



УКРАЇНА

(19) **UA** (11) **111866** (13) **C2**
(51) МПК**H04N 19/436** (2014.01)**H04N 19/17** (2014.01)**H04N 19/13** (2014.01)ДЕРЖАВНА СЛУЖБА
ІНТЕЛЕКТУАЛЬНОЇ
ВЛАСНОСТІ
УКРАЇНИ**(12) ОПИС ДО ПАТЕНТУ НА ВІНАХІД**

(21) Номер заявки:	а 2014 06112	(72) Винахідник(и):	Ван Є-Куй (US),
(22) Дата подання заявки:	01.11.2012		Кобан Мухаммед Зейд (US),
(24) Дата, з якої є чинними права на винахід:	24.06.2016		Карчевіч Марта (US)
(31) Номер попередньої заявки відповідно до Паризької конвенції:	61/555,932, 61/557,259, 13/548,825	(73) Власник(и):	КВЕЛКОММ ІНКОРПОРЕЙТЕД,
(32) Дата подання попередньої заявки відповідно до Паризької конвенції:	04.11.2011, 08.11.2011, 13.07.2012		Attn: International IP Administration, 5775 Morehouse Drive, San Diego, California 92121-1714, United States of America (US)
(33) Код держави-учасниці Паризької конвенції, до якої подано попередню заявку:	US, US, US	(74) Представник:	Мошинська Ніна Миколаївна, реєстр. №115
(41) Публікація відомостей про заявку:	10.09.2014, Бюл.№ 17	(56) Перелік документів, взятих до уваги експертизою:	BROSS B ET AL: "WD4: Working Draft 4 of High-Efficiency Video Coding", 6. JCT-VC MEETING; 97. MPEG MEETING; 14.07.2011 - 22.07.2011; TORINO; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC- SITE/, no. JCTVC-F803 THOMAS STOCKHAMMER ET AL: "Comments on H.26L NAL design", 12. VCEG MEETING; 09.01.2001 - 12.01.2001; EIBSEE, DE; (VIDEO CODINGEXPERTS GROUP OF ITU-T SG.16), no. VCEG--L26. 04.01.2001, XP 030003171 WO 2010050157 A1, 06.05.2010 FULDSETH (CISCO) A ET AL: "Tiles", 97. MPEG MEETING; 18.07.2011-22.07.2011; TORINO; (MOTION PICTURE EXPERT GROUP OR ISO/IES JTC1/SC/WG), no. m20757, 22.07.2011, XP 030049320
(46) Публікація відомостей про видачу патенту:	24.06.2016, Бюл.№ 12		
(86) Номер та дата подання міжнародної заявки, поданої відповідно до Договору РСТ	PCT/US2012/063027, 01.11.2012		

(54) ЗАПОВНЕННЯ НЕЗНАЧУЩОЮ ІНФОРМАЦІЄЮ СЕГМЕНТІВ У БЛОКАХ РІВНЯ АБСТРАКЦІЇ МЕРЕЖІ КОДОВАНОГО СЛАЙСУ**(57) Реферат:**

Відеокодер розділяє зображення на множину розділів зображення, таких як тайли або хвилі хвильової паралельної обробки (WPP). Розділи зображення зв'язані з неперекривними піднаборами деревоподібних блоків даного зображення. Відеокодер створює блок рівня абстракції мережі (NAL) кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, зв'язаних зі слайсом даного зображення. Кодовані деревоподібні блоки групуються в блоці NAL кодованого слайса в сегменти, зв'язані з різними розділами

UA 111866 C2

зображення. Відеокодер заповнює один або більше сегментів незначущою інформацією, так що кожний із сегментів починається на межі байта.

ОПИС

ПЕРЕХРЕСНІ ПОСИЛАННЯ НА СПОРІДНЕНІ ЗАЯВКИ

Дана заявка претендує на пріоритет попередньої заявки США № 61/557259, поданої 8 листопада 2011 року, зміст якої повністю включений сюди за допомогою посилання. Дана заявка також претендує на пріоритет попередньої заявки США № 61/555932, поданої 4 листопада 2011 року.

ГАЛУЗЬ ТЕХНІКИ, ДО ЯКОЇ НАЛЕЖИТЬ ВІНАХІД

Даний винахід належить до відеокодування (тобто кодування або декодування відеоданих).

РІВЕНЬ ТЕХНІКИ

Можливості цифрового відео можуть бути включені в пристрої найширшого застосування, в тому числі цифрові телевізійні приймачі, системи прямого цифрового мовлення, системи бездротового мовлення, персональні цифрові помічники (PDA), комп'ютери типу "лептоп" або настільні комп'ютери, цифрові камери, цифрові записуючі пристрої, цифрові медіаплеєри, відеоігрові пристрої, відеоігрові консолі, стільникові або супутникові радіотелефони, пристрої для проведення відео-телеконференцій тощо. Пристрої цифрового відео реалізують різні методи стиснення відео, які, наприклад, описані в стандартах, визначених як MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Частина 10, стандарт вдосконаленого відеокодування (AVC), стандарт високоефективного відеокодування (HEVC), що знаходиться в процесі розробки, а також розширення вказаних стандартів, для більш ефективної передачі, прийому і запам'ятовування цифрової відеоінформації.

Згідно з методиками стиснення відео виконується просторове (внутрішньокадрове) прогнозування і/або часове (міжкадрове) прогнозування з метою зменшення або виключення надмірності, властивої відеопослідовностям. Для відеокодування на блоковій основі відеослайс може бути розділений на відеоблоки, які також можуть називатися деревоподібними блоками, блоками кодування (CU), і/або вузлами кодування. Відеоблоки в слайсі зображення з внутрішньокадровим кодуванням (I) кодують з використанням просторового

прогнозування застосовно до опорних відліків у сусідніх блоках одного і того самого зображення. У відеоблоках в слайсі зображення з міжкадровим кодуванням (P або B) може використовуватися просторове прогнозування відносно опорних відліків у сусідніх блоках одного і того самого зображення або часове прогнозування відносно опорних відліків інших опорних зображень. Зображення можуть називатися кадрами, а опорні зображення можуть називатися опорними кадрами.

У документі "Comments on H.26L NAL Design" (VCEG-L26), Thomas Stockhammer, ITU-VCEG 12th meeting, Eibsee, Germany, січень 2001 року, запропоновано проектування NAL щодо його інтерфейсу за допомогою рівня відеокодування.

У документі WO 2010/050157 A1 розкритий спосіб кодування зображення, який може забезпечувати рівномірність якості зображення в цілому без пониження ефективності кодування за рахунок функціонування на високій швидкості за допомогою виконання перестановки макроблоків без зміни структури слайса.

У документі "WD4: Working Draft 4 °F High-Efficiency Video Coding" (JCTVC-F803_d0), Bross та ін., Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO / IEC JTC1 / SC29 / WG11, 6th Meeting: Torino, IT, 14-22 липня 2011 року, описаний робочий варіант проекту HEVC, що почалося в липні 2011 року.

СУТЬ ВІНАХОДУ

У загальному випадку у винаході описуються методики для кодування і декодування відеоданих. Відеокодер може розділити зображення на множину розділів зображення. Розділи зображення включають в себе неперекривні піднабори деревоподібних блоків даного зображення. Зразкові типи розділів зображення включають в себе тайли і хвилі хвильової паралельної обробки (WPP). Відеокодер може створити блок рівня абстракції мережі (NAL) кодованого слайса, який містить кодовані представлення деревоподібних блоків, зв'язаних з одним слайсом даного зображення. Відеокодер створює блок NAL кодованого слайса таким чином, що кодовані деревоподібні блоки групуються в блоці NAL кодованого слайса згідно з розділами зображення, до яких відносяться деревоподібні блоки. Відеокодер може заповнити один або більше сегментів незначущою інформацією, так щоб кожний із сегментів починався на межі байта. Відеокодер може декодувати кодовані деревоподібні блоки блока NAL кодованого слайса.

Згідно з одним аспектом винаходу описується спосіб кодування відеоданих. Спосіб містить розділення зображення на множину розділів зображення. Зображення містить множину деревоподібних блоків. Розділи зображення зв'язані з неперекривними під наборами деревоподібних блоків даного зображення. Спосіб додатково містить поділ зображення на

множину слайсів. Спосіб також містить створення блока NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, зв'язаних зі слайсом даного зображення, причому кодовані представлення деревоподібних блоків згруповані в блоці NAL кодованого слайса в сегменти, зв'язані з різними розділами зображення, де один або більше з вказаних сегментів заповнені незначущою інформацією, так що кожний із сегментів починається на межі байта.

Згідно з іншим аспектом описується спосіб декодування відеоданих. Спосіб містить запам'ятовування блока NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, зв'язаних зі слайсом зображення. Вказане зображення розбивають на множину розділів зображення. Кодовані представлення вказаних деревоподібних блоків групують у сегменти, зв'язані з різними розділами зображення. Один або більше з вказаних сегментів заповнюють незначущою інформацією, так що кожний з вказаних сегментів починається з межі байта. Спосіб також містить декодування кодованих представлень деревоподібних блоків.

Згідно з ще одним аспектом описується пристрій відеокодування, який кодує відеодані. Пристрій відеокодування містить один або більше процесорів, сконфігурованих для розділення зображення на множину розділів зображення. Зображення містить множину деревоподібних блоків. Розділи зображення зв'язані з неперекривними піднаборами деревоподібних блоків даного зображення. Зазначені один або більше процесорів також сконфігуровані для поділу зображення на множину слайсів. Вказані один або більше процесорів також сконфігуровані для створення блока NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, які зв'язані зі слайсом даного зображення. Кодовані представлення деревоподібних блоків, згруповані в блоці NAL кодованого слайса в сегменти, зв'язані з різними розділами зображення. Один або більше з вказаних сегментів заповнюють незначущою інформацією, так що кожний із сегментів починається на межі байта.

Згідно з наступним аспектом описується пристрій відеодекодування, який декодує відеодані. Пристрій відеодекодування містить пам'ять, в якій запам'ятовується блок NAL кодованого слайса, що включає в себе кодовані представлення деревоподібних блоків, зв'язаних зі слайсом зображення. Вказане зображення розбивають на множину розділів зображення. Кодовані представлення вказаних деревоподібних блоків групують у сегменти, зв'язані з різними розділами зображення. Один або більше з вказаних сегментів заповнюють незначущою інформацією, так що кожний з вказаних сегментів починається з межі байта. Пристрій відеодекодування також містить один або більше процесорів, які сконфігуровані для декодування кодованих представлень деревоподібних блоків.

Згідно з ще одним аспектом описується комп'ютерний програмний продукт, який містить один або більше машинозчитуваних носіїв інформації, що зберігають команди, які під час їх виконання одним або більше процесорами конфігурують пристрій відеокодування для розділення зображення на множину розділів зображення. Зображення містить множину деревоподібних блоків. Розділи зображення зв'язані з неперекривними піднаборами деревоподібних блоків даного зображення. Зазначені команди також конфігурують пристрій відеокодування для поділу зображення на множину слайсів. Вказані команди також конфігурують пристрій відеокодування для створення блока NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, які зв'язані зі слайсом даного зображення. Кодовані представлення деревоподібних блоків, згруповані в блоці NAL кодованого слайса в сегменти, зв'язані з різними розділами зображення. Один або більше з вказаних сегментів заповнюють незначущою інформацією, так що кожний із сегментів починається на межі байта.

Згідно з ще одним аспектом описується комп'ютерний програмний продукт, який містить один або більше машинозчитуваних носіїв інформації, що зберігають команди, які під час їх виконання одним або більше процесорами конфігурують пристрій відеодекодування для запам'ятовування блока NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, зв'язаних зі слайсом зображення. Вказане зображення розбивають на множину розділів зображення. Кодовані представлення вказаних деревоподібних блоків групують у сегменти, зв'язані з різними розділами зображення. Один або більше з вказаних сегментів заповнюють незначущою інформацією, так що кожний з вказаних сегментів починається з межі байта. Вказані команди також конфігурують пристрій відеодекодування для декодування кодованих представлень деревоподібних блоків.

Згідно з ще одним аспектом описується пристрій відеокодування, який виконує кодування відеоданих. Пристрій відеокодування містить засіб для розділення зображення на множину розділів зображення. Зображення містить множину деревоподібних блоків. Розділи зображення

зв'язані з неперекривними піднаборами деревоподібних блоків даного зображення. Пристрій відеокодування також містить засіб для створення блока NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, зв'язаних зі слайсом даного зображення. Кодовані представлення деревоподібних блоків, згруповані в блоці NAL кодованого слайса в сегменти, зв'язані з різними розділами зображення. Один або більше з вказаних сегментів заповнюють незначущою інформацією, так що кожний з вказаних сегментів починається з межі байта. Спосіб також містить декодування кодованих представлень деревоподібних блоків.

Згідно з ще одним аспектом описується пристрій відеокодування. Пристрій відеокодування містить один або більше процесорів, сконфігурованих для розділення зображення на множину розділів зображення. Зображення містить множину деревоподібних блоків. Розділи зображення зв'язані з неперекривними піднаборами деревоподібних блоків даного зображення. Вказані один або більше процесорів також сконфігуровані для створення блока NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, які зв'язані зі слайсом даного зображення. Кодовані представлення деревоподібних блоків, згруповані в блоці NAL кодованого слайса в сегменти, зв'язані з різними розділами зображення. Один або більше з вказаних сегментів заповнюють незначущою інформацією, так що кожний із сегментів починається на межі байта.

Згідно з наступним аспектом описується пристрій відеодекодування, який декодує відеодані. Пристрій відеодекодування містить пам'ять, в якій запам'ятовується блок NAL кодованого слайса, що включає в себе кодовані представлення деревоподібних блоків, зв'язаних зі слайсом зображення. Вказане зображення розбивають на множину розділів зображення. Кодовані представлення вказаних деревоподібних блоків групують у сегменти, зв'язані з різними розділами зображення. Один або більше з вказаних сегментів заповнюють незначущою інформацією, так що кожний з вказаних сегментів починається з межі байта. Пристрій відеодекодування також містить один або більше процесорів, які сконфігуровані для декодування кодованих представлень деревоподібних блоків.

Згідно з ще одним аспектом описується комп'ютерний програмний продукт, який містить один або більше машинозчитуваних носіїв інформації, що зберігають команди, які під час їх виконання одним або більше процесорами конфігурують пристрій відеокодування для розділення зображення на множину розділів зображення. Зображення містить множину деревоподібних блоків. Розділи зображення зв'язані з неперекривними піднаборами деревоподібних блоків даного зображення. Вказані команди також конфігурують пристрій відеокодування для створення блока NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, які зв'язані зі слайсом даного зображення. Кодовані представлення деревоподібних блоків, згруповані в блоці NAL кодованого слайса в сегменти, зв'язані з різними розділами зображення. Один або більше з вказаних сегментів заповнюють незначущою інформацією, так що кожний із сегментів починається на межі байта.

Згідно з ще одним аспектом описується комп'ютерний програмний продукт, який містить один або більше машинозчитуваних носіїв інформації, що зберігають команди, які під час їх виконання одним або більше процесорами конфігурують пристрій відеодекодування для запам'ятовування блока NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, зв'язаних зі слайсом зображення. Вказане зображення розбивають на множину розділів зображення. Кодовані представлення вказаних деревоподібних блоків групують у сегменти, зв'язані з різними розділами зображення. Один або більше з вказаних сегментів заповнюють незначущою інформацією, так що кожний з вказаних сегментів починається з межі байта. Вказані команди також конфігурують пристрій відеодекодування для декодування кодованих представлень деревоподібних блоків.

Згідно з ще одним аспектом описується пристрій відеокодування, який виконує кодування відеоданих. Пристрій відеокодування містить засіб для розділення зображення на множину розділів зображення. Зображення містить множину деревоподібних блоків. Розділи зображення зв'язані з неперекривними піднаборами деревоподібних блоків даного зображення. Пристрій відеокодування також містить засіб для створення блока NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, зв'язаних зі слайсом даного зображення. Кодовані представлення деревоподібних блоків, згруповані в блоці NAL кодованого слайса в сегменти, зв'язані з різними розділами зображення. Один або більше з вказаних сегментів заповнюють незначущою інформацією, так що кожний із сегментів починається на межі байта.

Згідно з наступним аспектом описується пристрій відеодекодування, який декодує відеодані. Пристрій відеодекодування містить засіб для зберігання блока NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, зв'язаних зі слайсом зображення. Вказане зображення розділяють на множину розділів зображення. Кодовані

представлення вказаних деревоподібних блоків групують у сегменти, зв'язані з різними розділами зображення. Один або більше з вказаних сегментів заповнюють незначущою інформацією, так що кожний з вказаних сегментів починається з межі байта. Пристрій відеодекодування також містить засіб для декодування кодованих представлень деревоподібних блоків.

Деталі одного або більше прикладів представлені на супровідних кресленнях і викладаються в наведеному нижче описі. Інші ознаки, цілі та переваги винаходу стануть очевидними із згаданого опису, креслень та з формули винаходу.

КОРОТКИЙ ОПИС КРЕСЛЕНЬ

Фіг. 1 - блок-схема, що ілюструє наведену як приклад систему відеокодування, в якій можуть використовуватися методики, описані в даному винаході;

Фіг. 2 - блок-схема, що ілюструє наведений як приклад відеокoder, який сконфігурований для реалізації методик, розкритих в даному винаході;

Фіг. 3 - блок-схема, що ілюструє наведений як приклад відеодекодер, який сконфігурований для реалізації методик, розкритих в даному винаході;

Фіг. 4 - блок-схема, що ілюструє зразкову операцію для створення даних для слайса зображення;

Фіг. 5 - блок-схема, що ілюструє зразкову операцію для декодування блока NAL кодованого слайса;

Фіг. 6 - концептуальна схема, що ілюструє хвильову паралельну обробку;

Фіг. 7 - концептуальна схема, що ілюструє зразковий порядок кодування, коли зображення розділене на множину тайлів;

Фіг. 8 - концептуальна схема, що ілюструє наведений як приклад блок NAL кодованого слайса.

ЗДІЙСНЕННЯ ВІНАХОДУ

Зображення включає в себе множину деревоподібних блоків. Деревоподібні блоки зв'язані з двомірними відеоблоками в даному зображенні. Відеокoder розділяє зображення на множину розділів зображення. Наприклад, відеокoder може розділити зображення на тайли або хвилі хвильової паралельної обробки (WPP). Іншими словами, в цьому винаході термін "розділ зображення" може використовуватися для посилань в загальному випадку на тайли або хвилі WPP. Розділи зображення зв'язані з неперекривними піднаборами деревоподібних блоків зображення. Наприклад, кожний деревоподібний блок даного зображення може бути зв'язаний рівно з одним із згаданих розділів зображення.

Відеокoder може створити блок рівня абстракції мережі (NAL) кодованого слайса. Блок NAL кодованого слайса може включати в себе кодовані представлення кожного деревоподібного блока, зв'язаного зі слайсом даного зображення. У даному описі винаходу можливі посилання на кодоване представлення деревоподібного блока як кодованого деревоподібного блока. Кодований деревоподібний блок може включати в себе послідовність бітів, яка представляє відеоблок, зв'язаний з деревоподібним блоком. Ця послідовність бітів у кодованому деревоподібному блоці може представляти послідовність синтаксичних елементів.

Відеокoder може згрупувати кодовані деревоподібні блоки в блоці NAL кодованого слайса в сегменти. Ці сегменти зв'язані з різними розділами зображення. Кожний з цих сегментів може являти собою послідовний ряд бітів, таких як біти, що представляють послідовність, яка складається з одного або більше кодованих деревоподібних блоків і зв'язаних з ними даних. Таким чином, кожний блок NAL кодованого слайса може включати в себе кодований деревоподібний блок, зв'язаний з першим розділом зображення, за яким йде кодований деревоподібний блок, зв'язаний з другим розділом зображення, за яким йде кодований деревоподібний блок, зв'язаний з третім розділом зображення, тощо.

Згідно з розкритими тут методиками, відеокoder може заповнити один або більше вказаних сегментів незначущою інформацією так, щоб кожний із сегментів починався на межі байта. Коли відеокoder заповнює сегмент, він може приєднати біти заповнення до цього сегмента. Біти заповнення не мають якого-небудь семантичного значення, а призначені для забезпечення того, щоб наступний сегмент починався з межі байта. Таким чином, відеокoder може забезпечити побайтове вирівнювання тайлів або хвиль WPP, коли ці тайли або хвилі WPP включені в один блок NAL кодованого слайса з метою паралельної обробки.

Відеодекодер може запам'ятати блок NAL кодованого слайса в пам'яті з байтовою адресацією. Відеодекодер може потім розподілити два або більше сегментів по різних потоках декодування, які діють паралельно. Кожний потік декодування декодує кодовані деревоподібні блоки сегмента, розподіленого для даного потоку декодування. Оскільки кожний сегмент починається з межі байта, відеодекодер може надати адресу пам'яті для сегмента потоку

декодування при розподілі даного сегмента для даного потоку декодування. Таким шляхом забезпечується початок кожного сегмента з межі байта, що дає можливість відеодекодерам декодувати сегменти паралельно більш простим способом, ніж у випадку, коли сегменти починаються з позицій, не співпадаючих з межею байта.

5 У цьому полягає їх відмінність від стандартних відеокодерів і стандартних відеодекодерів, які не забезпечують початок сегментів з меж байтів. Оскільки сегменти можуть починатися не з меж байтів, стандартний відеодекодер, що використовує побайтову адресацію пам'яті, можливо не зможе декодувати кодовані деревоподібні блоки у вказаних сегментах в паралельному режимі. Стандартний відеодекодер може використовувати побайтову адресацію пам'яті або
10 побайтову плюс побітову адресацію, щоб мати можливість декодувати кодовані деревоподібні блоки в сегментах паралельно, але при цьому ускладнюється технічна реалізація і процес обчислень.

На прикладених кресленнях показаний ряд прикладів. Елементи на прикладених кресленнях, позначені визначеними посилальними позиціями, відповідають елементам,
15 вказаним під тими самими посилальними позиціями в нижченаведеному описі. Тут елементи з назвами, що починаються з порядкових числівників (наприклад, "перший", "другий", "третій" тощо) не обов'язково передбачають, що ці елементи мають визначений порядок. Швидше навпаки, вказані порядкові числівники використовуються тут просто для посилань на різні елементи одного і того самого або подібного типу. Крім того, в подальшому описі термін
20 "поточне зображення" може відноситися до зображення, яке в даний момент зазнає кодування або декодування.

На Фіг. 1 представлена блок-схема, де показаний приклад системи 10 відеокодування, в якій можуть бути використані розкриті тут методики. Термін "відеокодер", що використовується тут, відноситься власне як до відеокодерів, так і до відеодекодерів. У цьому описі терміни
25 "відеокодування" або "кодування" можуть відноситися загалом до відеокодування і відеодекодування.

Як показано на Фіг. 1 система 10 відеокодування включає в себе пристрій-джерело 12 і пристрій-адресат 14. Пристрій-джерело 12 створює кодовані відеодані. Відповідно пристрій-джерело 12 може називатися пристроєм відеокодування. Пристрій-адресат 14 може декодувати
30 кодовані відеодані, створені пристроєм-джерелом 12. Відповідно пристрій-адресат 14 може називатися пристроєм відеодекодування. Пристрій-джерело 12 і пристрій-адресат 14 можуть являти собою приклади пристроїв відеокодування.

Пристрій-джерело 12 і пристрій-адресат 14 можуть містити будь-який з найрізноманітніших пристроїв, в тому числі, настільні комп'ютери, мобільні обчислювальні пристрої, комп'ютери типу "ноутбук" (наприклад, "лептоп"), планшетні комп'ютери, телевізійні приставки, телефонні трубки, так звані "смартфони", телевізійні приймачі, камери, пристрої відображення, цифрові медіаплеєри, відеоігрові консолі, автокомп'ютери або т.п. В деяких випадках пристрій-джерело 12 і пристрій-адресат 14 можуть бути обладнані засобами бездротового зв'язку.
35

Пристрій-адресат 14 може приймати кодовані відеодані від пристрою-джерела 12 через канал 16. Канал 16 може містити середовище або пристрій будь-якого типу, здатний
40 перемішувати кодовані відеодані від пристрою-джерела 12 на пристрій-адресат 14. В одному прикладі канал 16 може містити середовище зв'язку, що дозволяє пристрою-джерелу 12 передавати кодовані відеодані безпосередньо на пристрій-адресат 14 в режимі реального часу. Кодовані відеодані можуть бути модульовані пристроєм-джерелом 12 відповідно до стандарту зв'язку, такого як протокол бездротового зв'язку, і можуть передаватися на пристрій-адресат 14.
45 Середовище зв'язку може містити будь-який носій дротового або бездротового зв'язку, наприклад, радіочастотний (RF) спектр або одна або більше фізичних ліній передачі. Середовище зв'язку може становити частину мережі пакетної передачі, такої як локальна мережа, регіональна мережа або глобальна мережа, наприклад Інтернет. Середовище зв'язку
50 може включати в себе маршрутизатори, комутатори, базові станції або будь-яке інше обладнання, яке може бути корисне для здійснення зв'язку між пристроєм-джерелом 12 і пристроєм-адресатом 14.

В іншому прикладі канал 16 може відповідати носію інформації, що зберігає кодовані відеодані, створені пристроєм-джерелом 12. У цьому прикладі пристрій-адресат 14 може мати
55 доступ до вказаного носія інформації через диски або карти. Вказаний носій інформації може включати в себе найрізноманітніші носії інформації локального доступу, такі як диски типу Blu-ray, диски DVD, диски CD, флеш-пам'ять, або будь-який інший підходящий цифровий носій інформації для зберігання кодованих відеоданих. У ще одному прикладі канал 16 може включати в себе файловий сервер або інший проміжний запам'ятовуючий пристрій, який може
60 зберігати кодоване відео, створене пристроєм-джерелом 12. Пристрій-адресат 14 в цьому

прикладі може звертатися до кодованих відеоданих, запам'ятованих на файловому сервері або іншому проміжному запам'ятовуючому пристрої, використовуючи потокову передачу або завантаження відеоданих. Файловий сервер може являти собою сервер будь-якого типу, здатний запам'ятовувати і зберігати кодовані відеодані і передавати їх на пристрій-адресат 14.

5 Приклади файлових серверів включають в себе Web-сервери (наприклад, для Web-сайта), сервери Протоколу пересилання файлів (FTP), підключені до мережі запам'ятовуючі пристрої (NAS) і локальні накопичувачі на дисках. Пристрій-адресат 14 може мати доступ до кодованих відеоданих через будь-яке стандартне з'єднання для передачі даних, в тому числі Інтернет з'єднання. Такі з'єднання можуть включати в себе бездротові канали (наприклад, Wi-Fi з'єднання), дротові з'єднання (наприклад, DSL, кабельний модем тощо), або комбінацію обох видів з'єднань, яка підходить для доступу до кодованих відеоданих, що зберігаються на файловому сервері. Передача кодованих відеоданих від файлового сервера може бути реалізована у вигляді потокової передачі, передачі з вивантаженням або їх комбінації.

15 Розкриті тут методики не обов'язково обмежуються бездротовими додатками або установками. Ці методики можна застосувати до відеокодування для підтримки будь-якого з різноманітних мультимедійних додатків, таких як ефірне телевізійне мовлення, передачі кабельного телебачення, передачі супутникового телебачення, передачі потокового відео, наприклад, через Інтернет, кодування цифрового відео для запам'ятовування на носії даних, декодування цифрового відео, що зберігається на носії даних, або інші додатки. У деяких прикладах система 10 відеокодування може бути сконфігурована для підтримки однонаправленої або двонаправленої передачі відео для підтримки додатків, таких як потокове відео, відео відтворення, відеомовлення і/або відеотелефонія.

20 У прикладі за Фіг. 1 пристрій-джерело 12 включає в себе джерело 18 відео, відеокодер 20 і вихідний інтерфейс 22. У деяких випадках вихідний інтерфейс 22 може включати в себе модулятор/демодулятор (модем) і/або передавач. У пристрої-джерелі 12 джерело 18 відео може включати в себе таке джерело, як пристрій фіксації відео, наприклад, відеокамеру, відеоархів, що містять раніше зафіксоване відео, інтерфейс подачі відео для прийому відео від провайдера відеоконтенту і/або системи комп'ютерної графіки для створення відеоданих або комбінації вказаних джерел.

30 Зафіксоване в даний момент, заздалегідь зафіксоване або створене комп'ютером відео може бути кодоване відеокодером 20. Кодовані відеодані можуть передаватися безпосередньо на пристрій-адресат 14 через вихідний інтерфейс 22 пристрою-джерела 12. Кодовані відеодані також можуть запам'ятовуватися на носії даних або файловому сервері для звернення до них в майбутньому з боку пристрою-адресата 14 для декодування і/або відтворення.

35 У прикладі на Фіг. 1 пристрій-адресат 14 включає в себе вхідний інтерфейс 28, відеодекодер 30 і пристрій 32 відображення. У деяких випадках вхідний інтерфейс 28 може включати в себе приймач і/або модем. Вхідний інтерфейс 28 пристрою-адресата 14 приймає кодовані відеодані через канал 16. Кодовані відеодані можуть включати в себе різноманітні синтаксичні елементи, створені відеокодером 20, які представляють відеодані. Вказані синтаксичні елементи можуть бути включені в кодовані відеодані, які передаються через середовище зв'язку, запам'ятовуються на носії даних або запам'ятовуються на файловому сервері.

40 Пристрій 32 відображення може бути складовою частиною (або бути зовнішнім) пристрою-адресата 14. У деяких прикладах пристрій-адресат 14 може включати в себе інтегрований в нього пристрій відображення, а також бути сконфігурований для взаємодії із зовнішнім пристроєм відображення. В інших прикладах пристроєм-адресатом 14 може бути пристрій відображення. У загальному випадку пристрій 32 відображення відображає користувачу декодовані відеодані. Пристрій 32 відображення може містити будь-який з різноманітних пристроїв відображення, таких як рідкокристалічний дисплей (LCD), плазмовий дисплей, дисплей на органічних світловипромінюючих діодах (OLED) або пристрій відображення іншого типу.

50 Відеокодер 20 і відеодекодер 30 можуть функціонувати згідно зі стандартом стиснення відео, таким як стандарт високоефективного відеокодування (HEVC), що розробляється в цей час, і може відповідати тестовій моделі HEVC (HM). Останній проект стандарту HEVC, який називається "HEVC Working Draft 6" або "WD6" описаний в документі JCTVC-H1003, Bross і інш., "High efficiency video coding (HEVC) text specification draft 6" Об'єднаної команди розробників систем відеокодування (JCT-VC) стандартів ITU-T SG16 WP3 та ISO/IEC JTC1/SC29/WG11, 8th Meeting: SanJose, California USA, February, 2012, який станом на 1 травня 2012 року можна скачати з: http://phenix.int-evry.fr/jct/doc_end_user/documents/8_San%20Jose/wg11/JCTVC-H1003-v22.zip, і зміст якого повністю включений сюди за посиланням. Як альтернатива відеокодер 20 і відеодекодер 30 можуть працювати згідно з іншими комерційними або промисловими

стандартами, такими як стандарт ITU-T H.264, який також називається MPEG-4, частина 10, AVC (вдосконалене відеокодування) або розширеннями вказаних стандартів, коли в них включені методики розділення зображення на тайли або хвилі хвильової паралельної обробки. Однак методики, розкриті в цьому винаході, не обмежуються яким-небудь конкретним стандартом або методикою кодування. Інші приклади стандартів і методик стиснення відео включають в себе MPEG-2 та ITU-T H.263 і патентовані формати стиснення або формати стиснення з відкритих джерел, такі як VP8 і споріднені формати при використанні методик розділення зображення на тайли або хвилі хвильової паралельної обробки.

Хоча це на Фіг. 1 не показано, згідно з деякими аспектами відеокодер 20 і відеодекодер 30 можуть бути інтегровані кожний з аудіокодером і аудіодекодером і можуть включати в себе відповідні блоки MUX-DMUX (мультиплексор-демультиплексор) або інші апаратні і програмні засоби для кодування як аудіо, так і відео в загальному потоці даних або в окремих потоках даних. У деяких прикладах, якщо це має місце, блоки MUX-DMUX можуть відповідати Протоколу мультиплексора ITU H.223 або іншим протоколам, таким як Протокол дейтаграм користувача (UDP).

Потрібно зазначити, що Фіг. 1 є просто прикладом, і розкриті тут методики можна застосувати до установок відеокодування (наприклад, відеокодування або відеодекодування), які не обов'язково включають який-небудь обмін даними між пристроєм кодування і пристроєм декодування. В інших прикладах дані можуть витягуватися з локальної пам'яті, надходити у вигляді потоку через мережу або т.п. Пристрій кодування може кодувати і запам'ятовувати дані в пам'яті, і/або пристрій декодування може витягувати і декодувати дані з пам'яті. У багатьох прикладах кодування і декодування виконується пристроями, які не здійснюють обмін даними один з одним, а просто кодують дані в пам'ять і/або витягують і декодують дані з пам'яті.

Відеокодер 20 і відеодекодер 30 можуть бути реалізовані кожний у вигляді будь-яких підходящих схем кодера або декодера, таких як один або більше мікропроцесорів, цифрові процесори сигналів (DSP), прикладні спеціалізовані інтегральні схеми ASIC, вентиляльні матриці, що програмуються користувачем (FPGA), дискретні логічні схеми, апаратні засоби, або будь-які їх комбінації. При частковій реалізації вказаних методик програмними засобами пристрій може зберігати команди програмного забезпечення на підходящому зчитуваному комп'ютером носії тривалого зберігання для виконання цих команд апаратними засобами з використанням одного або більше процесорів, що реалізують розкриті у винаході методики. Як відеокодер 20, так і відеодекодер 30 можуть бути включені до складу одного або більше кодерів або декодерів, або кожний з них може бути інтегрований в об'єднаному кодері/декодері ("CODEC") у відповідному пристрої.

Як стисло згадувалося вище, відеокодер 20 кодує відеодані. Ці відеодані можуть містити одне або більше зображень. Кожне з цих зображень являє собою нерухоме зображення, яке утворює частину відео. У деяких випадках зображення може називатися "кадром" відео або "полем" відео. Коли відеокодер 20 кодує відеодані, він може створити бітовий потік. Цей бітовий потік може включати в себе послідовність бітів, які утворюють кодоване представлення відеоданих. Бітовий потік може включати в себе кодовані зображення і зв'язані з ними дані. Кодоване зображення є кодованим представленням зображення.

Для створення бітового потоку відеокодер 20 може виконати операції кодування для кожного зображення у відеоданих. Коли відеокодер 20 виконує операції кодування для зображень, він може створити ряд кодованих зображень і зв'язаних з ними даних. Ці зв'язані із зображеннями дані можуть включати в себе набори параметрів послідовності, набори параметрів зображення, набори параметрів адаптації та інші синтаксичні структури. Набір параметрів послідовності (SPS) може містити параметри, що застосовуються до нуля або більше послідовностей зображень. Послідовності зображень також можуть називатися кодованими відеопослідовностями, як в стандартах H.264/AVC та HEVC. Набір параметрів зображення (PPS) може містити параметри, що застосовуються до нуля або більше зображень. Набір параметрів адаптації (APS) може містити параметри, що застосовуються до нуля або більше зображень. Параметрами в наборі APS можуть бути параметри, імовірність зміни яких вище, ніж у параметрів в наборі PPS.

Для створення кодованого зображення відеокодер 20 може розділити зображення на відеоблоки однакового розміру. Відеоблок може являти собою двовірну матрицю відліків. Кожний з відеоблоків зв'язаний з деревоподібним блоком. У деяких випадках деревоподібний блок може називатися найбільшим блоком кодування (LCU) або деревоподібним блоком кодування. Деревоподібні блоки стандарту HEVC можуть в багатьох відношеннях бути подібними до макроблоків попередніх стандартів, таких як H.264/AVC. Однак деревоподібний блок не обов'язково обмежений конкретним розміром і може включати в себе один або більше

блоків кодування (CU). Відеокoder 20 може використовувати розбиття квадродререва для виділення відеоблоків деревоподібних блоків у відеоблоки, зв'язані з блоками CU, звідси назва "деревоподібні блоки".

У деяких прикладах відеокoder 20 може розбити зображення на множину слайсів. Кожний зі слайсів може включати в себе ціле число послідовно закодованих деревоподібних блоків. У деяких випадках кожний зі слайсів може включати в себе ціле число послідовно закодованих блоків CU. Як частину виконання операції кодування зображення, відеокoder 20 може виконувати операції кодування на кожному слайсі даного зображення. Під час виконання відеокoderом 20 операції кодування на слайсі відеокoder 20 може створити кодовані дані, зв'язані з даним слайсом. Кодовані дані, зв'язані з даним слайсом, можуть називатися "кодованим слайсом".

Для створення кодованого слайса відеокoder 20 може виконати операції кодування на кожному деревоподібному блоці в слайсі. Під час виконання відеокoderом 20 операції кодування на деревоподібному блоці він може створити кодований деревоподібний блок. Кодований деревоподібний блок може містити дані, що представляють кодовану версію деревоподібного блока.

При створенні кодованого слайса відеокoder 20 може виконати операцію кодування на деревоподібних блоках у цьому слайсі згідно з порядком растрового сканування. Іншими словами, відеокoder 20 може кодувати деревоподібні блоки слайса в порядку зліва направо через найверхній рядок деревоподібних блоків у слайсі, потім зліва направо через наступний рядок деревоподібних блоків, що лежить нижче, і так далі, доки відеокoder 20 не виконає кодування всіх деревоподібних блоків у даному слайсі.

В результаті кодування згідно з порядком растрового сканування будуть кодовані деревоподібні блоки, що знаходяться над і зліва від даного деревоподібного блока, а деревоподібні блоки, що знаходяться нижче і правіше даного деревоподібного блока, поки залишаться не кодованими. Потім відеокoder 20 може одержати доступ до інформації, створеної внаслідок кодування деревоподібних блоків, що знаходяться вище і зліва від даного деревоподібного блока при кодуванні даного деревоподібного блока. Однак відеокoder 20 не може одержати доступ до інформації, створеної внаслідок кодування деревоподібних блоків, що знаходяться нижче і праворуч від даного деревоподібного блока при кодуванні даного деревоподібного блока.

Для створення кодованого деревоподібного блока відеокoder 20 може рекурсивно виконувати розбиття квадродререва на відеоблоці деревоподібного блока для розділення відеоблока на відео блоки, що послідовно зменшуються. Кожний з відео блоків, що послідовно зменшуються, може бути зв'язаний зі своїм блоком CU. Наприклад, відеокoder 20 може розбити відеоблок деревоподібного блока на чотири підблоки однакового розміру, розбити один або більше з вказаних підблоків на чотири елементи підблока однакового розміру тощо. Виділений таким чином блок CU може являти собою блок CU, чий відеоблок розбитий на відеоблоки, зв'язані з іншими блоками CU. Нерозбитий блок CU - це блок CU, чий відеоблок не розбитий на відеоблоки, зв'язані з іншими блоками CU.

Один або більше синтаксичних елементів у бітовому потоці можуть вказувати максимальну кількість можливих розбиттів вказаного відеоблока деревоподібного блока, які виконуються відеокoderом 20. Відеоблок блока CU може мати форму квадрата. Розмір відеоблока блока CU може знаходитися у діапазоні від 8×8 пікселів до розміру відеоблока деревоподібного блока (тобто, розміру деревоподібного блока), що складає максимум 64×64 пікселя або більше.

Відеокoder 20 може виконувати операції кодування кожного блока CU деревоподібного блока згідно з порядком z-сканування. Іншими словами, відеокoder 20 може кодувати лівий верхній блок CU, правий верхній блок CU, лівий нижній блок CU, а потім правий нижній блок CU у вказаному порядку. Під час виконання операції кодування на виділеному блоці CU відеокoder 20 може кодувати блоки CU, зв'язані з підблоками відеоблока виділеного блока CU, згідно з порядком z-сканування. Іншими словами, відеокoder 20 може кодувати блок CU, зв'язаний з лівим нижнім підблоком, а потім блок CU, зв'язаний з правим нижнім підблоком у вказаному порядку.

В результаті кодування блоків CU деревоподібного блока згідно з порядком z-сканування можуть виявитися кодованими блоки CU вище, вище і лівіше, вище і правіше, лівіше і нижче і лівіше від даного блока CU. Не кодованими залишаться блоки CU нижче праворуч від даного блока CU. Потім відеокoder 20 одержує можливість доступу до інформації, створеної за допомогою кодування деяких блоків CU, які є сусідніми по відношенню до даного блока CU, при кодуванні даного блока CU. Однак відеокoder 20 може виявитися нездатним звертатися до

інформації, створеної шляхом кодування інших блоків CU, які є сусідніми по відношенню до даного блока CU, при кодуванні даного блока CU.

Коли відеокодер 20 виконує кодування неподіленого блока CU, відеокодер 20 може створити один або більше блоків прогнозування (PU) для цього блока CU. Кожний з блоків PU блока CU може бути зв'язаний з іншим відеоблоком у відеоблоці блока CU. Відеокодер 20 може створити прогнозований відеоблок для кожного блока PU блока CU. Прогнозований відеоблок блока PU може являти собою блок відліків. Відеокодер 20 може використовувати внутрішньокадрове прогнозування або міжкадрове прогнозування для створення прогнозованого відеоблока для PU.

Коли відеокодер 20 використовує внутрішньокадрове прогнозування для створення прогнозованого відеоблока блока PU, відеокодер 20 може створити прогнозований відеоблок блока PU на основі декодованих відліків зображення, зв'язаних з даним блоком PU. Якщо відеокодер 20 використовує внутрішньокадрове прогнозування для створення прогнозованих відеоблоків блока PU блока CU, блок CU є блоком CU з внутрішньокадровим прогнозуванням. Коли відеокодер використовує міжкадрове прогнозування для створення прогнозованого відеоблока блока PU, відеокодер 20 може створити прогнозований відеоблок блока PU на основі декодованих відліків одного або більше зображень, відмінних від зображення, зв'язаного з даним PU. Якщо відеокодер 20 використовує міжкадрове прогнозування для створення прогнозованих відеоблоків блоків PU блока CU, то блок CU являє собою блок CU з міжкадровим прогнозуванням.

Крім того, коли відеокодер 20 використовує міжкадрове прогнозування для створення прогнозованого відеоблока для блока PU, відеокодер 20 може створити інформацію про рух для даного блока PU. Інформація про рух для блока PU може вказувати один або більше опорних блоків блока PU. Кожний опорний блок блока PU може являти собою відеоблок в опорному зображенні. Опорним зображенням може бути зображення, відмінне від зображення, зв'язаного з блоком PU. У деяких випадках опорний блок блока PU також може називатися "опорним зразком" блока PU. Відеокодер 20 може створити прогнозований відеоблок для блока PU на основі опорних блоків даного PU.

Після створення відеокодером 20 прогнозованих відеоблоків для одного або більше PU блока CU відеокодер 20 може створити залишкові дані для блока CU на основі прогнозованих відеоблоків для блоків PU блока CU. Залишкові дані для блока CU можуть вказувати різницю між відліками в прогнозованих відеоблоках для блоків PU блока CU і вихідному відеоблоці блока CU.

Крім того, як частину процесу виконання операції кодування на неподіленому блоці CU, відеокодер 20 може виконати рекурсивне розбиття по квадродереву на залишкових даних блока CU для розділення залишкових даних блока CU на один або більше блоків залишкових даних (тобто, залишкових відеоблоків), зв'язаних з блоками перетворення (TU) блока CU. Кожний блок TU блока CU може бути зв'язаний з іншим залишковим відеоблоком.

Відеокодер 20 може застосувати одне або більше перетворень до залишкових відеоблоків, зв'язаних з блоками TU, для створення блоків коефіцієнтів перетворення, зв'язаних з блоками TU. Концептуально блок коефіцієнтів перетворення може являти собою двовірну (2D) матрицю коефіцієнтів перетворення.

Після створення блока коефіцієнтів перетворення відеокодер 20 може реалізувати процес квантування на блоці коефіцієнтів перетворення. Звичайно квантування відноситься до процесу, в якому виконується квантування коефіцієнтів перетворення з метою можливого скорочення обсягу даних, які використовуються для представлення коефіцієнтів перетворення, що забезпечує додаткове стиснення. Процес квантування може зменшити глибину в бітах, зв'язану з деякими або усіма коефіцієнтами перетворення. Наприклад, n-бітовий коефіцієнт перетворення можна округлити до m-бітового коефіцієнта перетворення в процес квантування, де n більше m.

Відеокодер 20 може зв'язати кожний блок CU зі значенням параметра квантування (QP). Значення QP, зв'язане з блоком CU, може визначити, яким чином відеокодер 20 виконує квантування блоків коефіцієнтів перетворення, зв'язаних з даним блоком CU. Відеокодер 20 може відрегулювати ступінь квантування, що застосовується до блоків коефіцієнтів перетворення, зв'язаних з блоком CU, шляхом регулювання значення QP, зв'язаного з даним блоком CU.

Після квантування блока коефіцієнтів перетворення відеокодером 20 він створює набори синтаксичних елементів, які представляють коефіцієнти трансформації в квантованому блоці коефіцієнтів трансформації. Відеокодер 20 може застосувати операції ентропійного кодування,

такі як операції контекстно-адаптивного бінарного арифметичного кодування (CABAC) до деяких з цих синтаксичних елементів.

Бітовий потік, створений відеокoderом 20, може включати в себе ряд блоків рівня абстракції мережі (NAL). Кожний з блоків NAL може являти собою синтаксичну структуру, що містить індикацію про тип даних в блоці NAL, і байти, що містять дані. Наприклад, блок NAL може містити дані, що представляють набір параметрів послідовності, набір параметрів зображення, кодований слайс, одного або більше повідомлень з додатковою розширеною інформацією (SEI), обмежувач блока доступу, дані фільтра, або інший тип даних. Дані в блоці NAL можуть включати в себе різні синтаксичні структури.

Відеокoder 30 може приймати бітовий потік, створений відеокoderом 20. Цей бітовий потік може включати в себе кодоване представлення відеоданих, кодованих відеокoderом 20. Приймаючи бітовий потік, відеокoder 30 може виконати операцію синтаксичного аналізу цього бітового потоку. Під час виконання операції синтаксичного аналізу відеокoder 30 може виділити синтаксичні елементи з бітового потоку. Відеокoder 30 може відновити зображення відеоданих на основі синтаксичних елементів, витягнутих з бітового потоку. Процес відновлення відеоданих на основі синтаксичних елементів у загальному випадку є оберненим по відношенню до процесу, що виконується відеокoderом 20 для створення синтаксичних елементів.

Після витягання відеокoderом 30 синтаксичних елементів, зв'язаних з блоком CU, відеокoder 30 може створити прогнозовані відеоблоки для блоків PU блока CU на основі цих синтаксичних елементів. Додатково, відеокoder 30 може виконати обернене квантування блоків коефіцієнтів перетворення, зв'язаних з блоками TU даного блока CU. Відеокoder 30 може виконати обернене перетворення блоків коефіцієнтів перетворення для відновлення залишкових відеоблоків, зв'язаних з блоками TU даного блока CU. Після створення прогнозованих відеоблоків і відновлення залишкових відеоблоків відеокoder 30 може відновити відеоблок блока CU на основі прогнозованих відеоблоків і залишкових відеоблоків. Відеокoder 30 може таким шляхом відновити відеоблоки блоків CU на основі вказаних синтаксичних елементів у бітовому потоці.

Відеокoder 20 може розділити поточне зображення на множину розділів зображення. Розділи зображення можуть бути зв'язані з неперекривними піднаборами деревоподібних блоків поточного зображення. Відеокoder 20 може розділити поточне зображення на множину розділів зображень різними шляхами. Як описано нижче, відеокoder 20 може розділити поточне зображення на множину тайлів або множину хвиль хвильової паралельної обробки (WPP). Термін "розділ зображення", що використовується в даному винаході, відноситься загалом як до тайлів, так і до хвиль WPP. Процес розділення поточного зображення на розділи зображення може називатися "розбиттям" поточного зображення на розділи зображення.

Як згадувалося вище, відеокoder 20 може розділити поточне зображення на один або більше тайлів. Кожний тайл може містити ціле число деревоподібних блоків у поточному зображенні. Відеокoder 20 може розділити поточне зображення на тайли шляхом визначення двох або більше вертикальних меж тайла і двох або більше горизонтальних меж тайла. Кожна вертикальна сторона поточного зображення може розглядатися як вертикальна межа тайла. Кожну горизонтальну сторону поточного зображення можна вважати горизонтальною межею тайла. Наприклад, якщо відеокoder 20 визначає чотири вертикальні межі тайлів і три горизонтальних межі тайлів для поточного зображення, то поточне зображення ділиться на шість тайлів.

Відеокoder, такий як відеокoder 20 або відеокoder 30, може кодувати тайли поточного зображення згідно з порядком растрового сканування. Крім того, при кодуванні тайла відеокoderом відеокoder може закодувати кожний деревоподібний блок у даному тайлі згідно з порядком растрового сканування. Відеокoder може таким шляхом закодувати кожний деревоподібний блок даного тайла поточного зображення, перш ніж кодувати який-небудь деревоподібний блок іншого тайла цього зображення. Отже, порядок кодування відеокoderом деревоподібних блоків поточного зображення, може відрізнятися, коли відеокoder розділяє поточне зображення на множину тайлів, від випадку, коли відеокoder не розділяє поточне зображення на множину тайлів.

Крім того, в деяких випадках відеокoder може використовувати інформацію, зв'язану із сусідніми в просторі блоками CU для виконання внутрішньокadрового прогнозування на даному блоці CU в поточному зображенні, якщо даний блок CU і сусідні в просторі блоки CU відносяться до одного і того самого тайлу. Сусідні в просторі блоки CU - це блоки CU, що відносяться до поточного слайсу поточного зображення. У деяких випадках відеокoder може використовувати інформацію, зв'язану із сусідніми в просторі блоками CU, для вибору контексту для кодування синтаксичного елемента даного блока CU за алгоритмом CABAC, якщо даний

блок CU і сусідні в просторі блоки CU знаходяться в одному і тому самому тайлі. Через вказані обмеження відеокoder здатний виконувати паралельне кодування деревоподібних блоків множини тайлів.

В інших прикладах відеокoder може закодувати поточне зображення, використовуючи хвильову паралельну обробку (WPP). При кодуванні поточного зображення з використанням WPP відеокoder може розділити деревоподібні блоки поточного зображення на множину "хвиль" WPP. Кожна хвиля WPP може відповідати своєму рядку деревоподібних блоків у поточному зображенні. При кодуванні поточного зображення з використанням WPP відеокoder може почати кодування з верхнього рядка деревоподібних блоків. Закодувавши два або більше деревоподібних блоків верхнього рядка, відеокoder може почати кодувати другий (зверху) рядок деревоподібних блоків паралельно з кодуванням верхнього рядка деревоподібних блоків. Закодувавши два або більше деревоподібних блоків з другого рядка зверху, відеокoder може почати кодувати третій зверху рядок деревоподібних блоків паралельно з кодуванням вищележачих рядків деревоподібних блоків. Такий порядок кодування може продовжуватися до нижніх рядків деревоподібних блоків у поточному зображенні.

При кодуванні відеокoderом поточного зображення з використанням WPP відеокoder може використовувати інформацію, зв'язану із сусідніми в просторі блоками CU поза поточним деревоподібним блоком, для виконання внутрішньокadрового прогнозування на даному блоці CU в поточному деревоподібному блоці, доки є сусідні в просторі блоки CU зліва, зліва і вище, вище або праворуч і вище від поточного деревоподібного блока. Якщо поточним деревоподібним блоком є крайній лівий деревоподібний блок в рядку, що не є найверхнім рядком, відеокoder може використовувати інформацію, зв'язану з другим деревоподібним блоком безпосередньо вищележачого рядка, для вибору контексту для кодування синтаксичного елемента поточного деревоподібного блока згідно з алгоритмом CABAC. В іншому випадку, якщо поточний деревоподібний блок не є крайнім лівим деревоподібним блоком у даному рядку, то відеокoder може використовувати інформацію, зв'язану з деревоподібним блоком, що знаходиться зліва від поточного деревоподібного блока, для вибору контексту для кодування синтаксичного елемента поточного деревоподібного блока, використовуючи алгоритм CABAC. Таким чином, відеокoder може використовувати стани CABAC рядка на основі станів CABAC, що відносяться до безпосередньо вищележачого рядка, після кодування двох або більше деревоподібних блоків безпосередньо вищележачого рядка.

У деяких прикладах при кодуванні відеокoderом поточного зображення з використанням WPP, горизонтальними і вертикальними межами поточного зображення є тільки межі тайлів поточного зображення. Таким чином, тільки тайл поточного зображення може мати той самий розмір, що і поточне зображення. Відеокoder може розділити поточне зображення, а означає, один тайл поточного зображення на множину хвиль WPP.

Як згадувалося вище, відеокoder 20 може створити блок NAL кодованого слайса, який включає в себе кодоване представлення слайса. Слайс може бути зв'язаний з цілим числом послідовно закодованих деревоподібних блоків. Блок NAL кодованого слайса може включати в себе заголовок слайса і дані слайса. Дані слайса можуть включати в себе кодовані представлення кожного деревоподібного блока, зв'язаного з цим слайсом. Відеокoder 20 може створити блок NAL кодованого слайса, так щоб вказані кодовані представлення деревоподібних блоків групувалися в даних слайса в сегментах згідно з розділами зображення, до яких належать дані деревоподібні блоки. Наприклад, кожний блок NAL кодованого слайса може включати в себе кодований деревоподібний блок, зв'язаний з першим розділом зображення, за яким йде деревоподібний блок, зв'язаний з другим розділом зображення, за яким йде кодований деревоподібний блок, зв'язаний з третім розділом зображення, тощо.

Згідно з розкритими тут методиками відеокoder 20 може заповнити один або більше сегментів незначущою інформацією, так щоб кожний сегмент починався на межі байта. Блок NAL кодованого слайса може бути розділений на ряд байтів. Сегмент може починатися на межі байта, коли його перший біт є першим бітом одного з байтів блока NAL кодованого слайса. Крім того, сегмент може бути вирівняний по байтах, якщо перший біт сегмента є першим бітом одного з байтів блока NAL кодованого слайса. При заповненні відеокoderом сегмента відеокoder 20 може прикласти до цього сегмента заповнюючі біти. Наприклад, відеокoder 20 може додати один або більше бітів заповнення до сегмента таким чином, щоб кількість бітів у сегменті ділилася на вісім без залишку. Біти заповнення не зобов'язані мати семантичне значення; вони служать для того, щоб забезпечити початок наступного сегмента з межі байта.

Одержавши блок NAL кодованого слайса, відеокoder 30 може запам'ятати цей блок NAL кодованого слайса в пам'яті. Для паралельного декодування розділів зображення відеокoder 30 може розподілити сегменти по різних потоках декодування, які виконуються паралельно.

Щоб розподілити сегменти по різних потоках декодування, відеодекодеру 30 буде потрібно вказати адреси пам'яті, зв'язані з початком кожного із сегментів. Відеодекодер 30 може використовувати побайтову адресацію пам'яті. Відповідно, відеодекодер 30 можливо не зможе вказати адресу пам'яті, зв'язану з початком сегмента, якщо початок сегмента виявився всередині байта. Отже, відеодекодер 30 не зможе в цьому випадку виконати паралельне декодування кодованих деревоподібних блоків у сегменті, якщо один або більше сегментів починаються всередині байта. Як альтернатива, відеодекодер 30 може використовувати побітову адресацію пам'яті або побайтову плюс побітову адресацію, щоб мати можливість паралельного декодування деревоподібних блоків у вказаних сегментах, але при цьому ускладнюється технічна реалізація та обчислення.

Відеокодер 20 може вказаним чином розділити зображення на множину розділів зображення. Зображення містить множину деревоподібних блоків. Розділи зображення зв'язані з неперекривними піднаборами деревоподібних блоків зображення. Відеокодер 20 може створити блок NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, які зв'язані зі слайсом зображення. Кодовані представлення деревоподібних блоків у блоці NAL кодованого слайса групуються в сегменти, зв'язані зі своїми розділами зображення. Один або більше сегментів заповнюють незначущою інформацією, так щоб кожний сегмент починався на межі байта.

Крім того, відеокодер 30 може запам'ятати блок NAL кодованого слайса, який включає в себе кодовані представлення деревоподібних блоків, зв'язаних зі слайсом зображення. Зображення можна розділити на множину розділів зображення. Кодовані представлення деревоподібних блоків можна згрупувати в сегменти, кожний з яких буде зв'язаний зі своїм розділом зображення. Один або більше сегментів заповнюють незначущою інформацією з тим, щоб кожний сегмент починався з межі байта. Відеокодер 30 може декодувати кодовані представлення деревоподібних блоків. У деяких випадках відеокодер 30 може декодувати кодовані представлення деревоподібних блоків у двох або більше сегментах паралельно.

На Фіг. 2 представлена блок-схема, що ілюструє зразковий відеокодер 20, сконфігурований для реалізації розкритих тут методик. Фіг. 2 наведена в роз'яснювальних цілях, і її не треба розглядати як обмеження вказаних методик, широко представлених та описаних тут як приклади. З метою роз'яснення винаходу відеокодер 20 описаний тут в контексті кодування згідно зі стандартом HEVC. Однак розкриті тут методики можна застосувати до інших стандартів або методів кодування.

У прикладі за Фіг. 2 відеокодер 20 включає в себе множину функціональних компонент. Функціональні компоненти відеокодера 20 включають в себе модуль 100 прогнозування, модуль 102 створення даних залишку, модуль 104 перетворення, модуль 106 квантування, модуль 108 оберненого квантування, модуль 110 оберненого перетворення, модуль 112 відновлення, модуль 113 фільтрів, буфер 114 декодованих зображень і модуль 116 ентропійного кодування. Модуль 100 прогнозування включає в себе модуль 121 міжкадрового прогнозування, модуль 122 оцінки руху, модуль 124 компенсації руху і модуль 126 внутрішньоканалового прогнозування. В інших прикладах відеокодер 20 може включати в себе меншу або більшу кількість функціональних компонент або інші функціональні компоненти. Крім того, модуль 122 оцінки руху і модуль 124 компенсації руху можуть бути виконані як єдиний модуль з високим ступенем інтеграції, але вони представлені в прикладі на Фіг. 2 окремо з метою пояснення їх функціонування.

Відеокодер 20 може приймати відеодані. Прийом відеоданих відеокодером 20 може здійснюватися від різних джерел. Наприклад, відеокодер 20 може приймати відеодані від джерела 18 відео (Фіг. 1) або іншого джерела. Відеодані можуть представляти ряд зображень. Для кодування відеоданих відеокодер 20 може виконати операцію кодування для кожного зображення. Як частину процесу виконання операції кодування зображення відеокодер 20 може виконати операції кодування на кожному слайсі даного зображення. Як частину виконання операції кодування на слайсі відеокодер 20 може виконати операції кодування на деревоподібних блоках у слайсі.

Як частину виконання операції кодування на деревоподібному блоці модуль 100 прогнозування може виконати розбиття на основі квадродерева на відеоблоці деревоподібного блока для розділення відеоблока на відео блоки, що зменшуються з кожним розбиттям. Кожний з відеоблоків меншого розміру може бути зв'язаний зі своїм блоком CU. Наприклад, модуль 100 прогнозування може розбити відеоблок деревоподібного блока на чотири підблоки однакового розміру, розбити один або більше з цих підблоків на чотири елементи підблока однакового розміру тощо.

Розміри відеоблоків, зв'язаних з блоками CU, можуть знаходитися в діапазоні від 8×8 відліків до розміру деревоподібного блока, що має максимум 64×64 відліків або більше. У даному описі вираз " $N \times N$ " та " N на N " можуть використовуватися як взаємозамінні для посилання на розмірності відліків відеоблока по вертикалі та горизонталі, наприклад, " 16×16 " відліків або " 16 на 16 " відліків. У загальному випадку відеоблок 16×16 має шістнадцять відліків у вертикальному напрямку ($y=16$) і шістнадцять відліків у горизонтальному напрямку ($x=16$). Аналогічним чином, блок $N \times N$ звичайно має N відліків у вертикальному напрямку і N відліків у горизонтальному напрямку, де N представляє ненегативне ціле число.

Крім того, як частину виконання операції кодування на деревоподібному блоці, модуль 100 прогнозування може створити структуру даних у вигляді ієрархічного квадродрева для даного деревоподібного блока. Наприклад, деревоподібний блок може відповідати кореневому вузлу структури даних у вигляді квадродрева. Якщо модуль 100 прогнозування розбиває відеоблок деревоподібного блока на чотири підблоки, то цей кореневий вузол буде мати чотири дочірніх вузли в структурі даних у вигляді квадродрева. Кожний з дочірніх вузлів відповідає блоку CU, зв'язаному з одним з підблоків. Якщо модуль 100 прогнозування розбиває один з підблоків на чотири елементи підблока, то вузол, що відповідає блоку CU, зв'язаному з даним підблоком, може мати чотири дочірніх вузли, кожний з яких відповідає блоку CU, зв'язаному з одним з елементів підблоків.

Кожний вузол структури даних у вигляді квадродрева може містити синтаксичні дані (наприклад, синтаксичні елементи) для відповідного деревоподібного блока або блока CU. Наприклад, вузол в квадродреві може включати в себе прапор розщеплення, який вказує, чи розбитий (тобто, чи розщеплений) відеоблок блока CU, що відповідає даному вузлу, на чотири підблоки. Синтаксичні елементи для блока CU можна визначити рекурсивно, і вони будуть залежати від того, чи розщеплений даний відеоблок блока CU на підблоки. Блок CU, чий відеоблок не розщеплений, може відповідати кінцевому вузлу в структурі даних типу квадродрева. Кодований деревоподібний блок може включати в себе дані на основі структурних даних типу квадродрева для відповідного деревоподібного блока.

Відеокодер 20 може виконати операції кодування на кожному неподіленому блоці CU деревоподібного блока. Під час виконання відеокодером 20 операції кодування на неподіленому блоці CU відеокодер 20 створює дані, що представляють кодоване представлення неподіленого блока CU.

Як частину виконання операції кодування на блоці CU модуль 100 прогнозування може розбити відеоблок блока CU по декількох (в граничному випадку по одному) блоках PU блока CU. Відеокодер 20 і відеодекодер 30 можуть підтримувати різні розміри блока PU. Якщо припустити, що розмір конкретного блока CU становить $2N \times 2N$, то відеокодер 20 і відеодекодер 30 можуть підтримувати розміри PU, що становлять $2N \times 2N$ або $N \times N$, і міжкадрове прогнозування для симетричних розмірів PU, що становлять $2N \times 2N$, $2N \times N$, $N \times 2N$, $2N \times N$, $N \times N$, $nL \times 2N$, $nR \times 2N$ або т.п. Відеокодер 20 і відеодекодер 30 також можуть підтримувати асиметричне розбиття для PU з розмірами, що становлять $2N \times nU$, $2N \times nD$, $nL \times 2N$ та $nR \times 2N$. У деяких прикладах модуль 100 прогнозування може виконувати геометричне розбиття з метою розбиття відеоблока блока CU між блоками PU даного блока CU вздовж межі, яка не перетинається зі сторонами відеоблока блока CU під прямими кутами.

Модуль 121 міжкадрового прогнозування може виконати міжкадрове прогнозування на кожному блоці PU блока CU. Міжкадрове прогнозування може забезпечити часове стиснення. Для виконання міжкадрового прогнозування на блоці PU модуль 122 оцінки руху може створити інформацію про рух для блока PU. Модуль 124 компенсації руху може створити прогнозований відеоблок для блока PU на основі інформації про рух і декодованих відліків зображень, відмінних від зображення, зв'язаного з даним блоком CU (тобто, опорні зображення). У даному описі прогнозований відеоблок, створений модулем 124 компенсації руху, може називатися відеоблоком з міжкадровим прогнозуванням.

Слайси можуть бути І слайсами, Р слайсами або В слайсами. Модуль 122 оцінки руху і модуль 124 компенсації руху можуть виконати різні операції для блока PU блока CU залежно від того, в слайсі якого типу (І, Р або В) знаходиться даний блок PU. У І слайсі всі блоки PU одержані з використанням внутрішньокadroвого прогнозування. Отже, якщо блок PU знаходиться в І слайсі, то модуль 122 оцінки руху і модуль 124 компенсації руху не виконують міжкадрове прогнозування на даному блоці PU.

Якщо блок PU знаходиться в Р слайсі, то зображення, яке містить цей бік PU, зв'язане зі списком опорних зображень, який називається "список 0". Кожне з опорних зображень в списку 0 містить відліки, які можна використовувати для міжкадрового прогнозування інших зображень. Під час виконання модулем 122 оцінки руху операції оцінки руху відносно блока PU в Р слайсі,

модуль 122 оцінки руху може здійснювати пошук опорних зображень у списку 0 для опорного блока для даного блока PU. Опорний блок блока PU може являти собою набір відліків, наприклад, блок відліків, який ближче за всіх відповідає відлікам у відеоблоці блока PU. Модуль 122 оцінки руху може використовувати найрізноманітніші метрики для визначення того, наскільки близько набір відліків в опорному зображенні, відповідає відлікам у відеоблоці блока PU. Наприклад, модуль 122 оцінки руху може визначити, наскільки близько набір відліків в опорному зображенні відповідає відлікам у відеоблоці блока PU, використовуючи підхід на основі суми абсолютних різниць (SAD), суми квадратів відхилень (SSD) або інших показників відхилень.

Після ідентифікації опорного блока для блока PU в Р слайсі модуль 122 оцінки руху може створити опорний індекс, який вказує опорне зображення в списку 0, що містить даний опорний блок, і вектор руху, який вказує просторове зміщення між блоком PU та опорним блоком. У різних прикладах модуль 122 оцінки руху може створювати вектори руху із змінною точністю. Наприклад, модуль 122 оцінки руху може створити вектори руху з точністю до чверті відліку, з точністю одна восьма відліку або з іншою дробовою точністю. У випадку дробової точності значення опорного блока можна інтерполювати, виходячи з цілочисельних значень відліків позицій в опорному зображенні. Модуль 122 оцінки руху може видати опорний індекс і вектор руху як інформацію про рух блока PU. Модуль 124 компенсації руху може створити прогнозований відеоблок блока PU на основі опорного блока, ідентифікованого інформацією про рух блока PU.

Якщо блок PU знаходиться в В слайсі, то зображення, яке містить цей блок PU, може бути зв'язане з двома списками опорних зображень, які називаються "список 0" та "список 1". У деяких прикладах зображення, що містить В слайс, може бути зв'язане з комбінацією списків, що є комбінацією списку 0 та списку 1.

Крім того, якщо блок PU знаходиться у В слайсі, то модуль 122 оцінки руху може виконати однонаправлене прогнозування або двонаправлене прогнозування для даного блока PU. Під час виконання модулем 122 оцінки руху однонаправленого прогнозування для блока PU цей модуль може виконати пошук опорних зображень зі списку 0 або зі списку 1 для опорного блока для блока PU. Потім модуль 122 може створити опорний індекс, який вказує опорне зображення в списку 0 або списку 1, що містить опорний блок, і вектор руху, який вказує просторове зміщення між блоком PU та опорним блоком. Модуль 122 оцінки руху може видати опорний індекс, індикатор напрямку прогнозування і вектор руху як інформацію про рух, що стосується блока PU. Індикатор напрямку прогнозування може вказати, чи вказує опорний індекс на опорне зображення в списку 0 або в списку 1. Модуль 124 компенсації руху може створити прогнозований відеоблок блока PU на основі опорного блока, вказаного в інформації про рух, що стосується блока PU.

Коли модуль 122 оцінки руху виконує двонаправлене прогнозування для PU, модуль 122 оцінки руху може здійснювати пошук опорних зображень в списку 0 для опорного блока для даного PU, а також може здійснювати пошук опорних зображень в списку 1 для іншого опорного блока для блока PU. Потім модуль 122 оцінки руху створює опорні індекси, які вказують опорні зображення в списку 0 і в списку 1, що містять опорні блоки і вектори руху, які вказують просторові зміщення між опорними блоками і блоком PU. Модуль 122 оцінки руху може видавати опорні індекси і вектори руху блока PU як інформацію про рух блока PU. Модуль 124 компенсації руху може створити прогнозований відеоблок блока PU на основі опорних блоків, вказаних в інформації про рух блока PU. У деяких випадках модуль 122 оцінки руху не видає повний набір інформації про рух для блока PU в модуль 116 ентропійного кодування. Навпаки, модуль 122 оцінки руху може повідомити інформацію про рух, що відноситься до блока PU, з посиланням на інформацію про рух, що відноситься до іншого блока PU. Наприклад, модуль 122 оцінки руху може визначити, що інформація про рух для даного PU подібна до інформації про рух для сусіднього блока PU. У цьому прикладі модуль 122 оцінки руху може вказати (в синтаксичній структурі, зв'язаній з даним PU) значення, яке вказує відеодекодеру 30, що даний блок PU має таку саму інформацію про рух, як сусідній блок PU. В іншому прикладі, модуль 122 оцінки руху може ідентифікувати (в синтаксичній структурі, зв'язаній з даним блоком PU) сусідній блок PU і різницю їх векторів руху (MVD). Різниця векторів руху вказує різницю між вектором руху блока PU і вектором руху вказаного сусіднього блока PU. Відеодекодер 30 може використовувати вектор руху вказаного сусіднього PU і різницю векторів руху для визначення вектора руху даного PU. Звернувшись до інформації про рух першого блока PU при сигналізації про інформацію про рух другого блока PU, відеокодер 20 має можливість повідомляти інформацію про рух для другого блока PU, використовуючи меншу кількість бітів.

Як частину виконання операції кодування на блоці CU модуль 126 внутрішньокадрового прогнозування може виконати внутрішньокадрове прогнозування на блоках PU блока CU. Внутрішньокадрове прогнозування може забезпечити просторове стиснення. Під час виконання внутрішньокадрового прогнозування на блоці PU модуль 126 внутрішньокадрового прогнозування може створити дані прогнозування для блока PU на основі декодованих відліків інших блоків PU в тому самому зображенні. Дані прогнозування для блока PU можуть включати в себе прогнозований відеоблок і різні синтаксичні елементи. Модуль 126 внутрішньокадрового прогнозування може виконати внутрішньокадрове прогнозування на блоках PU в I слайсах, P слайсах і B слайсах.

Для виконання внутрішньокадрового прогнозування на PU модуль 126 внутрішньокадрового прогнозування може використовувати множину режимів внутрішньокадрового прогнозування для створення множини наборів даних прогнозування для блока PU. При використанні режиму внутрішньокадрового прогнозування для створення набору даних прогнозування для даного блока PU модуль 126 внутрішньокадрового прогнозування може поширити відліки з відеоблоків сусідніх блоків PU через відеоблок даного блока PU в напрямку і/або за градієнтом, зв'язаним з режимом внутрішньокадрового прогнозування. Сусідні блоки PU можуть знаходитися вище, вище і правіше, вище і лівіше або лівіше даного блока PU, якщо припустити, що кодування для блоків PU, CU і деревоподібних блоків виконується в порядку зліва направо, зверху вниз. Модуль 126 внутрішньокадрового прогнозування може використовувати різну кількість режимів внутрішньокадрового прогнозування, наприклад, 33 режими направленої внутрішньокадрового прогнозування залежно від розміру даного блока PU.

Модуль 100 прогнозування може вибрати дані прогнозування для PU з числа даних прогнозування, створених модулем 124 компенсації руху для даного блока PU, або дані прогнозування, створені модулем 126 внутрішньокадрового прогнозування для даного блока PU. У деяких прикладах модуль 100 прогнозування вибирає дані прогнозування для даного блока PU на основі показника "витрата бітів/спотворення" наборів даних прогнозування.

Якщо модуль 100 прогнозування вибирає дані прогнозування, створені модулем 126 внутрішньокадрового прогнозування, він може просигналізувати про режим внутрішньокадрового прогнозування, використаний ним для створення даних прогнозування для блоків PU, тобто, про вибраний режим внутрішньокадрового прогнозування. Модуль 100 прогнозування може виконати сигналізацію про вибраний режим внутрішньокадрового прогнозування різними шляхами. Наприклад, досить імовірно, що вибраний режим внутрішньокадрового прогнозування співпадає з режимом внутрішньокадрового прогнозування сусіднього блока PU. Іншими словами, режим внутрішньокадрового прогнозування сусіднього блока PU може бути найбільш імовірним режимом для поточного блока PU. Таким чином, модуль 100 прогнозування може створити синтаксичний елемент для вказівки про те, що вибраний режим внутрішньокадрового прогнозування співпадає з режимом внутрішньокадрового прогнозування сусіднього блока PU.

Після вибору модулем 100 прогнозування даних прогнозування для блоків PU блока CU модуль 102 створення залишкових даних може створити залишкові дані для блока CU шляхом віднімання прогнозованих відеоблоків блоків PU блока CU з відеоблока блока CU. Залишкові дані блока CU можуть включати в себе залишкові відеоблоки в форматі 2D, які відповідають різним компонентам відліків у відеоблоці блока CU. Наприклад, залишкові дані можуть включати в себе залишковий відеоблок, який відповідає різницям між компонентами яскравості відліків у прогнозованих відеоблоках блоків PU блока CU і компонентами яскравості відліків у вихідному відеоблоці блока CU. Додатково, залишкові дані блока CU можуть включати в себе залишкові відеоблоки, які відповідають різницям між компонентами кольоровості відліків у прогнозованих відеоблоках блоків PU блока CU і компонентами кольоровості відліків у вихідному відеоблоці блока CU.

Модуль 100 прогнозування може виконати розбиття по квадродереву для розбиття залишкових відеоблоків блока CU на підблоки. Кожний неподілений залишковий відеоблок може бути зв'язаний з іншим блоком TU блока CU. Розміри і позиції залишкових відеоблоків, зв'язаних з блоками TU блока CU, можуть залежати, а можуть і не залежати від розмірів і позицій відеоблоків, зв'язаних з блоками PU блока CU. Деревоподібна структура, відома як "залишкове квадродререво" (RQT), може включати в себе режими, зв'язані з кожним із залишкових відеоблоків. Блоки TU блока CU можуть відповідати кінцевим вузлам дерева RQT.

Модуль 104 перетворення може створити один або більше блоків коефіцієнтів перетворення для кожного блока TU блока CU шляхом застосування одного або більше перетворень до залишкового відеоблока, зв'язаного з вказаним блоком TU. Кожний блок коефіцієнтів перетворення може являти собою 2D матрицю коефіцієнтів перетворення. Модуль 104

перетворення може застосувати різні перетворення до залишкового відеоблока, зв'язаного з блоком TU. Наприклад, модуль 104 перетворення може застосувати дискретне косинусне перетворення (DCT), направлене перетворення або концептуально подібне перетворення до залишкового відеоблока, зв'язаного з TU.

5 Після створення модулем 104 перетворення блока коефіцієнтів перетворення, зв'язаного з TU, модуль 106 квантування може виконати квантування коефіцієнтів перетворення в блоці коефіцієнтів перетворення. Модуль 106 квантування може виконати квантування блока коефіцієнтів перетворення, зв'язаного з TU блока CU, на основі значення QP, зв'язаного з даним блоком CU.

10 Відеокодер 20 може зв'язати значення QP з блоком CU різними шляхами. Наприклад, відеокодер 20 може виконати аналіз "витрата бітів-спотворення" на деревоподібному блоці, зв'язаному з CU. При аналізі "витрата бітів-спотворення" відеокодер 20 може створити множину кодованих представлень деревоподібного блока шляхом багаторазового виконання операції кодування на деревоподібному блоці. Відеокодер 20 може зв'язати різні значення QP з блоком CU при створенні різних кодованих представлень даного деревоподібного блока. Відеокодер 20 може просигналізувати про те, що дане значення QP зв'язане з CU, коли дане значення QP зв'язане з CU в кодованому представленні деревоподібного блока, що має мінімальний показник "витрата бітів-спотворення".

20 Модуль 108 оберненого квантування і модуль 110 оберненого перетворення можуть застосувати обернене квантування та обернене перетворення до блока коефіцієнтів перетворення відповідно для відновлення залишкового відеоблока з блока коефіцієнтів перетворення. Модуль 112 відновлення може додати відновлений залишковий відеоблок до відповідних відліків з одного або більше прогнозованих відеоблоків, створених модулем 100 прогнозування, для створення відновленого відеоблока, зв'язаного з TU. Відновивши таким 25 шляхом відеоблоки для кожного TU блока CU, відеокодер 20 може відновити відеоблок блока CU.

Після того, як модуль 112 відновлення відновить відеоблок блока CU, модуль 13 фільтрів може виконати операцію деблокування для заглушення артефактів блокування у відеоблоці, зв'язаному з даним блоком CU. Після виконання однієї або більше операцій деблокування 30 модуль 113 фільтрів може запам'ятати відновлений відеоблок блока CU в буфері 114 декодованих зображень. Модуль 122 оцінки руху і модуль 124 компенсації руху можуть використовувати опорне зображення, що містить відновлений відеоблок, для виконання міжкадрового прогнозування на блоках PU подальших зображень. Додатково, модуль 126 внутрішньокіадрового прогнозування може використовувати відновлені відеоблоки в буфері 114 35 декодованих зображень для виконання внутрішньокіадрового прогнозування на інших блоках PU в тому самому зображенні, що і CU.

Модуль 116 ентропійного кодування може приймати дані від інших функціональних компонент відеокодера 20. Наприклад, модуль 116 ентропійного кодування може приймати 40 блоки коефіцієнтів перетворення від модуля 106 квантування і може приймати синтаксичні елементи від модуля 100 прогнозування. Одержавши вказані дані, модуль 116 ентропійного кодування може виконати одну або більше операцій ентропійного кодування для створення кодованих даних. Наприклад, відеокодер 20 може виконати операцію контекстно-залежного адаптивного кодування із змінною довжиною слова (CAVLC), операцію контекстно-адаптивного бінарного арифметичного кодування (CABAC), операцію кодування з перетворенням слів змінної довжини (V2V), операцію синтаксично орієнтованого адаптивного бінарного арифметичного кодування (SBAC), операцію імовірнісного ентропійного кодування з 45 інтервальним розбиттям (PIPE), або операцію ентропійного кодування іншого типу на цих даних. Модуль 116 ентропійного кодування може вивести бітовий потік, який включає в себе дані, закодовані з використанням ентропійного кодування.

50 Як частину виконання операції ентропійного кодування на даних модуль 116 ентропійного кодування може вибрати контекстну модель. Якщо модуль 116 ентропійного кодування виконує операцію CABAC, то ця контекстна модель може вказати оцінки імовірностей того, що конкретні біни мають конкретні значення. У контексті алгоритму CABAC термін "бін" використовують для звернення до біта бінарної версії синтаксичного елемента.

55 Відеокодер 20 може створити блок NAL кодованого слайса для кожного слайса поточного зображення. Блок NAL кодованого слайса може включати в себе заголовок слайса і дані слайса. Дані слайса можуть містити множину сегментів. Кожний з цих сегментів включає в себе кодовані деревоподібні блоки, зв'язані з іншим розділом зображення. Відеокодер 20 може заповнити сегменти незначущою інформацією так, що кожен сегмент буде починатися з межі 60 байта в даних слайса. Наприклад, сегменти в блоці NAL кодованого слайса можуть включати в

себе заданий сегмент. У цьому прикладі відеокодер 20 може створити блок NAL кодованого слайса щонайменше частково за допомогою виконання операції заповнення незначущою інформацією, в якій до даного сегмента приєднуються біти, якщо наступний деревоподібний блок знаходиться всередині поточного слайса і зв'язаний з розділом зображення, відмінним від заданого сегмента.

У деяких прикладах відеокодер 20 може створити заголовок слайса блока NAL кодованого слайса, з тим щоб заголовок слайса вказував точки введення для сегментів у даних слайса блока NAL кодованого слайса. Ці точки введення можуть вказувати позиції в даних слайса вказаних сегментів. Наприклад, ці точки введення можуть вказувати байтові зміщення сегментів. У цьому прикладі байтові зміщення можуть відноситися до першого біта блока NAL кодованого слайса, першого біта даних слайса або іншого біта в блоці NAL кодованого слайса. В іншому прикладі вказані точки введення можуть вказувати кількість бітів або байт у кожному із сегментів. У деяких прикладах заголовок слайса не вказує точку введення для першого сегмента в даних слайса.

У деяких прикладах відеокодер 20 може визначити, чи має прапор перше значення (наприклад, 1). Якщо прапор має це перше значення, то відеокодер 20 може заповнити один або більше сегментів незначущою інформацією так, щоб кожний сегмент починався з межі байта. Коли прапор має друге значення (наприклад, 0), відеокодер 20 не заповнює сегменти незначущою інформацією. У результаті ці сегменти можуть починатися або не починатися з позиції, вирівняної по байту. У вказаних прикладах набір параметрів послідовності, набір параметрів зображення, набір параметрів адаптації або заголовок слайса можуть включати в себе вказаний прапор. Таким чином, в деяких прикладах відеокодер 20 може створити набір параметрів, зв'язаний з поточним зображенням, де цей набір параметрів включає в себе прапор. Коли цей прапор має перше значення, то один або більше сегментів заповнені незначущою інформацією, так що ці сегменти починаються з меж байтів. Коли цей прапор має друге значення, ці сегменти можуть починатися, а можуть і не починатися з меж байтів.

Крім того, в деяких прикладах відеокодер 20 може розбити поточне зображення на множину тайлів. Якщо відеокодер 20 допускає в зображенні прогнозування через межі тайла (тобто, коли два або більше тайлів залежать один від одного), відеокодер 20 не виконує вказане заповнення сегментів. У результаті ці сегменти можуть починатися, а можуть і не починатися з позиції, вирівняної по байту. Однак, якщо відеокодер 20 не допускає прогнозування через межі тайла, він може заповнити один або більше сегментів незначущою інформацією таким чином, що кожний сегмент буде починатися з межі байта. Таким чином, відеокодер 20 може створити блок NAL кодованого слайса щонайменше частково шляхом виконання операції заповнення незначущою інформацією, яка забезпечує знаходження початку кожного із сегментів на межі байта, але тільки після визначення того, що згадані тайли не залежать один від одного.

На Фіг. 3 представлена блок-схема, що ілюструє зразковий відеокодер 30, сконфігурований для реалізації розкритих тут методик. Фіг. 3 наведена тільки для роз'яснення і не обмежує вказані методики, широко представлені як приклади і детально описані в даному винаході. Для зручності роз'яснення відеокодер 30 описується тут в контексті кодування згідно зі стандартом HEVC. Однак розкриті тут методики можна застосувати до інших стандартів або способів кодування.

У прикладі за Фіг. 3 відеокодер 30 включає в себе множину функціональних компонент. Ці функціональні компоненти відеокодера 30 включають в себе модуль 150 ентропійного декодування, модуль 152 прогнозування, модуль 154 оберненого квантування, модуль 156 оберненого перетворення, модуль 158 відновлення, модуль 159 фільтрів і буфер 160 декодованих зображень. Модуль 152 прогнозування включає в себе модуль 162 компенсації руху і модуль 164 внутрішньокadroвого прогнозування. У деяких прикладах відеокодер 30 може виконати прогін декодування, який в загальному випадку є оберненим прогону кодування, описаному застосовно до відеокодеру 20 за Фіг. 2. В інших прикладах відеокодер 30 може включати в себе більше або менше функціональних компонент або інші функціональні компоненти.

Відеокодер 30 може приймати бітовий потік, що містить кодовані відеодані. Цей бітовий потік може включати в себе множину синтаксичних елементів. При прийомі відеокодером 30 бітового потоку модуль 150 ентропійного декодування може виконати операцію синтаксичного аналізу на цьому потоці. В результаті виконання операції синтаксичного аналізу на бітовому потоці модуль 150 ентропійного декодування може витягнути синтаксичні елементи з бітового потоку. Як частину виконання операції синтаксичного аналізу модуль 150 ентропійного декодування може виконати ентропійне декодування синтаксичних елементів, кодованих з використанням ентропійного кодування. Модуль 152 прогнозування, модуль 154 оберненого

квантування, модуль 156 оберненого перетворення, модуль 158 відновлення і модуль 159 фільтрів можуть виконати операцію відновлення, яка створює декодовані відеодані на основі синтаксичних елементів, витягнутих з бітового потоку.

Як обговорювалося вище, бітовий потік може містити ряд блоків NAL. Блоки NAL бітового потоку можуть включати в себе блоки NAL набору параметрів послідовності, блоки NAL набору параметрів зображення, блоки SEI NAL тощо. Як частину виконання операції синтаксичного аналізу на бітовому потоці, модуль 150 ентропійного декодування може виконати операції синтаксичного аналізу, які забезпечують витягання та ентропійне декодування наборів параметрів послідовності з блоків NAL параметрів послідовності, наборів параметрів зображення з блоків NAL набору параметрів зображення, даних SEI з блоків SEI NAL тощо.

Додатково, блоки NAL бітового потоку можуть включати в себе блоки NAL кодованого слайса. Як частину виконання операції синтаксичного аналізу на бітовому потоці відеодекодер 30 може виконати операції синтаксичного аналізу, що забезпечують витягання та ентропійне декодування кодованих слайсів з блоків NAL кодованого слайса. Кожний з цих кодованих слайсів може включати в себе заголовок слайса і дані слайса. Заголовок слайса може містити синтаксичні елементи, що належать слайсу. Ці синтаксичні елементи в заголовку слайса можуть включати в себе синтаксичний елемент, що ідентифікує набір параметрів зображення, зв'язаний із зображенням, яке містить даний слайс.

Дані блока NAL кодованого слайса можуть включати в себе множину сегментів. Кожний з цих сегментів може включати в себе кодовані деревоподібні блоки, зв'язані з іншим розділом зображення (наприклад, тайл або хвиля WPP). Один або більше з цих сегментів у даних слайса можуть бути заповнені незначущою інформацією таким чином, що кожний сегмент буде починатися з меж байта. Заголовок слайса блока NAL кодованого слайса може вказувати точки введення для сегментів. У цьому випадку, оскільки сегменти завжди починаються з меж байтів, відеодекодер 30 здатний розподілити простим чином різні сегменти по різних потоках декодування шляхом використання байтової адресації пам'яті. Різні потоки декодування можуть забезпечити синтаксичний аналіз кодованих деревоподібних блоків сегментів і відновити паралельно відеодані, зв'язані з відповідними деревоподібними блоками.

Як частину витягання даних слайса з блоків NAL кодованого слайса модуль 150 ентропійного декодування може виконати операції синтаксичного аналізу, які забезпечують витягання синтаксичних елементів з кодованих блоків CU. Витягнуті синтаксичні елементи можуть включати в себе синтаксичні елементи, зв'язані з блоками коефіцієнтів перетворення. Потім модуль 150 ентропійного декодування може виконати операції декодування CABAC на деяких синтаксичних елементах.

Після виконання модулем 150 ентропійного декодування операції синтаксичного аналізу на неподіленому блоці CU відеодекодер 30 може виконати операцію відновлення на неподіленому блоці CU. Для виконання операції відновлення на неподіленому блоці CU відеодекодер 30 може виконати операцію відновлення на кожному блоці TU блока CU. Виконавши операцію відновлення для кожного TU блока CU, відеодекодер 30 може відновити залишковий відеоблок, зв'язаний з даним CU.

Як частину виконання операції відновлення на блоці TU модуль 154 оберненого квантування може виконати обернене квантування, тобто "деквантування", блока коефіцієнтів перетворення, зв'язане з даним TU. Модуль 154 оберненого квантування може виконати обернене квантування блока коефіцієнтів перетворення, використовуючи процеси, аналогічні процесам оберненого квантування, запропонованим для стандарту HEVC або визначеним стандартом декодування H.264. Модуль 154 оберненого квантування може використовувати параметр QR квантування, обчислений відеокодером 20, для CU блока коефіцієнтів перетворення, щоб визначити ступінь квантування і аналогічним чином ступінь оберненого квантування для модуля 154 оберненого квантування.

Після виконання модулем 154 оберненого квантування операції оберненого квантування для блока коефіцієнтів перетворення модуль 156 оберненого перетворення може створити залишковий відеоблок для TU, зв'язаного з блоком коефіцієнтів перетворення. Модуль 156 оберненого перетворення може застосувати обернене перетворення до вказаного блока коефіцієнтів перетворення, щоб створити залишковий відеоблок для TU. Наприклад, модуль 156 оберненого перетворення може застосувати обернене перетворення DCT, обернене цілочисельне перетворення, обернене перетворення Кархунена-Лева (KLT), обернене поворотне перетворення, обернене направлене перетворення або інше обернене перетворення до блока коефіцієнтів перетворення.

У деяких прикладах модуль 156 оберненого перетворення може визначити обернене перетворення для застосування до блока коефіцієнтів перетворення на основі сигналізації з

боку відеокодера 20. У вказаних прикладах модуль 156 оберненого перетворення може визначити таке обернене перетворення на основі одержаного в результаті сигналізації перетворення в кореновому вузлі квадродерева для деревоподібного блока, зв'язаного з даним блоком коефіцієнтів перетворення. В інших прикладах модуль 156 оберненого перетворення може визначити обернене перетворення, виходячи з однієї або більше характеристик кодування, таких як розмір блока, режим кодування або т.п. В деяких прикладах модуль 156 оберненого перетворення може застосувати каскадне обернене перетворення.

У деяких прикладах модуль 162 компенсації руху може уточнити прогнозований відеоблок блока PU шляхом виконання інтерполяції на основі інтерполяційних фільтрів. Ідентифікатори для інтерполяційних фільтрів, що використовуються для компенсації руху з точністю до часток відліку, можуть бути включені в згадані синтаксичні елементи. Модуль 162 компенсації руху може використовувати ті самі інтерполяційні фільтри, які використовує відеокодер 20 під час створення прогнозованого відеоблока блока PU, для обчислення інтерпольованих значень для дробових відліків опорного блока. Модуль 162 компенсації руху може визначити інтерполяційні фільтри, що використовуються відеокодером 20 згідно з прийнятою синтаксичною інформацією, і використовувати ці інтерполяційні фільтри для створення прогнозованого відеоблока.

Якщо блок PU кодований з використанням внутрішньокadroвого прогнозування, то модуль 164 внутрішньокadroвого прогнозування може виконати внутрішньокadroве прогнозування для створення прогнозованого відеоблока для даного PU. Наприклад, модуль 164 внутрішньокadroвого прогнозування може визначити режим внутрішньокadroвого прогнозування для даного PU на основі синтаксичних елементів у бітовому потоці. Бітовий потік може включати в себе синтаксичні елементи, які може використовувати модуль 164 внутрішньокadroвого прогнозування для визначення режиму внутрішньокadroвого прогнозування для даного PU.

У деяких випадках вказані синтаксичні елементи можуть вказувати, що модуль 164 внутрішньокadroвого прогнозування повинен використовувати режим внутрішньокadroвого прогнозування іншого PU, щоб визначити режим внутрішньокadroвого прогнозування поточного PU. Наприклад, існує імовірність того, що режим внутрішньокadroвого прогнозування для поточного PU співпадає з режимом внутрішньокadroвого прогнозування для сусіднього PU. Іншими словами, режим внутрішньокadroвого прогнозування сусіднього PU може бути найбільш імовірним для поточного PU. Отже, в цьому прикладі бітовий потік може включати в себе невеликий синтаксичний елемент, який вказує, що режим внутрішньокadroвого прогнозування для даного PU співпадає з режимом внутрішньокadroвого прогнозування сусіднього PU. Тоді модуль 164 внутрішньокadroвого прогнозування може використовувати даний режим внутрішньокadroвого прогнозування для створення даних прогнозування (наприклад, прогнозованих відліків) для даного PU на основі відеоблоків для сусідніх у просторі блоків PU.

Модуль 158 відновлення може використовувати залишкові відеоблоки, зв'язані з блоками TU блока CU, і прогнозовані відеоблоки блоків PU блока CU, тобто, або дані внутрішньокadroвого прогнозування, або дані міжкadroвого прогнозування (залежно від того, що застосовується), для відновлення відеоблока для CU. Таким чином, відеокодер 30 може створити прогнозований відеоблок і залишковий відеоблок на основі синтаксичних елементів у бітовому потоці і може створити відеоблок на основі прогнозованого відеоблока і залишкового відеоблока.

Після відновлення модулем 158 відновлення відеоблока блока CU модуль 159 фільтрів може виконати операцію деблокування для заглушення артефактів блокування, зв'язаних з даним CU. Після виконання модулем 159 операції деблокування для заглушення артефактів блокування, зв'язаних з блоком CU, відеокодер 30 може запам'ятати цей відеоблок блока CU в буфері 160 декодованих зображень. Буфер 160 декодованих зображень може забезпечити опорні зображення для подальшої компенсації руху, внутрішньокadroвого прогнозування і представлення на пристрої відображення, такому як пристрій 32 відображення на Фіг. 1. Наприклад, відеокодер 30 може виконувати операції внутрішньокadroвого прогнозування або міжкadroвого прогнозування на блоках PU інших блоків CU на основі відеоблоків у буфері 160 декодованих зображень.

На Фіг. 4 представлена блок-схема, що ілюструє зразкову операцію 200 для створення даних для слайса. Операція 200 може бути виконана відеокодером, таким як відеокодер 20 (фігури 1 та 2). Блок-схема на Фіг. 4 є просто одним прикладом. В інших прикладах цієї операції можливе створення даних слайса іншими шляхами.

На початку операції 200 відеокодер може ініціалізувати адресу деревоподібного блока, яка ідентифікує початковий деревоподібний блок поточного слайса (202). Поточним слайсом може бути слайс, який в цей час кодується відеокодером. Початковим деревоподібним блоком

поточного слайса може бути перший деревоподібний блок, зв'язаний з поточним слайсом, згідно з порядком кодування деревоподібних блоків у поточному зображенні. Для спрощення пояснення тут мова йде про деревоподібний блок, ідентифікований за адресою деревоподібного блока як поточний деревоподібний блок.

Відеокодер може приєднати синтаксичні елементи для поточного деревоподібного блока до даних слайса блока NAL кодованого слайса для поточного слайса (204). Синтаксичні елементи для поточного деревоподібного блока можуть включати в себе синтаксичні елементи в квадродреві поточного деревоподібного блока. Синтаксичні елементи в квадродреві поточного деревоподібного блока можуть включати в себе синтаксичні елементи, які вказують режими внутрішньокadroвого прогнозування, інформацію про рух, синтаксичні елементи, які вказують рівні коефіцієнтів перетворення тощо.

Крім того, відеокодер може визначити, чи є ще дані в поточному слайсі (206). У поточному слайсі можуть ще бути дані, якщо деревоподібний блок, вказаний адресою деревоподібного блока, знаходиться в поточному слайсі. Відповідно до визначення того, чи є ще дані в поточному слайсі ("HI") на кроці (206), відеокодер може закінчити операцію 200, оскільки він додав у дані слайса всі необхідні синтаксичні елементи.

Відеокодер може визначити, чи є ще дані в поточному слайсі, різними шляхами. Наприклад, відеокодер може активувати функцію "coding_tree ()» для виведення синтаксичних елементів для деревоподібного блока. У цьому прикладі функція "coding_tree ()» може повернути прапор "moreDataFlag", який вказує, чи є ще дані в поточному слайсі.

Відповідно до визначення того, чи є ще дані, зв'язані з поточним слайсом ("ТАК" на кроці 206), відеокодер може визначити, чи є тайли поточного зображення незалежними, і визначити, чи знаходиться наступний деревоподібний блок поточного слайса в тайлі, відмінному від тайла поточного деревоподібного блока поточного слайса (208). Як було описано вище, тайли зображення можуть бути незалежними, якщо заборонене прогнозування в рамках зображення (наприклад, внутрішньокadroве прогнозування, міжкадрове прогнозування з використанням даних у поточному зображенні, і вибір контексту CABAC на основі даних з інших тайлів поточного зображення). Відеокодер може визначити, чи незалежні тайли поточного зображення, різними шляхами. Наприклад, набір параметрів послідовності, зв'язаний з поточним зображенням, може включати в себе синтаксичний елемент "tile_boundary_independence_idc". У цьому прикладі, якщо "tile_boundary_independence_idc" дорівнює 0, то тайли поточного зображення не є незалежними, і прогнозування через межі тайла дозволене. Якщо "tile_boundary_independence_idc" дорівнює 0, то прогнозування через межі слайса все ще заборонене. Якщо "tile_boundary_independence_idc" дорівнює 1, то тайли поточного зображення є незалежними, і прогнозування через межі тайла не дозволене.

Відеокодер може визначити різними шляхами, чи знаходиться наступний деревоподібний блок поточного слайса в іншому тайлі, відмінному від тайла поточного деревоподібного блока поточного слайса. Наприклад, відеокодер може визначити адресу наступного деревоподібного блока поточного слайса. У цьому прикладі відеокодер може активувати функцію "NewTile(...)", яка використовує адресу наступного деревоподібного блока як параметр і повертає значення "newTileflag", яке вказує, чи знаходиться наступний деревоподібний блок в тайлі, відмінному від тайла поточного деревоподібного блока.

Якщо тайли поточного зображення не є незалежними, або наступний деревоподібний блок знаходиться в тому самому тайлі, що і поточний деревоподібний блок ("HI" в блоці 208), то відеокодер може визначити, чи виконується кодування поточного зображення з використанням WPP, і чи знаходиться наступний деревоподібний блок поточного слайса в хвилі WPP, відмінній від хвилі поточного деревоподібного блока поточного слайса (210). Відеокодер може різними шляхами визначити, чи знаходиться наступний деревоподібний блок поточного слайса не в тій хвилі WPP, в якій знаходиться поточний деревоподібний блок поточного слайса. Наприклад, відеокодер може визначити адресу наступного деревоподібного блока поточного слайса. У даному прикладі відеокодер може активувати функцію "NewWave(...)", яка використовує адресу наступного деревоподібного блока як параметр і повертає значення "newWaveFlag", яке вказує, чи знаходиться наступний деревоподібний блок в хвилі WPP, відмінній від хвилі WPP поточного деревоподібного блока.

Відповідно до визначення того, що кодування поточного зображення виконується з використанням WPP, і що наступний деревоподібний блок знаходиться в хвилі WPP, відмінній від хвилі WPP поточного деревоподібного блока ("ТАК" в блоці 210), або відповідно до визначення того, що тайли поточного зображення є незалежними, і що наступний деревоподібний блок знаходиться в тайлі, відмінному від тайла поточного деревоподібного блока ("ТАК" в блоці 208), відеокодер може визначити, чи вирівняний поточний сегмент по

байтах. Іншими словами, відеокoder може визначити, чи закінчується поточний сегмент на межі байта. Поточним сегментом є сегмент, зв'язаний з розділом зображення (наприклад, тайл або хвиля WPP), з яким зв'язаний поточний деревоподібний блок. Відповідно до визначення того, що поточний сегмент не вирівняний по байтах ("HI" на кроці 212), відеокoder може приєднати біт заповнення до кінця поточного сегмента (214). Цей біт заповнення може мати різні значення. Наприклад, біт заповнення завжди може мати значення, рівне 1. В інших прикладах біт заповнення може завжди мати значення, рівне 0.

Після приєднання біта заповнення до кінця поточного сегмента відеокoder може знову визначити, чи вирівняний поточний сегмент по байтах (212). Таким шляхом відеокoder може продовжувати приєднання бітів заповнення до кінця даних слайса, доки не буде забезпечено вирівнювання поточного сегмента по байтах.

Відповідно до визначення того, що дані слайса вирівняні по байтах ("ТАК" на кроці 212), відеокoder може оновити адресу деревоподібного блока (216). Відеокoder може оновити адресу деревоподібного блока таким чином, щоб він вказував наступний деревоподібний блок згідно з порядком кодування деревоподібних блоків поточного зображення. Наприклад, при оновленні відеокoderом адреси деревоподібного блока ця адреса може ідентифікувати деревоподібний блок праворуч від деревоподібного блока, на який вказувала адреса деревоподібного блока. На Фіг. 7, детально описаній нижче, представлена концептуальна схема, що ілюструє зразковий порядок кодування деревоподібних блоків для одного зображення, яке розбите на множину тайлів.

Після оновлення адреси деревоподібного блока відеокoder може визначити, чи є ще дані в поточному слайсі (218). Відповідно до визначення того, чи є ще дані в поточному слайсі ("ТАК" на кроці 218), або відповідно до визначення того, що кодування поточного зображення з використанням WPP не виконується, і наступний деревоподібний блок знаходиться в тому самому тайлі, що і поточний деревоподібний блок ("HI" на кроці 210), відеокoder може приєднати синтаксичні елементи для поточного деревоподібного блока до даних слайса (204). Таким чином, відеокoder може приєднати синтаксичні елементи для кожного деревоподібного блока поточного слайса до даних слайса і може забезпечити таке заповнення сегментів незначущою інформацією, зв'язаних з іншими розділами зображення, при якому ці сегменти будуть починатися з меж байтів.

Відповідно до визначення того, чи є ще дані в поточному слайсі ("HI" на кроці 218), відеокoder може завершити операцію 200, оскільки він може приєднати всі синтаксичні елементи поточного слайса до даних слайса.

На Фіг. 5 представлена блок-схема, що ілюструє зразкову операцію 250 для декодування блока NAL кодованого слайса. Відеодекодер, такий як відеодекодер 30 (фіг. 1 та 3) може виконувати операцію 250. Показане на Фіг. 5, є просто одним з прикладів. В інших прикладах для декодування блоків NAL кодованого слайса можуть використовуватися інші операції.

У прикладі за Фіг. 5 відеодекодер може запам'ятати блок NAL кодованого слайса в пам'яті з байтовою адресацією (252). Блок NAL кодованого слайса може включати в себе заголовок слайса і дані слайса. Дані слайса можуть включати в себе множину сегментів. Один або більше сегментів можуть бути заповнені незначущою інформацією з тим, щоб кожний сегмент починався з межі байта.

Після запам'ятовування в пам'яті блока NAL кодованого слайса відеодекодер може ідентифікувати позиції сегментів у даних блока NAL кодованого слайса (254). Відеодекодер може ідентифікувати позиції сегментів різними способами. Наприклад, відеодекодер може ідентифікувати позиції сегментів на основі синтаксичних елементів у заголовку слайса блока NAL кодованого слайса, які вказують байтові зміщення сегментів. У даному прикладі заголовок слайса може не включати в себе байтове зміщення для першого сегмента даних слайса, оскільки позиція першого сегмента може безпосередньо слідувати за закінченням заголовка слайса. В іншому прикладі відеодекодер може ідентифікувати позиції сегментів на основі позначок точок введення в дані слайса. Позначки точок введення можуть являти собою значення, розташовані між сегментами.

Після ідентифікації позицій сегментів у даних слайса відеодекодер може розподілити два або більше сегментів по двох або більше різних потоках декодування (256). Кожний потік декодування може проаналізувати синтаксичні елементи кодованих деревоподібних блоків у сегменті, розподіленому для даного потоку декодування, і відновити відеоблоки для відповідних деревоподібних блоків, як було описано вище.

На Фіг. 6 представлена концептуальна схема, що ілюструє хвильову паралельну обробку. Як було описано вище, зображення може бути розбите на відеоблоки, кожний з яких зв'язаний з деревоподібним блоком. На Фіг. 6 показані відеоблоки, зв'язані з деревоподібними блоками у

вигляді сітки з білих квадратів. Зображення включає в себе рядки деревоподібних блоків 300A-300E (позначених разом як "рядки деревоподібних блоків 300").

У першому потоці виконується кодування деревоподібних блоків у рядку 300A деревоподібних блоків. Паралельно може виконуватися кодування деревоподібних блоків інших потоків у рядках 300B, 300C та 300D деревоподібних блоків. У прикладі за Фіг. 6 в першому потоці виконується кодування деревоподібного блока 302A, у другому потоці виконується кодування деревоподібного блока 302B, в третьому потоці виконується кодування деревоподібного блока 302C і в четвертому потоці виконується кодування деревоподібного блока 302D. Цей опис можна віднести до деревоподібних блоків 302A, 302B, 302C та 302D, позначених разом як "поточні деревоподібні блоки 302". Оскільки відеокодер може почати кодування рядка деревоподібних блоків після того, як буде виконане кодування більше двох деревоподібних блоків рядка, що знаходиться безпосередньо над даним рядком, поточні деревоподібні блоки 302 розміщуються по горизонталі один від одного на ширину двох деревоподібних блоків.

У прикладі за Фіг. 6 вказані потоки можуть використовувати дані з деревоподібних блоків, вказаних жирними сірими стрілками, під час виконання внутрішньокадрового прогнозування або міжкадрового прогнозування для блоків CU в поточних деревоподібних блоках 302 (Коли потоки виконують міжкадрове прогнозування для блоків CU, вони також можуть використовувати дані з одного або більше опорних кадрів). Коли потік кодує даний деревоподібний блок, він може відібрати один або більше контекстів CABAC на основі інформації, зв'язаної з раніше закодованими деревоподібними блоками. У даному потоці може використовуватися один або більше контекстів CABAC для виконання CABAC кодування на синтаксичних елементах, зв'язаних з першим CU даного деревоподібного блока. Якщо даний деревоподібний блок не є крайнім лівим деревоподібним блоком рядка, то даний потік може вибрати один або більше контекстів CABAC на основі інформації, зв'язаної з останнім CU деревоподібного блока зліва від даного деревоподібного блока. Якщо даний деревоподібний блок є крайнім лівим деревоподібним блоком рядка, то потік може вибрати один або більше контекстів CABAC на основі інформації, зв'язаної з останнім CU деревоподібного блока, що знаходиться вище, і двома деревоподібними блоками праворуч від даного деревоподібного блока. У вказаних потоках можуть використовуватися дані з останніх блоків CU деревоподібних блоків, вказаних тонкими чорними стрілками, для вибору контекстів CABAC для перших блоків CU поточних деревоподібних блоків 302.

На Фіг. 7 представлена концептуальна схема, що ілюструє зразковий порядок кодування деревоподібних блоків для зображення 350, яке розбите на множину тайлів 352A, 352B та 352C. Кожний квадратний білий блок в зображенні 350 представляє відеоблок, зв'язаний з деревоподібним блоком. Жирні вертикальні пунктирні лінії вказують зразкові вертикальні межі тайлів. Жирні сірі лінії вказують зразкову межу слайса.

Числа у відеоблоках вказують позиції відповідних деревоподібних блоків (блоків LCU) в порядку кодування деревоподібних блоків для зображення 350. Як показано в прикладі за Фіг. 7, кожний з деревоподібних блоків у крайньому лівому тайлі 352A з'являється в порядку кодування деревоподібних блоків перед будь-яким деревоподібним блоком в середньому тайлі 352B. Кожний з деревоподібних блоків у середньому тайлі 352B з'являється в порядку кодування деревоподібних блоків перед будь-яким деревоподібним блоком в крайньому правому тайлі 352C. Всередині кожного тайла 352A, 352B та 352C деревоподібні блоки кодуються згідно з порядком растрового сканування.

Відеокодер може створити два блоки NAL кодованого слайса для зображення 350. Перший блок NAL кодованого слайса може бути зв'язаний з лівим слайсом зображення 350. Перший блок NAL кодованого слайса може включати в себе кодовані представлення деревоподібних блоків 1-23. Дані слайса першого блока NAL кодованого слайса можуть включати в себе два сегменти. Перший сегмент може включати в себе кодовані представлення деревоподібних блоків 1-15. Другий сегмент може включати в себе кодовані представлення деревоподібних блоків 16-30. Згідно з розкритими тут методиками перший сегмент може бути заповнений незначущою інформацією, що забезпечує початок другого сегмента з межі байта.

Другий блок NAL кодованого слайса може бути зв'язаний з правим слайсом зображення 350. Другий блок NAL кодованого слайса може включати в себе кодовані представлення деревоподібних блоків 24-45. Дані слайса другого блока NAL кодованого слайса можуть включати в себе два сегменти. Перший сегмент може включати в себе кодовані представлення деревоподібних блоків 24-30. Другий сегмент може включати в себе кодовані представлення деревоподібних блоків 31-45. Перший сегмент може бути заповнений незначущою інформацією таким чином, щоб другий сегмент починався з межі байта.

На Фіг. 8 представлена концептуальна схема, що ілюструє зразковий блок 400 NAL кодованого слайса. Як показано в прикладі на Фіг. 8, блок 400 NAL кодованого слайса включає в себе заголовок 402 слайса і дані 404 слайса. Дані 404 слайса включають в себе перший сегмент 406 і другий сегмент 408. Сегмент 406 включає в себе кодовані деревоподібні блоки 410A-410N і дані 412 заповнення. Сегмент 408 включає в себе кодовані деревоподібні блоки 414A-414N.

В одному або більше прикладах описані функції можуть бути реалізовані апаратними засобами, програмними засобами, програмно-апаратними засобами або будь-якою їх комбінацією. При реалізації програмними засобами ці функції можуть зберігатися або передаватися у вигляді однієї або більше команд або коду на зчитуваному комп'ютером носії і виконуватися обробляючим апаратним блоком. Машинозчитуваний носій інформації може включати в себе машинозчитуваний носій інформації, який відповідає матеріальному носію, такому як носій для зберігання даних або носій для передачі, включаючи будь-який носій, який дозволяє забезпечити пересилання комп'ютерної програми з одного місця в інше, наприклад, згідно з яким-небудь протоколом зв'язку. У цьому аспекті машинозчитуваний носій інформації в загальному випадку може відповідати (1) матеріальному машинозчитуваному носію інформації, який забезпечує довготривале зберігання, або (2) середовищу зв'язку, такому як сигнал або несуча хвиля. Носієм для запам'ятовування даних може служити будь-який доступний носій, доступ до якого може здійснюватися одним або більше комп'ютерами або одним або більше процесорами для витягання команд, коду і/або структур даних для реалізації розкритих в цьому винаході методик. Комп'ютерний програмний продукт може включати в себе машинозчитуваний носій.

Як приклад, але не як обмеження, вказаний машинозчитуваний носій інформації може містити ОЗП (RAM), ПЗП (ROM), електрично стираєний програмований ПЗП (EEPROM), ПЗП на компакт-диску (CD-ROM) або інший запам'ятовуючий пристрій на оптичних дисках, запам'ятовуючий пристрій на магнітних дисках або інші магнітні запам'ятовуючі пристрої, флеш-пам'ять, або будь-який інший носій, який можна використовувати для запам'ятовування необхідного програмного коду у вигляді команд або структур даних, і до яких може мати доступ комп'ютер. Також машинозчитуваним носієм інформації буде правильно назвати будь-яке з'єднання. Наприклад, якщо команди передаються з Web-сайта, сервера або іншого віддаленого джерела з використанням коаксіального кабелю, оптоволоконного кабелю, скрученої пари, цифрової абонентської лінії (DSL) або бездротових технологій, таких як інфрачервоний зв'язок, радіозв'язок і мікрохвильовий зв'язок, то тоді коаксіальний кабель, оптоволоконний кабель, скручена пара, лінії DSL або такі бездротові технології, як інфрачервоний зв'язок, радіозв'язок і мікрохвильовий зв'язок, входять у визначення носія. Однак потрібно розуміти, що терміни: "машинозчитуваний носій інформації" і "носій для зберігання даних" не включають в себе з'єднання, несучі хвилі, сигнали або інші часові носії, а відносяться до матеріального носія довготривалого зберігання. Термін "disc" (диск), що використовується тут, відноситься до компакт-диска (CD), лазерного диска, оптичного диска, цифрового універсального диска (DVD), флоппі-диска і диска Blu-ray, які відтворюють дані оптичним шляхом з використанням лазерів, в той час як термін "disk" (диск) звичайно відноситься до дисків, які відтворюють дані магнітним шляхом. У поняття машинозчитуваного носія інформації також потрібно віднести комбінації з вищепереліченого.

Команди можуть виконуватися одним або більше процесорами, такими як один або більше цифрових процесорів сигналів (DSP), мікропроцесорів загального призначення, прикладних спеціалізованих інтегральних схем (ASIC), вентильних матриць, що програмується користувачем (FPGA), або інших еквівалентних інтегральних або дискретних логічних схем. Відповідно, термін "процесор", що використовується тут, може відноситися до будь-якої з вищеописаних структур або будь-якої іншої структури, підходящої для реалізації вищеописаних методик. Додатково, в деяких аспектах описані тут функціональні можливості можуть бути забезпечені в рамках спеціалізованих апаратних і/або програмних модулів, сконфігурованих для кодування і декодування, або включених в склад комбінованого кодека. Також розкриті тут методики можна повністю реалізувати в одній або більше схемах або логічних елементах.

Розкриті тут методики можуть бути реалізовані в найрізноманітніших пристроях або установках, включаючи бездротову телефонну трубку, інтегральну схему (IC) або набір інтегральних схем (наприклад, мікропроцесорний набір). Різні компоненти, модулі або блоки описані тут таким чином, щоб підкреслити функціональні аспекти пристроїв, сконфігурованих для виконання розкритих тут методик, причому для цього не потрібна їх реалізація за допомогою різних апаратних блоків. Швидше навпаки, як обговорювалося вище, різні блоки можуть бути скомбіновані в апаратному блоці-кодеку або забезпечені у вигляді набору

взаємодіючих апаратних блоків, включаючи один або більше процесорів, як було описано вище, в поєднанні з підходящим програмним і/або програмно-апаратним забезпеченням.

Тут були описані різні приклади. Ці та інші приклади входять в обсяг нижченаведеної формули винаходу.

5	Посилальні позиції
	10 система відеокодування
	12 пристрій-джерело
	14 пристрій-адресата
10	16 канал
	18 джерело відео
	20 відеокодер
	22 вихідний інтерфейс
	28 вхідний інтерфейс
15	30 відеодекодер
	32 пристрій відображення
	100 модуль прогнозування
	102 модуль створення даних залишку
	104 модуль перетворення
20	106 модуль квантування
	108 модуль оберненого квантування
	110 модуль оберненого перетворення
	112 модуль відновлення
	113 модуль фільтрів
25	114 буфер декодованих зображень
	116 модуль ентропійного кодування
	121 модуль міжкадрового прогнозування
	122 модуль оцінки руху
	124 модуль компенсації руху
30	126 модуль внутрішньокадрового прогнозування
	150 модуль ентропійного декодування
	152 модуль прогнозування
	154 модуль оберненого квантування
	156 модуль оберненого перетворення
35	158 модуль відновлення
	159 модуль фільтрів
	160 буфер декодованих зображень
	162 модуль компенсації руху
40	164 модуль внутрішньокадрового прогнозування

ФОРМУЛА ВИНАХОДУ

45	1. Спосіб кодування відеоданих, причому спосіб включає: створення блока (400) рівня абстракції мережі (NAL) кодованого слайсу, який включає в себе кодовані представлення відеоблока (410A-N, 414A-N), які пов'язані зі слайсом зображення відеоданих, причому зображення розділене на множину слайсів, і кодування зображення, використовуючи хвильову паралельну обробку (WPP), який відрізняється тим, що додатково включає:
50	розділення зображення на множину хвиль хвильової паралельної обробки (WPP), причому кожна з хвиль WPP відповідає різному ряду відеоблоків в зображенні, при цьому кодовані представлення відеоблоків згруповані в блоці NAL кодованого слайсу в сегменти, пов'язані з різними хвилями WPP, і
55	заповнення незначущою інформацією одного або більше сегментів таким чином, що кожен із сегментів починається на границі байта, при цьому один або більше сегментів включають в себе заданий сегмент, і
60	заповнення незначущою інформацією одного або більше сегментів включає виконання операції заповнення незначущою інформацією, яка приєднує біти до заданого сегмента, якщо наступний відеоблок знаходиться всередині слайсу і пов'язаний з іншою хвилею WPP, ніж заданий сегмент.

2. Спосіб за п. 1, в якому створення блока NAL кодованого слайсу включає створення заголовка слайсу, який вказує точки введення одного або більше сегментів.

3. Спосіб за п. 2, в якому точки введення для сегментів вказують байтові зміщення сегментів.

4. Спосіб за п. 1, який додатково включає створення набору параметрів, пов'язаного із зображенням,

при цьому набір параметрів включає в себе прапор, що має перше значення, яке вказує, що один або більше сегментів заповнені незначущою інформацією таким чином, що кожен із сегментів починається на границі байта, і

при цьому в тому випадку, коли прапор має друге значення, сегменти можуть починатися або можуть не починатися на границях байтів.

5. Спосіб за п. 1,

в якому сегменти включають в себе заданий сегмент; і

в якому створення блока NAL кодованого слайсу включає виконання операції заповнення незначущою інформацією, яка приєднує біти до заданого сегмента, якщо наступний відеоблок знаходиться всередині слайсу і пов'язаний з іншою хвилею WPP, ніж заданий сегмент.

6. Спосіб декодування відеоданих, що включає зображення, розділене на множину слайсів, причому спосіб включає:

запам'ятовування блока рівня абстракції мережі (NAL) кодованого слайсу, який включає в себе кодовані представлення відеоблоків, пов'язаних зі слайсом зображення,

при цьому зображення розбите на множину хвиль хвильової паралельної обробки (WPP), причому кожна з хвиль WPP відповідає різному ряду відеоблоків в зображенні, причому кодовані представлення відеоблоків згруповані в сегменти, пов'язані з різними хвилями WPP,

причому один або більше сегментів заповнені незначущою інформацією таким чином, що кожен із сегментів починається на границі байта, один або більше сегментів включають в себе

заданий сегмент, і біти приєднуються до заданого сегмента, якщо наступний відеоблок знаходиться всередині слайсу і пов'язаний з іншою хвилею WPP, ніж заданий сегмент; і

декодування зображення, використовуючи WPP.

7. Спосіб за п. 6, в якому блок NAL кодованого слайсу включає заголовок слайсу, який вказує точки введення для одного або більше сегментів.

8. Спосіб за п. 7, в якому точки введення для сегментів вказують байтові зміщення сегментів.

9. Спосіб за п. 6, який додатково включає запам'ятовування набору параметрів, пов'язаного із зображенням, при цьому набір параметрів включає в себе прапор, що має перше значення, яке вказує, що один або більше сегментів заповнені незначущою інформацією таким чином, що кожний із зазначених сегментів починається на границі байта, і

при цьому в тому випадку, коли прапор має друге значення, сегменти можуть починатися або можуть не починатися на межах байтів.

10. Спосіб за п. 6, в якому декодування зображення включає паралельне декодування кодованих представлень відеоблоків у двох або більше сегментах.

11. Пристрій відеокодування, який кодує відеодані, причому пристрій відеокодування містить:

пам'ять, сконфігуровану для зберігання кодової версії відеоданих; і

один або більше процесорів, сконфігурованих для:

створення блока (400) рівня абстракції мережі (NAL) кодованого слайсу, який включає в себе кодовані представлення відеоблоків (410A-N, 414A-N), які пов'язані зі слайсом зображення відеоданих, причому зображення розділене на множину слайсів, і

кодування зображення, використовуючи WPP,

який **відрізняється** тим, що один або більше процесорів додатково сконфігуровані для:

розділення зображення на множину хвиль хвильової паралельної обробки (WPP), причому кожна з хвиль WPP відповідає різному ряду відеоблоків в зображенні, причому кодовані представлення відеоблоків згруповані в блоці NAL кодованого слайсу в сегменти, пов'язані з

різними хвилями WPP, і заповнення незначущою інформацією одного або більше сегментів таким чином, що кожен із сегментів починається на границі байта,

при цьому один або більше сегментів включають в себе заданий сегмент, і заповнення незначущою інформацією одного або більше сегментів містить виконання операції заповнення

незначущою інформацією, яка приєднує біти до заданого сегмента, якщо наступний відеоблок знаходиться всередині слайсу і пов'язаний з іншою хвилею WPP, ніж заданий сегмент.

12. Пристрій відеокодування за п. 11, в якому один або більше процесорів сконфігуровані для створення блока NAL кодованого слайсу таким чином, що блок NAL кодованого слайсу включає в себе заголовок слайсу, який вказує точки введення для одного або більше сегментів.

13. Пристрій відеокодування за п. 12, в якому точки введення для сегментів вказують байтові зміщення сегментів.

14. Пристрій відеокодування за п. 11, в якому один або більше процесорів сконфігуровані для створення набору параметрів, пов'язаного із зображенням, при цьому набір параметрів включає в себе прапор, що має перше значення, яке вказує, що один або більше сегментів заповнені незначущою інформацією таким чином, що кожен з сегментів починається на границі байта, і
- 5 при цьому в тому випадку, коли прапор має друге значення, сегменти можуть починатися або можуть не починатися на границях байтів.
15. Пристрій відеокодування за п. 11, при цьому пристрій відеокодування являє собою мобільний обчислювальний пристрій.
- 10 16. Пристрій відеокодування за п. 11, при цьому пристрій відеокодування містить щонайменше одне з наступного:
- інтегральну схему, мікропроцесор або пристрій бездротового зв'язку.
17. Пристрій відеокодування за п. 11, який додатково містить камеру, сконфігурований для отримання відеоданих.
- 15 18. Пристрій відеодекодування, який декодує відеодані, що містять зображення, розділене на множину слайсів, причому пристрій відеодекодування містить:
- пам'ять, в якій зберігається блок рівня абстракції мережі (NAL) кодованого слайсу, який включає в себе кодовані представлення відеоблоків, пов'язаних зі слайсом зображення, при цьому зображення розділене на множину хвиль хвильової паралельної обробки (WPP),
- 20 причому кожна з хвиль WPP відповідає різному ряду відеоблоків в зображенні, при цьому кодовані представлення відеоблоків згруповані в сегменти, пов'язані з різними хвилями WPP, при цьому один або більше сегментів заповнені незначущою інформацією таким чином, що кожен із сегментів починається на границі байта, один або більше сегментів включають в себе заданий сегмент, і біти приєднані до заданого сегмента, якщо наступний відеоблок знаходиться всередині слайсу і пов'язаний з іншою хвилею WPP, ніж заданий сегмент; і
- 25 один або більше процесорів сконфігуровані для декодування зображення, використовуючи WPP.
19. Пристрій відеодекодування за п. 18, в якому блок NAL кодованого слайсу містить заголовок слайсу, який вказує точки введення для одного або більше сегментів.
- 30 20. Пристрій відеодекодування за п. 19, в якому точки введення для сегментів вказують байтові зміщення сегментів.
21. Пристрій відеодекодування за п. 18, в якому в пам'яті зберігається набір параметрів, пов'язаний із зображенням, при цьому набір параметрів включає в себе прапор, що має перше значення, яке вказує, що один або більше сегментів заповнені незначущою інформацією таким
- 35 чином, що кожен з зазначених сегментів починається на границі байта, і
- при цьому в тому випадку, коли прапор має друге значення, сегменти можуть починатися або можуть не починатися на границях байтів.
22. Пристрій відеодекодування за п. 18, при цьому пристрій відеодекодування являє собою мобільний обчислювальний пристрій.
- 40 23. Пристрій відеодекодування за п. 18, в якому один або більше процесорів сконфігуровані для паралельного декодування кодованих представлень відеоблока у двох або більше сегментах.
24. Пристрій відеодекодування за п. 18, при цьому пристрій відеодекодування містить щонайменше одне з наступного:
- інтегральну схему, мікропроцесор або пристрій бездротового зв'язку.
- 45 25. Пристрій відеодекодування за п. 18, який додатково містить дисплей, сконфігурований для відображення декодованих відеоданих.
26. Машиночитаний носій інформації, який зберігає команди, які при їх виконанні одним або більше процесорами конфігурують пристрій відеокодування, призначений для кодування відеоданих, для:
- 50 створення блока (400) рівня абстракції мережі (NAL) кодованого слайсу, який включає в себе кодовані представлення відеоблоків (410A-N, 414A-N), які пов'язані зі слайсом зображення відеоданих, причому зображення розділене на множину слайсів, і
- кодування зображення, використовуючи WPP,
- який **відрізняється** тим, що інструкції при їх виконанні одним або більше процесорами
- 55 конфігурують пристрій відеокодування для:
- розділення зображення на множину хвиль хвильової паралельної обробки (WPP), причому кожна з хвиль WPP відповідає різному ряду відеоблоків в зображенні, причому кодовані представлення відеоблоків згруповані в блоці NAL кодованого слайсу в сегменти, пов'язані з різними хвилями WPP, і

заповнення незначущою інформацією одного або більше сегментів таким чином, що кожен із сегментів починається на границі байта,

при цьому один або більше сегментів включають в себе заданий сегмент; і заповнення незначущою інформацією одного або більше сегментів містить виконання операції заповнення незначущою інформацією, яка приєднує біти до заданого сегмента, якщо наступний відеоблок знаходиться всередині слайсу і пов'язаний з іншою хвилею WPP, ніж заданий сегмент.

27. Машиночитаний носій інформації, який зберігає команди, які при їх виконанні одним або більше процесорами конфігурують пристрій відеодекодування, призначений для декодування відеоданих, що містять зображення, розділене на множину слайсів, для:

запам'ятовування блока рівня абстракції мережі (NAL) кодованого слайсу, який включає в себе кодовані представлення відеоблоків, пов'язаних зі слайсом зображення,

при цьому зображення розділене на множину хвиль хвильової паралельної обробки (WPP), причому кожна з хвиль WPP відповідає різному ряду відеоблоків в зображенні,

при цьому кодовані представлення відеоблоків згруповані в сегменти, пов'язані з різними хвилями WPP, причому один або більше сегментів заповнені незначущою інформацією таким чином, що кожен із сегментів починається на границі байта,

при цьому один або більше сегментів включають в себе заданий сегмент; і біти приєднуються до заданого сегмента, якщо наступний відеоблок знаходиться всередині слайсу і пов'язаний з іншою хвилею WPP, ніж заданий сегмент; і

декодування зображення, використовуючи WPP.

28. Пристрій відеокодування, який кодує відеодані, причому пристрій відеокодування містить:

засіб для створення блока (400) рівня абстракції мережі (NAL) кодованого слайсу, який включає в себе кодовані представлення відеоблока (410A-N, 414A-N), які пов'язані зі слайсом зображення відеоданих, причому зображення розділене на множину слайсів, і

засіб для кодування зображення, використовуючи WPP, який **відрізняється** тим, що містить:

засіб для розділення зображення на множину хвиль хвильової паралельної обробки (WPP), при цьому кожна з хвиль WPP відповідає різному ряду відеоблоків в зображенні, причому кодовані представлення відеоблоків згруповані в блоці NAL кодованого слайсу в сегменти, пов'язані з різними хвилями WPP, і

засіб для заповнення незначущою інформацією одного або більше сегментів таким чином, що кожен із сегментів починається на границі байта, при цьому один або більше сегментів включають в себе заданий сегмент, і заповнення незначущою інформацією одного або більше сегментів містить виконання операції заповнення незначущою інформацією, яка приєднує біти до заданого сегмента, якщо наступний відеоблок знаходиться всередині слайсу і пов'язаний з іншою хвилею WPP, ніж заданий сегмент.

29. Пристрій відеодекодування, який декодує відеодані, що містять зображення, розділене на множину слайсів, причому пристрій відеодекодування містить:

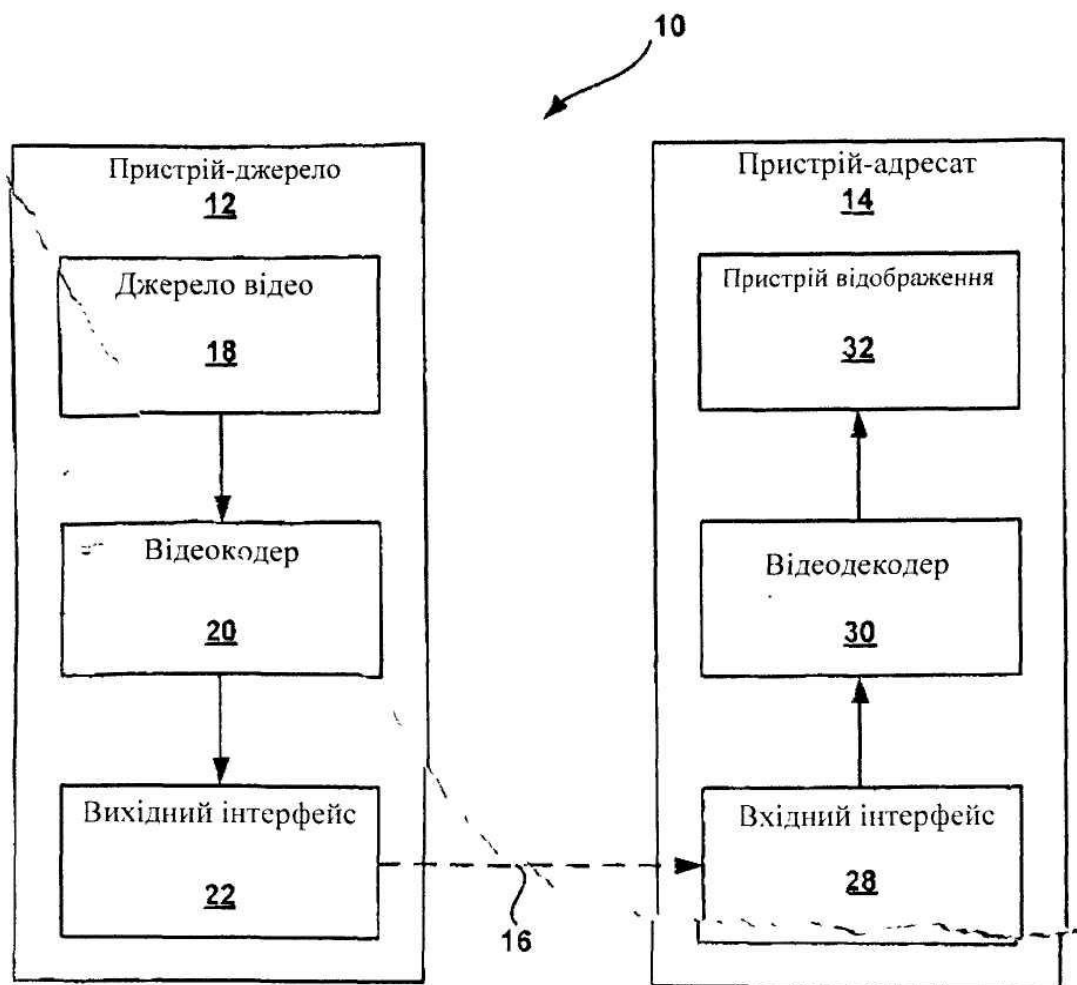
засіб для зберігання блока рівня абстракції мережі (NAL) кодованого слайсу, який включає в себе кодовані представлення відеоблоків, пов'язаних зі слайсом зображення, при цьому

зображення розділене на множину хвиль хвильової паралельної обробки (WPP), при цьому кожна з хвиль WPP відповідає різному ряду відеоблоків в зображенні, причому кодовані

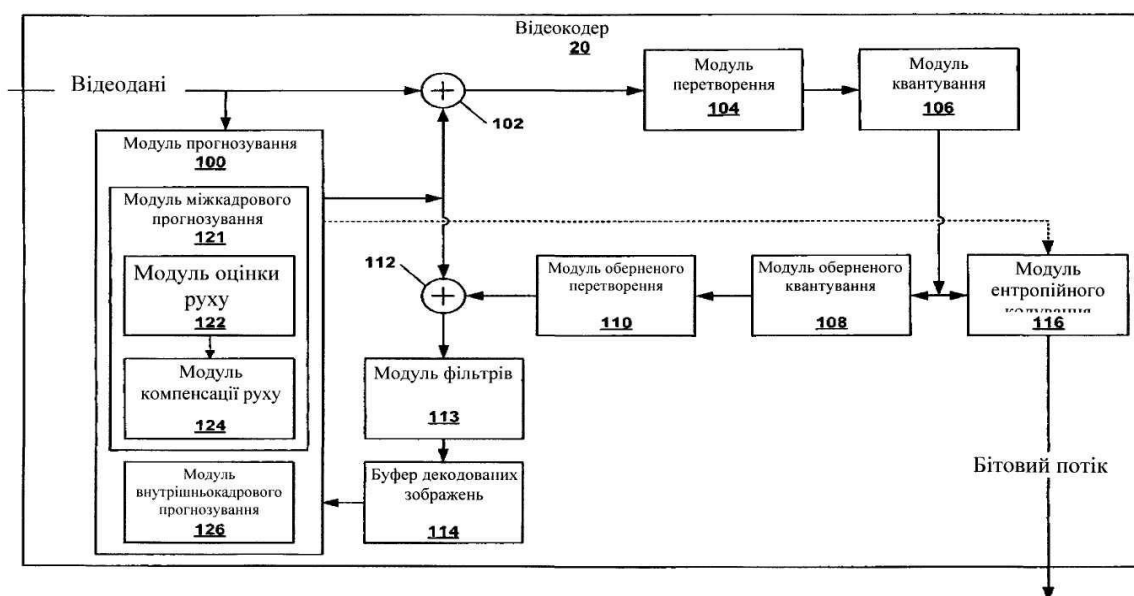
представлення відеоблоків згруповані в сегменти, пов'язані з різними хвилями WPP, причому один або більше сегментів заповнені незначущою інформацією таким чином, що кожен із

сегментів починається на границі байта, один або більше сегментів включають в себе заданий сегмент, і біти приєднуються до заданого сегмента, якщо наступний відеоблок знаходиться всередині слайсу і пов'язаний з іншою хвилею WPP, ніж заданий сегмент; і

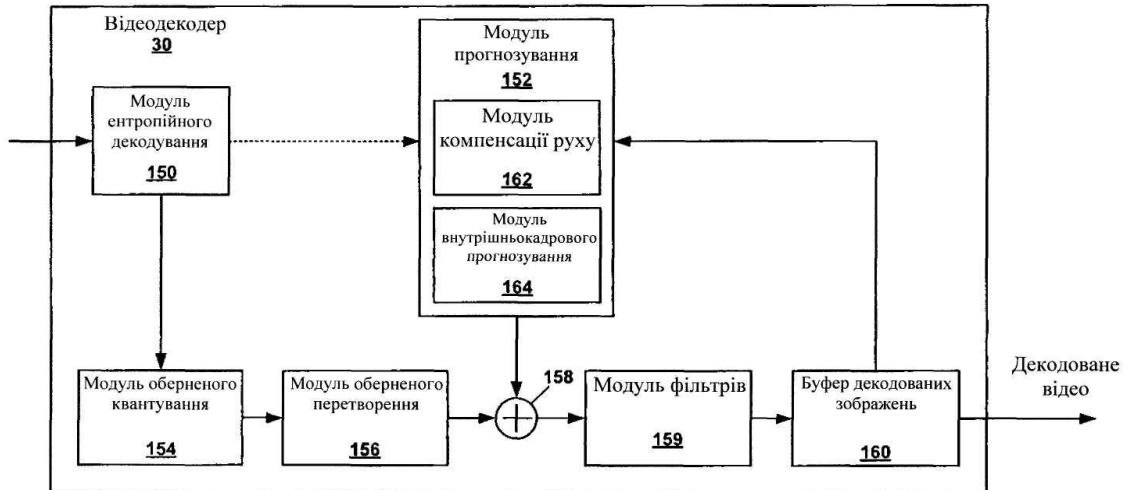
засіб для декодування зображення, використовуючи WPP.



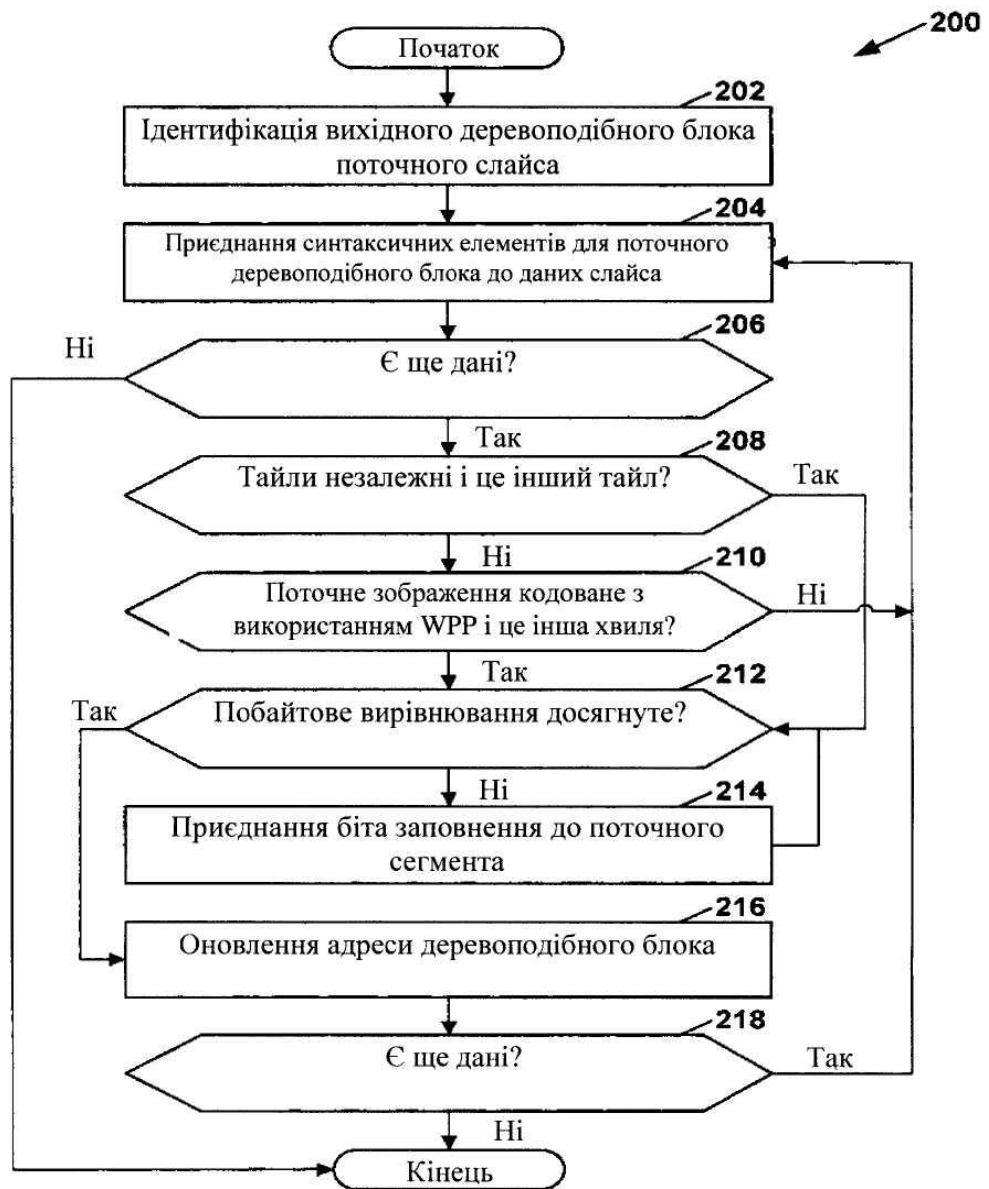
Фіг. 1



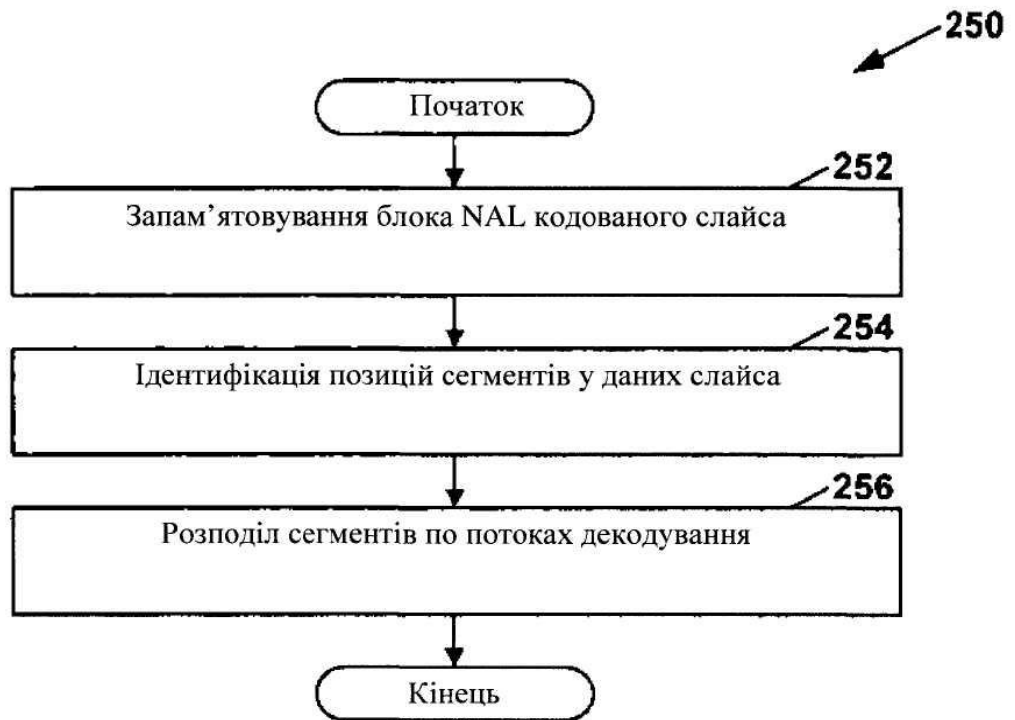
Фіг. 2



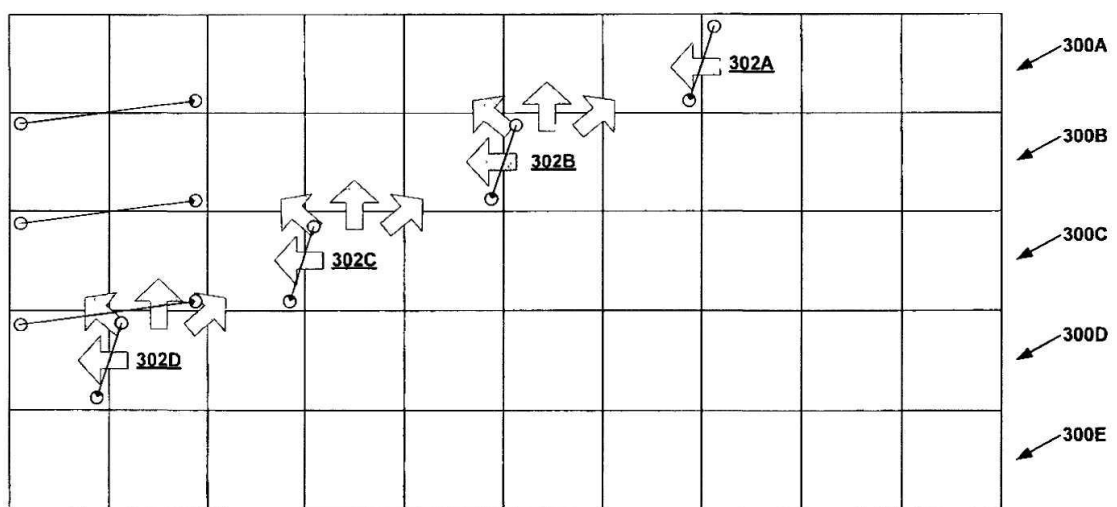
Фіг. 3



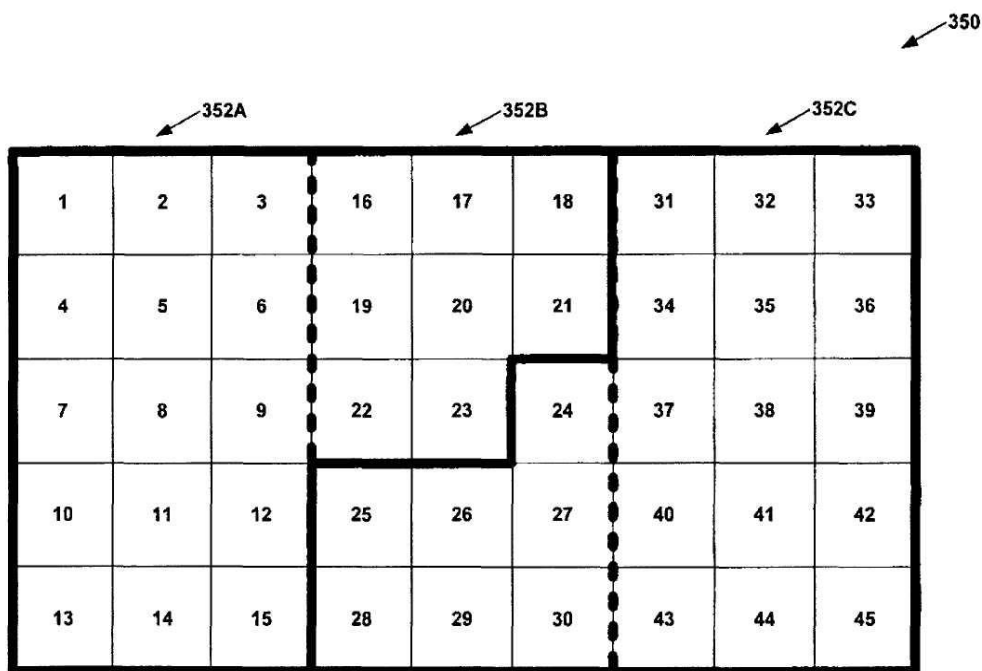
Фіг. 4



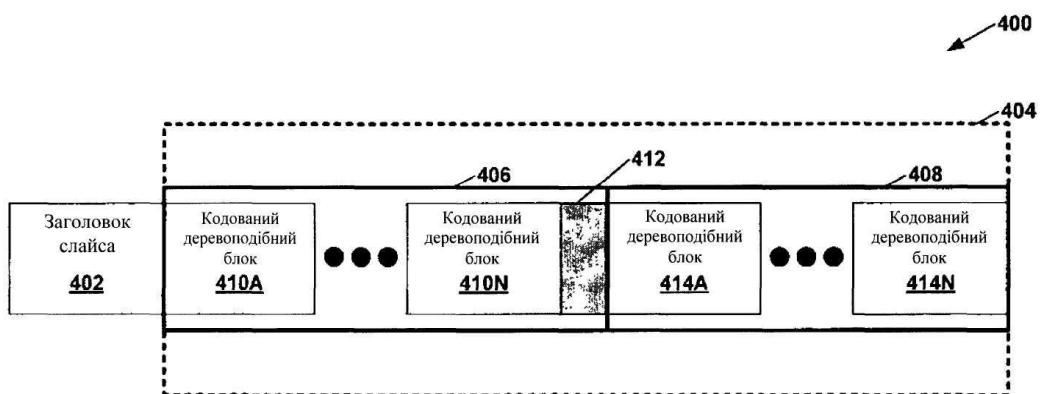
Фіг. 5



Фіг. 6



Фіг. 7



Фіг. 8

Комп'ютерна верстка І. Скворцова

Державна служба інтелектуальної власності України, вул. Василя Липківського, 45, м. Київ, МСП, 03680, Україна

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