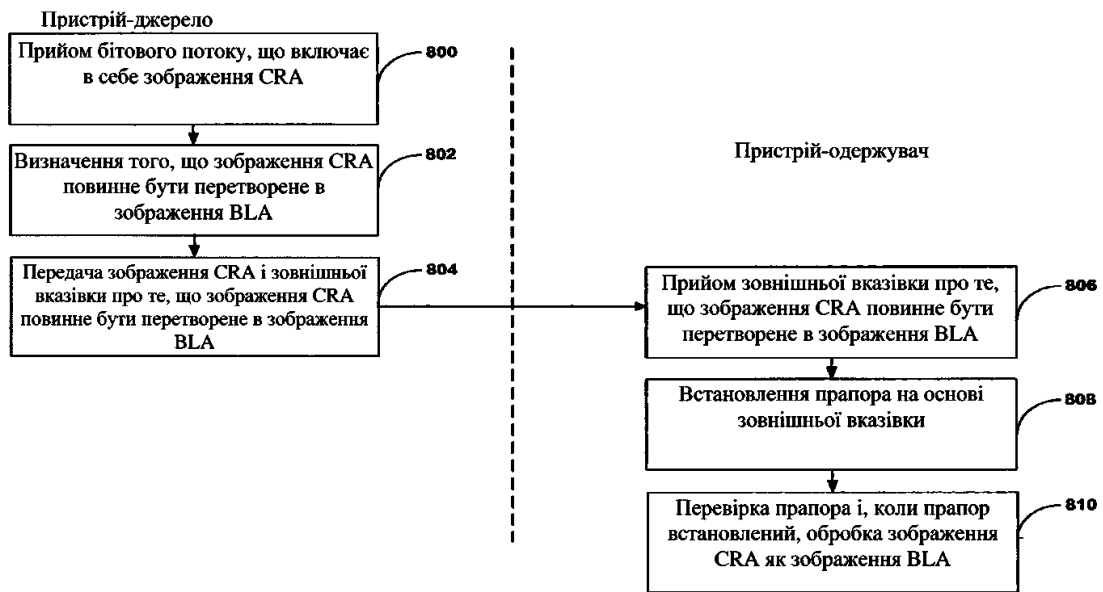


Fig. 8

ДАНА ЗАЯВКА ЗАЯВЛЯЄ ПРІОРИТЕТ:

попередньої заявки США № 61/665,667, поданої 28 Червня, 2012, повний вміст якої включений в даний документ за допомогою посилання.

ГАЛУЗЬ ТЕХНІКИ

- 5 В цілому дане розкриття належить до обробки відеоданих і, більш конкретно, до прийомів для підтримки довільного доступу в стиснутих відеопотоках.

РІВЕНЬ ТЕХНІКИ

- Можливості цифрового відео можуть бути включені в широкий діапазон пристроїв, які включають в себе засоби цифрового телебачення, системи цифрового прямого мовлення, системи бездротового мовлення, кишенькові персональні комп'ютери (PDA), переносні або настільні комп'ютери, планшетні комп'ютери, засоби читання електронних книг, цифрові камери, пристрої цифрового запису, цифрові мультимедійні програвачі, відео-ігрові пристрої, консолі відеоігор, стільникові або супутникові радіотелефони, так звані "смартфони", пристрої відеоконференцзв'язку, пристрої потокової передачі відео, транскодери, маршрутизатори або інші мережні пристрої, і тому подібне. Цифрові відеопристрої реалізують прийоми стиснення відео подібні тим, що описані в стандартах, визначених в MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Частина 10, Вдосконаленого Відеокодування (AVC), стандарті Високоєфективного Відеокодування (HEVC), що розробляється в цей час, приватних стандартах, відкритих форматах стиснення відео, таких як VP8, і розширеннях таких стандартів, прийомів або форматів. Відеопристрої можуть передавати, приймати, кодувати, декодувати, і/або зберігати цифрову відеоінформацію більш ефективно за допомогою реалізації таких прийомів стиснення відео.

- Прийоми стиснення відео виконують просторове (всередині зображення) прогнозування і/або часове (між зображеннями) прогнозування для зменшення або видалення надмірності, властивій відеопослідовностям. Для кодування відео на основі блоків, відеослайс (тобто, відеозображення або частина відеозображення) може бути розділений на відеоблоки, які можуть також згадуватися як деревоподібні блоки, одиниці (CU) кодування і/або вузли кодування. Відеоблоки у внутрішньо кодованому (I) слайсі зображення кодуються з використанням просторового прогнозування відносно опорних вибірок у сусідніх блоках в тому самому зображенні. Відеоблоки у зовні кодованому (P або B) слайсі зображення можуть використовувати просторове прогнозування відносно опорних вибірок у сусідніх блоках в тому самому зображенні або часове прогнозування відносно опорних вибірок в інших опорних зображеннях. Зображення можуть згадуватися як кадри, і опорні зображення можуть згадуватися як опорні кадри.

- Просторове або часове прогнозування має результатом предиктивний блок для блока, який повинен бути кодований. Залишкові дані представляють піксельні різниці між вихідним блоком, який повинен бути кодований, і предиктивним блоком. Зовні кодований блок кодується відповідно до вектора руху, який вказує на блок опорних вибірок, що формують предиктивний блок, і залишкових даних, які вказують різницю між кодованим блоком і предиктивним блоком. Внутрішньо кодованим блок кодується згідно з режимом внутрішнього кодування і залишковими даними. Для додаткового стиснення залишкові дані можуть бути перетворені з піксельної ділянки в ділянку перетворення, маючи результатом залишкові коефіцієнти перетворення, які потім можуть бути квантовані. Квантовані коефіцієнти перетворення, спочатку впорядковані в двовимірному масиві, можуть бути проскановані для створення одновимірного вектора коефіцієнтів перетворення і ентропійне кодування може бути застосоване для досягнення ще більшого стиснення.

СУТЬ ВІНАХОДУ

- В одному прикладі прийоми даного розкриття належать до обробки зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі зовнішньої вказівки. Наприклад, відеодекодер або інший пристрій можуть приймати зовнішню вказівку. Відеодекодер може потім обробляти зображення CRA як зображення BLA на основі зовнішньої вказівки. У деяких прикладах, прапор задається для зображення CRA і зовнішня вказівка вказує, чи повинен прапор бути встановлений у відеодекодері. Відповідно, відеодекодер може встановлювати прапор на основі зовнішньої вказівки. Декодер або деякі внутрішні функціональні засоби, такі як блок обробки зовнішньої вказівки або модуль прогнозування можуть потім перевірити прапор. Наприклад, модуль прогнозування може обробляти зображення CRA як зображення BLA на основі зовнішньої вказівки. Наприклад, декодер може обробляти зображення CRA як зображення BLA на основі прапора.

В одному прикладі дане розкриття описує спосіб обробки відеоданих, який включає в себе прийом зовнішньої вказівки на відеодекодері та обробку зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі зовнішньої вказівки.

В іншому прикладі дане розкриття описує відеодекодер для обробки відеоданих, який
5 включає в себе процесор, сконфігурований для прийому зовнішньої вказівки на відеодекодері та розгляду зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі зовнішньої вказівки.

В іншому прикладі дане розкриття описує відеодекодер для обробки відеоданих, який
10 включає в себе засіб для прийому зовнішньої вказівки на відеодекодері та засіб для обробки зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі зовнішньої вказівки.

В іншому прикладі дане розкриття описує зчитуваний комп'ютером носій даних. Зчитуваний комп'ютером носій даних має збережені на ньому інструкції, які при виконанні спонукають один
15 або більше процесорів пристрою приймати зовнішню вказівку на відеодекодері та обробляти зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі зовнішньої вказівки.

Подробиці одного або більше прикладів викладені в супровідних кресленнях та описі нижче за текстом. Інші ознаки, цілі та переваги будуть очевидні з опису і креслень та з формули винаходу.

20 КОРОТКИЙ ОПИС КРЕСЛЕНЬ

Фіг. 1 є блок-схемою, що ілюструє зразкову систему кодування і декодування відео, яка може використовувати прийоми, описані в даному розкритті.

Фіг. 2 є блок-схемою, що ілюструє зразковий відеокодер, який може реалізовувати прийоми, описані в даному розкритті.

25 Фіг. 3 є блок-схемою, що ілюструє зразковий відеодекодер, який може реалізовувати прийоми, описані в даному розкритті.

Фіг. 4 є блок-схемою, що ілюструє зразковий набір пристроїв, які формують частину мережі.

Фіг. 5 є блок-схемою послідовності операцій, що ілюструє приклад способу відповідно до
одного або більше прикладів, описаних в даному розкритті.

30 Фіг. 6 є блок-схемою послідовності операцій, що ілюструє зразковий спосіб відповідно до одного або більше прикладів, описаних в даному розкритті.

Фіг. 7 є блок-схемою послідовності операцій, що ілюструє зразковий спосіб відповідно до одного або більше прикладів, описаних в даному розкритті.

35 Фіг. 8 є блок-схемою послідовності операцій, що ілюструє зразкову роботу з відправлення першим пристроєм зовнішньої вказівки і дії у відповідь другого пристрою з прийому зовнішньої вказівки.

ДОКЛАДНИЙ ОПИС

Дане розкриття описує прийоми для адаптації потокової передачі на основі зображень чистого довільного доступу (CRA). Описані різні поліпшені проектні рішення відеокодування, які
40 можуть бути зв'язані з адаптацією потокової передачі на основі зображень CRA, виведенням зображень до зображень точки довільного доступу (RAP) і сигналізацією інформації синхронізації зображень.

Спочатку описується короткий рівень техніки деяких стандартів відеокодування. Стандарти відеокодування включають в себе ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 або ISO/IEC
45 MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual і ITU-T H.264 (також відомий як ISO/IEC MPEG-4 AVC), що включає в себе розширення: Масштабоване Відеокодування (SVC) і Багаторакурсне Відеокодування (MVC).

Додатково, є новий стандарт відеокодування, який називається Високоєфективним Кодуванням Відео (HEVC), що розробляється Об'єднаною Командою з Кодування Відео (JCT-VC) Експертної Групи з Кодування Відео ITU-T (VCEG) і Експертною Групою з Рухомих
50 Зображень ISO/IEC (MPEG). Робочий Проект (WD) HEVC, що згадується як HEVC WD7 в цьому документі, доступний за посиланням http://phenix.int-evry.fr/jct/doc_end_user/documents/9_Geneva/wg11/JCTVC-I1003-v5.zip, повний зміст якого включений в цей документ за допомогою посилання.

55 Більш новий Робочий Проект (WD) HEVC, що згадується як HEVC WD9 в цьому документі, доступний за посиланням http://phenix.int-evry.fr/jct/doc_end_user/documents/9_Geneva/wg11/JCTVC-I1003-v10.zip, повний зміст якого включений в цей документ за допомогою посилання.

60 В одному прикладі прийоми даного розкриття належить до обробки зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі

зовнішньої вказівки. Наприклад, відеодекодер або інший пристрій можуть приймати зовнішню вказівку. Відеодекодер може потім обробляти зображення CRA як зображення BLA на основі зовнішньої вказівки. У деяких прикладах прапор задається для зображення CRA і зовнішня вказівка вказує, чи повинен прапор бути встановлений у відеодекодері. Відповідно, відеодекодер може встановлювати прапор на основі зовнішньої вказівки. Декодер або деякі внутрішні функціональні засоби, такі як блок обробки зовнішньої вказівки або модуль прогнозування можуть потім перевірити прапор. В одному прикладі модуль прогнозування може обробляти зображення CRA як зображення BLA на основі зовнішньої вказівки. Наприклад, декодер може обробляти зображення CRA як зображення BLA на основі прапора.

В іншому прикладі прапор задається для зображення CRA і декодер або інший пристрій можуть приймати зовнішню вказівку про те, що прапор повинен бути встановлений. Декодер або інший пристрій можуть потім встановити прапор на основі зовнішньої вказівки. Декодер може потім перевірити прапор. Коли прапор встановлений, декодер може обробляти зображення CRA як зображення BLA.

Довільний доступ належить до декодування бітового потоку відео, що починається з кодованого зображення, яке не є першим кодованим зображенням в бітовому потоці. Довільний доступ до бітового потоку необхідний в багатьох практичних застосуваннях відео, таких як мовлення і потокова передача, наприклад, для перемикавання користувачами між різними каналами, для швидкого переходу до конкретних частин відео, або для перемикавання на інший бітовий потік для адаптації потоку (наприклад, частота проходження бітів, частоти кадрів, просторового розрізнення і так далі). Ця особливість може бути забезпечена за допомогою вставки зображень довільного доступу або точок довільного доступу багаторазово з постійними інтервалами в бітовий потік відео.

З'єднання бітових потоків належить до зв'язку двох або більше бітових потоків або їх частин. Наприклад, перший бітовий потік може бути приєднаний до другого бітового потоку, можливо з деякими модифікаціями або одного, або обох бітових потоків, для генерування з'єданого бітового потоку. Перше кодоване зображення у другому бітовому потоці також згадується як точка з'єднання. Тому зображення, що йдуть за точкою з'єднання в з'єданому бітовому потоці, походять з другого бітового потоку, тоді як зображення, що передують точці з'єднання в з'єданому бітовому потоці, походять з першого бітового потоку.

Засоби з'єднання бітових потоків можуть виконувати з'єднання бітових потоків. Засоби з'єднання бітових потоків часто є менш складними, менш сучасними і/або менш інтелектуальними, ніж кодери. Наприклад, вони можуть не бути оснащені можливостями ентропійного кодування і декодування. Засоби з'єднання бітових потоків можуть бути включені в будь-який з пристроїв, описаних в цьому документі, що включають в себе пристрої кодування або мережні пристрої.

Перемикання бітового потоку може бути використане в середовищах адаптивної потокової передачі. Операція перемикання бітового потоку на деяке зображення в бітовому потоці, на який здійснюється перемикання, є ефективною операцією з'єднання бітових потоків, в якій точкою з'єднання є точка перемикання бітового потоку, тобто, перше зображення з бітового потоку, на який здійснюється перемикання.

Зображення миттєвого оновлення декодування (IDR), визначені в AVC або HEVC, можуть бути використані для довільного доступу. Однак, оскільки зображення, що йдуть за зображенням IDR в порядку декодування, не можуть використовувати зображення, декодовані до зображення IDR як посилання, бітові потоки, що покладаються на зображення IDR для довільного доступу, можуть значно знизити ефективність кодування.

Для поліпшення ефективності кодування поняття зображень чистого довільного доступу (CRA) було введено в HEVC, щоб дозволити зображенням, які йдуть за зображенням CRA в порядку декодування, але передують йому в порядку виведення, використовувати зображення, декодовані до зображення CRA, як посилання. Зображення, які йдуть за зображенням CRA в порядку декодування, але передують зображенню CRA в порядку виведення, згадуються як ведучі зображення, асоційовані із зображенням CRA (або ведучі зображення для зображення CRA). Ведучі (початкові) зображення для зображення CRA є коректно декодованими, якщо декодування починається із зображення CRA або IDR перед поточним зображенням CRA. Однак, ведучі зображення для зображення CRA можуть не бути коректно декодованими, коли виникає довільний доступ із зображення CRA. Отже, декодери типово відкидають ведучі зображення під час декодування з довільним доступом. Для запобігання поширенню помилки від опорних зображень, які можуть не бути доступні залежно від того, де починається декодування, всі зображення, які йдуть за зображенням CRA як в порядку декодування так і в порядку виведення, не повинні використовувати яке-небудь зображення, яке передус

зображенню CRA або в порядку декодування, або в порядку виведення (який включає в себе ведучі зображення) як посилання.

Поняття зображення доступу з розірваним посиланням (BLA) було додатково введене в HEVC після введення зображень CRA і на основі поняття зображень CRA. Зображення BLA

5 типowo виникає із з'єднання бітових потоків у позиції зображення CRA, і в з'єднаному бітовому потоці зображення CRA точки з'єднання може бути змінено на зображення BLA. Зображення IDR, зображення CRA і зображення BLA спільно згадуються як зображення точки довільного доступу (RAP).

Однією відмінністю між зображеннями BLA і зображеннями CRA є наступне. Для

10 зображення CRA асоційовані ведучі зображення є коректно декодованими, якщо декодування починається із зображення RAP перед згаданим зображенням CRA в порядку декодування. Зображення CRA може не бути коректно декодованим, коли виникає довільний доступ із зображення CRA. Наприклад, коли декодування починається із зображення CRA або, іншими словами, коли зображення CRA є першим зображенням в бітовому потоці. Для зображення BLA

15 асоційовані ведучі зображення можуть не бути коректно декодованими у всіх випадках, навіть коли декодування починається із зображення RAP перед згаданим зображенням BLA в порядку декодування.

Для деякого конкретного зображення BLA або CRA деякі з асоційованих ведучих зображень є коректно декодованими навіть, коли зображення BLA або CRA є першим зображенням в бітовому потоці. Ці ведучі зображення згадуються як декодовані ведучі зображення (DLP) і інші ведучі зображення згадуються як недековані ведучі зображення (NLP) або декодовані ведучі зображення довільного доступу (RADL). NLP також згадуються як відмічені для відкидання зображення (TFD) або ведучі зображення, що пропускають довільний доступ (RASL).

У деяких випадках нижченаведені проблеми можуть бути асоційовані з деякими існуючими

25 способами (1) в адаптації потокової передачі на основі зображень CRA, зміну зображення CRA на зображення BLA звичайно необхідно виконувати за допомогою медіа-сервера або проміжного мережного елемента, наприклад обізнаного про середовище мережного елемента (MANE) або навіть не обізнаного про середовище мережного елемента, такого як кеш HTTP або веб-проксі, MANE, який звичайно, переважно, є менш складним, менш сучасним і/або менш інтелектуальним і може не мати можливості зміни бітового потоку зовсім, (2) виведення зображень перед зображенням BLA або IDR в порядку декодування може, яким-небудь чином, керуватися за допомогою використання `no_output_of_prior_pics_flag`. Коли `no_output_of_prior_pics_flag` встановлений в "1" або виведений рівним 1, всі зображення, декодовані в порядку декодування раніше, ніж зображення BLA або IDR, відкидаються після

35 декодування зображення BLA або IDR без виведення/відображення. Однак, іноді відображення більшої кількості цих зображень може забезпечити краще сприйняття користувачем. У даний момент, відсутній спосіб для забезпечення виведення/відображення більшої кількості зображень в подібних ситуаціях, (3) зображенням DLP дозволено виводитися. Оскільки їх порядок виведення або час виведення є більш ранніми, ніж у асоційованого зображення RAP, найбільш ранній час представлення в ході довільного доступу із зображення RAP не може бути відомий за допомогою простої перевірки одиниці доступу, що містить зображення RAP. Однак, в ході довільного доступу із зображення RAP система повинна спробувати обчислити найбільш ранній початок відтворення, щоб з'ясувати, чи відповідає зображення RAP запиту довільного доступу від користувача.

У даному розкритті викладена деяка кількість прийомів, які можуть, загалом, вирішити або перевершити одну або більше з ідентифікованих вище за текстом проблем. Деяка кількість шляхів обробки подібного повідомлення, прийнятого або виведеного, є можливими. Декілька прикладів розглядаються нижче за текстом; вони включають в себе (1) розгляд зображення CRA як зображення BLA, (2) зміну зображення CRA на зображення BLA і (3) розгляд зображення CRA як зображення CRA, яке починає бітовий потік.

50

У деякому прикладі декодер може розглядати зображення CRA як зображення BLA. Декодер може бути сконфігурований таким чином, що зображення CRA розглядається як зображення BLA, коли це вказане зовнішнім засобом. Такою зовнішньою вказівкою може бути повідомлення, як описано вище за текстом (що деяке зображення CRA повинне бути розглянуте як зображення BLA), яке передається декодеру за допомогою функції сторони декодера, через виведення або прийом від сервера або проміжного мережного елемента.

55

Більш конкретно, процес декодування може бути змінений таким чином. Окрема змінна, яка може бути асоційована з кожним зображенням CRA, може бути використана. Наприклад, змінна `HandleCraAsBlaFlag` асоціюється з кожним зображенням CRA. Іншими словами, кожне зображення CRA може мати змінну `HandleCraAsBlaFlag` (що також згадується як прапор),

60

асоційовану з ним. Значення `HandleCraAsBlaFlag` для деяких зображень CRA може бути точно визначене зовнішніми засобами. Коли значення `HandleCraAsBlaFlag` для деякого конкретного зображення CRA не точно визначене зовнішніми засобами, воно може бути встановлене в "0" (наприклад, `HandleCraAsBlaFlag` зображення CRA за замовчуванням рівне "0" зі значенням "0",
 5 яке вказує, що зображення CRA не обробляється як зображення BLA). У подібному прикладі значення "1" може вказувати, що зображення CRA обробляється як зображення BLA. В інших прикладах істиною може бути протилежне і значення "1" може вказувати, що зображення CRA не обробляється як зображення BLA і значення "0" може вказувати, що зображення CRA обробляється як зображення BLA.

10 Нижченаведений приклад передбачає випадок, коли `HandleCraAsBlaFlag` за замовчуванням має значення "0", яке вказує, що зображення CRA не обробляється як зображення BLA і значення "1", яке вказує, що зображення CRA обробляється як зображення BLA. Коли декодують (в тому числі аналізують) одиницю NAL кожного кодованого слайса, якщо `HandleCraAsBlaFlag` рівний "1", наприклад, розгляд зображення CRA як зображення BLA і
 15 `nal_unit_type` вказує зображення CRA (наприклад, значення рівне "4" або "5" згідно з HEVC WD7), застосовується нижченаведене, (1) значення `nal_unit_type` змінюється для вказівки зображення BLA (наприклад, значення збільшується на 2 згідно з HEVC WD7), (2) значення `po_output_of_prior_pics_flag` встановлюється в 1, (3) якщо попереднім зображенням в порядку декодування є зображення RAP і `rap_pic_id` поточного слайса рівний `rap_pic_id` попереднього зображення, застосовується нижченаведене. По-перше, якщо наступним зображенням в порядку декодування не є зображення RAP, значення `rap_pic_id` поточного слайса змінюється, щоб бути відмінним від `rap_pic_id` попереднього зображення в порядку декодування, але все ще знаходиться в дозволеному діапазоні значень синтаксичного елемента. По-друге, інакше (наступним зображенням в порядку декодування є зображення RAP), значення `rap_pic_id`
 20 поточного зображення змінюється, щоб бути значенням, яке відрізняється від `rap_pic_id` і попереднього зображення і наступного зображення в порядку декодування, але все ще знаходиться в дозволеному діапазоні значень синтаксичного елемента.

Як альтернатива, коли змінюють зображення CRA на зображення BLA, декодер може виконувати нижченаведене, якщо присутні повідомлення SEI синхронізації зображень і час
 30 виведення з DPB для всіх зображень в DPB менше, ніж час виведення з DPB поточного зображення, значення `po_output_of_prior_pics_flag` встановлюється в 1; інакше значення `po_output_of_prior_pics_flag` встановлюється в "0".

У деяких прикладах `HandleCraAsBlaFlag` може бути першим прапором і `po_output_of_prior_pictures_flag` може бути другим прапором. У деяких прикладах
 35 `po_output_of_prior_pictures_flag` може бути змінною контексту.

З вищезгаданими змінами до HEVC WD7 існує можливість додатково видаляти особливі процеси декодування для зображень CRA, яке є першим зображенням в бітовому потоці, і асоційованих зображень TFD. У цьому випадку, коли бітовий потік починається із зображення CRA, перше зображення CRA в бітовому потоці повинне бути розглянуте як зображення BLA
 40 шляхом встановлення значення `HandleCraAsBlaFlag` в "1" для зображення CRA, що починає бітовий потік, незалежно від того, чи здійснене точне визначення значення за допомогою зовнішнього засобу, якщо є, і застосування вищезгаданого зміненого процесу декодування.

Як альтернатива, коли декодують (в тому числі аналізують) одиницю NAL кожного кодованого слайса, якщо поточне зображення є першим зображенням в бітовому потоці і
 45 `nal_unit_type` вказує зображення CRA (наприклад, значення рівне "4" або "5" згідно з HEVC WD7), може застосовуватися нижченаведене, значення `nal_unit_type` змінюється для вказівки зображення BLA (наприклад, значення збільшується на 2 згідно з HEVC WD5). У даному прикладі, немає необхідності змінювати значення `po_output_of_prior_pics_flag` та `rap_pic_id`. Як альтернатива, значення `HandleCraAsBlaFlag` може бути вказане за допомогою синтаксичного елемента в бітовому потоці, наприклад, нового синтаксичного елемента, який може бути
 50 включений в заголовок слайса або нове повідомлення SEI.

Один приклад належить до адаптації потокової передачі на основі зображень CRA. У такому прикладі замість того щоб покладатися на зміну сервером або проміжним мережним елементом зображення BLA на зображення CRA, сервер або проміжний мережний елемент може
 55 генерувати повідомлення, яке повинне бути відправлене на сторону декодера (тобто клієнта). Повідомлення може повідомити декодер, наприклад, що відбулася операція перемикання бітового потоку в деякому зображенні CRA і що зображення CRA повинне бути розглянуте як зображення BLA. У контексті динамічної адаптивної потокової передачі по HTTP (DASH), сторона декодера може також сама виводити таке повідомлення за допомогою зміни

уніфікованого покажчика ресурсу (URL), що використовується нею для запиту даних потоку, і прийому даних мультимедіа, асоційованих із зміненим URL.

В іншому прикладі зображення CRA може бути змінене таким чином, що якщо присутні повідомлення SEI синхронізації зображень і час виведення з DPB для всіх зображень в DPB менше, ніж час виведення з DPB поточного зображення. Значення `no_output_of_prior_pics_flag` може бути встановлене в 1. Інакше, значення `no_output_of_prior_pics_flag` може бути встановлене в "0".

Фіг. 1 є блок-схемою, що ілюструє зразкову систему 10 кодування і декодування відео, яка може використовувати прийоми, описані в даному розкритті. Як показано на Фіг. 1, система 10 включає в себе пристрій-джерело 12, який генерує закодовані відеодані для їх декодування згодом пристроєм-адресатом 14. Прийоми, описані в цьому документі, загалом належать до обробки зображення CRA як зображення BLA на основі зовнішньої вказівки. Відповідно, ці прийоми можуть загалом застосовуватися до пристрою-адресата 14, який може загалом приймати зовнішню вказівку і у відповідь на таку зовнішню вказівку може обробляти зображення CRA, прийняте пристроєм-адресатом, як зображення BLA в ході його обробки пристроєм-адресатом. Однак, в деяких прикладах пристрій-джерело 12 або інший мережний пристрій, такий як MANE, може забезпечити зовнішню вказівку пристрою-адресату 14, яка спонукає пристрій-адресат 14 обробляти зображення CRA, прийняте пристроєм-адресатом, як зображення BLA.

Пристрій-джерело 12 і пристрій-адресат 14 можуть являти собою будь-який з широкого діапазону пристроїв, що включають в себе настільні комп'ютери, блокнотні (тобто, переносні) комп'ютери, планшетні комп'ютери, телевізійні приставки, мікротелефонні трубки, такі як так звані "інтелектуальні" телефони, так звані "інтелектуальні" планшети, телевізійні приймачі, камери, пристрої відображення, цифрові мультимедійні програвачі, консолі для відеоігор, пристрої потокової передачі відео або подібне. У деяких випадках пристрій-джерело 12 і пристрій-адресат 14 можуть бути обладнані для бездротового зв'язку.

Пристрій-адресат 14 може приймати закодовані відеодані, які повинні бути декодовані за допомогою лінії 16 зв'язку. Лінія 16 зв'язку може являти собою будь-який тип носія або пристрою здатного переміщувати закодовані відеодані з пристрою-джерела 12 в пристрій-адресат 14. В одному прикладі лінія 16 зв'язку може являти собою середовище зв'язку для забезпечення можливості пристрою-джерела 12 передавати закодовані відеодані безпосередньо до пристрою-адресата 14 в реальному часі. Модулятор може модулювати закодовані відеодані згідно зі стандартом зв'язку, таким як протокол бездротового зв'язку, і передавати пристрою-адресату 14. Середовище зв'язку може являти собою будь-яке бездротове або дротове середовище зв'язку, таке як радіочастотний (RF) спектр або одна або більше фізичних ліній передачі. Середовище зв'язку може формувати частину пакетної мережі, такої як локальна мережа, широкомасштабна мережа або глобальна мережа, така як Інтернет. Середовище зв'язку може включати в себе маршрутизатори, комутатори, базові станції або інше обладнання, яке може бути придатним для сприяння зв'язку від пристрою-джерела 12 до пристрою-адресата 14.

Як альтернатива, закодовані дані можуть бути виведені від інтерфейсу 22 виведення до запам'ятовуючого пристрою 32. Аналогічним чином, вхідний інтерфейс може здійснити доступ до закодованих даних із запам'ятовуючого пристрою 32. Запам'ятовуючий пристрій 36 може включати в себе будь-який з різноманітних поширюваних або локально доступних носіїв даних, таких як жорсткий диск, Blu-Ray диски, DVD, CD-ROM, флеш-пам'ять, енергозалежна або енергонезалежна пам'ять, або будь-які інші придатні цифрові носії даних для зберігання закодованих відеоданих. У додатковому прикладі запам'ятовуючий пристрій 36 може відповідати файловому серверу або іншому проміжному запам'ятовуючому пристрою, який може зберігати закодоване відео, згенероване пристроєм-джерелом 12. Пристрій-адресат 14 може здійснювати доступ до відеоданих, що зберігаються, із запам'ятовуючого пристрою 36 шляхом потокової передачі або завантаження. Файловим сервером може бути будь-який тип сервера, здатний зберігати закодовані відеодані і передавати ці закодовані відеодані пристрою-адресату. Зразкові файлові сервери включають в себе веб-сервер (наприклад, для веб-сайта), FTP сервер, з'єднані з мережею запам'ятовуючі пристрої (NAS) або локальний дисковий накопичувач. Пристрій-адресат 14 може здійснювати доступ до закодованих відеоданих через будь-яке стандартне з'єднання для передачі даних, яке включає в себе Інтернет-з'єднання. Це може включати в себе бездротовий канал (наприклад, Wi-Fi з'єднання), дротове з'єднання (наприклад, DSL, кабельний модем) або їх комбінацію, яка є придатною для здійснення доступу до закодованих відеоданих, що зберігаються на файловому сервері. Передачею закодованих

відеоданих від запам'ятовуючого пристрою 36 може бути потокова передача, передача завантаження або їх комбінацією.

Прийоми даного розкриття необов'язково обмежені бездротовими додатками або установками. Прийоми можуть бути застосовані до кодування відео в підтримку будь-якого з різноманіття мультимедійних додатків, таких як мовні телепередачі по радіоінтерфейсу, кабельні телепередачі, супутникові телепередачі, потокові передачі відео, наприклад, через Інтернет, кодування цифрового відео для зберігання на носії даних, декодування цифрового відео, що зберігається на носії даних, або інші додатки. У деяких прикладах система 10 може бути сконфігурована для підтримки односторонньої або двосторонньої передачі відео для підтримки додатків, таких як потокова передача відео, відтворення відео, мовлення відео і/або відеотелефонія.

У прикладі на Фіг. 1 пристрій-джерело 12 включає в себе джерело 18 відео, відеокодер 20 та інтерфейс 22 виведення. У деяких випадках інтерфейс 22 виведення може включати в себе модулятор/демодулятор (модем) і/або передавач. У пристрої-джерелі 12 джерело 18 відео може включати в себе джерело таке як пристрій відеозахоплення. Наприклад, відеокамера, відеоархів, що містить раніше захоплене відео, інтерфейс подачі відео для прийому відео від постачальника відеоконтенту і/або систему комп'ютерної графіки для генерування даних комп'ютерної графіки як вихідне відео або комбінацію таких джерел. Як один приклад, якщо джерелом 18 відео є відеокамера, пристрій-джерело 12 і пристрій-адресат 14 можуть формувати так звані телефони з камерами або відео-телефони. Однак, прийоми, описані в цьому розкритті, можуть бути застосовані до кодування відео загалом і можуть бути застосовані до бездротових і/або дротових додатків.

Відеокодер 20 може кодувати захоплюване, раніше захоплене або згенероване комп'ютером відео. Закодовані відеодані можуть бути передані безпосередньо пристрою-адресату 14 через інтерфейс 22 виведення пристрою-джерела 12. Як альтернатива, закодовані відеодані можуть бути збережені на запам'ятовуючий пристрій 36 для подальшого доступу пристроєм-адресатом 14 або іншими пристроями для декодування і/або відтворення. В інших прикладах може бути виконане і те і інше, вказане вище за текстом.

Пристрій-адресат 14 включає в себе інтерфейс 28 введення, відеодекодер 30 і пристрій 32 відображення. У деяких випадках інтерфейс 28 введення може включати в себе приймач і/або модем. Інтерфейс 28 введення пристрою-адресата 14 приймає закодовані відеодані по лінії 16 зв'язку. Закодовані відеодані, що передаються по лінії 16 зв'язку або забезпечені на запам'ятовуючому пристрої 36, можуть включати в себе різноманіття синтаксичних елементів, згенерованих за допомогою відеокодера 20 для використання відеодекодером, таким як відеодекодер 30, при декодуванні відеоданих. Такі синтаксичні елементи можуть бути включені в закодовані відеодані, які передаються по середовищу зв'язку, що зберігаються на носії даних або зберігаються на файловому сервері.

В одному прикладі відеодекодер 30 або інший пристрій можуть приймати зовнішню вказівку. Відеодекодер 30 може потім обробляти зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі зовнішньої вказівки. У деяких прикладах зовнішня вказівка вказує, чи повинен прапор бути встановлений у відеодекодері. Відповідно, відеодекодер 30 може встановлювати прапор на основі зовнішньої вказівки. Відеодекодер 30 або деякі внутрішні функціональні засоби, такі як блок 72 обробки зовнішньої вказівки або модуль 81 прогнозування, можуть потім перевірити прапор. Як приклад модуль 81 прогнозування може обробляти зображення CRA як зображення BLA на основі зовнішньої вказівки, яка вказує, що зображення CRA повинне бути оброблене як зображення BLA на основі прапора.

В іншому прикладі відеодекодер 30 або інший пристрій може приймати зовнішню вказівку, що прапор повинен бути встановлений. Відеодекодер 30 або інший пристрій може потім встановити прапор на основі зовнішньої вказівки. Декодер 30 може потім перевірити прапор. Коли прапор встановлений, відеодекодер 30 обробляє зображення CRA як зображення BLA.

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 інтегрований пристрій відображення і також може бути сконфігурований для взаємодії із зовнішнім пристроєм відображення. В інших прикладах пристрій-адресат 14 може бути пристроєм відображення. Загалом пристрій 32 відображення відображає декодовані відеодані користувачу і може містити будь-який з різноманіття пристроїв відображення, таких як рідкокристалічний пристрій відображення (LCD), плазмовий пристрій відображення, пристрій відображення на органічних світлодіодах (OLED) або інший тип пристрою відображення.

Відеокодер 20 і відеодекодер 30 можуть функціонувати згідно зі стандартом стиснення відео, таким як стандарт Високоєфективного Відеокодування (HEVC), що розробляється в цей час, і можуть відповідати Тестовій Моделі HEVC (HM). Останній проект HEVC, від 27 червня 2012, доступний за посиланням http://wg11.sc29.org/jct/doc_end_user/current_document.php?id=5885/JCTVC-I1003-v5, повний вміст якого включений в цей документ за допомогою посилання. Як альтернатива, відеокодер 20 і відеодекодер 30 можуть функціонувати згідно з іншими приватними або промисловими стандартами, такими як стандарт ITU-T H.264, що як альтернатива згадується як MPEG-4, Частина 10, Вдосконалене Відеокодування (AVC) або розширення таких стандартів. Прийоми даного розкриття, однак, не обмежені яким-небудь конкретним стандартом кодування. Інші приклади стандартів стиснення відео включають в себе MPEG-2 та ITU-T H.263, а також відкриті формати, такі як VP8.

Незважаючи на те, що не показано на Фіг. 1, в деяких аспектах кожний з відеокодера 20 і відеодекодера 30 може бути об'єднаний з аудіо-кодером і -декодером, і може включати в себе належні блоки MUX-DEMUX, або інше апаратне і програмне забезпечення, щоб обробляти кодування як аудіо, так і відео, в загальному потоці даних або в окремих потоках даних. Якщо застосовно, в деяких прикладах блоки MUX-DEMUX можуть відповідати протоколу мультимплектора ITU H.223 або іншим протоколам, таким як протокол користувацьких дейтаграм (UDP).

Кожний з відеокодера 20 і відеодекодера 30 може бути реалізований як будь-яка з різноманітних придатних схем кодера, таких як один або більше мікропроцесорів, цифрових сигнальних процесорів (DSP), спеціалізованих інтегральних схем (ASIC), програмованих користувачем вентильних матриць (FPGA), дискретної логіки, програмного забезпечення, апаратного забезпечення, програмно-апаратного забезпечення або будь-які їх комбінації. При частковій реалізації прийомів у програмному забезпеченні, пристрій може зберігати інструкції для програмного забезпечення на придатному нечасовому зчитуваному комп'ютером носії і виконувати інструкції в апаратному забезпеченні, використовуючи один або більше процесорів, щоб виконувати прийоми даного розкриття. Кожний з відеокодера 20 і відеодекодера 30 може бути включений в один або більше кодерів або декодерів, будь-який з яких може бути інтегрований як частина комбінованого кодера/декодера (КОДЕКА) у відповідному пристрої.

JCT-VC працює над розробкою стандарту HEVC. Зусилля стандартизації HEVC основані на моделі пристрою відеокодування, що розвивається, яка згадується як Тестова Модель HEVC (HM). HM передбачає декілька додаткових можливостей пристроїв відеокодування відносно існуючих пристроїв згідно з, наприклад, ITU-T H.264/AVC. Наприклад, в той час як H.264 забезпечує дев'ять режимів кодування з внутрішнім прогнозуванням, HM може забезпечувати аж до тридцяти трьох режимів кодування з внутрішнім прогнозуванням.

Загалом робоча модель HM описує, що відеокадр або зображення можуть бути розділені на послідовність блоків дерева кодування або деревоподібних блоків або найбільших одиниць кодування (LCU), які включають в себе як вибірки (відліки) яскравості, так і вибірки кольоровості. Деревоподібний блок може мати призначення аналогічне макроблоку зі стандарту H.264. Слайс включає в себе деяке число послідовних деревоподібних блоків у порядку кодування. Відеокадр або зображення можуть бути розділені на один або більше слайсів. Кожний деревоподібний блок може бути розбитий на одиниці кодування (CU) згідно з квадродревом. Наприклад, деревоподібний блок, як кореневий вузол квадродреву, може бути розбитий на чотири дочірніх вузли, і кожний дочірній вузол може, в свою чергу, бути батьківським вузлом і бути розбитий на інші чотири дочірніх вузли. Кінцевий дочірній вузол, що не розбивається, як кінцевий вузол квадродреву, містить вузол кодування, тобто, кодований відеоблок. Синтаксичні дані, асоційовані з кодованим бітовим потоком, можуть задавати максимальну кількість разів, коли деревоподібний блок може бути розбитий, і також можуть задавати мінімальний розмір вузлів кодування.

CU включає в себе вузол кодування і одиниці прогнозування (PU) і одиниці перетворення (TU), асоційовані з вузлом кодування. Розмір CU відповідає розміру вузла кодування і повинен бути квадратним за формою. Розмір CU може знаходитися в діапазоні від 8×8 пікселів аж до розміру деревоподібного блока з максимумом в 64×64 пікселів або більше. Кожна CU може містити одну або більше PU і одну або більше TU. Синтаксичні дані, асоційовані з CU, можуть описувати, наприклад, розділення CU на одну або більше PU. Режими розділення можуть розрізняватися тим часом, чи закодована CU в прямому режимі або режимі пропускання, закодована в режимі внутрішнього прогнозування або закодована в режимі зовнішнього прогнозування. PU можуть бути розділені, щоб бути не квадратними за формою. Синтаксичні

дані, асоційовані з CU, також можуть описувати, наприклад, розділення CU на одну або більше TU згідно з квадродревом. TU може бути квадратною або неквадратною за формою.

Стандарт HEVC робить можливим перетворення згідно з TU, які можуть розрізнятися для різних CU. TU типово встановлені за розміром на основі розміру PU в межах даної CU, заданої для розділеної LCU, хоча це може не завжди мати місце. TU звичайно має той самий розмір, що PU, або менше з. У деяких прикладах залишкові вибірки, що відповідають CU, можуть бути поділені на менші одиниці з використанням структури квадродреву, відомої як "залишкове квадродреву" (RQT). Кінцеві вузли RQT можуть згадуватися як одиниці перетворення (TU). Значення піксельних різниць, асоційовані з TU, можуть бути перетворені для створення коефіцієнтів перетворення, які можуть бути квантовані.

Загалом, PU включає в себе дані, зв'язані з процесом прогнозування. Наприклад, коли PU закодована у внутрішньому режимі, PU може включати в себе дані, що описують режим внутрішнього прогнозування для PU. Як інший приклад, коли PU закодована у зовнішньому режимі, PU може включати в себе дані, які задають вектор руху для PU. Дані, які задають вектор руху для PU, можуть описувати, наприклад, горизонтальну компоненту вектора руху, вертикальну компоненту вектора руху, розрізнення для вектора руху (наприклад, точність в одну четверту пікселя або точність в одну восьму пікселя), опорне зображення, на яке вказує вектор руху і/або список опорних зображень (наприклад, Список 0, Список 1 або Список C) для вектора руху.

Загалом, TU використовується для процесів перетворення і квантування. Дана CU, що має одну або більше PU, також може включати в себе одну або більше одиниць перетворення (TU). Услід за прогнозуванням, відеокодер 20 може обчислювати залишкові значення, що відповідають PU. Залишкові значення містять значення піксельних різниць, які можуть бути перетворені в коефіцієнти перетворення, квантовані і проскановані з використанням TU для створення впорядкованих коефіцієнтів перетворення для ентропійного кодування. Дане розкриття звичайно використовує термін "відеоблок" для посилання на вузол кодування з CU. У деяких особливих випадках дане розкриття також може використовувати термін "відеоблок" для посилання на деревоподібний блок, тобто, LCU, або CU, яка включає в себе вузол кодування, а також PU та TU.

Відеопослідовність звичайно включає в себе ряд відеокadrів або зображень. Група зображень (GOP) звичайно містить ряд з одного або більше зображень відео. GOP може включати в себе синтаксичні дані в заголовку GOP, заголовку одного або більше зображень, або в іншому місці, яке описує деяке число зображень, включених в GOP. Кожний слайс зображення може включати в себе синтаксичні дані слайса, які описують режим кодування для відповідного слайса. Відеокодер 20 звичайно здійснює роботу над відеоблоками, в межах індивідуальних слайсів відео, для того, щоб закодувати відеодані. Відеоблок може відповідати вузлу кодування в CU. Відеоблоки можуть мати незмінні розміри або розміри, що міняються, і можуть відрізнятися за розміром згідно з точно визначеним стандартом кодування.

Як приклад НМ підтримує прогнозування в різних розмірах PU. Передбачаючи, що розмір деякої конкретної CU є $2N \times 2N$, НМ підтримує внутрішнє прогнозування в розмірах PU $2N \times 2N$ або $N \times N$ і зовнішнє прогнозування в симетричних розмірах PU $2N \times 2N$, $2N \times N$, $N \times 2N$ або $N \times N$. НМ також підтримує асиметричне розділення для зовнішнього прогнозування в розмірах PU $2N \times nU$, $2N \times nD$, $nL \times 2N$ та $nR \times 2N$. При асиметричному розділенні один напрямок CU не розділяється, в той час як інший напрямок розділяється на 25 % та 75 %. Частина CU, що відповідає 25 %-ій частині, вказується за допомогою "n", за чим йде вказівка "Up" (вгору), "Down" (вниз), "Left" (ліворуч) або "Right" (праворуч). Таким чином, наприклад, " $2N \times nU$ " належить до CU $2N \times 2N$, яка розділена горизонтально з PU $2N \times 0,5N$ зверху і PU $2N \times 1,5N$ знизу.

У даному розкритті " $N \times N$ " та " N на N " може використовуватися взаємозамінно для позначення розмірності пікселів відеоблока виходячи з вертикальної і горизонтальної розмірності, наприклад, 16×16 пікселів або 16 на 16 пікселів. Звичайно блок 16×16 буде мати 16 пікселів у вертикальному напрямку ($y=16$) і 16 пікселів у горизонтальному напрямку ($x=16$). Подібно, блок $N \times N$ загалом має N пікселів у вертикальному напрямку і N пікселів у горизонтальному напрямку, де N являє собою ціле значення, що не є негативним. Пікселі в блоці можуть бути впорядковані в рядках і стовпцях. Більше того, необов'язково, щоб блоки мали таку саму кількість пікселів у горизонтальному напрямку, як і у вертикальному напрямку. Наприклад, блоки можуть містити $N \times M$ пікселів, де M не обов'язково дорівнює N .

Нижченаведене кодування з внутрішнім прогнозуванням або зовнішнім прогнозуванням використовує PU з CU, відеокодер 20 може обчислювати залишкові дані для TU з CU. PU може містити дані пікселів у просторовій ділянці (яка також називається ділянкою пікселів), а TU може містити коефіцієнти в ділянці перетворення з подальшим застосуванням перетворення,

наприклад, дискретного косинусного перетворення (DCT), цілочисельного перетворення, вейвлет-перетворення, або концептуально подібного перетворення до залишкових відеоданих. Залишкові дані можуть відповідати піксельним різницям між пікселями незакодованого зображення і значеннями прогнозування, що відповідають PU. Відеокодер 20 може формувати

5 TU, що включають в себе залишкові дані для CU, а потім перетворювати TU для створення коефіцієнтів перетворення для CU.

Услід за будь-якими перетвореннями для створення коефіцієнтів перетворення відеокодер 20 може виконувати квантування коефіцієнтів перетворення. Квантування в цілому належить до процесу, в якому коефіцієнти перетворення квантуються для можливого скорочення обсягу

10 даних, що використовуються для представлення коефіцієнтів, що забезпечують додаткове стиснення. Процес квантування може скорочувати бітову глибину, асоційовану з деякими або усіма коефіцієнтами. Наприклад, n -бітове значення може округлятися в менший бік до m -бітового значення в ході квантування, де n більше m .

У деяких прикладах відеокодер 20 може застосовувати заздалегідь заданий порядок

15 сканування для сканування квантованих коефіцієнтів перетворення для створення впорядкованого вектора, який може бути ентропійно закодований. В інших прикладах відеокодер 20 може виконувати адаптивне сканування. Після сканування квантованих коефіцієнтів перетворення для формування одновимірний вектора відеокодер 20 може ентропійно кодувати одновимірний вектор, наприклад, згідно з контекстно-адаптивним

20 кодуванням із змінною довжиною кодового слова (CAVLC), контекстно-адаптивним двійковим арифметичним кодуванням (CABAC), синтаксичним контекстно-адаптивним двійковим арифметичним кодуванням (SBAC), ентропійним кодуванням на основі розділення інтервалу імовірності (PIPE) або іншою методологією ентропійного кодування. Відеокодер 20 також може ентропійно кодувати синтаксичні елементи, асоційовані із закодованими відеоданими для

25 використання за допомогою відеодекодера 30 при декодуванні відеоданих.

Для виконання CABAC відеокодер 20 може призначати контекст в межах контекстної моделі, символу, який повинен бути переданий. Контекст може належати до, наприклад, того, чи є сусідні значення символу ненульовими чи ні. Для виконання CAVLC відеокодер 20 може

30 вибирати код із змінною довжиною для символу, який повинен бути переданий. Кодові слова в VLC можуть бути побудовані так, що відносно більш короткі коди відповідають більш імовірним символам, в той час як більш довгі коди відповідають менш імовірним символам. Таким чином, використання VLC може досягнути економії бітів у порівнянні з, наприклад, використанням кодових слів рівної довжини для кожного символу, який повинен бути переданий. Визначення імовірності може бути ґрунтоване на контексті, призначеному символу.

Відповідно до даного розкриття пристрій-джерело 12 (або можливо інший проміжний пристрій, не показаний на Фіг. 1) може забезпечити зовнішню вказівку 34 пристрою-адресату 14, яка спонукає пристрій-адресат 14 обробляти зображення CRA, прийняте пристроєм-адресатом, як зображення BLA. Наприклад, пристрій-джерело 12 (або можливо інший проміжний пристрій, не показаний на Фіг. 1) може визначити, що зміна була зроблена користувачем, наприклад,

40 запит відео іншого розрізнення або якості, або може виникнути розірване посилання. Коли відбувається зміна розрізнення або виникає розірване посилання, наприклад, це може означати, що зображення CRA повинне бути оброблене як зображення BLA, оскільки інформація будь-якого попереднього зображення, що зберігається на приймальному пристрої, може не бути допустимою для декодування вхідного бітового потоку.

ФІГ. 2 є блок-схемою, що ілюструє приклад відеодекодера 20, який може реалізовувати прийоми, описані в даному розкритті. Як розглядалося вище за текстом, прийоми, описані в цьому документі, в цілому належать до обробки зображення CRA як зображення BLA на основі зовнішньої вказівки, прийнятої на пристрої-адресаті 14. Однак, в деяких прикладах пристрій-джерело 12 або інший мережний пристрій, такий як MANE, може забезпечити зовнішню вказівку

50 пристрою-адресату 14, яка спонукає пристрій-адресат 14 обробляти зображення CRA, прийняте на пристрої-адресаті, як зображення BLA.

Відеокодер 20 може виконувати внутрішнє і зовнішнє кодування відеоблоків у межах слайсів відео. Внутрішнє кодування покладається на просторове прогнозування для скорочення або видалення просторової надмірності відео в межах даного відеокадру або зображення. Зовнішнє

55 кодування покладається на часове прогнозування, щоб скорочувати або видаляти часову надмірність у відео в межах суміжних кадрів або зображень відеопослідовності. Внутрішній режим (I режим) може належати до будь-якого з декількох режимів просторового стиснення. Зовнішні режими, такі як однонаправлене прогнозування (P режим) або біпрогнозування (B режим), можуть належати до будь-якого з декількох режимів часового стиснення.

У прикладі на Фіг. 2 відеокодер 20 включає в себе модуль 35 розділення, модуль 41 прогнозування, модуль 63 фільтра, пам'ять 64 опорних зображень, суматор 50, модуль 52 перетворення, модуль 54 квантування і модуль 56 ентропійного кодування. Модуль 41 прогнозування включає в себе модуль 42 оцінки руху, модуль 44 компенсації руху і модуль 46 внутрішнього прогнозування. Для відновлення блока відео відеокодер 20 також включає в себе модуль 58 оберненого квантування, модуль 60 оберненого перетворення і суматор 62. Модуль 63 фільтра призначений для представлення одного або більше контурних фільтрів, таких як фільтр видалення блочності, адаптивний контурний фільтр (ALF) і фільтр адаптивного до вибірки зміщення (SAO). Хоча модуль 63 фільтра показаний на Фіг. 2 як такий, що є внутрішньоконтурним фільтром, в інших конфігураціях, модуль 63 фільтра може бути реалізований як пост-контурний фільтр.

Пристрій-джерело 12 або інший мережний пристрій, такий як MANE, може забезпечити зовнішню вказівку 34 пристрою-адресату 14, яка спонукає пристрій-адресат 14 обробляти зображення CRA, прийняте на пристрої-адресаті, як зображення BLA. Наприклад, зовнішня вказівка 34, яка, звичайно є зовнішньою по відношенню до пристрою-адресата 14 і, звичайно не передається як частина бітового потоку, може бути згенероване за допомогою модуля 41 прогнозування, який може мати доступ до вказівок, зв'язаних зі статусом бітового потоку. Це є тільки одним прикладом, однак, інші одиниці або модулі в пристрої-джерелі 12 або в інших пристроях, зовнішніх по відношенню до пристрою-джерела 12, можуть також генерувати зовнішню вказівку.

Як показано на Фіг. 2, відеокодер 20 приймає відеодані і модуль 35 розділення розділяє дані на відеоблоки. Це розділення також може включати в себе розділення на слайси, фрагменти або інші більші одиниці, а також розділення відеоблока, наприклад, згідно зі структурою квадродревця LCU та CU. Відеокодер 20 звичайно ілюструє компоненти, які кодують відеоблоки в межах слайса відео, який повинен бути закодований. Слайс може бути розділений на множинні відеоблоки (і можливо на набори відеоблоків, що згадуються як фрагменти). Модуль 41 прогнозування може вибирати один з множини можливих режимів кодування, таких як один з множини режимів внутрішнього кодування або один з множини режимів зовнішнього кодування, для поточного відеоблока на основі результатів помилки (наприклад, швидкості кодування і рівня спотворення). Модуль 41 прогнозування може забезпечувати одержаний в результаті внутрішньо або зовні кодований блок суматора 50 для генерування даних залишкового блока і суматора 62 для відновлення закодованого блока для використання як опорного зображення.

Модуль 46 внутрішнього прогнозування, в межах модуля 41 прогнозування, може виконувати кодування з внутрішнім прогнозуванням поточного відеоблока відносно одного або більше сусідніх блоків в одному і тому самому кадрі або слайсі як поточний блок, який повинен бути кодований для забезпечення просторового стиснення. Модуль 42 оцінки руху і модуль 44 компенсації руху, в межах модуля 41 прогнозування, виконують кодування із зовнішнім прогнозуванням поточного відеоблока відносно одного або більше прогнозованих блоків, в одному або більше опорних зображеннях для забезпечення часового стиснення.

Модуль 42 оцінки руху може бути сконфігурований для визначення режиму зовнішнього прогнозування для слайса відео згідно із зумовленим шаблоном для відеопослідовності. Зумовлений шаблон може позначати слайси відео в послідовності як Р слайси, В слайси або GPB слайси. Модуль 42 оцінки руху і модуль 44 компенсації руху можуть бути високо інтегровані, але проілюстровані окремо в концептуальних цілях. Оцінка руху, що виконується за допомогою модуля 42 оцінки руху, є процесом генерування векторів руху, які оцінюють рух для відеоблоків. Вектор руху, наприклад, може вказувати зміщення PU відеоблока в межах поточного відеокдру або зображення відносно предиктивного блока в межах опорного зображення.

Предиктивний блок є блоком, який знайдений як близько відповідний PU відеоблока, який повинен бути кодований, з точки зору піксельної різниці, яка може бути визначена за допомогою суми абсолютних різниць (SAD), суми квадратів різниці (SSD) або інших різницевих метрик. У деяких прикладах відеокодер 20 може обчислювати значення для позицій суб-цілих пікселів опорних зображень, збережених в пам'яті 64 опорних зображень. Наприклад, відеокодер 20 може інтерполювати значення позицій однієї четвертої пікселя, позицій однієї восьмої пікселя, або інших позицій дробового пікселя опорного зображення. Тому модуль 42 оцінки руху може виконувати пошук руху відносно позицій повних пікселів і позицій дробових пікселів і виведення вектора руху з точністю дробового пікселя.

Модуль 42 оцінки руху обчислює вектор руху для PU відеоблока у зовні кодованому слайсі за допомогою порівняння позиції PU з позицією предиктивного блока опорного зображення. Опорне зображення може бути вибране з першого списку опорних зображень (Список 0) або

другого списку опорних зображень (Список 1), кожний з яких ідентифікує одне або більше опорних зображень, збережених в пам'яті 64 опорних зображень. Модуль 42 оцінки руху відправляє обчислений вектор руху модулю 56 ентропійного кодування і модулю 44 компенсації руху.

5 Компенсація руху, що виконується за допомогою модуля 44 компенсації руху, може залучати до себе вибірку або генерування предиктивного блока на основі вектора руху, що визначається шляхом оцінки руху, можливо виконуючи інтерполяції до субпіксельної точності. Після прийому вектора руху для PU поточного відеоблока модуль 44 компенсації руху може розташовувати предиктивний блок, на який вказує вектор руху, в одному зі списків опорних зображень.

10 Відеокодер 20 формує залишковий відеоблок шляхом віднімання піксельних значень предиктивного блока з піксельних значень поточного кодованого відеоблока, формуючи значення піксельних різниць. Значення піксельних різниць формують залишкові дані для блока і можуть включати в себе компоненти різниць як яскравості, так і кольоровості. Суматор 50 являє собою компонент або компоненти, які виконують дану операцію віднімання. Модуль 44 компенсації руху також може генерувати синтаксичні елементи, асоційовані з відеоблоками і відеослайсом, для використання за допомогою відеодекодера 30 при декодуванні відеоблоків слайса відео.

Модуль 46 внутрішнього прогнозування може внутрішньо прогнозувати поточний блок як альтернативу зовнішньому прогнозуванню, що виконується за допомогою модуля 42 оцінки руху і модуля 44 компенсації руху, як описано вище за текстом. Зокрема, модуль 46 внутрішнього прогнозування може визначати режим внутрішнього прогнозування для використання для кодування поточного блока. У деяких прикладах модуль 46 внутрішнього прогнозування може кодувати поточний блок, використовуючи різні режими внутрішнього прогнозування, наприклад, під час окремих проходів кодування, і модуль 46 внутрішнього прогнозування (або модуль 40 вибору режиму, в деяких прикладах) може вибирати належний режим внутрішнього прогнозування для використання з режимів, що тестуються. Наприклад, модуль 46 внутрішнього прогнозування може обчислювати значення "швидкість передачі - спотворення", використовуючи аналіз "швидкість передачі - спотворення" для різних режимів внутрішнього прогнозування, що тестуються, і вибирати режим внутрішнього прогнозування, що має кращі характеристики "швидкість передачі - спотворення" серед режимів, що тестуються. Аналіз "швидкість передачі - спотворення" звичайно визначає обсяг спотворення (або помилку) між закодованим блоком і вихідним, незакодованим блоком, який був закодований для створення закодованого блока, а також частоту проходження бітів (тобто, кількість бітів), що використовується для створення закодованого блока. Модуль 46 внутрішнього прогнозування може обчислювати співвідношення за спотвореннями і швидкостями передачі для різних закодованих блоків для визначення, який режим внутрішнього прогнозування показує краще значення "швидкість передачі - спотворення" для блока.

У будь-якому випадку, після вибору режиму внутрішнього прогнозування для блока модуль 46 внутрішнього прогнозування може забезпечувати інформацію, яка вказує вибраний режим внутрішнього прогнозування для блока модулю 56 ентропійного кодування. Модуль 56 ентропійного кодування може кодувати інформацію, яка вказує вибраний режим внутрішнього прогнозування відповідно до прийомів даного розкриття. Відеокодер 20 може включати в себе дані конфігурації в бітовому потоці, що передається. Дані конфігурації можуть включати в себе множину таблиць індексів режиму внутрішнього прогнозування і множину модифікованих таблиць індексів режиму внутрішнього прогнозування (що також згадуються як таблиці відображення кодових слів), визначень контекстів кодування для різних блоків і вказівки найбільш імовірного режиму внутрішнього прогнозування, таблиць індексів режиму внутрішнього прогнозування і модифікованих таблиць індексів режиму внутрішнього прогнозування для використання для кожного з контекстів.

50 Після того як модуль 41 прогнозування генерує предиктивний блок для поточного відеоблока за допомогою або зовнішнього прогнозування, або внутрішнього прогнозування, відеокодер 20 формує залишковий відеоблок за допомогою віднімання предиктивного блока з поточного відеоблока. Залишкові відеодані в залишковому блоці можуть бути включені в один або більше TU і застосовуватися для модуля 52 перетворення. Модуль 52 перетворення перетворює залишкові відеодані в залишкові коефіцієнти перетворення, використовуючи перетворення, таке як дискретне косинусне перетворення (DCT) або концептуально аналогічне перетворення. Модуль 52 перетворення може конвертувати залишкові відеодані з піксельної ділянки в ділянку перетворення, таку як частотна ділянка.

Модуль 52 перетворення може відправляти одержані в результаті коефіцієнти перетворення модулю 54 квантування. Модуль 54 квантування квантує коефіцієнти перетворення, щоб

додатково зменшувати частоту проходження бітів. Процес квантування може зменшувати бітову глибину, асоційовану з деякими або всіма коефіцієнтами. Ступінь квантування може бути модифікований шляхом регулювання параметра квантування. У деяких прикладах модуль 54 квантування може потім виконувати сканування матриці, що включає в себе квантовані коефіцієнти перетворення. Як альтернатива, модуль 56 ентропійного кодування може виконувати сканування.

Услід за квантуванням модуль 56 ентропійного кодування ентропійно кодує квантовані коефіцієнти перетворення. Наприклад, модуль 56 ентропійного кодування може виконувати контекстно-адаптивне кодування із змінною довжиною кодового слова (CAVLC), контекстно-адаптивне двійкове арифметичне кодування (CABAC), синтаксичне контекстно-адаптивне двійкове арифметичне кодування (SBAC), ентропійне кодування на основі розділення інтервалу імовірності (PIPE) або іншу методологію або прийом ентропійного кодування. Услід за ентропійним кодуванням за допомогою модуля 56 ентропійного кодування, закодований бітовий потік може бути переданий відеокодеру 30 або вміщений в архів для подальшої передачі або витягання відеодекодером 30. Модуль 56 ентропійного кодування може також ентропійно закодувати вектори руху та інші синтаксичні елементи для поточного кодованого слайса відео.

Модуль 58 оберненого квантування і модуль 60 оберненого перетворення застосовують обернене квантування та обернене перетворення, відповідно, для відновлення залишкового блока в піксельній ділянці для подальшого використання як опорного блока опорного зображення. Модуль 44 компенсації руху може обчислювати опорний блок шляхом додавання залишкового блока до предиктивного блока одного з опорних зображень в межах одного зі списків опорних зображень. Модуль 44 компенсації руху також може застосовувати один або більше інтерполяційних фільтрів до відновленого залишкового блока для обчислення субцілочисельних піксельних значень для використання в оцінці руху. Суматор 62 додає відновлений залишковий блок до предиктивного блока з компенсованим рухом, створеного модулем 44 компенсації руху, для створення опорного блока для збереження в пам'яті 64 опорних зображень. Опорний блок може бути використаний модулем 42 оцінки руху і модулем 44 компенсації руху як опорний блок для зовнішнього прогнозування блока в подальшому відеокадрі або зображенні.

Відеокодер 20 на Фіг. 2 являє собою приклад відеокодера, який може бути сконфігурований для сигналізування того, що зображення чистого довільного доступу (CRA) повинне бути оброблене як зображення доступу з розірваним посиланням (BRA), як описано в цьому документі.

Фіг. 3 є блок-схемою, що ілюструє зразковий відеодекодер 30, який може реалізовувати прийоми даного розкриття, які, звичайно належить до обробки зображення CRA як зображення BLA на основі зовнішньої вказівки 70, яка може бути згенерована за допомогою мережного об'єкта 29, такого як MANE або деякого іншого зовнішнього пристрою (не показаного). В одному прикладі відеодекодер 30 приймає зовнішню вказівку 70 про те, що прапор 74 повинен бути встановлений. Зовнішня вказівка 70 приймається відеодекодером 30. В інших прикладах зовнішня вказівка 70 може бути прийнята та оброблена зовні відносно відеокодера 30. Блок 72 обробки зовнішньої вказівки встановлює прапор 74 на основі зовнішньої вказівки. Прапор потім передається в модуль 81 прогнозування. У проілюстрованому прикладі блок 72 обробки зовнішньої вказівки знаходиться в межах відеокодера 30. В інших прикладах блок 72 обробки зовнішньої вказівки може бути зовнішнім відносно і окремим від відеокодера 30. У відеокодері 30 модуль (81) прогнозування перевіряє прапор і, коли прапор встановлений, обробляє одне зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA).

У деяких прикладах значенням за замовчуванням прапора є "0" і встановленим значенням прапора є "1". В інших прикладах істиною може виявитися обернене, значенням за замовчуванням прапора є "1" і встановленим значенням прапора є "0". Іншими словами, прапор може бути активним високим ("1") або активним низьким ("0").

У деяких прикладах, коли декодують одиницю рівня абстракції мережі (NAL) кодованого слайса, якщо встановлений перший прапор, модуль 81 прогнозування може змінювати тип одиниці NAL одиниці NAL. Коли декодують одиницю рівня абстракції мережі (NAL) кодованого слайса, якщо встановлений перший прапор, модуль 81 прогнозування може змінювати значення другого прапора. Другим прапором може бути `no_output_of_prior_pics_flag`. Додатково, коли декодують одиницю рівня абстракції мережі (NAL) кодованого слайса, якщо встановлений прапор, модуль прогнозування може встановлювати значення другого прапора в "1".

У прикладі, коли поточним зображенням є зображення CRA і коли деяка зовнішня вказівка є доступною для встановлення змінної, яка вказує, що зображення CRA повинне бути розглянуте

як зображення BLA (наприклад, `HandleCraAsBlaFlag`), потім змінна (наприклад, `HandleCraAsBlaFlag`) може бути встановлена в значення, що забезпечується зовнішнім засобом. Інакше, значення змінної (наприклад, `HandleCraAsBlaFlag`) може бути встановлене для вказівки, що зображення CRA не повинне бути розглянуте як зображення BRA. Наприклад, `HandleCraAsBlaFlag` може бути встановлено в "1" для вказівки, що зображення CRA повинне бути розглянуте як зображення BRA і встановлене в "0" для вказівки, що зображення CRA не повинне бути розглянуте як зображення BRA.

Потрібно зазначити, що, в той час як деякі приклади зовнішньої вказівки можуть бути описані в цьому документі, вони не призначені щоб бути вичерпними. Багато можливих зовнішніх вказівок могли б бути використані.

У деяких прикладах, коли поточним зображенням є зображення CRA і змінна, яка вказує, що зображення CRA повинне бути розглянуте як зображення BLA (наприклад, `HandleCraAsBlaFlag`), є рівною "1", де "1" вказує, що зображення CRA повинне бути розглянуте як зображення BLA, значення `no_output_of_prior_pics_flag` може бути встановлене в "1" і нижченаведене застосовується під час процесів аналізу і декодування для кожної одиниці NAL кожного сегмента кодового слайса.

Як приклад `no_output_of_prior_pics_flag` точно визначає, як раніше декодовані зображення в буфері декодованих зображень обробляються після декодування зображення BLA або IDR. У прикладі, коли зображення BLA або IDR є першим зображенням в бітовому потоці, значення `no_output_of_prior_pics_flag` не впливає на процес декодування. Коли зображення BLA або IDR не є зображенням в бітовому потоці і значення `pic_width_in_luma_samples` або `pic_height_in_luma_samples` або `sps_max_dec_pic_buffering [sps_max_temporal_layers_minus1]`, витягнуте з активного набору параметрів послідовності, є відмінним від значення `pic_width_in_luma_samples` або `pic_height_in_luma_samples` або `sps_max_dec_pic_buffering [sps_max_temporal_layers_minus1]`, витягнутого з набору параметрів послідовності активного для попереднього зображення, `no_output_of_prior_pics_flag`, рівне "1", може (але не повинне) бути виведене декодером незалежно від фактичного значення `no_output_of_prior_pics_flag`.

У прикладі на Фіг. 3 відеодекодер 30 включає в себе модуль 80 ентропійного декодування, модуль 81 прогнозування, модуль 86 оберненого квантування, модуль 88 оберненого перетворення, суматор 90, модуль 91 фільтра і пам'ять 92 опорних зображень. Модуль 81 прогнозування включає в себе модуль 82 компенсації руху і модуль 84 внутрішнього прогнозування. Відеодекодер 30 відео може, в деяких прикладах, виконувати прохід декодування, в цілому обернений проходу кодування, описаному відносно відеокодера 20 з Фіг. 2.

У ході процесу декодування відеодекодер 30 приймає закодований бітовий потік відео, який являє собою відеоблоки закодованого відеослайса і зв'язані синтаксичні елементи з відеокодера 20. Відеодекодер 30 може приймати закодований бітовий потік відео від мережного об'єкта 29. Мережним об'єктом 29 може, наприклад, бути сервер, MANE, редактор/засіб з'єднання відео або інший такий пристрій, сконфігурований для реалізації одного або більше прийомів, описаних вище за текстом. Як описано вище за текстом, деякі з прийомів, описаних в даному розкритті, можуть бути реалізовані за допомогою мережного об'єкта 29 до моменту передачі мережним об'єктом 29 закодованого бітового потоку відео до відеокодера 30. У деяких системах декодування відео, мережний об'єкт 29 і відеодекодер 30 можуть бути частинами окремих пристроїв, в той час як в інших прикладах функціональність, описана відносно мережного об'єкта 29, може бути виконана за допомогою одного і того самого пристрою, який містить відеодекодер 30.

Як розглянуто вище, мережний пристрій, такий як мережний об'єкт 29, яким може бути MANE, може забезпечити зовнішню вказівку 34 пристрою-адресату 14, яка спонукає пристрій-адресата 14 обробляти зображення CRA, прийняте на пристрої-адресаті, як зображення BLA. Наприклад, зовнішня вказівка 34, яка, звичайно є зовнішньою по відношенню до пристрою-адресата 14 і, звичайно не передається як частина бітового потоку, може бути згенерована за допомогою модуля 41 прогнозування, який може мати доступ до вказівок, зв'язаних зі статусом бітового потоку. Це є тільки одним прикладом, однак, інші блоки або модулі в пристрої-джерелі 12 або в інших пристроях, зовнішніх по відношенню до пристрою-джерела 12, можуть також генерувати зовнішню вказівку.

Модуль 80 ентропійного декодування відеокодера 30 ентропійно декодує бітовий потік для генерування квантованих коефіцієнтів, векторів руху та інших синтаксичних елементів. Модуль 80 ентропійного декодування переадресовує вектори руху та інші синтаксичні елементи модулю 81 прогнозування. Відеодекодер 30 може приймати синтаксичні елементи на рівні відеослайсів і/або на рівні відеоблоків.

Коли відеослайс кодований як внутрішньо кодований (I) слайс, модуль 84 внутрішнього прогнозування модуля 81 прогнозування може генерувати дані прогнозування для відеоблока поточного слайса відео на основі сигналізованого режиму внутрішнього прогнозування і даних з раніше декодованих блоків поточного кадру або зображення. Коли відеокادر кодований як зовнішній (тобто, B, P або GPB) слайс, модуль 82 компенсації руху модуля 81 прогнозування створює предиктивні блоки для відеоблока поточного слайса відео на основі векторів руху та інших синтаксичних елементів, прийнятих від модуля 80 ентропійного декодування. Предиктивні блоки можуть бути створені з одного з опорних зображень в межах одного зі списків опорних зображень. Відеодекодер 30 може будувати списки опорних кадрів, Список 0 і Список 1, використовуючи стандартні прийоми побудови на основі опорних зображень, що зберігаються в пам'яті 92 опорних зображень.

Модуль 82 компенсації руху визначає предиктивну інформацію для відеоблока поточного відеослайса шляхом аналізу векторів руху та інших синтаксичних елементів, і використовує предиктивну інформацію для створення предиктивних блоків для поточного декодованого відеоблока. Наприклад, модуль 82 компенсації руху може використовувати деякі з прийнятих синтаксичних елементів для визначення режиму прогнозування (наприклад, внутрішнього або зовнішнього прогнозування), використаного для кодування відеоблоків слайса відео, типу слайса зовнішнього прогнозування (наприклад, слайс B, слайс P або слайс GPB). Додатково, модуль 82 компенсації руху може використовувати інформацію побудови для одного або більше списків опорних зображень слайса, векторів руху для кожного зовні закодованого відеоблока слайса, статусу зовнішнього прогнозування для кожного зовні кодованого відеоблока слайса та іншу інформацію для декодування відеоблоків у поточному відеослайсі.

Модуль 82 компенсації руху також може виконувати інтерполяцію на основі інтерполяційних фільтрів. Модуль 82 компенсації руху може використовувати інтерполяційні фільтри як такі, що використовуються за допомогою відеокодера 20 під час кодування відеоблоків, для обчислення інтерпольованих значень для субцілочисельних пікселів опорних блоків. У такому випадку, модуль 82 компенсації руху може визначати інтерполяційні фільтри, використані відеокодером 20, з прийнятих синтаксичних елементів і використовувати інтерполяційні фільтри для створення предиктивних блоків.

Модуль 86 оберненого квантування обернено квантує, тобто деквантує, квантовані коефіцієнти перетворення, забезпечені в бітовому потоці і декодовані модулем 80 ентропійного декодування. Процес оберненого квантування може включати в себе використання параметра квантування, що обчислюється відеокодером 20 для кожного відеоблока у відеослайсі, для визначення ступеня квантування і, точно так само, ступеня оберненого квантування, який повинен бути застосований. Модуль 88 оберненого перетворення застосовує обернене перетворення, наприклад, обернене DCT, обернене цілочисельне перетворення або концептуально схожий процес оберненого перетворення, до коефіцієнтів перетворення для створення залишкових блоків у піксельній ділянці.

Після того як модуль 82 компенсації руху генерує предиктивний блок для поточного відеоблока на основі векторів руху та інших синтаксичних елементів, відеодекодер 30 формує декодований відеоблок шляхом підсумовування залишкових блоків модуля 88 оберненого перетворення з відповідними предиктивними блоками, згенерованими модулем 82 компенсації руху. Суматор 90 являє собою компонент або компоненти, які виконують дану операцію підсумовування. При необхідності контурні фільтри (або в контурі кодування, або після контуру кодування) також можуть бути використані для згладжування піксельних переходів або, в іншому випадку, поліпшення якості відео. Модуль 91 фільтра призначений для представлення одного або більше контурних фільтрів, таких як деблокуючий фільтр, адаптивний контурний фільтр (ALF) і фільтр адаптивного до вибірки зміщення (SAO). Хоча модуль 91 фільтра показаний на Фіг. 3 як внутрішньоконтурний фільтр, в інших конфігураціях, модуль 91 фільтра може бути реалізований як пост-контурний фільтр. Декодовані відеоблоки в даному кадрі або зображенні потім зберігаються в пам'яті 92 опорних зображень, яка зберігає опорні зображення, що використовуються для подальшої компенсації руху. Пам'ять 92 опорних зображень також зберігає декодоване відео для подальшого представлення на пристрої відображення, такому як пристрій 32 відображення на Фіг. 1.

Відеодекодер 30 на Фіг. 3 являє собою приклад відеокодера, сконфігурованого для обробки одного зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиленням (BRA), як описано в цьому документі.

Фіг. 4 є блок-схемою, що ілюструє зразковий набір пристроїв, які формують частину мережі 100. У цьому прикладі мережа 100 включає в себе пристрої 104A, 104B маршрутизації (пристрої 104 маршрутизації) і пристрій 106 транскодування. Пристрої 104 маршрутизації і пристрій 106

транскодування призначені для представлення невеликої кількості пристроїв, які можуть формувати частину мережі 100. Інші мережні пристрої, такі як комутатори, концентратори, шлюзи, міжмережні екрани, мости та інші подібні пристрої також можуть бути включені в мережу 100. Більше того, можуть бути забезпечені додаткові мережні пристрої в мережному тракті між пристроєм 102 сервера і клієнтським пристроєм 108. Пристрій 102 сервера може відповідати пристрою-джерелу 12 (Фіг. 1), в той час як клієнтський пристрій 108 може відповідати пристрою-адресату 14 (Фіг. 1), в деяких прикладах. Відповідно, пристрій 102 сервера звичайно не приймає зовнішню вказівку для обробки зображення CRA як зображення BLA. Однак, сервер 102 може забезпечити зовнішню вказівку 34 до клієнтського пристрою 108, яка спонукає клієнтський пристрій 108 обробляти зображення CRA, прийняте на пристрої-адресаті, як зображення BLA. Аналогічним чином, однак, пристрої 104A, 104B маршрутизації (пристрої 104 маршрутизації) і пристрій 106 транскодування звичайно не приймають зовнішню вказівку обробляти зображення CRA як зображення BLA, але можуть забезпечити зовнішню вказівку 34 клієнтському пристрою 108, щоб клієнтський пристрій 108 обробляв зображення CRA, прийняте в пристрої-адресаті, як зображення BLA. Деякі приклади, описані в цьому документі, можуть включати в себе одне або більше з нижченаведеного: мережні пристрої, сервери MANE, кеші протоколу передачі гіпертексту ("HTTP") або веб-проксі.

У деяких прикладах клієнтський пристрій 108 може встановлювати прапор після прийому повідомлення про те, що відбулася зміна в частоті проходження бітів бітового потоку. Відповідно, клієнтський пристрій може встановлювати прапор на основі зміни частоти проходження бітів. У деяких прикладах декодер в клієнтському пристрої 108 може декодувати одиницю NAL кодованого слайса. Модуль прогнозування в декодері в клієнтському пристрої 108 може аналізувати одиницю NAL кожного кодованого слайса для ідентифікації типу одиниці NAL. Додатково, модуль прогнозування може декодувати одиницю NAL кодованого слайса на основі типу одиниці NAL.

Звичайно пристрої 104 маршрутизації реалізують один або більше протоколів маршрутизації для обміну мережними даними по мережі 100. У деяких прикладах пристрої 104 маршрутизації можуть бути сконфігуровані для виконання операцій кешування або посередництва. Тому, в деяких прикладах пристрої 104 маршрутизації можуть згадуватися як пристрої-посередники. Звичайно пристрої 104 маршрутизації виконують протоколи маршрутизації для виявлення маршрутів через мережу 100. Шляхом виконання таких протоколів маршрутизації, пристрій 104B маршрутизації може виявляти мережний маршрут від себе до пристрою 102 сервера через пристрій 104A маршрутизації.

Прийоми даного розкриття можуть бути реалізовані мережними пристроями, такими як пристрої 104 маршрутизації і пристрій 106 транскодування, але також можуть бути реалізовані клієнтським пристроєм 108. Таким чином, пристрої 104 маршрутизації, пристрій 106 транскодування і клієнтський пристрій 108 являють собою приклади пристроїв, сконфігурованих для виконання прийомів даного розкриття, в тому числі прийомів, викладених в розділі ФОРМУЛА ВИНАХОДУ даного розкриття. Більше того, пристрої на Фіг. 1 і кодер, показаний на Фіг. 2 і декодер, показаний на Фіг. 3, також є зразковими пристроями, які можуть бути сконфігуровані для виконання прийомів даного розкриття, в тому числі прийомів, викладених в розділі ФОРМУЛА ВИНАХОДУ даного розкриття.

Фіг. 5 є блок-схемою послідовності операцій, що ілюструє зразковий спосіб відповідно до одного або більше прикладів, описаних в даному розкритті. Як приклад відеодекодер 30 або інший пристрій приймає зовнішню вказівку (500). Потім відеодекодер 30 обробляє зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі зовнішньої вказівки (502). У деяких прикладах зовнішня вказівка вказує, чи повинен прапор бути встановлений у відеодекодер. Відповідно, відеодекодер може встановлювати прапор на основі зовнішньої вказівки, що буде обговорюватися більш детально відносно Фіг. 6. Декодер або деякий внутрішній функціональний засіб, такий як блок обробки зовнішньої вказівки або модуль прогнозування, може потім перевірити прапор. В одному прикладі модуль прогнозування може обробляти зображення CRA як зображення BLA на основі зовнішньої вказівки, яка вказує, що зображення CRA повинне бути розглянуте як зображення BLA на основі прапора.

Фіг. 6 є блок-схемою послідовності операцій, що ілюструє інший зразковий спосіб відповідно до одного або більше прикладів, описаних в даному розкритті. У прикладі, що ілюструється, відеодекодер 30 приймає зовнішню вказівку 70 про те, що прапор 74 повинен бути встановлений (600). У прикладі, що ілюструється, зовнішня вказівка 70 приймається відеодекодером 30. В інших прикладах зовнішня вказівка 70 може бути прийнята та оброблена зовні відносно відеодекодера 30.

Блок 72 обробки зовнішньої вказівки встановлює перший прапор 74 на основі зовнішньої вказівки (602). Перший прапор потім передається модулю 81 прогнозування. У проілюстрованому прикладі блок 72 обробки зовнішньої вказівки знаходиться в межах відеодекодера 30. В інших прикладах блок 72 обробки зовнішньої вказівки може бути зовнішнім відносно і окремим від відеодекодера 30.

У декодері 30 модуль 81 прогнозування перевіряє прапор і, коли прапор встановлений, обробляє зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиленням (BLA) (604). У прикладі, коли декодують одиницю рівня абстракції мережі (NAL) кодованого слайса, якщо перший прапор рівний "1" і тип одиниці NAL одиниці NAL кодованого слайса вказує зображення CRA (наприклад, значення рівне "4" або "5" згідно з HEVC WD7) блок 72 обробки зовнішньої вказівки або інший блок в межах відеодекодера 30 змінює значення типу одиниці NAL для вказівки зображення BLA (наприклад збільшує значення типу одиниці NAL на 2 згідно з HEVC WD7). Додатково, модуль 81 прогнозування встановлює значення другого прапора в 1. Якщо попереднім зображенням в порядку декодування є зображення RAP і `gar_pic_id` поточного слайса рівний `gar_pic_id` попереднього зображення, застосовується нижченаведене. Якщо наступним зображенням в порядку декодування не є зображення RAP, змінюється значення `gar_pic_id` поточного слайса, щоб стати відмінним від `gar_pic_id` попереднього зображення в порядку декодування. Інакше, значення `gar_pic_id` поточного зображення змінюється з тим, щоб стати значенням, яке відрізняється від `gar_pic_id` як попереднього зображення, так і наступного зображення в порядку декодування.

В іншому прикладі у відеодекодері 30, модуль 81 прогнозування або інший блок у межах відеодекодера 30 декодує одиницю NAL кодованого слайса. Якщо перший прапор рівний "1" і тип одиниці NAL одиниці NAL кодованого вказує зображення CRA (наприклад, значення рівне "4" або "5" згідно з HEVC WD7), модуль прогнозування (або інший блок) змінює значення типу одиниці NAL для вказівки зображення BLA (наприклад, збільшує значення типу одиниці NAL на 2 згідно з HEVC WD7). Можливо додатково, якщо є повідомлення SEI синхронізації зображень і час виведення з DPB для всіх зображень в DPB менше, ніж час виведення з DPB поточного зображення, модуль 81 прогнозування або інший блок встановлює значення другого прапора в 1. Інакше, якщо значення другого прапора встановлене в "0" і якщо попереднім зображенням в порядку декодування є зображення RAP і `gar_pic_id` поточного слайса рівний `gar_pic_id` попереднього зображення, застосовується нижченаведене. Якщо наступним зображенням в порядку декодування не є зображення RAP, модуль 81 прогнозування або інший блок змінює значення `gar_pic_id` поточного слайса, щоб воно стало відмінним від `gar_pic_id` попереднього зображення в порядку декодування. Інакше, значення `gar_pic_id` модуля прогнозування або іншого блока змінює поточне зображення, щоб воно стало значенням, яке відрізняється від `gar_pic_id` як попереднього зображення, так і наступного зображення в порядку декодування.

Фіг. 7 є блок-схемою послідовності операцій, що ілюструє зразковий спосіб відповідно до одного або більше прикладів, описаних в даному розкритті. Пристрій, такий як мережний пристрій, наприклад MANE, приймає бітовий потік, що включає в себе зображення CRA (700). Мережний пристрій визначає, що зображення CRA повинне бути оброблене як зображення BLA (702). Наприклад, мережний пристрій може визначати, що зображення CRA повинне бути оброблене як зображення BLA для забезпечення можливості виведення і/або відображення більшої кількості зображень, коли відкидаються всі декодовані зображення, що виводяться раніше в порядку декодування, ніж зображення BLA або IDR, після декодування зображення BLA або IDR без виведення і/або відображення. Іноді відображення більшої кількості таких зображень може забезпечити краще сприйняття користувачем. Відповідно, мережний пристрій передає зображення CRA і зовнішню вказівку про те, що зображення CRA повинне бути перетворене в зображення BLA (704).

Фіг. 8 є блок-схемою послідовності операцій, що ілюструє зразкову роботу з відправки першим пристроєм зовнішньої вказівки і дії у відповідь другого пристрою з прийому зовнішньої вказівки. Пристрій-джерело, такий як мережний пристрій, наприклад MANE, приймає бітовий потік, що включає в себе зображення CRA (800). Мережний пристрій визначає, що зображення CRA повинне бути оброблене як зображення BLA (802). Відповідно, мережний пристрій передає зображення CRA і зовнішню вказівку про те, що зображення CRA повинне бути перетворене в зображення BLA (804). Відеодекодер 30 приймає зображення CRA і зовнішню вказівку 70 про те, що зображення CRA повинне бути перетворене в зображення BLA (806).

Блок 72 обробки зовнішньої вказівки встановлює прапор 74 на основі зовнішньої вказівки (808). Прапор потім може бути переданий модулю 81 прогнозування. В одному прикладі блок 72 обробки зовнішньої вказівки знаходиться в межах відеодекодера 30. В інших прикладах блок 72 обробки зовнішньої вказівки може бути зовнішнім відносно і окремим від відеодекодера 30. У

декодері 30 модуль 81 прогнозування перевіряє прапор і, коли прапор встановлений, обробляє зображення CRA як зображення BLA (810).

У прикладі на Фіг. 8 прапор використовується для вказівки, що зовнішня вказівка була прийнята. В інших прикладах, аналогічних Фіг. 5, відеодекодер 30 або інший пристрій приймає зовнішню вказівку і потім обробляє зображення CRA як зображення BLA на основі зовнішньої вказівки.

В одному прикладі декодер змінює зображення CRA на зображення BLA як функцію на стороні декодера. При прийомі або виведенні такого повідомлення одна функція сторони декодера може виконувати зміну ідентифікованого зображення CRA на зображення BLA бітового потоку до кодованого зображення до декодера для декодування.

Зображення CRA може бути змінено на зображення BLA. Для одиниці NAL кожного кодованого слайса, якщо `nal_unit_type` вказує зображення CRA, наприклад, значення рівне "4" або "5" згідно з HEVC WD7, застосовується наступне, (1) значення `nal_unit_type` змінюється для вказівки зображення BLA, наприклад, значення збільшується на 2, (2) значення `no_output_of_prior_pics_flag` встановлюється в 1, (3) якщо попереднім зображенням в порядку декодування є зображення RAP і `gar_pic_id` поточного слайса рівний `gar_pic_id` попереднього зображення, застосовується нижченаведене: (а) якщо наступним зображенням в порядку декодування не є зображення RAP, значення `gar_pic_id` поточного слайса змінюється, щоб стати відмінним від `gar_pic_id` попереднього зображення в порядку декодування, але все ще в дозволеному діапазоні значень синтаксичного елемента, або (b) інакше (наступним зображенням в порядку декодування є зображення RAP), значення `gar_pic_id` поточного зображення змінюється, щоб стати значенням, відмінним від `gar_pic_id` як для попереднього зображення, так і наступного зображення в порядку декодування, але все ще в дозволеному діапазоні значень синтаксичного елемента.

Нижче описаний розгляд зображення CRA як зображення CRA, яке починає бітовий потік. Вказівка, що деяке конкретне зображення CRA повинне бути розглянуте як зображення BLA, як описано вище, також може бути змінена на або інтерпретована як вказівка про те, що конкретне зображення CRA повинне бути розглянуте як зображення CRA, яке є першим зображенням в бітовому потоці, за умови, що зміни, описані нижче зроблені в проектній специфікації HEVC.

В одному прикладі змінна `CraIsFirstPicFlag` асоційована з кожним зображенням CRA. Значення `CraIsFirstPicFlag` для деяких зображень CRA може бути точно визначене зовнішніми засобами. Якщо зображення CRA є першим зображенням в бітовому потоці, тоді значення `CraIsFirstPicFlag` для зображення CRA встановлюється в 1, незалежно від значення, вказаного зовнішньою вказівкою (коли присутня). Інакше, коли значення `CraIsFirstPicFlag` для зображення CRA не визначене точно зовнішніми засобами, воно встановлюється в "0".

У ході декодування (в тому числі аналізу) одиниці NAL кожного кодованого слайса, якщо `CraIsFirstPicFlag` рівний "1" і `nal_unit_type` рівний "4" або "5", значення `no_output_of_prior_pics_flag` може бути встановлене в 1. Якщо попереднім зображенням в порядку декодування є зображення RAP і `gar_pic_id` поточного слайса рівний `gar_pic_id` попереднього зображення, тоді, якщо наступним зображенням в порядку декодування не є зображення RAP, значення `gar_pic_id` поточного слайса змінюється щоб бути відмінним від `gar_pic_id` попереднього зображення в порядку декодування, але все ще в дозволеному діапазоні значення синтаксичного елемента. Інакше (наступним зображенням в порядку декодування є зображення RAP), значення `gar_pic_id` поточного слайса змінюється, щоб бути значенням, яке відрізняється від `gar_pic_id` як попереднього зображення, так і наступного зображення в порядку декодування, але все ще знаходиться в дозволеному діапазоні значень синтаксичного елемента.

Як альтернатива, замість встановлення значення `no_output_of_prior_pics_flag` може бути встановлене в 1, модуль 81 прогнозування може, якщо є повідомлення SEI синхронізації зображень і час виведення з DPB для всіх зображень в DPB менше, ніж час виведення з DPB поточного зображення, значення `no_output_of_prior_pics_flag` встановлюється в 1, інакше значення `no_output_of_prior_pics_flag` встановлюється в "0".

В інших прикладах різні визначення рахунку порядку зображення, відмічені для відкидання (TFD) зображення, можуть бути змінені з HEVC WD9 або інших робочих проектів стандарту. Відповідно, визначення забезпечені нижче за текстом можуть відрізнятися від стандарту. Ці визначення можуть не застосовуватися до деяких або всіх прикладах, описаних в цьому документі.

У деяких прикладах кованою відеопослідовністю є послідовність одиниць доступу, яка може включати в себе, в порядку декодування, одиницю доступу CRA, яка може мати `CraIsFirstPicFlag` рівний 1, одиницю доступу IDR або одиницю доступу BLA, за якою йде нуль

або більше одиниць доступу не-BLA і не-IDR, що включають в себе всі подальші одиниці доступу аж до, але без включення в склад, будь-якої подальшої одиниці доступу BLA або IDR.

У деяких прикладах рахунок порядку зображення може бути змінною, яка може бути асоційована з кожним кодованим зображенням і має значення, яке збільшується із збільшенням позиції зображення в порядку виведення відносно одного з наступних кодованих зображень: (1) попереднє зображення IDR в порядку декодування, якщо таке є, (2) попереднє зображення BLA в порядку декодування, якщо таке є, і (3) попереднє зображення CRA в порядку декодування, якщо таке є і, в деяких прикладах, якщо попереднє зображення CRA має `CralsFirstPicFlag` рівний 1.

У деяких прикладах, якщо представлено більше одного з вищезгаданих кодованих зображень, рахунок порядку зображення належить до останнього з подібних кодованих зображень в порядку декодування. Відмічене для відкидання (TFD) зображення: кодоване зображення для якого кожний слайс має `nal_unit_type` рівний 2; зображення TFD асоційоване з попереднім зображенням CRA або зображенням BLA в порядку декодування і передусь асоційованому зображенню в порядку виведення; коли асоційованим зображенням є зображення BLA або коли асоційованим зображенням є зображення CRA, яке може мати `CralsFirstPicFlag` рівний 1, зображення TFD може не бути коректно декодованим і не виводитися.

У деяких прикладах семантика `no_output_of_prior_pics_flag` може бути змінена таким чином, що `no_output_of_prior_pics_flag` точно визначає те, як раніше декодовані зображення в буфері декодованих зображень обробляються після декодування зображення CRA з `CralsFirstPicFlag`, рівного 1 або зображення BLA або IDR.

У деяких прикладах, коли зображенням CRA з `CralsFirstPicFlag`, рівним 1, або зображенням BLA або IDR, є перше зображення в бітовому потоці, значення `no_output_of_prior_pics_flag` не впливає на процес декодування. Коли зображення CRA з `CralsFirstPicFlag`, рівним 1, або зображення BLA або IDR, не є першим зображенням в бітовому потоці і значення `pic_width_in_luma_samples` або `pic_height_in_luma_samples` або `sps_max_dec_pic_buffering[sps_max_temporal_layers_minus1]`, витягнуте з активного набору параметрів послідовності, відрізняється від значення `pic_width_in_luma_samples` або `pic_height_in_luma_samples` або `sps_max_dec_pic_buffering[sps_max_temporal_layers_minus1]`, витягнутого з набору параметрів послідовності активного для попереднього зображення, `no_output_of_prior_pics_flag`, рівний 1, може (але не повинен) бути виведений декодером, незалежно від фактичного значення `no_output_of_prior_pics_flag`.

У деяких прикладах зміна може бути зроблена по відношенню до нижченаведеного в підпункті 8.1 HEVC WD7, наприклад, зміна: якщо першим кодованим зображенням в бітовому потоці є зображення CRA і поточним зображенням є зображення TFD, асоційоване із зображенням CRA або якщо попереднім зображенням RAP, що передусь поточному зображенню в порядку декодування, є зображення BLA і поточним зображенням є зображення TFD, асоційоване із зображенням BLA, `PicOutputFlag` встановлюється рівним "0" і викликається процес декодування для генерування недоступних опорних зображень, точно визначений в підпункті 8.3.3 (тільки необхідно, щоб він був викликаний для одного слайса зображення), НА: якщо зображення CRA має `CralsFirstPicFlag`, рівний 1, і поточним зображенням є зображення TFD, асоційоване із зображенням CRA або якщо попереднім зображенням RAP, що передусь поточному зображенню в порядку декодування, є зображення BLA і поточним зображенням є зображення TFD, асоційоване із зображенням BLA, `PicOutputFlag` встановлюється рівним "0" і викликається процес декодування для генерування недоступних опорних зображень, точно визначений в підпункті 8.3.3 (тільки необхідно, щоб він був викликаний для одного слайса зображення).

У деяких прикладах зміна нижченаведеного в підпункті 8.3.1 HEVC WD7 може бути зроблена, наприклад, зміна: поточним зображенням є зображення CRA і є першим кодованим зображенням в бітовому потоці НА поточним зображенням є зображення CRA з `CralsFirstPicFlag`, рівний 1.

У деяких прикладах зміна нижченаведеного в підпункті 8.3.1 HEVC WD7 може бути зроблена, наприклад, зміна: якщо поточним зображенням є зображення BLA або IDR або, якщо першим кодованим зображенням в бітовому потоці є зображення CRA і поточне зображення є першим кодованим зображенням в бітовому потоці, `PicOrderCntMsb` встановлюється рівним "0". Інакше, `PicOrderCntMsb` витягується, як точно визначене нижченаведеним псевдо-кодом, НА, якщо поточним зображенням є зображення BLA або IDR або зображення CRA з `CralsFirstPicFlag` рівним 1, `PicOrderCntMsb` встановлюється рівним "0". Інакше, `PicOrderCntMsb` витягується, як точно визначене нижченаведеним псевдо-кодом.

У деяких прикладах зміна нижченаведеного в підпункті 8.3.2 HEVC WD7 може бути зроблена, наприклад, зміна ПРИМІТКА 4 - можуть бути одне або більше опорних зображень, які включені в набір опорних зображень, але не присутні в буфері декодованих зображень. Записи в RefPicSetStFoll або RefPicSetLtFoll, які рівні "немає опорного зображення" повинні бути проігноровані. Доки будь-яка з нижченаведених двох умов не є істиною, ненавмисна втрата зображення повинна бути виведена для кожного запису в RefPicSetStCurrBefore, RefPicSetStCurrAfter та RefPicSetLtCurr, яка рівна "немає опорного зображення": а) першим кодованим зображенням в бітовому потоці є зображення CRA і поточним кодованим зображенням є зображення TFD, асоційоване з першим кодованим зображенням в бітовому потоці; б) попереднім зображенням RAP, що передує поточному кодованому зображенню в порядку декодування, є зображення BLA і поточним кодованим зображенням є зображення TFD, асоційоване із зображенням BLA, НА ПРИМІТКУ 4 - Можуть бути одне або більше опорних зображень, які включені в набір опорних зображень, але не є в буфері декодованих зображень. Записи в RefPicSetStFoll або RefPicSetLtFoll, які рівні "немає опорного зображення" повинні бути проігноровані. Доки попереднім зображенням RAP, що передує поточному кодованому зображенню в порядку декодування, є зображення CRA з CralsFirstPicFlag, рівний "1" або зображення BLA і поточним кодованим зображенням є зображення TFD, асоційоване з попереднім зображенням RAP, ненавмисна втрата зображення повинна бути виведена для кожного запису в RefPicSetStCurrBefore, RefPicSetStCurrAfter та RefPicSetLtCurr, яка рівна "немає опорного зображення".

У деяких прикладах зміна нижченаведеного в підпункті 8.3.2. HEVC WD7 може бути зроблена, наприклад, зміна: Доки будь-яка з нижченаведених умов не є істинною, не повинно бути ніякого запису в RefPicSetStCurrBefore, RefPicSetStCurrAfter або RefPicSetLtCurr, яка рівна "немає опорного зображення": а) першим кодованим зображенням в бітовому потоці є зображення CRA і поточним кодованим зображенням є зображення TFD, асоційоване з першим кодованим зображенням в бітовому потоці; б) попереднім зображенням RAP, що передує поточному кодованому зображенню в порядку декодування, є зображення BLA і поточним кодованим зображенням є зображення TFD, асоційоване із зображенням BLA, НА доки попереднім зображенням RAP, що передує поточному кодованому зображенню в порядку декодування, не є зображення CRA з CralsFirstPicFlag, рівний "1" або зображення BLA і поточним кодованим зображенням не є зображення TFD, асоційоване з попереднім зображенням RAP, не повинно бути ніякого запису в RefPicSetStCurrBefore, RefPicSetStCurrAfter або RefPicSetLtCurr, яка рівна "немає опорного зображення".

У деяких прикладах зміна в перших трьох абзацах в підпункті 8.3.3.1 HEVC WD7 може бути зроблена таким чином: цей процес викликається один раз для кожного кодованого зображення, після виклику процесу декодування для набору опорних зображень, як точно визначено в підпункті 8.3.2., коли попереднім зображенням RAP, що передує поточному кодованому зображенню в порядку декодування, є зображення CRA з CralsFirstPicFlag, рівний "1", або зображення BLA і поточним кодованим зображенням є зображення TFD, асоційоване з попереднім зображенням RAP. ПРИМІТКА 1 - Повна специфікація, в даному документі, процесу декодування для зображень TFD, асоційованих із зображенням CRA на початку бітового потоку або для зображень TFD, асоційованих із зображенням BLA, включена тільки для цілей точного визначення обмежень на допустимий вміст синтаксису таких зображень. У фактичних декодерах, будь-які зображення TFD, асоційовані із зображенням CRA на початку бітового потоку або будь-які зображення TFD, асоційовані із зображенням BLA, можуть бути просто проігноровані (видалені з бітового потоку і відкинуті), оскільки вони не визначені точно для виведення і не впливають на процес декодування будь-яких інших зображень, які точно визначені для виведення. Коли попереднім зображенням RAP, що передує поточному кодованому зображенню в порядку декодування, є зображення CRA з CralsFirstPicFlag, рівний "1", або зображення BLA і поточним кодованим зображенням є зображення TFD, асоційоване з попереднім зображенням RAP, застосовується нижченаведене.

У деяких прикладах зміна нижченаведеного в підпункті C.4 HEVC WD7 може бути зроблена, наприклад, зміна: ПРИМІТКА 1 - Дане обмеження гарантує декодованість зображення TFD, якщо його асоційоване зображення RAP є зображенням CRA і якщо дане зображення CRA не є першим зображенням, що кодується в бітовому потоці НА ПРИМІТКУ 1 - Дане обмеження гарантує декодованість зображення TFD, якщо його асоційоване зображення RAP є зображенням CRA і якщо дане зображення CRA має CralsFirstPicFlag, рівний "0".

У деяких прикладах зміна в третій абзац в підпункті C.3.1 HEVC WD7 може бути зроблена таким чином: Якщо поточним зображенням є зображення CRA з CralsFirstPicFlag, рівний 1, або зображення BLA або IDR, застосовується нижченаведене, коли зображення CRA з

CralsFirstPicFlag, рівним 1, або зображення BLA або IDR, не є декодованим зображенням і значення `pic_width_in_luma_samples` або `pic_height_in_luma_samples` або `sps_max_dec_pic_buffering [i]` для будь-якого можливого значення `i`, витягнуте з активного набору параметрів послідовності, відрізняється від значення `pic_width_in_luma_samples` або `pic_height_in_luma_samples` або `sps_max_dec_pic_buffering [i]`, витягнутого з набору параметрів послідовності, який був активним для попереднього зображення, відповідно, `no_output_of_prior_pics_flag` виводиться щоб бути рівним "1" за допомогою HRD незалежно від фактичного значення `no_output_of_prior_pics_flag`. ПРИМІТКА 1 - Реалізації декодера повинні намагатися обробляти зображення або зміни розміру DPB більш тонко, ніж HRD відносно змін в `pic_width_in_luma_samples`, `pic_height_in_luma_samples` або `sps_max_dec_pic_buffering [i]`. Коли `no_output_of_prior_pics_flag` рівний "1" або виведений як рівний 1, всі буфери зберігання зображень в DPB спустошуються без виведення зображень, які вони містять, і наповненість DPB встановлюється в "0".

У деяких прикладах зміна всього підпункту C.5.2. HEVC WD7 може бути зроблена таким чином, видалення зображень з DPB до декодування поточного зображення (але після аналізу заголовка слайса першого слайса поточного зображення) відбувається миттєво, коли перша одиниця декодування одиниці доступу, що містить поточне зображення, видаляється з CPB і відбувається наступним чином. Викликається процес декодування для набору опорних зображень, як точно визначено в підпункті 8.3.2. Якщо поточним зображенням є зображення CRA з `CralsFirstPicFlag` рівним "1" або зображення BLA або IDR, застосовується нижченаведене. Коли зображення CRA з `CralsFirstPicFlag` рівним 1 або зображення BLA або IDR, не є першим декодованим зображенням і значення `pic_width_in_luma_samples` або `pic_height_in_luma_samples` або `sps_max_dec_pic_buffering [i]` для будь-якого можливого значення `i`, витягнуте з активного набору параметрів послідовності, відрізняється від значення `pic_width_in_luma_samples` або `pic_height_in_luma_samples` або `sps_max_dec_pic_buffering [i]`, витягнутого з набору параметрів послідовності, який був активним для попереднього зображення, відповідно, `no_output_of_prior_pics_flag` виводиться щоб бути рівним "1" за допомогою HRD незалежно від фактичного значення `no_output_of_prior_pics_flag`. ПРИМІТКА - Реалізації декодера повинні намагатися обробляти зображення або зміни розмірів DPB більш тонко, ніж HRD відносно змін в `pic_width_in_luma_samples`, `pic_height_in_luma_samples` або `sps_max_dec_pic_buffering [i]`. Коли `no_output_of_prior_pics_flag` рівний 1 або виводиться, щоб бути рівним 1, всі буфери зберігання зображень в DPB спустошуються без виведення зображень, які вони містять. Інакше, буфери зберігання зображень, що містять зображення, яке помічене як "не потрібно для виведення" або "таке, що не використовується для посилання", спустошуються (без виведення). Коли одна або більше з нижченаведених умов є істинною, процес "різкої зміни параметрів", точно визначений в підпункті C.5.2.1, неодноразово викликається до появи порожнього буфера зберігання зображень для збереження поточного декодованого зображення. Кількість зображень в DPB, які відмічені як "потрібні для виведення", є більшою ніж `sps_num_reorder_pics [temporal_id]`. Кількість зображень в DPB з `temporal_id`, менший або рівний `temporal_id` поточного зображення, рівне `sps_max_dec_pic_buffering [temporal_id]`.

У деяких прикладах, коли поточним зображенням є зображення CRA з `CralsFirstPicFlag`, рівний "1" або зображення BLA або IDR, для якого `no_output_of_prior_pics_flag` не рівний "1" і не виведений як рівний 1, виконуються нижченаведені два етапи. Буфери зберігання зображень, що містять зображення, яке відмічене як "не потрібне для виведення" або "таке, що не використовується для посилання", спустошуються (без виведення). Всі не спустошені буфери зберігання зображень в DPB спустошуються шляхом багаторазового виклику процесу "різкої зміни параметрів", точно визначеного в підпункті C.5.2.1.

Деякі приклади можуть включати в себе процес "різкої зміни параметрів". Процес "різкої зміни параметрів" може бути викликаний в нижченаведених випадках: (1) поточним зображенням є зображення CRA з `CralsFirstPicFlag`, рівний "1" або зображення BLA або IDR і `no_output_of_prior_pics_flag` не рівний "1" і не виведений рівним 1, як точно визначено в підпункті C.5.2, (2) кількість зображень в DPB, які помічені як "потрібні для виведення", більше, ніж `sps_num_reorder_pics [temporal_id]`, як точно визначено в підпункті C.5.2, і (3) кількість зображень в DPB з `temporal_id` меншим або рівним `temporal_id` поточного зображення рівна `sps_max_dec_pic_buffering [temporal_id]`, як точно визначена в підпункті C.5.2.

Процес "різкої зміни параметрів" може включати в себе нижченаведені впорядковані етапи: (1) зображення, яке є першим для виведення, вибирається як зображення, яке має найменше значення `PicOrderCntVal` з усіх зображень в DPB, помічених як "потрібні для виведення", (2) зображення кадрується, за допомогою використання кадруючого прямокутника, що точно

визначається в активному наборі параметрів послідовності для зображення, кадроване зображення виводиться і зображення позначається як "не потрібне для виведення", (3) якщо буфер зберігання зображень, який включає в себе зображення, яке було кадроване і виведення містить зображення, помічене як "таке, що не використовується для посилання", буфер зберігання зображень спустошується.

У деяких прикладах спільно з вищезгаданими змінами в проектній специфікації HEVC, може бути можливим додаткове видалення всіх текстових повідомлень для підтримки зображень BLA.

Зараз буде описане вдосконалене виведення зображень. Як приклад припускається, що `no_output_of_prior_pics_flag` змінюється на `output_all_prior_pics_flag`, даний прапор, рівний "1", має еквівалентне значення, коли `no_output_of_prior_pics_flag` рівний "0". Коли даний прапор рівний "0", крім того, кількість попередніх зображень, які можуть бути використані для виведення/відображення, сигналізується як `num_output_pics`. `num_output_pics` може бути сигналізоване як `u(v)`, даний синтаксичний елемент знаходиться в діапазоні від 0 до `MaxDpbSize` не включно. Зображення `num_output_pics`, які повинні бути виведені/відображені, є зображеннями, що мають порядок відображення близький до зображення IDR або BLA і в першому бітовому потоці. `num_output_pics` може бути пов'язане з кількістю ведучих зображень, яким не потрібно бути виведеними.

Як альтернатива, `num_output_pics` може бути сигналізоване як `ue(v)`. Як альтернатива, `no_output_of_prior_pics_flag`, `output_all_prior_pics_flag`, або `num_output_pics` не сигналізується і `num_prior_discard_pics` безпосередньо сигналізується як `u(v)` або `ue(v)` `num_prior_discard_pics` знаходиться в діапазоні від 0 до `MaxDpbSize` не включно. Це вказує кількість попередніх зображень, які повинні бути відкинуті. Зображення `num_prior_discard_pics`, які повинні бути відкинуті (таким чином, не відображені) є зображеннями, що мають порядок відображення більш віддалений відносно зображення IDR або BLA і в першому бітовому потоці.

Як альтернатива, повідомлення SEI може бути додане під час з'єднання для вказівки додаткової пам'яті виходячи з кількості кадрів у першому бітовому потоці, необхідної для відображення всіх зображень в першому бітовому потоці, які не були відображені.

Нижче описане сигналізування синхронізації зображень. Вказівка однієї або більше різної інформації синхронізації, наприклад, найбільш ранній час представлення (тобто, найбільш ранній час виведення з DPB) і найменше значення рахунку порядку зображення всіх зображень TFD, асоційованих з одним зображенням CRA або BLA, може бути включений в бітовий потік. Інформація може бути включена в одне або більше із заголовка слайса і повідомлення SEI (наприклад, повідомлення SEI точки відновлення або повідомлення SEI періоду буферизації або повідомлення SEI синхронізації зображень). Один або більше з нижченаведених синтаксичних елементів можуть бути включені в заголовок слайса зображення RAP або повідомлення SEI, асоційоване із зображенням RAP, для сигналізування інформації: (1) `delta_earliest_presentation_time`, що вказує різницю між часом виведення з DPB зображення RAP і найбільш раннім часом виведення з DPB будь-якого зображення, коли зображення RAP є першим зображенням в бітовому потоці (тобто, найбільш раннім часом виведення з DPB з всіх DLP, асоційованих із зображенням RAP) в одиницях тактів синхронізації, як точно визначено в Додатку 3 HEVC WD7. Синтаксичний елемент може бути кодованим `u(v)` і кількість бітів, що використовуються для представлення синтаксичного елемента, дорівнює `crpb_removal_delay_length_minus1+1` бітів. Значення "0" вказує, що зображення RAP не має асоційованих DLP, (2) `delta_earliest_pos`, що вказує різницю між значенням `PicOrderCntVal` зображення RAP і найменшим значенням `PicOrderCntVal` якого-небудь зображення, коли зображення RAP є першим зображенням в бітовому потоці (тобто, найменшим і найбільш раннім значенням `PicOrderCntVal` з усіх DLP, асоційованих із зображенням RAP. Синтаксичний елемент може бути кодованим `ue(v)` і діапазон значень може бути від 0 до `MaxPicOrderCntLsb/2-1` включно.

В одному або більше прикладах описані функції можуть бути реалізовані в апаратному забезпеченні, програмному забезпеченні, програмно-апаратному забезпеченні, або будь-якій їх комбінації. Якщо реалізовані в програмному забезпеченні, функції можуть бути збережені на або передані до, як одна або більше інструкцій або коду, зчитуваному комп'ютером носію і виконані за допомогою, оснований на апаратному забезпеченні, блока обробки. Зчитуваний комп'ютером носій може включати в себе зчитуваний комп'ютером носій даних, який відповідає речовинному носію, такому як носій зберігання даних, або носій зв'язку, що включає в себе будь-який носій, який сприяє перенесенню комп'ютерної програми з одного місця в інше, наприклад згідно з протоколом зв'язку. Таким чином, зчитуваний комп'ютером носій, звичайно може відповідати (1) речовинному зчитуваному комп'ютером носію даних, який є нечасовим, або (2) середовищу зв'язку, такому як сигнал або несуча. Носієм зберігання даних може бути

будь-який з доступних носіїв, який може бути доступний за допомогою одного або більше комп'ютерів або одного або більше процесорів для витягання інструкцій, коду і/або структур даних для реалізації прийомів, описаних в даному розкритті. Комп'ютерний програмний продукт може включати в себе зчитуваний комп'ютером носій.

5 У деяких прикладах будь-яке з повідомлення, і синтаксичного елемента, включає в себе одне з нижченаведеного: (1) `delta_earliest_presentation_time`, що вказує різницю у часі, асоційовану з одним або більше CRA, що обробляється як зображення BRA; або (2) `delta_earliest_pos`, що вказує різницю в значення порядку зображень, асоційованому з одним або більше CRA, що обробляється як зображення BRA.

10 У все ще інших прикладах це розкриття передбачає зчитуваний комп'ютером носій, що містить структуру даних, збережену на ньому, в якому структура даних включає в себе закодований бітовий потік відповідно до цього розкриття. Зокрема, структури даних можуть включати в себе проекти одиниці NAL, описані в цьому документі.

Як приклад зображення CRA може бути оброблене як зображення BRA. Відеодекодер 30
15 може змінювати значення типу рівня абстракції мережі (NAL), встановлювати значення, яке керує виведенням попередніх зображень і змінювати ідентифікаційне значення (ID) зображення, асоційоване з наступним зображенням. Відеодекодер 30 може приймати синтаксичний елемент для вказівки операції перемикавання. Синтаксичний елемент або повідомлення, яке є частиною стиснутого бітового потоку і операції перемикавання, інструктує декодеру обробляти одне або
20 більше зображень CRA як зображення BRA. Декодер може потім декодувати бітовий потік частково на основі синтаксичного елемента.

В одному прикладі відеокодер 20 може генерувати синтаксичний елемент або повідомлення для вказівки операції перемикавання. Операція перемикавання інструктує декодеру обробляти одне або більше зображень CRA як зображення BRA. Відеокодер 20 може відправити синтаксичний
25 елемент до пристрою декодування як частину стиснутого бітового потоку.

Шляхом прикладу, а не обмеження, такі зчитувані комп'ютером носії даних можуть містити RAM, ROM, EEPROM, CD-ROM або інший запам'ятовуючий пристрій на оптичному диску, запам'ятовуючий пристрій на магнітному диску або інші магнітні запам'ятовуючі пристрої, флеш-пам'ять або будь-який інший носій, який може бути використаний для зберігання бажаного
30 програмного коду в формі інструкцій або структур даних і який може бути доступний за допомогою комп'ютера. Також, будь-яке з'єднання належним чином іменується зчитуванням комп'ютером носієм. Наприклад, якщо інструкції передаються з веб-сайта, сервера або іншого віддаленого джерела з використанням коаксіального кабелю, оптоволоконного кабелю, витой пари, цифрової абонентської лінії зв'язку (DSL) або бездротових підходів, таких як
35 інфрачервоні, радіо і мікрохвильові, то коаксіальний кабель, оптоволоконний кабель, вита пара, DSL або бездротові технології, такі як інфрачервоні, радіо і мікрохвильові включаються у визначення носія. Потрібно розуміти, однак, що зчитувані комп'ютером носії даних і носії зберігання даних не включають в себе з'єднання, несучі хвилі, сигнали, або інші часові носії, але звичайно направлені на нечасові, часові носії зберігання. Диск (disk) і диск (disc), як
40 використовуються в даному документі, включає в себе компакт-диск (CD), лазерний диск, оптичний диск, цифровий універсальний диск (DVD), флоппі диск і Blu-ray диск, де диски звичайно відтворюють дані магнітно, тоді як диски відтворюють дані оптично за допомогою лазерів. Комбінації вищеприписаного також повинні бути включені в обсяг зчитуваних комп'ютером носіїв.

Інструкції можуть бути виконані одним або більше процесорами, такими як один або більше цифрових сигнальних процесорів (DSP), мікропроцесорів загального призначення, спеціалізованих інтегральних схем (ASIC), програмованих користувачем вентильних матриць (FPGA), або іншими еквівалентними інтегрованими або дискретними логічними схемами. Відповідно термін "процесор", як використовується в даному документі, може належати до будь-якої з вищенаведених структур або будь-якої іншої структури, придатної для реалізації
50 прийомів, описаних в даному документі. Додатково, в деяких аспектах, функціональність, описана в даному документі, може бути забезпечена в спеціалізованих модулях апаратного забезпечення і/або програмного забезпечення, сконфігурованих для кодування і декодування, або включена в об'єднаний кодек. Також, підходи можуть бути повністю реалізовані в одній або
55 більше схемах або логічних елементах.

Прийоми даного розкриття можуть бути реалізовані в широкому ряді пристроїв або приладів, в тому числі в бездротових телефонних трубках, інтегральних схемах (IC) або наборах IC (наприклад наборах інтегральних схем). Різні компоненти, модулі, або блоки описані в цьому розкритті, щоб підкреслити функціональні аспекти пристроїв, сконфігурованих для
60 виконання розкритих підходів, але необов'язково вимагають реалізації за допомогою різних

блоків апаратного забезпечення. Навпаки, як описано вище за текстом, різні блоки можуть бути об'єднані в блок апаратного забезпечення кодека або бути забезпечені сукупністю взаємодіючих блоків апаратного забезпечення, що включають в себе один або більше процесорів, як описано вище за текстом, разом з придатним програмним забезпеченням і/або програмно-апаратним забезпеченням.

Були описані різні приклади. Ці та інші приклади знаходяться в обсязі нижченаведеної формули винаходу.

Посилальні позиції

- 10 система кодування і декодування відео
- 12 пристрій-джерело
- 14 пристрій-адресат
- 16 лінія зв'язку
- 18 джерело відео
- 20 відеокодер
- 15 22 інтерфейс виведення
- 28 інтерфейс введення
- 29 мережний об'єкт
- 30 відеодекодер
- 32 пристрій відображення
- 20 36 запам'ятовуючий пристрій
- 34 зовнішня вказівка
- 35 модуль розділення
- 41 модуль прогнозування
- 42 модуль оцінки руху
- 25 44 модуль компенсації руху
- 46 модуль внутрішнього прогнозування
- 50 суматор
- 52 модуль перетворення
- 54 модуль квантування
- 30 56 модуль ентропійного кодування
- 58 модуль оберненого квантування
- 60 модуль оберненого перетворення
- 62 суматор
- 63 модуль фільтра
- 35 64 пам'ять опорних зображень
- 70 зовнішня вказівка
- 72 блок обробки зовнішньої вказівки
- 74 прапор
- 80 модуль ентропійного декодування
- 40 81 модуль прогнозування
- 82 модуль компенсації руху
- 84 модуль внутрішнього прогнозування
- 86 модуль оберненого квантування
- 88 модуль оберненого перетворення
- 45 90 суматор
- 91 модуль фільтра
- 92 пам'ять опорних зображень
- 100 мережа
- 102 пристрій сервера
- 50 104 пристрої маршрутизації
- 106 пристрій транскодування
- 108 клієнтський пристрій

ФОРМУЛА ВИНАХОДУ

55

1. Спосіб обробки відеоданих, який включає:
прийом повідомлення, яке містить зовнішню вказівку від мережевого об'єкта, на пристрої декодування відео, причому мережевий об'єкт є відмінним і окремим від пристрою декодування відео;

прийом бітового потоку відео на пристрої декодування відео, при цьому потік бітів відео містить стиснуті відеодані, і при цьому потік бітів відео приймається окремо від повідомлення, яке містить зовнішню вказівку; і

обробку зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі зовнішньої вказівки.

2. Спосіб за пунктом 1, в якому зовнішня вказівка вказує, чи має значення прапора бути встановлене за допомогою пристрою декодування відео в одне зі значення за замовчуванням або встановленого значення, причому спосіб додатково включає:

встановлення значення прапора в значення, задане мережевим об'єктом, при цьому обробка зображення CRA як зображення BLA на основі зовнішньої вказівки включає обробку зображення CRA як зображення BLA на основі значення прапора, встановленого в значення, задане мережевим об'єктом.

3. Спосіб за пунктом 2, в якому прапор асоційований із зображенням CRA.

4. Спосіб за пунктом 2, в якому значення за замовчуванням прапора вказує, що зображення CRA не повинно оброблятися як зображення BLA.

5. Спосіб за пунктом 2, який додатково включає декодування одиниці рівня абстракції мережі (NAL) кодованого слайса, і коли декодують одиницю NAL кодованого слайса, зміну типу одиниці NAL власне одиниці NAL кодованого слайса на основі прапора, встановлюваного у встановлене значення.

6. Спосіб за пунктом 2, в якому прапор містить перший прапор, причому спосіб додатково включає, коли декодують одиницю NAL кодованого слайса, на основі першого прапора, що встановлюється у встановлене значення, зміну значення другого прапора, причому другий прапор керує виведенням попередніх декодованих зображень.

7. Спосіб за пунктом 2, в якому прапор містить перший прапор, причому спосіб додатково включає, коли декодують одиницю NAL кодованого слайса, на основі першого прапора, що встановлюється у встановлене значення, встановлення значення другого прапора в "1".

8. Спосіб за пунктом 5, в якому декодування одиниці NAL кодованого слайса включає в себе аналіз одиниці NAL кодованого слайса для ідентифікації типу одиниці NAL.

9. Спосіб за пунктом 2, в якому прапор містить перший прапор, і в якому коли декодують одиницю рівня абстракції мережі (NAL) кодованого слайса, якщо перший прапор дорівнює 1 і тип одиниці NAL власне одиниці NAL кодованого слайса вказує зображення CRA, спосіб додатково включає зміну значення типу одиниці NAL для вказівки зображення BLA.

10. Спосіб за пунктом 2, в якому прапором є HandleCraAsBlaFlag, причому HandleCraAsBlaFlag вказує, чи має конкретне зображення CRA бути оброблено як зображення BLA.

11. Спосіб за пунктом 2, який додатково включає, на основі значення прапора, встановлення значення другого прапора, причому другий прапор вказує, чи має щонайменше одне зображення, декодувати перед декодуванням зображення CRA, виводитися.

12. Пристрій декодування відео, який містить: пам'ять, сконфігуровану для зберігання відеоданих; і

процесор, пов'язаний з пам'яттю, причому процесор сконфігурований для: прийому повідомлення, яке містить зовнішню вказівку від мережевого об'єкта, на згаданому пристрої декодування відео, причому мережевий об'єкт є відмінним і окремим від згаданого пристрою декодування відео;

прийому бітового потоку відео на згаданому пристрої декодування відео, при цьому потік бітів відео містить стиснуті відеодані, і при цьому потік бітів відео приймається окремо від повідомлення, яке містить зовнішню вказівку; і

обробки зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі зовнішньої вказівки.

13. Пристрій декодування відео за пунктом 12, в якому зовнішня вказівка вказує, чи має значення прапора бути встановлено за допомогою згаданого пристрою декодування відео в одне зі значень за замовчуванням або встановленого значення, і при цьому процесор додатково сконфігурований для:

встановлення значення прапора в значення, задане мережевим об'єктом, при цьому обробка зображення CRA як зображення BLA на основі зовнішньої вказівки містить обробку зображення CRA як зображення BLA на основі значення прапора, встановленого в значення, задане мережевим об'єктом.

14. Пристрій декодування відео за пунктом 13, в якому прапор асоційований із зображенням CRA.

15. Пристрій декодування відео за пунктом 13, в якому значення за замовчуванням прапора вказує, що зображення CRA не повинно оброблятися як зображення BLA.

16. Пристрій декодування відео за пунктом 13, причому процесор додатково сконфігурований для, коли декодують одиницю рівня абстракції мережі (NAL) кодованого слайса, зміни типу одиниці NAL власне одиниці NAL кодованого слайса на основі прапора, що встановлюється у встановлене значення.

5 17. Пристрій декодування відео за пунктом 13, в якому прапор містить перший прапор і процесор згаданого пристрою декодування відео додатково сконфігурований для, коли декодують одиницю NAL кодованого слайса, на основі першого прапора, що встановлюється у встановлене значення, зміни значення другого прапора.

10 18. Пристрій декодування відео за пунктом 13, в якому прапор містить перший прапор, причому процесор додатково сконфігурований для, коли декодують одиницю рівня абстракції мережі (NAL) кодованого слайса, на основі першого прапора, що встановлюється у встановлене значення, встановлення значення другого прапора в "1".

19. Пристрій декодування відео за пунктом 16, причому процесор додатково сконфігурований для аналізу кожної одиниці NAL кодованого слайса для ідентифікації типу одиниці NAL.

15 20. Пристрій декодування відео за пунктом 19, при цьому прапор містить перший прапор, і при цьому, коли декодують одиницю NAL кодованого слайса, якщо перший прапор дорівнює 1 і тип одиниці NAL власне одиниці NAL кодованого слайса вказує зображення CRA, процесор додатково сконфігурований для зміни значення типу одиниці NAL для вказівки зображення BLA.

20 21. Пристрій декодування відео за пунктом 13, в якому прапором є HandleCraAsBlaFlag, причому HandleCraAsBlaFlag вказує, чи має зображення CRA бути оброблено як зображення BLA.

22. Пристрій декодування відео за пунктом 13, при цьому процесор додатково сконфігурований для, на основі значення прапора, встановлення значення першого прапора і другого прапора, причому другий прапор вказує, чи має щонайменше одне зображення, декодоване перед декодуванням зображення CRA, виводитися.

23. Пристрій декодування відео, що містить: пам'ять;

засіб, пов'язаний з пам'яттю, для прийому повідомлення, яке містить зовнішню вказівку від мережевого об'єкта, на згаданому пристрої декодування відео, причому мережевий об'єкт є відмінним і окремим від згаданого пристрою декодування відео;

30 засіб, пов'язаний з пам'яттю, для прийому бітового потоку відео на згаданому пристрої декодування відео, при цьому потік бітів відео містить стиснуті відеодані, і при цьому потік бітів відео приймається окремо від повідомлення, яке містить зовнішню вказівку; і

засіб, пов'язаний з пам'яттю, для обробки зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі зовнішньої вказівки.

35 24. Пристрій декодування відео за пунктом 23, в якому зовнішня вказівка вказує, чи має значення прапора бути встановлено за допомогою згаданого пристрою декодування відео в одне зі значення за замовчуванням або встановленого значення, причому пристрій декодування відео додатково містить:

засіб для встановлення значення прапора в значення, задане мережевим об'єктом,

40 при цьому обробка зображення CRA як зображення BLA на основі зовнішньої вказівки містить обробку зображення CRA як зображення BLA на основі значення прапора, встановленого в значення, задане мережевим об'єктом.

25. Пристрій декодування відео за пунктом 24, в якому прапор асоційований із зображенням CRA.

45 26. Пристрій декодування відео за пунктом 24, в якому значення за замовчуванням прапора вказує, що зображення CRA не повинно оброблятися як зображення BLA.

27. Пристрій декодування відео за пунктом 24, який додатково містить засіб, коли декодують одиницю рівня абстракції мережі (NAL) кодованого слайса, для зміни типу одиниці NAL власне одиниці NAL кодованого слайса на основі прапора, що встановлюється у встановлене значення.

50 28. Пристрій декодування відео за пунктом 24, причому прапор містить перший прапор і згаданий пристрій декодування відео додатково містить засіб, коли декодують одиницю NAL кодованого слайса, і на основі першого прапора, що встановлюється у встановлене значення, для зміни значення другого прапора, причому другий прапор керує виведенням попередніх декодованих зображень.

55 29. Пристрій декодування відео за пунктом 24, причому прапор містить перший прапор, причому пристрій декодування відео додатково містить засіб, коли декодують одиницю NAL кодованого слайса, і на основі першого прапора, що встановлюється у встановлене значення, для установки значення другого прапора в "1", причому другий прапор керує виведенням попередніх декодованих зображень.

30. Пристрій декодування відео за пунктом 27, в якому декодування одиниці NAL кодованого слайса включає в себе аналіз кожної одиниці NAL кодованого слайса для ідентифікації типу одиниці NAL.

5 31. Пристрій декодування відео за пунктом 24, при цьому прапор містить перший прапор, і при цьому коли декодують одиницю рівня абстракції мережі (NAL) кодованого слайса, якщо перший прапор дорівнює 1 і тип одиниці NAL власне одиниці NAL кодованого слайса вказує зображення CRA, згаданий пристрій декодування відео змінює значення типу одиниці NAL для вказівки зображення BLA.

10 32. Пристрій декодування відео за пунктом 24, при цьому прапором є HandleCraAsBlaFlag, причому HandleCraAsBlaFlag вказує, чи має зображення CRA бути оброблено як зображення BLA.

15 33. Пристрій декодування відео за пунктом 24, при цьому згаданий пристрій декодування відео містить засіб для встановлення, на основі значення прапора, значення першого прапора і другого прапора, причому другий прапор вказує, чи має щонайменше одне зображення, декодоване перед декодуванням зображення CRA, виводитися.

34. Зчитуваний комп'ютером носій даних, який зберігає виконуваний комп'ютером код, що містить інструкції, які при виконанні спонукають один або більше процесорів на одному пристрої декодування відео:

20 приймати повідомлення, яке містить зовнішню вказівку, від мережевого об'єкта, на згаданому пристрої декодування відео, причому мережевий об'єкт є відмінним і окремим від згаданого пристрою декодування відео;

приймати потік бітів відео на згаданому пристрої декодування відео, при цьому потік бітів відео містить стиснуті відеодані, і при цьому потік бітів відео приймається окремо від повідомлення, яке містить зовнішню вказівку; і

25 обробляти зображення чистого довільного доступу (CRA) як зображення доступу з розірваним посиланням (BLA) на основі зовнішньої вказівки.

35 35. Зчитуваний комп'ютером носій даних за пунктом 34, в якому зовнішня вказівка вказує, чи має значення прапора бути встановлено за допомогою пристрою декодування відео в одне зі значення за замовчуванням або встановленого значення, при цьому інструкції при виконанні

30 додатково спонукають один або більше процесорів на одному пристрої: встановлювати значення прапора в значення, задане мережевим об'єктом, при цьому обробка зображення CRA як зображення BLA на основі зовнішньої вказівки містить обробку зображення CRA як зображення BLA на основі значення прапора, встановленого в значення, задане мережевим об'єктом.

35 36. Зчитуваний комп'ютером носій даних за пунктом 35, в якому прапор асоційований із зображенням CRA.

40 37. Зчитуваний комп'ютером носій даних за пунктом 35, при цьому інструкції додатково спонукають один або більше процесорів, коли декодують одиницю NAL кодованого слайса, змінювати тип одиниці NAL власне одиниці NAL кодованого слайса на основі прапора, що встановлюється у встановлене значення.

45 38. Зчитуваний комп'ютером носій даних за пунктом 35, в якому прапор містить перший прапор і причому зчитуваний комп'ютером носій додатково сконфігурований для спонукання одного або більше процесорів, коли декодують одиницю NAL кодованого слайса, і на основі першого прапора, що встановлюється у встановлене значення, змінювати значення другого прапора, причому другий прапор керує виведенням попередніх декодованих зображень.

39. Зчитуваний комп'ютером носій даних за пунктом 37, додатково сконфігурований для спонукання одного або більше процесорів аналізувати одиницю NAL кодованого слайса для ідентифікації типу одиниці NAL.

50 40. Зчитуваний комп'ютером носій даних за пунктом 35, в якому прапор містить перший прапор, причому згаданий зчитуваний комп'ютером носій даних додатково зберігає інструкції, які при виконанні спонукають один або більше процесорів згаданого пристрою, коли декодують одиницю NAL кодованого слайса, і на основі першого прапора, що встановлюється у встановлене значення, змінювати значення другого прапора, причому другий прапор керує виведенням попередніх декодованих зображень.

55 41. Зчитуваний комп'ютером носій даних за пунктом 35, причому згаданий зчитуваний комп'ютером носій даних додатково зберігає інструкції, які при виконанні спонукають один або більше процесорів згаданого пристрою, на основі значення прапора, встановлювати значення другого прапора, причому другий прапор вказує, чи має щонайменше одне зображення, декодоване перед декодуванням зображення CRA, виводитися.

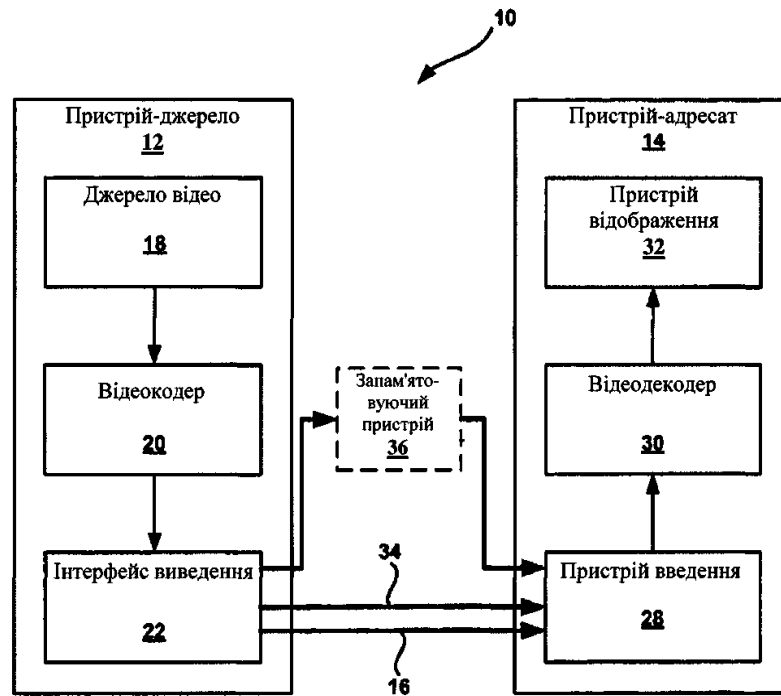


Fig. 1

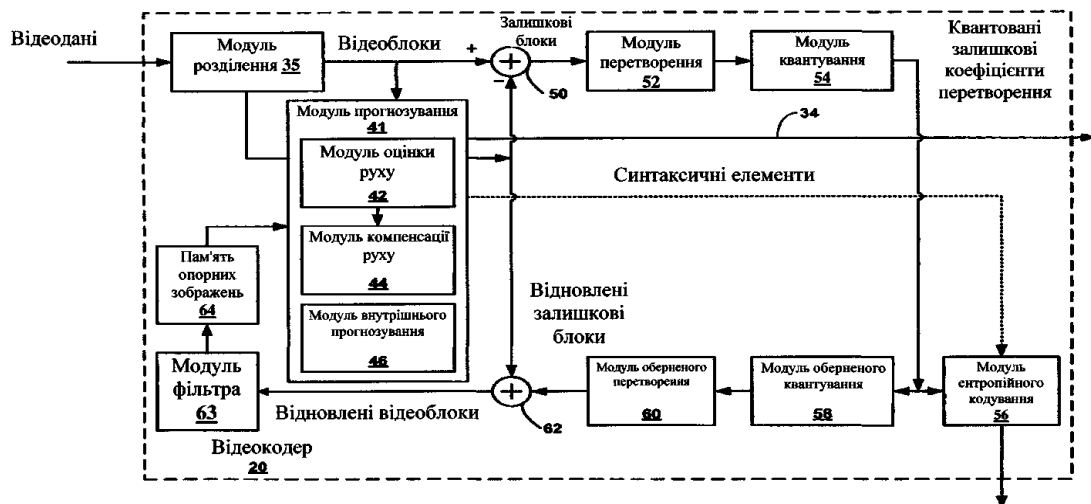
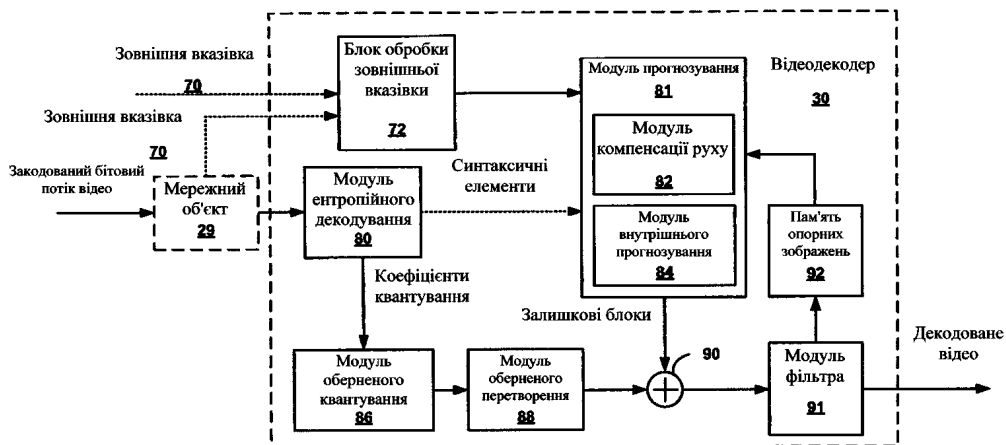
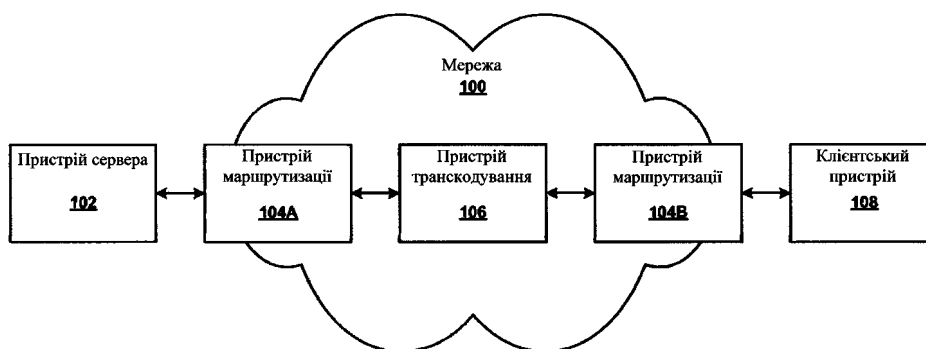


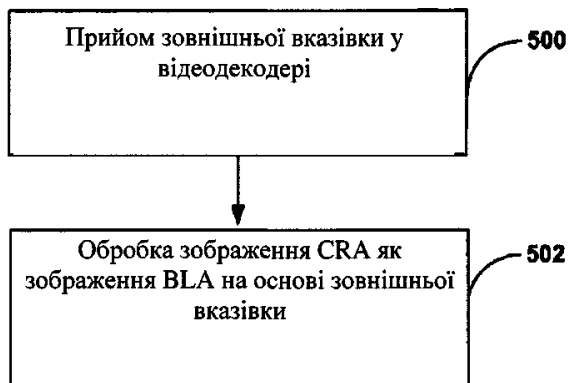
Fig. 2



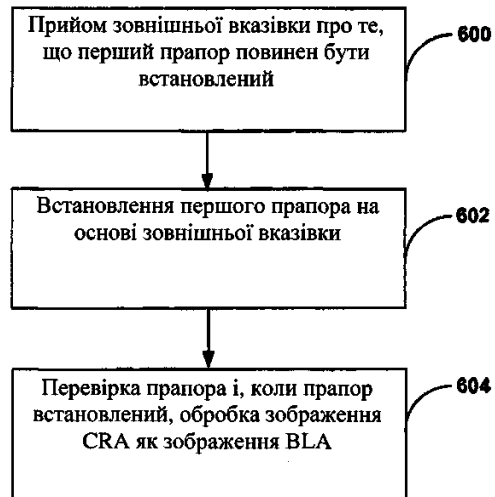
Фіг. 3



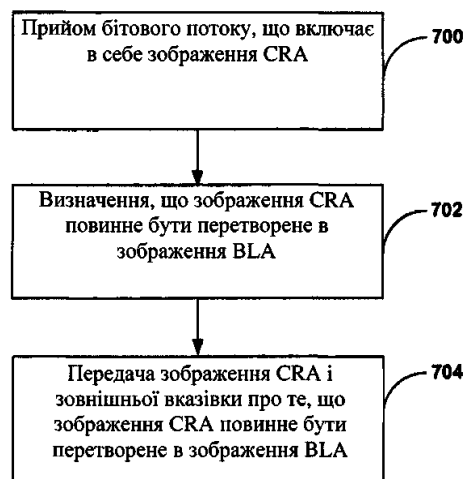
Фіг. 4



Фіг. 5



Фіг. 6



Фіг. 7

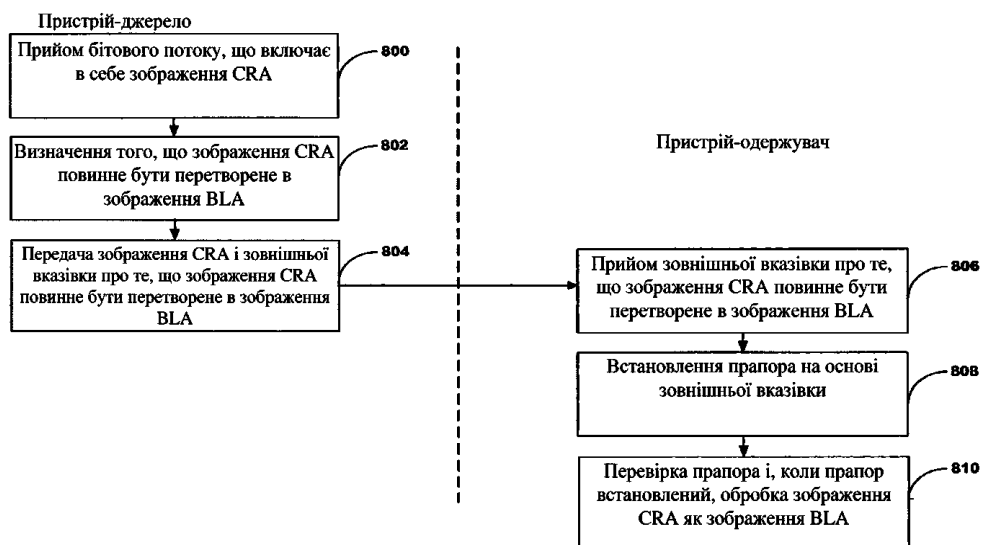


Fig. 8

Комп'ютерна верстка О. Рябо

Міністерство економічного розвитку і торгівлі України, вул. М. Грушевського, 12/2, м. Київ, 01008, Україна

ДП "Український інститут інтелектуальної власності", вул. Глазунова, 1, м. Київ – 42, 01601