



УКРАЇНА

(19) UA

(11) 110637

(13) C2

(51) МПК

H04N 19/117 (2014.01)

H04N 19/169 (2014.01)

ДЕРЖАВНА СЛУЖБА  
ІНТЕЛЕКТУАЛЬНОЇ  
ВЛАСНОСТІ  
УКРАЇНИ

(12) ОПИС ДО ПАТЕНТУ НА ВІНАХІД

(21) Номер заявки: а 2013 11226

(22) Дата подання заявки: 22.02.2012

(24) Дата, з якої є чинними  
права на винахід: 25.01.2016

(31) Номер попередньої  
заявки відповідно до  
Паризької конвенції: 61/445,967,  
61/448,771,  
61/473,713,  
61/476,260,  
61/478,287,  
61/503,426,  
61/503,434,  
61/503,440,  
61/527,463,  
61/531,571,  
13/401,685

(32) Дата подання  
попередньої заявки  
відповідно до  
Паризької конвенції: 23.02.2011,  
03.03.2011,  
08.04.2011,  
16.04.2011,  
22.04.2011,  
30.06.2011,  
30.06.2011,  
30.06.2011,  
25.08.2011,  
06.09.2011,  
21.02.2012

(33) Код держави-учасниці  
Паризької конвенції,  
до якої подано  
попередню заявку: US,  
US,  
US,  
US,  
US,  
US,  
US,  
US,  
US,  
US,  
US

(41) Публікація відомостей  
про заявку: 25.02.2014, Бюл.№ 4

(46) Публікація відомостей  
про видачу патенту: 25.01.2016, Бюл.№ 2

(86) Номер та дата  
подання міжнародної  
заявки, поданої  
відповідно до  
Договору РСТ РСТ/US2012/026166,  
22.02.2012

(72) Винахідник(и):  
Чонг Ін Сук (US),  
Карчевіч Марта (US)

(73) Власник(и):  
КВЕЛКОММ ІНКОРПОРЕЙТЕД,  
5775 Morehouse Drive, ATTN: International IP  
Administration, San Diego, California 92121-  
1714, United States of America (US)

(74) Представник:  
Мошинська Ніна Миколаївна, реєстр.  
№115

(56) Перелік документів, взятих до уваги  
експертизою:  
ХР 030007566, 18.05.2010  
ХР 030003708, 03.07.2009  
ХР 030007715, 30.08.2010  
ХР 030047440, 24.01.2011

(54) МУЛЬТИМЕТРИЧНА ФІЛЬТРАЦІЯ

UA 110637 C2

---

**(57)** Реферат:

Модуль фільтра відеокодера або відеодекодера може визначати першу метрику для набору пікселів у блоці пікселів на основі порівняння піднабору пікселів у цьому блоці з іншими пікселями в цьому блоці; визначати фільтр на основі першої метрики; і генерувати відфільтроване зображення шляхом застосування фільтра до набору пікселів. Піднабір пікселів може бути вибраний так, щоб не включати в себе пікселі на межі блока пікселів.

## МУЛЬТИМЕТРИЧНА ФІЛЬТРАЦІЯ

За даною заявкою заявляється пріоритет:

попередньої заявки на патент США № 61/445,967, поданої 23 лютого 2011 року;

попередньої заявки на патент США № 61/448,771, поданої 3 березня 2011 року;

5 попередньої заявки на патент США № 61/473,713, поданої 8 квітня 2011 року;

попередньої заявки на патент США № 61/476,260, поданої 16 квітня 2011 року;

попередньої заявки на патент США № 61/478,287, поданої 22 квітня 2011 року;

попередньої заявки на патент США № 61/503,426, поданої 30 червня 2011 року;

попередньої заявки на патент США № 61/503,434, поданої 30 червня 2011 року;

10 попередньої заявки на патент США № 61/503,440, поданої 30 червня 2011 року;

попередньої заявки на патент США № 61/527,463, поданої 25 серпня 2011 року;

попередньої заявки на патент США № 61/531,571, поданої 6 вересня 2011 року;

повний зміст яких увімкнений сюди в повному обсязі за допомогою посилання.

## ГАЛУЗЬ ТЕХНІКИ, ДО ЯКОЇ НАЛЕЖИТЬ ВІНАХІД

15 Дане розкриття стосується цифрового кодування відео на основі блоків, що використовуються для стиснення відеоданих, зокрема - методик фільтрації відеоблоків.

## РІВЕНЬ ТЕХНІКИ

Можливості цифрового відео можуть впроваджуватися для широкого діапазону пристроїв, включаючи цифрове телебачення, системи цифрового прямого мовлення, пристрої бездротового зв'язку, такі як радіотелефонні гарнітури, бездротові системи мовлення, персональні цифрові помічники (PDA), портативні комп'ютери, стаціонарні комп'ютери, планшетні комп'ютери, цифрові камери, цифрові записуючі пристрої, пристрої для відеоігор, ігрові відеоприставки, тощо. Цифрові відеопристрої реалізують методики стиснення відео, такі як MPEG-2, MPEG-4, ITU-T H.264/MPEG-4, Частина 10, вдосконалене кодування відео (AVC), щоб більш ефективно передавати і приймати цифрове відео. Методики стиснення відео здійснюють просторове і часове прогнозування, щоб зменшити надмірність, властиву відеопослідовностям. Нові відеостандарти, такі як стандарт високоефективного кодування відео (HEVC), який розробляється "Спільною об'єднаною командою кодування відео" (JCTVC), що є спільною роботою MPEG та ITU-T, продовжують з'являтися і розвиватися. Цей новий стандарт HEVC також іноді іменується H.265.

Методики стиснення відео на основі блоків можуть здійснювати просторове прогнозування і/або часове прогнозування. Внутрішнє (intra-) кодування спирається на просторове прогнозування, щоб зменшити або усунути просторову надмірність між відеоблоками всередині заданого елемента кодованого відео, який може містити відеокадр, вирізку відеокадру тощо. На відміну від цього, зовнішнє (inter-) кодування спирається на часове прогнозування, щоб зменшити або усунути часову надмірність між відеоблоками послідовних елементів кодування відеопослідовності. Для внутрішнього кодування відеокoder здійснює просторове прогнозування, щоб стиснути дані на основі інших даних в межах одного і того самого елемента кодованого відео. Для зовнішнього кодування відеокoder здійснює оцінку руху та компенсацію руху, щоб відстежити рух відповідних відеоблоків двох або більше сусідніх елементів кодованого відео.

Кодований відеоблок може бути представлений інформацією прогнозування, яка може використовуватися для створення або ідентифікації блока прогнозування і залишкового блока даних, що показує відмінності між блоком, який піддається кодуванню, і прогнозованим блоком. У випадку зовнішнього кодування один або більше векторів руху використовуються для ідентифікації блока прогнозування даних з попереднього або подальшого елемента кодування, тоді як у випадку внутрішнього кодування режим прогнозування може використовуватися для генерування блока прогнозування на основі даних в межах CU, зв'язаного з відеоблоком, що кодується. І внутрішнє кодування і зовнішнє кодування можуть визначати декілька різних режимів прогнозування, які можуть визначати різні розміри блоків і/або методики прогнозування, що використовуються при кодуванні. Додаткові типи синтаксичних компонентів також можуть бути включені як частина кодованих відеоданих з метою керувати методиками кодування або параметрами, що використовуються в процесі кодування, або визначати їх.

Після кодування з прогнозуванням на основі блоків відеокoder може здійснювати перетворення, квантування та ентропійне кодування, щоб додатково зменшити швидкість передачі бітів, що стосується повідомлення залишкового блока. Методики перетворення можуть містити дискретні косинусні перетворення (DCT) або концептуально схожі процеси, такі як вейвлетні перетворення, цілочисельні перетворення або інші типи перетворень. У процесі дискретного косинусного перетворення, як приклад, процес перетворення перетворює набір значень різниці пікселів у коефіцієнти перетворення, які можуть представляти енергію значень

пікселів у частотній ділянці. Квантування застосовується відносно коефіцієнтів перетворення і загалом включає в себе процес, який обмежує кількість бітів, зв'язаних з будь-яким заданим коефіцієнтом перетворення. Ентропійне кодування містить один або більше процесів, які спільно стискають послідовність квантованих коефіцієнтів перетворення.

5 Фільтрація відеоблоків може виконуватися як частина циклів кодування і декодування або як частина постфільтраційного процесу у відношенні реконструйованих відеоблоків. Фільтрація звичайно використовується, наприклад, щоб зменшити блочність або інші артефакти, поширені при кодуванні відео на основі блоків. Коефіцієнти фільтра (що іноді називаються відведеннями фільтра) можуть визначатися або вибиратися з метою підтримувати бажані рівні фільтрації відеоблока, які можуть зменшити блочність і/або поліпшити якість відеозображення іншими способами. Набір коефіцієнтів фільтра, наприклад, може визначати, як фільтрація виконується по межах відеоблоків або в інших точках в межах відеоблоків. Різні коефіцієнти фільтра можуть спричиняти досягнення різних рівнів фільтрації по відношенню до різних пікселів відеоблоків. Фільтрація, наприклад, може згладжувати або загострювати відмінності в інтенсивності значень сусідніх пікселів, щоб допомогти уникнути небажаних артефактів.

#### СУТЬ ВИНАХОДУ

Дане розкриття описує методики, пов'язані з фільтрацією відеоданих в процесі кодування і/або декодування відео. Відповідно до цього розкриття фільтрація здійснюється в кодері, і інформація фільтрації кодується в бітовому потоці, щоб дозволити декодеру ідентифікувати фільтрацію, яка виконувалася в кодері. Декодер приймає кодовані відеодані, які включають в себе інформацію фільтрації, декодує відеодані та здійснює фільтрацію на основі інформації фільтрації. Таким чином, декодер здійснює ту саму фільтрацію, яка застосовувалася в кодері. Відповідно до методик даного розкриття на покадровій основі, на основі вирізка за вирізкою або на основі LCU за LCU кодер може вибирати один або більше наборів фільтрів і на основі кодований елемент за кодованим елементом кодер може визначати, чи здійснювати фільтрацію. Для кодованих елементів (CU), які підлягають фільтрації, кодер може здійснювати фільтрацію на піпиксельній основі або на основі група за групою, де група може, наприклад, бути блоком пікселів  $2 \times 2$  або блоком пікселів  $4 \times 4$ .

В одному прикладі спосіб кодування відео включає в себе визначення першої метрики для блока пікселів, при цьому перша метрика визначається на основі порівняння піднабору пікселів у блоці з іншими пікселями в блоці; на основі першої метрики визначення фільтра для блока пікселів; і генерування відфільтрованого зображення шляхом застосування фільтра відносно блока пікселів.

В іншому прикладі пристрій кодування відео включає в себе елемент фільтра, виконаний з можливістю визначати першу метрику для блока пікселів, при цьому перша метрика визначається на основі порівняння піднабору пікселів у блоці з іншими пікселями в блоці, визначати фільтр для блока пікселів на основі першої метрики і генерувати відфільтроване зображення шляхом застосування фільтра відносно блока пікселів; і запам'ятовувачий пристрій, виконаний з можливістю зберігати відфільтрований результат елемента фільтра.

В іншому прикладі апарат включає в себе засіб для визначення першої метрики для блока пікселів, при цьому перша метрика визначається на основі порівняння піднабору пікселів у блоці з іншими пікселями в блоці, засіб для визначення фільтра для блока пікселів на основі першої метрики і засіб для генерування відфільтрованого зображення шляхом застосування фільтра відносно блока пікселів.

В іншому прикладі машинозчитуваний носій даних зберігає команди, які при виконанні ініціюють визначення одним або більше процесорами першої метрики для блока пікселів, при цьому перша метрика визначається на основі порівняння піднабору пікселів у блоці з іншими пікселями в блоці, визначення фільтра для блока пікселів на основі першої метрики і генерування відфільтрованого зображення шляхом застосування фільтра відносно блока пікселів.

Деталі одного або більше прикладів викладені нижче в прикладених кресленнях та описі. Інші ознаки, цілі та переваги будуть зрозумілі з опису та креслень, а також з формули винаходу.

#### КОРОТКИЙ ОПИС КРЕСЛЕНЬ

Фіг. 1 являє собою блок-схему, яка ілюструє зразкову систему кодування і декодування відео.

Фіг. 2A та фіг. 2B являють собою концептуальні схеми, що ілюструють приклад розділення квадродререва, що застосовується відносно найбільшого елемента кодування (LCU).

Фіг. 2C та фіг. 2D являють собою концептуальні схеми, що ілюструють приклад схеми фільтра для серії відеоблоків, яка відповідає зразковому розділенню квадродререва з фіг. 2A та фіг. 2B.

Фіг. 3 являє собою блок-схему, яка ілюструє зразковий відеокoder, сумісний з даним розкриттям.

Фіг. 4А являє собою концептуальну схему, що ілюструє приведення у відповідність діапазонів для двох метрик і фільтрів.

5 Фіг. 4В являє собою концептуальну схему, що ілюструє приведення у відповідність діапазонів для метрики активності та метрики напрямку для фільтрів.

Фіг. 5 являє собою блок-схему, яка ілюструє зразковий відеодекодер відповідно до даного розкриття.

Фіг. 6А, фіг. 6В та фіг. 6С демонструють концептуальні схеми блока пікселів  $4 \times 4$ .

10 Фіг. 7 являє собою блок-схему послідовності операцій, що ілюструє методики кодування відповідно до даного розкриття.

Фіг. 8А та фіг. 8В являють собою блок-схеми послідовності операцій, що ілюструють методики кодування відповідно до даного розкриття.

15 Фіг. 9А та фіг. 9В являють собою блок-схеми послідовності операцій, що ілюструють методики кодування відповідно до даного розкриття.

Фіг. 10 являє собою блок-схему послідовності операцій, що ілюструє методики кодування відповідно до даного розкриття.

Фіг. 11 являє собою блок-схему послідовності операцій, що ілюструє методики кодування відповідно до даного розкриття.

## 20 ДОКЛАДНИЙ ОПИС

Дане розкриття описує методики, пов'язані з фільтрацією відеоданих в процесі кодування і/або декодування відео. Відповідно до цього розкриття фільтрація здійснюється в кодері, і інформація фільтрації кодується в бітовому потоці, щоб дозволити декодеру ідентифікувати фільтрацію, яка виконувалася в кодері. Декодер приймає кодовані відеодані, які включають в себе інформацію фільтрації, декодує відеодані та здійснює фільтрацію на основі інформації фільтрації. Таким чином, декодер здійснює ту саму фільтрацію, яка застосовувалася в кодері. Відповідно до методик даного розкриття на покадровій основі, на основі вирізка за вирізкою або на основі LCU за LCU кодер може вибирати один або більше наборів фільтрів і на основі кодований елемент за кодованим елементом кодер може визначати, чи здійснювати 25 фільтрацію. Для кодованих елементів (CU), які підлягають фільтрації, кодер може здійснювати фільтрацію на попиксельній або погруповій основі, де група може, наприклад бути блоком пікселів  $2 \times 2$  або блоком пікселів  $4 \times 4$ .

Відповідно до методик цього розкриття відеодані можуть кодуватися в елементах, що називаються кодованими елементами (CU). CU можуть розділятися на менші CU або піделементи з використанням схеми розділення квадродререва. Синтаксис, що ідентифікує 35 схему розділення квадродререва для конкретного CU, може передаватися від кодера до декодера. Множинні введення, пов'язані з кожним піделементом заданого CU, можуть фільтруватися під час процесу декодування і реконструкції кодованих відеоданих. Відповідно до методик цього розкриття синтаксис опису фільтра може описувати набір фільтрів, як то - скільки 40 фільтрів у наборі або яку форму приймають фільтри. Додатковий синтаксис в бітовому потоці, що приймається декодером, може ідентифікувати фільтри (тобто коефіцієнти фільтрів), що використовуються в кодері для конкретного піделементу. Фільтр, що використовується для конкретного введення, може бути вибраний на основі двох або більше метрик, де деякі комбінації значень для двох або більше метрик індексуються для конкретних фільтрів у межах 45 набору фільтрів. В інших прикладах дві або більше метрик можуть об'єднуватися для формування єдиної метрики. Відповідність метрик фільтрам може також сигналізуватися в бітовому потоці.

Різні типи фільтрації можуть застосовуватися відносно пікселів або блоків пікселів на основі двох або більше метрик, визначених для відеоданих. Фільтр, що використовується для 50 конкретного пікселя, може бути вибраний на основі двох або більше метрик, таких як деяка комбінація метрики активності та метрики напрямку. Метрика активності, наприклад, може визначати кількісно активність, зв'язану з одним або більше блоками пікселів у відеоданих. Метрика активності може містити метрику дисперсії, що показує дисперсію пікселя в межах набору пікселів. Метрика активності може бути як специфічною для напрямку, так і не специфічною для напрямку. Наприклад, метрика активності, не специфічна для напрямку, може 55 включати в себе значення Лапласіана із зміненою сумою, як пояснюється нижче більш детально.

Приклади метрик активності, специфічних для напрямку, включають в себе метрику горизонтальної активності, метрику вертикальної активності, метрику 45-градусної активності та 60 метрику 135-градусної активності. Метрика напрямку може визначати кількісно для блока

пікселів будь-яку з горизонтальної активності, вертикальної активності або діагональної активності пікселя або групи пікселів або, метрика напрямку може включати в себе порівняння горизонтальної активності, вертикальної активності і/або діагональної активності, де горизонтальна активність загалом стосується змін піксельних значень в горизонтальному напрямку, вертикальна активність загалом стосується змін піксельних значень у вертикальному напрямку, а діагональна активність загалом стосується змін піксельних значень в діагональному напрямку.

Відповідно до методик цього розкриття при визначенні фільтра для блока пікселів піднабір пікселів у межах блока може використовуватися, щоб зменшити складність кодування та декодування. Наприклад, при визначенні фільтра для блока пікселів  $4 \times 4$  може бути необов'язково використовувати всі шістнадцять пікселів з блока  $4 \times 4$ . Крім того, відповідно до методик цього розкриття піднабір пікселів у межах поточного блока, що кодується, може бути вибраний так, що метрики обчислюються з використанням тільки піксельних значень поточного блока, але не піксельних значень сусідніх блоків. Наприклад, метрика для пікселя, що оцінюється, може обчислюватися на основі порівняння пікселя з розташованими поблизу пікселями. У деяких випадках один або більше пікселів, розташовані поблизу від пікселя, що оцінюється, можуть бути в блоці, відмінному від того, в якому піксель, що оцінюється. В інших випадках, однак, один або більше пікселів, розташованих поблизу від згаданого пікселя, можуть бути в тому самому блоці, що і згаданий піксель. Відповідно до методик цього розкриття піднабір пікселів може вибиратися так, щоб включати в себе пікселі, у яких немає розташованих поблизу пікселів у сусідніх блоках. Додатково або як альтернатива піднабір пікселів може включати в себе пікселі, у яких є розташовані поблизу пікселі в сусідніх блоках, але ці розташовані поблизу пікселі в сусідніх блоках можуть не використовуватися при визначенні метрики. Шляхом базування визначення конкретної метрики на пікселях в межах поточного блока, але не на пікселях із сусідніх блоків, потреба в буферах в кодері і/або декодері може, в деяких випадках, бути зменшена або зовсім усунена.

У деяких випадках, відповідно до методик цього розкриття, піднабір пікселів з поточного кодованого блока може бути вибраний так, що метрики обчислюються з використанням тільки значень пікселів поточного блока і лівих і правих сусідніх блоків, але не значень пікселів з верхніх сусідніх блоків або нижніх сусідніх блоків. Внаслідок порядку растрового сканування, що використовується при кодуванні відеоблоків, лінійні буфери для верхніх і нижніх сусідніх блоків прагнуть до збереження набагато більшої кількості піксельних значень, ніж лінійні буфери для зберігання піксельних значень лівих і правих сусідніх блоків.

Відповідно до методик цього розкриття елемент фільтра, такий як адаптивний внутрішньоконтурний фільтр, може бути сконфігурований з можливістю використовувати декілька фільтрів, основуючись на мультиметричній відповідності фільтрів. Декілька фільтрів можуть використовуватися разом з єдиним введенням або множинними введеннями. Як буде описано більш детально нижче, множинні введення, описані в даному розкритті, загалом стосуються проміжних даних відеоблока або даних зображення, які здійснюються під час процесів кодування і декодування. Множинні введення, зв'язані із заданим відеоблоком, можуть включати в себе, наприклад, реконструйований блок або зображення (RI), заздалегідь деблокований реконструйований блок або зображення (pRI), блок або зображення прогнозування (PI) і/або квантоване зображення помилки прогнозування (EI). У схемі одиничного введення фільтр може застосовуватися тільки відносно одного з представленого вище введення, такого як RI. Також, як буде більш детально пояснено нижче, методики фільтрації можуть застосовуватися відносно CU різних розмірів з використанням схеми розділення квадродрева. При використанні декількох фільтрів з мультиметричною відповідністю фільтрів для CU, розділеного із застосуванням схеми розділення квадродрева, продуктивність кодування відео, як виміряно за допомогою одним або обома з коефіцієнта стиснення та якості відновленого відео, може бути поліпшена.

Для реалізації методик мультиметричної фільтрації, описаних вище, кодер підтримує визначення відповідності комбінацій діапазонів і фільтрів шляхом генерування, оновлення, збереження та іншими способами. Як один приклад комбінація першого діапазону для першої метрики і першого діапазону для другої метрики може приводитися у відповідність до першого фільтра. Комбінація першого діапазону для першої метрики і другого діапазону для другої метрики може також приводитися у відповідність до першого фільтра або ж може приводитися у відповідність до другого фільтра. Якщо перша метрика має вісім діапазонів і друга метрика має чотири діапазони, наприклад, тоді перша і друга метрики можуть мати тридцять дві комбінації діапазонів, і кожна з тридцяти двох комбінацій може приводитися у відповідність до фільтра. Кожна комбінація, однак, не обов'язково приводиться у відповідність до єдиного фільтра. Так,

тридцять дві комбінації можуть приводитися у відповідність до чотирьох фільтрів, вісьмох фільтрів, десятих фільтрів або деякої іншої кількості фільтрів. З метою застосувати ті самі фільтри, що і кодер, декодер може також підтримувати ту саму відповідність комбінацій діапазонів фільтрам.

Розкриття описує методики для сигналізації від кодера до декодера в кодованому бітовому потоці відповідності комбінацій діапазонів фільтрам. Приведення у відповідність може, наприклад, зв'язувати кожну комбінацію діапазонів з ідентифікацією фільтра (ID). Один простий спосіб сигналізувати про цю відповідність - використати одне кодове слово для кожного ID фільтра і потім для кожної комбінації діапазонів відправити кодове слово відповідного ID фільтра. Ця методика, однак, звичайно неефективна. Методики за даним розкриттям можуть використовувати кореляції в межах відповідності шляхом використання способів різницевого кодування. Комбінації діапазонів, які спільно використовують загальний діапазон, іноді використовують один і той самий фільтр. Як один приклад комбінація першого діапазону для першої метрики і першого діапазону для другої метрики і комбінація першого діапазону для першої метрики і другого діапазону для другої метрики спільно використовують загальний діапазон (перший діапазон першої метрики). Так, ці дві комбінації можуть в деяких випадках відповідати одному і тому самому ID фільтра. Використовуючи цю кореляцію, методики даного розкриття можуть зменшувати кількість бітів, необхідну для сигналізування відповідності комбінацій діапазонів різним ID фільтрів від кодера до декодера.

Додатково до сигналізації відповідності комбінацій діапазонів на множину ID фільтрів це розкриття також описує методики для сигналізації в кодованому бітовому потоці коефіцієнтів фільтра для фільтрів. Методики даного розкриття включають в себе використання способів різницевого кодування для сигналізації коефіцієнтів фільтра від кодера до декодера. Таким чином, коефіцієнти фільтра для другого фільтра можуть бути повідомлені декодеру як інформація різниці, де інформація різниці описує, як модифікувати коефіцієнти фільтра першого фільтра способом, який створює коефіцієнти фільтра другого фільтра. Методики різницевого кодування можуть бути більш ефективними (тобто можуть сприяти більшому збереженню бітів), коли коефіцієнти фільтра першого і другого фільтра більш схожі, в порівнянні з випадком коли коефіцієнти фільтра першого і другого фільтра менш схожі. Методики цього розкриття включають в себе визначення порядку послідовності, в якому необхідно сигналізувати коефіцієнти фільтра для фільтра. Черговості, визначені з використанням методик, описаних в цьому розкритті, можуть давати в результаті поліпшене різничеве кодування коефіцієнтів фільтра і, таким чином, можуть в деяких випадках давати в результаті економію бітів при сигналізації коефіцієнтів фільтра.

Хоча методики цього розкриття іноді можуть описуватися в зв'язку з внутрішньоконтурною фільтрацією, вони можуть застосовуватися до внутрішньоконтурної фільтрації, пост-контурної фільтрації та інших схем фільтрації, таких як фільтрація, що перемикається. Внутрішньоконтурна фільтрація загалом стосується фільтрації, при якій відфільтровані дані є частиною контурів кодування і декодування так, що відфільтровані дані використовуються для внутрішнього або зовнішнього кодування з прогнозуванням. Пост-контурна фільтрація стосується фільтрації, яка застосовується у відношенні реконструйованих відеоданих після контуру кодування. При пост-контурній фільтрації невідфільтровані дані, на противагу відфільтрованим даним, використовуються для внутрішнього або зовнішнього кодування з прогнозуванням. У деяких варіантах здійснення тип фільтрації може перемикається між пост-контурною фільтрацією і внутрішньоконтурною фільтрацією на, наприклад, покадровій основі, основі вирізка за вирізкою або іншій такій основі, і рішення про те, чи використовувати пост-контурну фільтрацію або внутрішньоконтурну фільтрацію, може сигналізуватися від кодера до декодера для кожного кадру, вирізки тощо. Методики цього розкриття не обмежуються внутрішньоконтурною фільтрацією або постфільтрацією і можуть ввести в дію широкий діапазон фільтрації, що застосовується під час кодування відео.

У цьому розкритті термін "кодування" стосується кодування або декодування. Подібним чином, термін "кодер" загалом стосується будь-якого відеокодера, відеодекодера або комбінованого кодера/декодера (кодека). Відповідно, термін "кодер" використовується тут відносно спеціального комп'ютерного пристрою або апарату, який здійснює кодування відео або декодування відео.

Додатково в цьому розкритті термін "фільтр" загалом стосується набору коефіцієнтів фільтра. Наприклад, фільтр  $3 \times 3$  може визначатися набором з 9 коефіцієнтів фільтра, фільтр  $5 \times 5$  може визначатися набором з 25 коефіцієнтів фільтра, фільтр  $9 \times 5$  визначатися набором з 45 коефіцієнтів фільтра і так далі. Термін "набір фільтрів" загалом стосується набору з більше ніж одного фільтра. Наприклад, набір з двох фільтрів  $3 \times 3$  може включати в себе перший набір

з 9 коефіцієнтів фільтра і другий набір з 9 коефіцієнтів фільтра. Відповідно до методик цього розкриття, для серії відеоблоків, таких як кадр, вирізка або найбільший елемент кодування (LCU), інформація, що ідентифікує набори фільтрів, сигналізується від кодера до декодера в заголовку серії відеоблоків. Термін "форма", який іноді називається "підтримкою фільтра", загалом стосується кількості рядків коефіцієнтів фільтра і кількості стовпців коефіцієнтів фільтра для конкретного фільтра. Наприклад,  $9 \times 9$  являє собою приклад першої форми,  $9 \times 5$  - це приклад другої форми, а  $5 \times 9$  - приклад третьої форми. У деяких випадках фільтри можуть приймати непрямокутні форми, включаючи ромбічні форми, ромбоподібні форми, круглі форми, подібні до круглих форм, шестикутні форми, восьмикутні форми, хрестоподібні форми, Х-подібні форми, Т-подібні форми, інші геометричні форми або численні інші форми або конфігурації.

Фіг. 1 являє собою блок-схему, яка ілюструє зразкову систему 110 кодування і декодування відео, яка може реалізовувати методики цього розкриття. Як показано на фіг. 1, система 110 включає в себе пристрій-джерело 112, який передає кодовані відеодані до пристрою-адресата 116 за допомогою каналу 115 зв'язку. Пристрій-джерело 112 і пристрій-адресат 116 можуть містити будь-який з широкого діапазону пристроїв. У деяких випадках пристрій-джерело 112 і пристрій-адресат 116 можуть містити бездротові портативні пристрої зв'язку, такі як так звані стільникові або супутникові радіотелефони. Методики цього розкриття, однак, які загалом більше застосовуються до фільтрації відеоданих, не обов'язково обмежуються бездротовими додатками та установками і можуть застосовуватися до небездротових пристроїв, що включають в себе можливості кодування і/або декодування відео.

У прикладі з фіг. 1 пристрій-джерело 112 включає в себе відеоджерело 120, відеокодер 122, модулятор/демодулятор (модем) 123 і передавач 124. Пристрій-адресат 116 включає в себе приймач 126, модем 127, відеодекодер 128 і пристрій 130 відображення. Відповідно до цього розкриття відеокодер 122 пристрою-джерела 112 може бути виконаний з можливістю вибирати один або більше наборів коефіцієнтів фільтра для множинних введень в процесі фільтрації відеоблока і потім кодувати вибраний один або більше наборів коефіцієнтів фільтра. Конкретні фільтри з одного або більше наборів коефіцієнтів фільтра можуть бути вибрані на основі однієї або більше метрик для одного або більше введень, і коефіцієнти фільтра можуть використовуватися для фільтрації одного або більше введень. Методики фільтрації цього розкриття загалом сумісні з будь-якими методиками кодування або сигналізації коефіцієнтів фільтра в кодованому бітовому потоці.

Відповідно до методик цього розкриття пристрій, який включає в себе відеокодер 122, може сигналізувати пристрою, що включає в себе відеодекодер 128, один або більше наборів коефіцієнтів фільтра для серії відеоблоків, як то кадр або вирізка. Для серії відеоблоків відеокодер 122 може, наприклад, сигналізувати один набір фільтрів, що підлягають використанню з усіма введеннями, або може сигналізувати декілька наборів фільтрів, що підлягають використанню з множинними введеннями (один набір на кожне введення, наприклад). Кожний відеоблок або CU в межах серії відеоблоків потім може містити додатковий синтаксис, щоб ідентифікувати, який фільтр або фільтри з набору фільтрів підлягають використанню для кожного введення цього відеоблока, або відповідно до методик цього розкриття, то, який фільтр або фільтри з набору фільтрів підлягають використанню, може бути визначено на основі двох або більше метрик, зв'язаних з одним або більше введеннями.

Точніше, відеокодер 122 пристрою-джерела 112 може вибирати один або більше наборів фільтрів для серії відеоблоків, застосовувати фільтри з набору (наборів) до пікселів або груп пікселів з введень, зв'язаних з CU серії відеоблоків, під час процесу кодування і потім кодувати набори фільтрів (тобто набори коефіцієнтів фільтра) для зв'язку з відеодекодером 128 пристрою-адресата 116. Відеокодер 122 може визначати одну або більше метрик, зв'язаних з введеннями CU, кодованих з метою вибрати, який фільтр (які фільтри) з набору (наборів) фільтрів використовувати з пікселями або групами пікселів для того конкретного CU. Відеокодер 122 може також сигналізувати відеодекодеру 128, як частина кодованого бітового потоку, відповідність комбінацій діапазонів фільтрам у наборі фільтрів.

На боці декодера, відеодекодер 128 може визначати коефіцієнти фільтра на основі інформації фільтра, прийнятої в синтаксисі бітового потоку. Відеодекодер 128 може декодувати коефіцієнти фільтра на основі прямого декодування або декодування з прогнозуванням залежно від того, як були кодовані коефіцієнти фільтра, що може бути сигналізовано як частина синтаксису бітового потоку. Додатково бітовий потік може включати в себе інформацію синтаксису опису фільтра, щоб описати фільтри для набору фільтрів. На основі синтаксису опису фільтра декодер 128 може реконструювати коефіцієнти фільтра, основуючись на додатковій інформації, прийнятій від кодера 122. Проілюстрована система 110 з фіг. 1 - тільки зразкова. Методики фільтрації цього розкриття можуть реалізовуватися будь-якими пристроями



кодування або декодування. Пристрій-джерело 112 і пристрій-адресат 116 - це тільки приклади пристроїв кодування, які можуть підтримувати подібні методики. Відеодекодер 128 також може визначати відповідність комбінацій діапазонів фільтрам на основі інформації фільтра, прийнятої в синтаксисі бітового потоку.

5 Відеодекодер 122 пристрою-джерела 112 може кодувати відеодані, прийняті від відеоджерела 120, використовуючи методики цього розкриття. Відеоджерело 120 може містити пристрій захоплення відео, такий як відеокамера, відеоархів, що містить раніше захоплене відеозображення, або відеосигнал поданий від провайдера відеоконтенту. Як додаткова  
10 альтернатива відеоджерело 120 може генерувати дані на основі комп'ютерної графіки як вихідне відео або в комбінації з "живим" відео, архівованим відео і згенерованим комп'ютером відео. У деяких випадках, якщо відеоджерело 120 є відеокамерою, пристрій-джерело 112 і пристрій-адресат 116 можуть формувати так звані камерофони або відеофони. У кожному випадку, захоплене, заздалегідь захоплене або згенероване комп'ютером відеозображення може бути кодоване відеокодером 122.

15 Коли відеодані кодовані відеокодером 122, інформація кодованого відео потім може модулюватися модемом 123 відповідно до стандарту зв'язку, наприклад, такого як множинний доступ з кодовим розділенням каналів (CDMA), множинний доступ з частотним розділенням каналів (FDMA), мультимплексування з ортогональним частотним розділенням каналів (OFDM), або будь-якого іншого стандарту або методики зв'язку і передаватися до пристрою-адресата  
20 116 за допомогою передавача 124. Модем 123 може включати в себе різні блоки змішування, фільтри, підсилювачі та інші компоненти, створені для модуляції сигналу. Передавач 124 може включати в себе схеми, створені для передачі даних, включаючи підсилювачі, фільтри і одну або більше антен.

Приймач 126 пристрою-адресата 116 приймає інформацію по каналу 115, і модем 127  
25 демодулює цю інформацію. Процес декодування відео, що виконується відеодекодером 128, може включати в себе фільтрацію, наприклад, як частину внутрішньоконтурного декодування або як етап постфільтрації, що йде за контуром декодування. Так чи інакше, набір фільтрів, що застосовується відеодекодером 128 для конкретної вирізки або кадру, може бути декодований з використанням методик цього розкриття. Інформація декодованого фільтра може включати в  
30 себе ідентифікацію синтаксису опису фільтра в кодованому бітовому потоці. Якщо, наприклад, кодування з прогнозуванням використовується для коефіцієнтів фільтра, схожість між різними коефіцієнтами фільтра може використовуватися, щоб скоротити кількість інформації, що переноситься по каналу 115. Зокрема, фільтр (тобто набір коефіцієнтів фільтра) може піддаватися кодуванню з прогнозуванням як значення різниці відносно іншого набору  
35 коефіцієнтів фільтра, зв'язаного з іншим фільтром. Інший фільтр може, наприклад, бути зв'язаним з іншими вирізкою або кадром. У такому випадку відеодекодер 128 може приймати кодований бітовий потік, що містить відеоблоки та інформацію фільтра, яка ідентифікує інший кадр або вирізку, з яким зв'язаний інший фільтр. Інформація фільтра також включає в себе значення різниці, які визначають поточний фільтр відносно фільтра іншого CU. Зокрема,  
40 значення різниці можуть містити значення різниці коефіцієнтів фільтра, які визначають коефіцієнти фільтра для поточного фільтра відносно коефіцієнтів фільтра іншого фільтра, що використовується для іншого CU.

Відеодекодер 128 декодує відеоблоки, генерує коефіцієнти фільтра і фільтрує декодовані  
45 відеоблоки на основі згенерованих коефіцієнтів фільтра. Відеодекодер 128 може генерувати коефіцієнти фільтра на основі синтаксису опису фільтра, витягнутого з бітового потоку. Декодовані та відфільтровані відеоблоки можуть бути зібрані у відеокадри, щоб сформувати декодовані відеодані. Пристрій 128 відображення відображає декодовані відеодані користувачу і може містити будь-який з різноманітних пристроїв відображення, таких як електронно-променева трубка (CRT), рідкокристалічний дисплей (LCD), плазмовий дисплей, дисплей на  
50 органічних світлодіодах (OLED), або пристрій відображення іншого типу.

Канал 115 зв'язку може містити будь-який бездротовий або дротовий засіб зв'язку, такий як радіочастотний (RF) спектр або одна або більше фізичних ліній передачі або будь-яку  
55 комбінацію бездротових і дротових засобів. Канал 115 зв'язку може формувати частину пакетної мережі, такої як локальна мережа, глобальна мережа, така як Інтернет. Канал 115 зв'язку загалом являє собою будь-яке прийнятне середовище зв'язку або сукупність різних середовищ зв'язку для передачі відеоданих від пристрою-джерела 112 або пристрою-адресата 116. Знову, фіг. 1 - тільки приклад і методики цього розкриття можуть застосовуватися до установок кодування відео (наприклад, кодування відео або декодування відео), які не обов'язково включають в себе яке-небудь повідомлення даних між пристроями кодування і декодування. В

інших прикладах дані могли б бути витягнуті з локального запам'ятовуючого пристрою, направлені по мережі тощо.

Як альтернатива кодовані дані можуть виводитися від відеокодера 122 до пристрою 132 зберігання даних. Подібним чином, кодовані дані можуть бути доступні із запам'ятовуючого пристрою 132 відеодекодеру 128. Запам'ятовуючий пристрій 132 може включати в себе будь-який з розподілених або локально доступних засобів для зберігання даних, таких як жорсткий диск, диски Blu-ray, DVD, CD-ROM, флеш-пам'ять, енергозалежний або енергонезалежний запам'ятовуючий пристрій, або будь-яких інших прийнятних цифрових носіїв для зберігання кодованих відеоданих. У додатковому прикладі пристрій 132 зберігання даних може відповідати файловому серверу або іншому проміжному пристрою зберігання даних, який може зберігати кодоване відео, згенероване пристроєм-джерелом 112. Пристрій-адресат 116 може здійснювати доступ до збережених відеоданих з пристрою 132 зберігання даних за допомогою потокового режиму або завантаження. Файловий сервер може бути сервером будь-якого типу, здатним зберігати кодовані відеодані і передавати ці кодовані відеодані до пристрою-адресата 116. Зразкові файлові сервери включають в себе веб-сервер (наприклад, для веб-сайта), FTP-сервер, пристрої-накопичувачі (NAS), що підключаються до мережі або локальний накопичувач на дисках. Пристрій-адресат 14 може здійснювати доступ до кодованих відеоданих за допомогою будь-якого стандартного інформаційного з'єднання, включаючи інтернет-з'єднання. Це може включати в себе бездротовий канал (наприклад, Wi-Fi-з'єднання), дротове з'єднання (наприклад, DSL, кабельний модем тощо) або комбінацію обох варіантів, яка є прийнятною для доступу до кодованих відеоданих, що зберігаються на файловому сервері. Передача кодованих відеоданих від пристрою 132 зберігання даних може бути потоковою передачею, передачею завантаження або їх комбінацією.

Методики цього розкриття не обов'язково обмежуються бездротовими додатками або установками. Методики можуть застосовуватися відносно кодування відео при підтримці будь-якого з різноманіття мультимедійних додатків, таких як ефірне телевізійне мовлення, кабельні телепередачі, супутникові телепередачі, передачі потокового відео, наприклад, через Інтернет, кодування цифрового відео для зберігання на засобі зберігання даних, декодування цифрового відео, збереженого на засобі зберігання даних, або інших додатків. У деяких прикладах система 110 може бути виконана з можливістю підтримувати односторонню або двосторонню передачу відео, щоб підтримувати додатки, такі як потокове відео, відтворення відеозапису, телевізійне мовлення і/або відеотелефонія.

Відеокoder 122 та відеодекодер 128 можуть функціонувати відповідно до стандарту стиснення відео, такого як стандарт ITU-T H.264, який як альтернатива іменується MPEG-4, Частина 10, вдосконалене кодування відео (AVC), який буде використовуватися в частинах цього розкриття в роз'яснювальних цілях. Однак багато які з методик цього розкриття можуть легко застосовуватися до будь-якого з різноманіття стандартів кодування відео, включаючи стандарт HEVC, що нещодавно з'явився. Загалом, будь-який стандарт, який допускає фільтрацію в кодері та декодері, може одержувати користь з різних аспектів ідеї цього розкриття.

Хоча це і не показано на фіг. 1, в деяких аспектах відеокoder 122 і відеодекодер 128 - кожний з них - можуть об'єднуватися з аудіокoderом і декодером і можуть включати в себе відповідні елементи MUX-DEMUX або інше апаратне та програмне забезпечення, щоб здійснити кодування як аудіо, так і відео в загальному потоці даних або окремих потоках даних. Якщо застосовно, елементи MUX-DEMUX можуть відповідати протоколу мультиплексування ITU H.223 або іншим протоколам, таким як протокол користувацьких дейтаграм (UDP).

Відеокoder 122 та відеодекодер 128 - кожний з них - можуть бути реалізовані як один або більше мікропроцесорів, цифрових процесорів сигналів (DSP), спеціалізованих інтегральних схем (ASIC), програмованих користувачем вентильних матриць (FPGA), блоків дискретної логіки, програмного забезпечення, апаратного забезпечення, програмно-апаратного забезпечення або будь-яких їх комбінацій. Кожний з відеокодера 122 та відеодекодера 128 може бути увімкнений в один або більше кодерів або декодерів, кожний з яких може бути вбудований як частина комбінованого кодера/декодера (CODEC) у відповідний мобільний пристрій, абонентський пристрій, пристрій мовлення, сервер тощо.

У деяких випадках пристрої 112, 116 можуть функціонувати в основному симетрично. Наприклад, кожний з пристроїв 112, 116 може включати в себе компоненти кодування і декодування відео. Отже, система 110 може підтримувати односторонню або двосторонню передачу відео між пристроями 112, 116, наприклад, для потокового відео, відтворення відеозапису, телевізійного мовлення або відеотелефонії.

Під час процесу кодування відеокодер 122 може виконувати деяку кількість методик або етапів кодування. Загалом, відеокодер 122 здійснює функціонування відносно відеоблоків, з окремими відеокадрами, щоб кодувати відеодані. В одному прикладі відеоблок може відповідати макроблоку або фрагменту макроблока. Макроблоки являють собою один тип відеоблока, визначеного стандартом ITU H.264 та іншими стандартами. Макроблоки звичайно стосуються блоків даних  $16 \times 16$ , хоча цей термін також іноді використовується загалом для будь-якого відеоблока розміру  $N \times N$  або  $N \times M$ . Стандарт ITU-T H.264 підтримує внутрішнє прогнозування в блоках різних розмірів, таких як  $16 \times 16$ ,  $8 \times 8$ ,  $4 \times 4$  для компонент яскравості та  $8 \times 8$  для компонент кольоровості, а також зовнішнє прогнозування в блоках різних розмірів, таких як  $16 \times 16$ ,  $16 \times 8$ ,  $8 \times 16$ ,  $8 \times 8$ ,  $8 \times 4$ ,  $4 \times 8$  та  $4 \times 4$  для компонент яскравості, і відповідних масштабованих розмірів для компонент кольоровості. У цьому розкритті "N×N" стосується піксельних вимірювань блока з точки зору вертикальних і горизонтальних вимірювань, наприклад,  $16 \times 16$  пікселів. Загалом, блок  $16 \times 16$  буде містити 16 пікселів у вертикальному напрямку і 16 пікселів у горизонтальному напрямку. Подібним чином блок  $N \times N$  загалом містить N пікселів у вертикальному напрямку і N пікселів у горизонтальному напрямку, де N представляє позитивне ціле значення. Пікселі в блоці можуть бути розподілені в рядки і стовпці.

Стандарт HEVC, що з'явився, визначає нові терміни для відеоблоків. Зокрема, відеоблоки (або їх фрагменти) можуть іменуватися "елементами кодування" (або CU). З використанням стандарту HEVC найбільші елементи кодування (LCU) можуть розділятися на менші CU відповідно до схеми розбиття квадродререва, і різні CU, які визначаються в цій схемі, можуть додатково фрагментуватися на так звані елементи прогнозування (PU). LCU, CU та PU - це все відеоблоки в межах значення цього розкриття. Також можуть використовуватися інші типи відеоблоків, сумісні зі стандартом HEVC або іншими стандартами кодування відео. Так, фраза "відеоблоки" стосується відеоблока будь-якого розміру. Окремі CU можуть бути включені в компоненти яскравості та масштабовані розміри для компонент кольоровості для заданого пікселя, хоча інші колірні простори також можуть використовуватися.

Відеоблоки можуть мати фіксовані або змінні розміри і можуть розрізняватися за розміром відповідно до визначеного стандарту кодування. Кожний відеокадр може включати в себе множину вирізок. Кожна вирізка може включати в себе множину відеоблоків, які можуть бути розподілені на фрагменти, що також називаються субблоками. Відповідно до схеми фрагментування квадродререва, посилання на яку було зроблене вище і яка більш детально буде описана нижче, перший CU  $N/2 \times N/2$  може містити субблок LCU  $N \times N$ , другий CU  $N/4 \times N/4$  може також містити субблок першого CU. PU  $N/8 \times N/8$  може містити субблок другого CU. Подібним чином, як додатковий приклад розміри блоків, які менше, ніж  $16 \times 16$ , можуть іменуватися фрагментами відеоблока  $16 \times 16$  або субблоками відеоблока  $16 \times 16$ . Подібно до цього, для блока  $N \times N$ , розміри блока менше, ніж  $N \times N$ , можуть іменуватися фрагментами  $N \times N$  або субблоками  $N \times N$ . Відеоблоки можуть містити блоки даних пікселя в ділянці пікселя або блоки коефіцієнтів перетворення в ділянці перетворення, наприклад, здійснюючи застосування перетворення, таке як дискретне конусне перетворення (DCT), цілочисельне перетворення, вейвлетне перетворення або концептуально подібне перетворення, до залишкового відеоблока даних, що представляє різницю пікселів між кодованими відеоблоками і прогнозованими відеоблоками. У деяких випадках відеоблок може містити блоки квантованих коефіцієнтів перетворення в ділянці перетворення.

Дані синтаксису в межах бітового потоку можуть визначати LCU для кадру або вирізки, який є найбільшим елементом кодування з точки зору кількості пікселів для цього кадру або вирізки. Загалом, LCU або CU має схоже призначення для макроблока, кодованого згідно з H.264, за винятком того, що LCU та CU не мають визначеної відмінності за розміром. Замість цього розмір LCU може визначатися на покадровій основі або на основі вирізка за вирізкою, і LCU може розбиватися на CU. Загалом, посилання в цьому розкритті на CU можуть відноситися до LCU зображення або до суб-CU LCU. LCU може бути розбитий на суб-CU, і кожний суб-CU може бути розбитий на суб-CU. Дані синтаксису для бітового потоку можуть визначати максимальну кількість разів, на яку LCU може бути розбитий, що називається глибиною CU. Відповідно, бітовий потік також може визначати найменший елемент кодування (SCU). Це розкриття також використовує терміни "блок" і "відеоблок", щоб посылатися на будь-який з LCU, CU, PU, SCU та TU.

Як було представлено вище, LCU може бути зв'язаний зі структурою даних квадродререва. Загалом, структура даних квадродререва включає в себе один вузол на кожний CU, де кореневий вузол відповідає LCU. Якщо CU розбивається на чотири суб-CU, вузол, що відповідає CU, включає в себе чотири листових вузли, кожний з яких відповідає одному з суб-CU. Кожний вузол структури даних квадродререва може забезпечувати дані синтаксису для

відповідного CU. Наприклад, вузол в квадродереві може включати в себе прапор розбиття, що вказує, чи розбивається CU, що відповідає вузлу, на суб-CU. Компоненти синтаксису для CU можуть визначатися рекурсивно і можуть залежати від того, чи розбивається CU на суб-CU.

CU, який не розбивається, може включати в себе один або більше елементів прогнозування (PU). Загалом, PU представляє всі або частину відповідного CU і включає в себе дані для витягання опорного зразка для PU. Наприклад, коли PU кодований у внутрішньому режимі, він може включати в себе дані, що описують режим внутрішнього прогнозування для PU. Як інший приклад, коли PU, кодований у зовнішньому режимі, він може включати в себе дані, що визначають вектор руху для PU. Дані, що визначають вектор руху, можуть описувати, наприклад, горизонтальну компоненту вектора руху, вертикальну компоненту вектора руху, розрізнення для вектора руху (наприклад, точність до чверті пікселя або точність до однієї восьмої пікселя), опорний кадр, на який вказує вектор руху, і/або опорний список (наприклад, список 0 або список 1) для вектора руху. Дані для CU, що визначають PU (або більше PU), можуть також описувати, наприклад, фрагментування CU на один або більше PU. Режими фрагментування можуть відрізнятися в плані того, чи є CU некодованим, кодованим в режимі внутрішнього прогнозування або кодованим в режимі зовнішнього прогнозування.

CU, що має один або більше PU, також може включати в себе один або більше елементів перетворення (TU). TU містять структуру даних, яка включає в себе залишкові коефіцієнти перетворення, які звичайно квантуються. Зокрема, після прогнозування з використанням PU відеокoder може обчислювати залишкові значення для фрагмента CU, що відповідає PU. Залишкові значення можуть перетворюватися, квантуватися, скануватися та зберігатися в TU, який може мати змінні розміри, що відповідають розміру перетворення, яке було здійснене. Відповідно, TU не обов'язково обмежується розміром PU. Так, TU можуть бути більше або менше, ніж відповідні PU для одного і того самого CU. У деяких прикладах максимальний розмір TU може бути розміром відповідного CU. Ще раз варто зазначити, що TU може містити структури даних, які включають в себе залишкові коефіцієнти перетворення, зв'язані із заданим CU.

Фіг. 2A та фіг. 2B являють собою концептуальні схеми, що ілюструють зразкове квадродереве 250 і відповідний LCU 272. Фіг. 2A ілюструє зразкове квадродереве 250, яке включає в себе вузли, розподілені ієрархічно. Кожний вузол в квадродереві, такому як квадродереве 250, може бути кінцевою вершиною без потомків або мати вузли-потомки. У прикладі фіг. 2A квадродереве 250 включає в себе кореневий вузол 252. Кореневий вузол 252 має чотири вузли-потомки 256A-256C (кінцеві вершини 256) і вузол 254. Оскільки вузол 254 не є кінцевою вершиною, він включає в себе чотири вузли-потомки, які в цьому прикладі - кінцеві вершини 258A-258D (кінцеві вершини 258).

Квадродереве 250 може включати в себе дані, що описують характеристики відповідного LCU, такого як LCU 272, наприклад. Наприклад, квадродереве 250 за своєю структурою може описувати розбиття LCU на суб-CU. Припустимо, що LCU 272 має розмір  $2N \times 2N$ . LCU 272 в цьому прикладі має чотири суб-CU 276A-276C (суб-CU 276) і 274, кожний розміру  $N \times N$ . Суб-CU 274 додатково розбивається на чотири суб-CU 278A-278D (суб-CU 278), кожний розміру  $N/2 \times N/2$ . Структура квадродерева 250 відповідає розбиттю LCU 272 в цьому прикладі. Тобто кореневий вузол 252 відповідає LCU 272, кінцеві вершини 256 відповідають суб-CU 276, вузол 254 відповідає суб-CU 274, а кінцеві вершини 258 відповідають суб-CU 278.

Дані для вузлів квадродерева 250 можуть описувати, чи розбивається CU, що відповідає вузлу. Якщо CU розбивається, в квадродереві 250 можуть бути присутніми чотири додаткових вузли. У деяких прикладах вузол квадродерева може бути реалізований подібно до того, як в наступному псевдокоді:

```
quadtree_node {
    boolean split_flag(1);
    // signaling data
    if (split_flag) {
        quadtree_node child1;
        quadtree_node child2;
        quadtree_node child3;
        quadtree_node child4;
    }
}
вузол_квадродерева {
    булевий прапор_розбиття(1);
    // повідомлення даних
```

```

якщо (прапор_розбиття) {
    потомок1 вузла_квадродерева;
    потомок2 вузла_квадродерева;
    потомок3 вузла_квадродерева;
5    потомок4 вузла_квадродерева;
}
}

```

Значення прапора\_розбиття може бути однобітовим значенням, що показує, чи розбивається CU, який відповідає поточному вузлу. Якщо CU не розбивається, значення прапора\_розбиття може бути "0", тоді як якщо CU розбивається, значення прапора\_розбиття може бути "1". Що стосується прикладу квадродерева 250, масив значень прапора розбиття може бути 1010000.

У деяких прикладах кожний з суб-CU 276 та суб-CU 278 може бути підданим кодуванню з внутрішнім прогнозуванням з використанням одного і того самого режиму внутрішнього прогнозування. Відповідно, відеокодер 122 може забезпечувати індикацію режиму внутрішнього прогнозування в кореновому вузлі 252. Більше того, визначені розміри суб-CU можуть мати множинні можливі перетворення для конкретного режиму внутрішнього прогнозування. Відеокодер 122 може забезпечувати індикацію перетворення, що підлягає використанню для таких суб-CU в кореновому вузлі 252. Наприклад, суб-CU розміру  $N/2 \times N/2$  може мати множину доступних можливих перетворень. Відеокодер 122 може сигналізувати перетворення, яке треба використовувати в кореновому вузлі 252. Відповідно, відеокодер 128 може визначати перетворення, яке потрібно застосувати до суб-CU 278, на основі режиму внутрішнього прогнозування, сигналізованого в кореновому вузлі 252, і перетворення, сигналізованого в кореновому вузлі 252.

Як такий відеокодер 122 не потребує сигналізації перетворень, що підлягають застосуванню до суб-CU 276 та суб-CU 278, в кінцевих вершинах 256 та кінцевих вершинах 258, але може замість цього просто сигналізувати режим внутрішнього прогнозування і, в деяких прикладах, перетворення, що підлягає застосуванню до визначених розмірів суб-CU, в кореновому вузлі 252, відповідно до методик цього розкриття. Таким чином, ці методики можуть знизити витрати на сигналізування функцій перетворення для кожного суб-CU LCU, такого як LCU 272.

У деяких прикладах режими внутрішнього прогнозування для суб-CU 276 і/або суб-CU 278 можуть бути відмінними від режимів внутрішнього прогнозування для LCU 272. Відеокодер 122 та відеокодер 130 можуть бути сконфігуровані з функціями, які приводять у відповідність режим внутрішнього прогнозування, сигналізований в кореновому вузлі 252, до доступного режиму внутрішнього прогнозування для суб-CU 276 і/або суб-CU 278. Функція може забезпечувати відповідність "множина-до-одного" режимів внутрішнього прогнозування, доступних для LCU 272, режимам внутрішнього прогнозування для суб-CU 276 і/або суб-CU 278.

Вирізка може бути розділена на відеоблоки (або LCU), і кожний відеоблок може бути фрагментований відповідно до структури квадродерева, описаної відносно фіг. 2A - фіг. 2B. Додатково, як показано на фіг. 2C, субблоки квадродерева, відмічені "включено" ("ON") можуть бути відфільтровані циклічними фільтрами, що описуються тут, тоді як субблоки квадродерева, відмічені "вимкнено" ("OFF"), можуть не бути відфільтровані. Рішення про те, фільтрувати чи ні заданий блок або субблок, може бути прийняте в кодері шляхом порівняння відфільтрованого результату і невідфільтрованого результату відносно оригінального блока, що піддається кодуванню. Фіг. 2D - це дерево рішень, що представляє рішення про фрагментування, що приводить до фрагментування квадродерева, показаного на фіг. 2C. Дійсна фільтрація, що застосовується до будь-яких пікселів для "включених" блоків, може визначатися на основі метрик, що обговорюються тут.

Зокрема, фіг. 2C може представляти відносно великий відеоблок, який фрагментується відповідно до схеми розбиття квадродерева на менші відеоблоки змінних розмірів. Кожний відеоблок позначається (увімкнений або вимкнений) на фіг. 2C, щоб проілюструвати, чи повинна фільтрація застосовуватися або обминути даний відеоблок. Відеокодер може визначити цю відповідність фільтра шляхом порівняння відфільтрованих і невідфільтрованих версій кожного відеоблока з оригінальним відеоблоком, що кодується.

Ще раз зазначимо, що фіг. 2D - це дерево рішень, що відповідають рішенням про фрагментування, які призводять до фрагментування квадродерева, показаного на фіг. 2C. На фіг. 2D кожне коло може відповідати CU. Якщо коло включає в себе прапор "1", то CU додатково фрагментується на ще чотири CU, але якщо коло включає в себе прапор "0", то CU більше не фрагментується. Кожне коло (наприклад, відповідний CU), також включає в себе відповідний ромб. Якщо прапор в ромбі для заданого CU встановлений на 1, то фільтрація

переходить на "включено" для цього CU, але якщо прапор в ромбі для заданого CU встановлений на 0, то фільтрація вимикається. Таким чином, фіг. 2C та фіг. 2D можуть окремо або разом розглядатися як схема фільтра, яка може бути згенерована в кодері та повідомлена до декодера щонайменше один раз за вирізку кодованих відеоданих, щоб повідомити рівень

фрагментування квадродерева для заданого відеоблока (наприклад, LCU), чи застосовувати фільтрацію до кожного фрагментованого відеоблока (наприклад, кожного CU в межах LCU). Менші відеоблоки можуть забезпечувати краще розрізнення і можуть використовуватися для розташування відеокадру, які включають в себе високий рівень деталей. Більші відеоблоки можуть забезпечувати більш високу ефективність кодування і можуть використовуватися для точок розташування відеокадру, які включають в себе низький рівень деталей. Вирізка може розглядатися як множина відеоблоків і/або субблоків. Кожна вирізка може бути незалежно декодованою серією відеоблоків відеокадру. Як альтернатива самі кадри можуть бути декодованими серіями відеоблоків. Термін "серія відеоблоків" може відноситися до будь-якої незалежно декодованої частини відеокадру, такої як цілий кадр, вирізка кадру, група зображень (GOP), що також іменується послідовністю, або іншого незалежно декодованого елементу, визначеного відповідно до застосовних методик кодування. Аспекти цього розкриття могли б бути описані з посиланням на кадри або вирізки, але такі посилання лише зразкові. Потрібно розуміти, що загалом будь-яка серія відеоблоків може використовуватися замість кадру або вирізки.

Дані синтаксису можуть бути визначені на основі наявності в кожному кодованому елементі так, що кожний CU включає в себе відповідні дані синтаксису. Інформація фільтра, описана тут, може бути частиною такого синтаксису для CU, але більш ймовірно могла б бути частиною синтаксису для серії відеоблоків, таких як кадр, вирізка, GOP, LCU або послідовність відеокадрів, ніж для CU. Дані синтаксису можуть вказувати набір або набори фільтрів, що підлягають використанню з CU вирізки або кадру. Додатково не вся інформація фільтра обов'язково повинна бути включена в заголовок загальної серії відеоблоків. Наприклад, синтаксис опису фільтра може передаватися в заголовку кадру, тоді як інша інформація фільтра сигналізується в заголовку для LCU.

Відеокодер 122 може здійснювати кодування з прогнозуванням, в якому відеоблок, що кодується, порівнюється з прогнозованим кадром (або іншим CU), щоб ідентифікувати прогнозований блок. Відмінності між поточним відеоблоком, що кодується, і прогнозованим блоком кодуються як залишковий блок, і синтаксис прогнозування використовується, щоб ідентифікувати прогнозований блок. Залишковий блок може бути перетворений і квантований. Методики перетворення можуть містити процес DCT або концептуально подібний процес, цілочисельні перетворення, вейвлетні перетворення або інші типи перетворень. У процесі DCT, як приклад, процес перетворення перетворює набір піксельних значень в коефіцієнти перетворення, які можуть представляти енергію піксельних значень в частотній ділянці. Квантування звичайно застосовується до коефіцієнтів фільтра і загалом включає в себе процес, який обмежує кількість бітів, зв'язаних з будь-яким заданим коефіцієнтом перетворення.

Після перетворення і квантування до квантованих і перетворених залишкових відеоблоків може застосовуватися ентропійне кодування. Синтаксичні компоненти, такі як інформація фільтра і вектори прогнозування, визначені під час кодування, також можуть бути включені в підданий ентропійному кодуванню бітовий потік для кожного CU. Загалом, ентропійне кодування містить один або більше процесів, які спільно стискають послідовність квантованих коефіцієнтів перетворення і/або іншу інформацію синтаксису. Методики сканування, такі як методики зигзагоподібного сканування, реалізуються відносно квантованих коефіцієнтів перетворення, наприклад, як частина процесу ентропійного кодування, щоб визначити один або більше перетворених в послідовну форму одновимірних векторів коефіцієнтів з двовимірних відеоблоків. Інші методики сканування, що включають в себе порядки сканування або адаптивні скани, також можуть використовуватися і, можливо, сигналізуватися в кодованому бітовому потоці. У будь-якому випадку скановані коефіцієнти потім піддаються ентропійному кодуванню нарівні з інформацією синтаксису, наприклад, за допомогою контекстно-адаптивного кодування із змінною довжиною (CAVLC) або контекстно-адаптивного двійкового арифметичного кодування (CABAC) або іншого процесу ентропійного кодування.

Як частина процесу кодування, кодовані відеоблоки можуть декодуватися з метою генерувати відеодані, що використовуються для подальшого кодування на основі прогнозування у відношенні подальших відеоблоків. На цьому етапі фільтрація може здійснюватися з метою поліпшити якість відео і, наприклад, усунути артефакти блочності з декодованого відео. Відфільтровані дані можуть використовуватися для прогнозування інших відеоблоків, в цьому випадку фільтрація називається "внутрішньоконтурною" фільтрацією. Як альтернатива

прогнозування інших відеоблоків може бути основане на невідфільтрованих даних, в цьому випадку фільтрація називається "постфільтрацією".

На покадровій основі, основі вирізка за вирізкою або на основі LCU-за-LCU відеокoder 122 може вибирати один або більше наборів фільтрів і на основі кодований елемент за кодованим елементом відеокoder може визначати, чи здійснювати фільтрацію. Для CU, які підлягають фільтрації, кодер може здійснювати фільтрацію на попиксельній або погруповій основі, де група могла б, наприклад бути блоком пікселів  $2 \times 2$  або блоком пікселів  $4 \times 4$ . Ці вибори можуть бути зроблені так, щоб підтримувати якість відео. Такі набори фільтрів можуть вибиратися із заздалегідь визначених наборів фільтрів або можуть адаптивно визначатися, щоб підтримувати якість відео. Як приклад відеокoder 122 може вибирати або визначати декілька наборів фільтрів для заданого кадру або вирізки так, що різні фільтри використовуються для різних пікселів або груп пікселів CU цього кадру або вирізки. Зокрема, для кожного введення, зв'язаного з CU, може бути визначено декілька наборів коефіцієнтів фільтра, і дві або більше метрик, зв'язаних з пікселями CU, можуть бути використані, щоб визначити, який фільтр з набору фільтрів застосовувати з такими пікселями або групами пікселів.

У деяких випадках відеокoder 122 може застосовувати декілька наборів коефіцієнтів фільтра і вибирати один або більше наборів, які створюють відео кращої якості з точки зору кількості спотворень між кодованим блоком та оригінальним блоком і/або найвищих рівнів стиснення. У будь-якому випадку, будучи одного разу вибраним, набір коефіцієнтів фільтра, що застосовується відеокoderом 122 для кожного CU, може бути кодований і повідомлений до відеокoderу 128 пристрою-адресата 118 так, що відеокoder 128 може застосовувати таку саму фільтрацію, що і застосована під час процесу кодування, для кожного заданого CU.

Коли дві або більше метрик використовуються для визначення того, який фільтр використовувати з конкретним введенням для CU, вибір фільтра для цього конкретного CU не обов'язково повинен повідомлятися відеокoderу 128. Навпаки, відеокoder 128 також може обчислювати дві або більше метрик і на основі інформації фільтра, раніше наданій відеокoderом 122, узгоджувати комбінацію з двох або більше метрик для конкретного фільтра.

Фіг. 3 являє собою блок-схему, яка ілюструє відеокoder 350, відповідно до даного розкриття. Відеокoder 350 може відповідати відеокoderу 122 пристрою 120 або відеокoderу іншого пристрою. Як показано на фіг. 3, відеокoder 350 включає в себе елемент 332 прогнозування, суматори 348 та 351 і запам'ятовуючий пристрій 334. Відеокoder 350 також включає в себе елемент 338 перетворення і елемент 340 квантування, як і елемент 342 оберненого квантування і елемент 344 оберненого перетворення. Відеокoder 350 також включає в себе фільтр 347 видалення блочності та елемент 349 адаптивного фільтра. Відеокoder 350 також включає в себе елемент 346 ентропійного кодування. Елемент 349 фільтра відеокodера 350 може здійснювати операції фільтрації і також може включати в себе елемент 353 вибору фільтра (FSU) для ідентифікації бажаного або переважного фільтра або набору фільтрів, що підлягають використанню для декодування. Елемент 349 фільтра також може генерувати інформацію фільтра, що ідентифікує вибрані фільтри так, що вибрані фільтри можуть ефективно повідомлятися як інформація фільтра до іншого пристрою, що підлягає використанню під час операції декодування.

Під час процесу кодування відеокoder 350 приймає відеоблок, такий як LCU, що підлягає кодуванню, і модуль 332 прогнозування реалізовує методики кодування з прогнозуванням у відношенні відеоблока. З використанням схеми фрагментування квадродревця, розглянутої вище, модуль 332 прогнозування може фрагментувати відеоблок і реалізовувати методики кодування з прогнозуванням у відношенні CU різних розмірів. Для зовнішнього кодування модуль 332 прогнозування порівнює відеоблок, що підлягає кодуванню, який включає в себе субблоки відеоблока, з різними блоками в одному або більше опорних відеокадрах або вирізок з метою визначити прогнозований блок. Для внутрішнього кодування модуль 332 прогнозування генерує блок прогнозування на основі суміжних даних в межах того самого CU. Модуль 332 прогнозування виводить прогнозований блок, і суматор 348 віднімає прогнозований блок з відеоблока, що кодується, щоб генерувати залишковий блок.

Для зовнішнього кодування модуль 332 прогнозування може містити елементи оцінки руху і компенсації руху, які ідентифікують вектор руху, який вказує на прогнозований блок і генерує прогнозований блок на основі вектора руху. Звичайно оцінка руху розглядається як процес генерування вектора руху, який оцінює рух. Наприклад, вектор руху може представляти зміщення прогнозованого блока в межах прогнозованого кадру відносно поточного блока, що кодується в межах поточного кадру. Компенсація руху звичайно розглядається як процес вибірки або генерування прогнозованого блока на основі вектора руху, визначеного оцінкою руху. Для внутрішнього кодування модуль 332 прогнозування генерує прогнозований блок на

основі суміжних даних в межах того самого CU. Один або більше режимів внутрішнього прогнозування можуть визначати, як може бути визначений блок внутрішнього прогнозування.

Після цього модуль 332 прогнозування виводить блок прогнозування і суматор 348 віднімає блок прогнозування з відеоблока, що кодується, щоб генерувати залишковий блок, елемент 338 перетворення застосовує перетворення до залишкового блока. Перетворення може містити дискретне косинусне перетворення (DCT) або концептуально подібне перетворення, таке як визначене стандартом кодування, таким як стандарт HEVC. Вейвлетні перетворення, цілочисельні перетворення, субсмугові перетворення або інші типи перетворень також можуть використовуватися. У будь-якому випадку, елемент 338 перетворення застосовує перетворення до залишкового блока, виробляючи блок залишкових коефіцієнтів перетворення. Перетворення може перетворити залишкову інформацію з піксельної ділянки в частотну ділянку.

Елемент 340 квантування потім квантує залишкові коефіцієнти перетворення, щоб додатково знизити швидкість передачі бітів. Елемент 340 квантування, наприклад, може обмежувати кількість бітів, що використовуються, щоб кодувати кожний з коефіцієнтів. Після квантування елемент 346 ентропійного кодування сканує квантований блок коефіцієнтів з двомірного представлення в один або більше перетворених в послідовну форму одновимірних векторів. Порядок сканування може бути заздалегідь запрограмованим так, щоб реалізовуватися у визначеному порядку (як то зигзагоподібне сканування, горизонтальне сканування, вертикальне сканування, комбінації або інший заздалегідь визначений порядок) або, можливо, адаптивно визначеним на основі попередньої статистики кодування.

Після цього процесу сканування елемент 346 ентропійного кодування кодує квантовані коефіцієнти перетворення (нарівні із синтаксичними компонентами) відповідно до методології ентропійного кодування, як то CAVLC або CABAC, щоб додатково стиснути дані. Синтаксичні компоненти, включені в ентропійно кодований бітовий потік, можуть включати в себе синтаксис прогнозування з модуля 332 прогнозування, такий як вектори руху для зовнішнього кодування або режими прогнозування для внутрішнього кодування. Синтаксичні компоненти, включені в ентропійно кодований бітовий потік, можуть також включати в себе інформацію фільтра від елемента 349 фільтра, яка може бути кодована способом, описаним тут.

CAVLC - це один тип методики ентропійного кодування, що підтримується стандартом ITU H.264/MPEG-4, AVC, який може застосовуватися на векторизованій основі елементом 346 ентропійного кодування. CAVLC використовує таблиці кодування із змінною довжиною (VLC) таким чином, який сприяє ефективному стисненню перетворених в послідовну форму "ранів" коефіцієнтів перетворення і/або синтаксичних компонентів. CABAC - це інший тип методики ентропійного кодування, що підтримується стандартом ITU H.264/MPEG-4, AVC, який може застосовуватися на векторизованій основі елементом 346 ентропійного кодування. CABAC включає в себе декілька етапів, включаючи перетворення в двійкову форму, вибір моделі контексту і двійкове арифметичне кодування. У цьому випадку елемент 346 ентропійного кодування кодує коефіцієнти перетворення і синтаксичні компоненти відповідно до CABAC. Подібно до стандарту ITU H.264/MPEG-4, AVC, стандарт HEVC, що з'явився, також може підтримувати як ентропійне кодування CAVLC, так і ентропійне кодування CABAC. Крім того, існують і багато інших типів методик ентропійного кодування, і нові методики ентропійного кодування, ймовірно, з'являться в майбутньому. Це розкриття не обмежується якою-небудь конкретною методикою ентропійного кодування.

Після ентропійного кодування елементом 346 ентропійного кодування кодоване відео може передаватися до іншого пристрою або архівуватися для передачі або витягання пізніше. Знову варто зазначити, що кодоване відео може містити ентропійно кодовані вектори і різний синтаксис, який може бути використаний декодером, щоб належним чином конфігурувати процес декодування. Елемент 342 оберненого квантування та елемент 344 оберненого перетворення здійснюють обернене квантування та обернене перетворення відповідно, щоб реконструювати залишковий блок в ділянці пікселя. Суматор 351 додає реконструйований залишковий блок до блока прогнозування, створеного модулем 332 прогнозування, щоб створити заздалегідь деблокований реконструйований відеоблок, що іноді іменується заздалегідь деблокованим реконструйованим зображенням. Фільтр 347 видалення блочності може застосовувати фільтрацію до заздалегідь деблокованого реконструйованого відеоблока, щоб поліпшити якість відео шляхом усунення блочності та інших артефактів. Виведення деблокуючого фільтра 347 може іменуватися згодом деблокованим відеоблоком, реконструйованим відеоблоком або реконструйованим зображенням.

Модуль 349 фільтра може бути виконаний з можливістю приймати єдине введення або множинні введення. У прикладі з фіг. 3 модуль 349 фільтра приймає як введення згодом деблоковане реконструйоване зображення (RI), заздалегідь деблоковане реконструйоване



зображення (pRI), зображення прогнозування (PI) і реконструйований залишковий блок (EI). Модуль 349 фільтра може використовувати будь-які з цих введень, як окремо, так і в поєднанні, щоб створити реконструйоване зображення, щоб зберігати в пам'яті 334. Додатково, як буде нижче розглядатися більш детально, на основі двох або більше метрик один або більше

5 фільтрів можуть бути вибрані для застосування до введення (введень). В одному прикладі виведення модуля 349 фільтра може бути одним додатковим фільтром, що застосовується до RI. В іншому прикладі виведення модуля 349 фільтра може бути одним додатковим фільтром, що застосовується до pRI. В інших прикладах, однак, виведення модуля 349 фільтра може бути

10 основане на множинних введеннях. Наприклад, модуль 349 фільтра може застосовувати перший фільтр до pRI і потім використовувати відфільтровану версію pRI спільно з відфільтрованими версіями EI та PI, щоб створити реконструйоване зображення. У випадках, коли виведення модуля 349 фільтра - це продукт одного додаткового фільтра, що застосовується до єдиного введення, модуль 349 фільтра може фактично застосовувати

15 фільтри до інших введень, але ті фільтри могли б мати всі нульові коефіцієнти. Подібним чином, якщо виведення модуля 349 фільтра - це продукт застосування трьох фільтрів до трьох введень, модуль 349 фільтра може фактично застосовувати фільтр до четвертого введення, але той фільтр може мати всі нульові коефіцієнти.

Модуль 349 фільтра також може бути виконаний з можливістю приймати єдине введення. Наприклад, хоча фіг. 3 показує PI, EI, pRI та RI такими, що вводяться в модуль 349 фільтра, в

20 деяких варіантах здійснення RI може бути єдиним введенням, прийнятим модулем 349 фільтра. У такому варіанті здійснення модуль 349 фільтра може застосовувати фільтр до RI так, що відфільтрована версія RI більше схожа з оригінальним зображенням, ніж невідфільтрована версія RI. В інших варіантах здійснення модуль 349 фільтра і фільтр 347 видалення блочності можуть бути скомбіновані в єдиний елемент фільтрації, який застосовує фільтрацію до pRI.

25 Методики цього розкриття, які загалом стосуються побудови відповідності фільтрів на мультиметричній основі, сумісні як зі схемами фільтрації з єдиним введенням, так і зі схемами фільтрації з множиною введень, які використовують множинні фільтри.

Фільтрація модулем 349 фільтра може поліпшити стиснення шляхом генерування прогнозованих відеоблоків, які більш точно відповідають відеоблокам, що кодуються, ніж

30 невідфільтровані прогнозовані відеоблоки. Після фільтрації реконструйований відеоблок може використовуватися модулем 332 прогнозування як опорний блок, щоб піддати зовнішньому кодуванню блок в подальшому відеокадрі або інший CU. Хоча модуль 349 фільтра і показаний "внутрішньоконтурним", методики цього розкриття також можуть застосовуватися з пост-фільтрами, в цьому випадку невідфільтровані дані (на відміну від відфільтрованих даних) могли б використовуватися з метою прогнозування даних в подальших CU.

35

Для серії відеоблоків, такої як вирізка або кадр, модуль 349 фільтра може вибирати набори фільтрів для кожного введення способом, який підтримує якість відео. Наприклад, модуль 349

40 фільтра може вибирати набори фільтрів із заздалегідь заданих наборів коефіцієнтів або може адаптивно задавати фільтри, щоб підтримувати якість відео або поліпшене стиснення. Модуль 349 фільтра може вибирати або задавати один або більше наборів фільтрів для заданого CU так, що один і той самий набір (набори) фільтрів використовується (використовуються) для пікселів з різних відеоблоків того CU. Для конкретного кадру, вирізка або LCU модуль 349

45 фільтра може застосовувати декілька наборів фільтрів до множинних введень, і FSU 353 може вибирати набір, який створює відео кращої якості або найвищі рівні стиснення. Як альтернатива FSU 353 може готувати новий фільтр шляхом аналізу автокореляцій і крос-кореляцій між множинними введеннями та оригінальним зображенням. Новий набір фільтрів може, наприклад, бути визначений шляхом вирішення рівнянь Вінера - Хопфа на основі авто- і крос-кореляцій. Незалежно від того, чи підготовлюється новий набір фільтрів або вибирається існуючий набір

50 фільтрів, модуль 349 фільтра генерує синтаксис для включення в бітовий потік, що дозволяє декодеру також ідентифікувати набір або набори фільтрів, що підлягають використанню для конкретного кадру або вирізки.

Відповідно до цього розкриття для кожного пікселя CU в межах серії відеоблоків модуль 349

55 фільтра може вибирати, який фільтр з набору фільтрів використовувати, на основі двох або більше метрик, які визначають кількість властивостей, зв'язаних з одним або більше наборами пікселів у межах CU. Таким чином, FSU 353 може визначати набори фільтрів для кодованого блока більш високого рівня, як то кадр або вирізка, тоді як модуль 349 фільтра визначає, який фільтр (які фільтри) з набору (наборів) підлягає використанню для конкретного пікселя кодованого елемента більш низького рівня, на основі двох або більше метрик, зв'язаних з пікселями цього кодованого елемента більш низького рівня.

Набір М-фільтрів може використовуватися для кожного введення. Залежно від переваг виконання М може, наприклад, бути лише 2 або майже як 16, або навіть більше. Велика кількість фільтрів на введення може поліпшити якість відео, але також може збільшити витрати, зв'язані з наборами сигналізації у фільтрів від кодера до декодера. Набір М-фільтрів може визначатися FSU 353, як описується вище, і сигналізуватися до декодера для кожного кадру або вирізки. Схема сегментації може використовуватися для індикації того, як сегментуються CU, і того, чи підлягає фільтрації конкретний піделемент CU. Схема сегментації може, наприклад, включати в себе масив прапорів розбиття для суб-CU, як описується вище, як і додатковий біт, що сигналізує про те, чи підлягає кожний суб-CU фільтрації. Для кожного введення, зв'язаного з пікселем CU, який підлягає фільтрації, може бути вибраний конкретний фільтр з набору фільтрів на основі двох або більше метрик. Комбінації значень для двох або більше метрик можуть бути індексовані для конкретних фільтрів з набору М-фільтрів.

Фіг. 4А являє собою концептуальну схему, що ілюструє діапазони значень для двох метрик, індексованих для фільтрів з набору фільтрів. Конкретний приклад з фіг. 4А демонструє вісім фільтрів (тобто Фільтр 1, Фільтр 2 ... Фільтр 8), але також може використовуватися і більша або менша кількість фільтрів. Фіг. 4А демонструє дві метрики, які можуть використовуватися для вибору фільтра відповідно до методик цього розкриття. Дві метрики можуть, наприклад, визначати кількість властивостей даних пікселя, що стосуються активності, не зв'язаної з напрямком (наприклад, значення Лапласіана із зміненою сумою), і напрямку, активності, зв'язаної з напрямком, і виявлення країв, метрики напрямку і метрики краю, метрики горизонтальної активності і метрики вертикальної активності або двох інших подібних метрик. У деяких випадках три або більше метрик могло б використовуватися, в цьому випадку концептуальна схема з фіг. 4А включала б в себе третє вимірювання для діапазонів відповідності метрик фільтрам з набору фільтрів.

У прикладі з фіг. 4А перша метрика (Метрика 1) має чотири діапазони (Діапазони 1-1, 1-2, 1-3 та 1-4) і друга метрика (Метрика 2) також має чотири діапазони (Діапазони 2-1, 2-2, 2-3 та 2-4). Отже, приклад з фіг. 4А має шістнадцять комбінацій діапазонів для Метрики 1 та Метрики 2. Як можна побачити з фіг. 4А, однак кожна комбінація не обов'язково зв'язана з унікальним фільтром. Комбінація Діапазону 1-1 та Діапазону 2-1, як і комбінації 1-1 та 2-2, і 1-1 і 2-3, наприклад, всі приведені у відповідність до фільтра 1, в прикладі з фіг. 4А. Фільтр 4, навпаки, приведений у відповідність до тільки однієї комбінації (1-1 та 2-4). Хоча діапазони з фіг. 4А показані як приблизно рівні, розміри діапазонів можуть варіюватися. Наприклад, в деяких варіантах здійснення Діапазон 1-1 може охоплювати більший діапазон значень, ніж Діапазон 1-2. Додатково, хоча фіг. 4А демонструє Метрику 1 та Метрику 2 як такі, що мають однакову кількість діапазонів, кількість діапазонів для першої метрики і кількість діапазонів для другої метрики не обов'язково повинні бути рівними. Якщо, наприклад, Метрика 1 - метрика дисперсії, а Метрика 2 - метрика напрямку, Метрика 1 могла б використовувати вісім діапазонів, тоді як Метрика 2 використовує три діапазони.

У деяких прикладах діапазони Метрики 1 та Метрики 2 можуть представляти безперервний спектр значень. Наприклад, якщо Метрика 1 є значенням Лапласіана із зміненою сумою, Діапазон 1-2 може відповідати більшій активності, ніж Діапазон 1-1, але меншій активності, ніж Діапазон 1-3, і Діапазон 1-4 може відповідати більшій активності, ніж Діапазон 1-3. У межах діапазону кількість активності, визначена для конкретного пікселя або групи пікселів, може подібним чином збільшуватися вздовж осі Метрики 1. В інших прикладах діапазони Метрики 1 та Метрики 2 можуть не представляти дійсних діапазонів, але замість цього можуть представляти дискретні визначення. Наприклад, якщо Метрика 2 - метрика напрямку, Діапазон 1-1 може відповідати визначенню відсутності напрямку, Діапазон 2-2 може відповідати визначенню горизонтального напрямку, Діапазон 2-3 може відповідати визначенню вертикального напрямку, а Діапазон 2-4 може відповідати визначенню діагонального напрямку. Як буде описано більш детально нижче, відсутність напрямку, горизонтальний напрямок, вертикальний напрямок і діагональний напрямок можуть бути дискретними визначеннями, і таким чином, діапазони для Метрики 2 могли б не представляти безперервний спектр значень таким саме чином, як роблять діапазони Метрики 1.

Фіг. 4В являє собою концептуальну схему, що ілюструє діапазони значень для метрики активності та метрики напрямку. У прикладі Фіг. 4В метрика напрямку включає в себе три дискретних визначення (Відсутність Напрямку, Горизонтальний, Вертикальний). Методики для визначення відсутності напрямку, горизонтального і вертикального, як і методики визначення активності, більш детально будуть пояснені нижче. Конкретний приклад з фіг. 4В демонструє шість фільтрів (тобто Фільтр 1, Фільтр 2 ... Фільтр 6), але більша або менша кількість фільтрів також може використовуватися. Як можна побачити з фіг. 4В, дві метрики (активності і

напрямку) створюють 15 комбінацій, ідентифікованих як комбінації 421-435. У деяких випадках, однак, додаткові комбінації, не показані детально на фіг. 4В, також можуть використовуватися. Наприклад, комбінація, що відповідає відсутності активності, може бути 16-ю комбінацією, яка також має відповідний фільтр.

Модуль 349 фільтра може зберігати відповідність фільтрів комбінаціям діапазонів двох або більше метрик, як то зразкова відповідність з фіг. 4А та фіг. 4В, і використовувати побудову відповідностей, щоб визначити, який фільтр з набору фільтрів застосовувати до конкретного пікселя або групи пікселів в СУ. Відповідність фільтрів комбінаціям діапазонів двох або більше метрик може, наприклад, визначатися модулем 349 фільтра як частина процесу вибору фільтра, описаного вище. Незалежно від того, як визначається відповідність, модуль 349 фільтра може генерувати інформацію, що дозволяє декодеру реконструювати відповідність. Ця інформація може бути включена в кодований бітовий потік, щоб сигналізувати відповідність комбінацій діапазонів фільтрам. Сигналізована відповідність комбінацій діапазонів фільтра може приводити у відповідність комбінації діапазонів до ідентифікаційного ID фільтра. Дійсні коефіцієнти для конкретного фільтра можуть сигналізуватися окремо.

Щоб генерувати таку інформацію, модуль 349 фільтра спочатку визначає порядок передачі для комбінацій. Порядок передачі загалом стосується порядку, в якому фільтри будуть сигналізовані для комбінацій діапазонів. Використовуючи фіг. 4А як приклад, модуль 349 фільтра може використовувати порядок передачі зліва-направо, зверху-вниз, де фільтр для комбінації 401 сигналізується першим, фільтр для комбінації 402 сигналізується другим, і комбінації, що залишилися, сигналізуються в порядку 403=>404=>405=>406=>407=>408=>409=>410=>411=>412=>413=>414=>415=>416. Модуль 349 фільтра може також використовувати порядок передачі зверху-вниз, зигзагоподібно, де фільтри для комбінацій сигналізуються в порядку 401=>402=>403=>404=>408=>407=>406=>405=>409=>410=>411=>412=>416=>415=>414=>413. Модуль 349 фільтра може також використовувати порядок передачі зверху-вниз, зліва-направо, де фільтри для комбінацій сигналізуються в порядку 401=>405=>409=>413=>402=>406=>410=>414=>403=>407=>411=>415=>404=>408=>412=>416. Модуль 349 фільтра може також використовувати порядок передачі зліва-направо, зигзагоподібно, де фільтри для комбінацій сигналізуються в порядку 401=>405=>409=>413=>414=>410=>406=>402=>403=>407=>411=>415=>416=>412=>408=>404. З посиланням на Фіг. 4В модуль 349 фільтра може використовувати порядок передачі зліва-направо, знизу-вгору так, що порядок сигналізації являє собою 421=>422=>423=>424=>425=>426=>427=>428=>429=>430=>431=>432=>433=>434=>435. Як можна представити, це лише деякі з множини порядків передачі, які є можливими.

Відповідно до методики цього розкриття модуль 349 фільтра може використовувати серію кодових слів, щоб сигналізувати відповідність декодеру. Наприклад, модуль 349 фільтра може генерувати перше кодове слово для індикації того, якщо поточна декодована комбінація відповідає тому самому фільтру, що і найостанніша декодована комбінація, яка спільно використовує той самий діапазон для першої метрики. Якщо поточна декодована комбінація відповідає тому самому фільтру, що і найостанніша декодована комбінація, яка спільно використовує той самий діапазон для другої метрики, то модуль 349 фільтра може генерувати друге кодове слово замість першого кодового слова. Якщо поточна декодована комбінація не відповідає тому самому фільтру, що і будь-яка з цих найостанніших декодованих комбінацій, то модуль 349 фільтра може генерувати третє кодове слово замість першого кодового слова і другого кодового слова, яке вказує фільтр, що відповідає поточній декодованій комбінації. Перше кодове слово і друге кодове слово з даного прикладу можуть бути відносно короткими в порівнянні з третім кодовим словом. Наприклад, перше кодове слово і друге кодове слово могли б являти собою два біти кожне (наприклад, 00 та 01 відповідно), тоді як третє кодове слово - це більше бітів (перший біт 1, плюс додаткові біти). У цьому конкретному контексті поточна декодована комбінація або попередня декодована комбінація стосується частини процесів кодування і декодування, де побудова відповідності фільтрів комбінаціям діапазонів сигналізується кодером або реконструюється декодером, і не обов'язкова до передачі або декодування самої комбінації.

Приклади методик, описані вище, тепер будуть представлені з посиланням на фіг. 4А і порядок передачі зверху-вниз, зліва-направо. Якщо, наприклад, комбінація 407 - це комбінація, декодована в поточний момент, то комбінація 406 - це найостанніша декодована комбінація, яка спільно використовує той самий діапазон для Метрики 1, і комбінація 403 - це найостанніша декодована комбінація, яка спільно використовує той самий діапазон для Метрики 2. Якщо комбінація 407 відповідає тому самому фільтру (Фільтр 7 на фіг. 4А), що і найостанніша декодована комбінація, яка спільно використовує той самий діапазон для другої метрики (тобто

Діапазон 2-3 для Метрики 2), модуль 349 фільтра може передавати друге кодове слово (наприклад, 01) для індикації того, що поточна декодована комбінація (комбінація 407) відповідає тому самому фільтру, що і найостанніша декодована комбінація, яка спільно використовує той самий діапазон для другої метрики (комбінація 403).

5 Якщо, наприклад, комбінація 410 є поточною декодованою комбінацією, то комбінація 409 - це найостанніша декодована комбінація, яка спільно використовує той самий діапазон для Метрики 1, і комбінація 406 - це найостанніша декодована комбінація, яка спільно використовує той самий діапазон для Метрики 2. Якщо комбінація 410 відповідає тому самому фільтру (Фільтр 2 на фіг. 4А), що і найостанніша декодована комбінація, яка спільно використовує той  
10 самий діапазон для першої метрики (тобто Діапазон 1-2 для Метрики 1), модуль 349 фільтра може передавати перше кодове слово (наприклад, 00) для індикації того, що поточна декодована комбінація (комбінація 410) відповідає тому самому фільтру (Фільтр 2), що і найостанніша декодована комбінація, яка використовує той самий діапазон для першої метрики (комбінація 409).

15 Якщо, наприклад, комбінація 411 є поточною декодованою комбінацією, то комбінація 410 - це найостанніша декодована комбінація, яка спільно використовує той самий діапазон для Метрики 1, і комбінація 407 - це найостанніша декодована комбінація, яка спільно використовує той самий діапазон для Метрики 2. Якщо комбінація 411 не відповідає тому самому фільтру, що і будь-яка з комбінації 410 і комбінації 407, модуль 349 фільтра може передавати третє кодове  
20 слово (наприклад, 1 + додаткові біти) для індикації того, що поточна декодована комбінація (комбінація 411) відповідає іншому фільтру (Фільтр 3), ніж обидві - і найостанніша декодована комбінація, яка спільно використовує той самий діапазон для Метрики 1, і найостанніша декодована комбінація, яка спільно використовує той самий діапазон для Метрики 2.

Для тих поточних комбінацій, де комбінація, яка спільно використовує той самий діапазон  
25 для Метрики 1, або комбінація, яка спільно використовує той самий діапазон для Метрики 2, ще не були декодовані, ті опції можуть або розглядатися як недоступні або можуть бути замінені іншою комбінацією. Якщо, наприклад, комбінація 409 - це поточна комбінація, що підлягає декодуванню, то комбінація 405 - це найостанніша декодована комбінація, яка спільно використовує той самий діапазон для Метрики 2, але ніяка комбінація, яка спільно використовує  
30 діапазон для Метрики 1, ще не була декодована. У таких випадках найостанніша декодована комбінація, яка спільно використовує той самий діапазон для Метрики 1, може вважатися не відповідною тому самому фільтру, що і поточна комбінація, що підлягає декодуванню. Таким чином, в цьому випадку перше кодове слово не буде використовуватися для комбінації 409. Як альтернатива, комбінація, яка спільно використовує діапазон для Метрики 1, може бути  
35 замінена іншою комбінацією, такою як найостанніша декодована комбінація або інша раніше декодована комбінація. У такому випадку найостанніша декодована комбінація до комбінації 409 могла б бути комбінацією 408. Таким чином, якщо комбінація 408 відповідає тому самому фільтру, що і комбінація 409, модуль 349 фільтра може генерувати перше кодове слово. Аналогічні методики можуть використовуватися для тих комбінацій, де попередня комбінація,  
40 що спільно використовує загальний діапазон для Метрики 1, ще не була декодована.

Для першої комбінації в порядку передачі (наприклад, комбінації 401 в прикладі з фіг. 4А), де ні комбінація, яка спільно використовує той самий діапазон для Метрики 1, ні комбінація, яка спільно використовує той самий діапазон для Метрики 2, не були декодовані, модуль 349  
45 фільтра може генерувати кодове слово, яке вказує фільтр, який відповідає першій комбінації. Фільтр може бути, наприклад, сигналізований з використанням третього кодового слова або може бути сигналізований з використанням іншої методики, у цьому випадку методики, описані в цьому розкритті, могли б початися з другої комбінації в порядку передачі або більш пізньої комбінації.

Відповідно до іншої методики цього розкриття, модуль 349 фільтра може використовувати  
50 серію кодових слів, щоб сигналізувати відповідність декодеру. У деяких варіантах здійснення модуль 349 фільтра може генерувати перше кодове слово для індикації того, якщо поточна декодована комбінація відповідає тому самому фільтру, що і найостанніша декодована комбінація, яка спільно використовує той самий діапазон для першої метрики. Якщо поточна декодована комбінація не відповідає тому самому фільтру, що найостанніша декодована  
55 комбінація, яка спільно використовує той самий діапазон для першої метрики, то модуль 349 фільтра може генерувати друге кодове слово, замість першого кодового слова, яке вказує фільтр, який відповідає поточній декодованій комбінації. Наприклад, перше кодове слово може являти собою один біт (наприклад, 0), тоді як друге кодове слово - це більше бітів (наприклад, перший біт 1, плюс додаткові біти). На відміну від попередньої методики, де коротке кодове  
60 слово могло б генеруватися, якщо поточна комбінація відповідає тому самому фільтру, що і

раніше декодована комбінація, яка спільно використовує той самий діапазон для будь-якої з Метрики 1 або Метрики 2, ця методика включає в себе тільки генерування короткого кодового слова, якщо поточна комбінація відповідає тому самому фільтру, що і раніше декодована комбінація, яка спільно використовує той самий діапазон для Метрики 1. Таким чином, навіть

якщо поточна комбінація відповідає тому самому фільтру, що і раніше декодована комбінація, яка спільно використовує той самий діапазон для Метрики 2, модуль 349 фільтра все ще генерує друге кодове слово (наприклад, 1 + додаткові біти). Хоча це розкриття використовує Метрику 1 з метою пояснення, ті самі методики також можуть застосовуватися з використанням тільки Метрики 2.

Відповідно ще до однієї методики цього розкриття модуль 349 фільтра може використовувати різні серії кодових слів, щоб сигналізувати відповідність до декодера. Наприклад, модуль 349 фільтра може генерувати перше кодове слово для індикації того, якщо поточна декодована комбінація відповідає тому самому фільтру, що і найостанніша декодована комбінація, незалежно від того, який діапазон, якщо він був, був для поточної комбінації загальним з раніше декодованою комбінацією. Якщо поточна декодована комбінація не відповідає тому самому фільтру, що і найостанніша декодована комбінація, то модуль 349 фільтра може генерувати друге кодове слово, що ідентифікує фільтр, який відповідає поточній комбінації. У цьому конкретному варіанті здійснення перше слово може бути відносно коротким в порівнянні з другим кодовим словом. Наприклад, перше кодове слово може являти собою один біт (наприклад, 0), тоді як друге кодове слово - це більше бітів (наприклад, перший біт 1, плюс додаткові біти).

Знову варто зазначити, що при використанні прикладу з фіг. 4A і порядку передачі зверху-вниз, зліва-направо комбінація 401 була б найостаннішою декодованою комбінацією, якщо комбінація 402 декодується в поточний момент, комбінація 402 була б найостаннішою декодованою комбінацією, якщо комбінація 403 - поточна комбінація, і так далі. 404 була б найостаннішою декодованою комбінацією, якщо комбінація 405 - це комбінація, яка декодується в поточний момент. Так, модуль 349 фільтра може генерувати перше кодове слово, якщо комбінація 402 відповідає тому самому фільтру, що і комбінація 401, якщо комбінація 403 відповідає тому самому фільтру, що і комбінація 402 тощо. В іншому випадку модуль 349 фільтра може генерувати друге кодове слово, що ідентифікує фільтр, який відповідає поточній комбінації.

Відповідно ще до однієї методики цього розкриття модуль 349 фільтра може використовувати два кодових слова, щоб сигналізувати відповідність фільтрів комбінаціям. Перше кодове слово, таке як "0", може використовуватися для сигналізування того, що поточна комбінація використовує той самий фільтр, що і попередня комбінація. Друге кодове слово, таке як "1", може використовуватися для сигналізування того, що поточна комбінація має інший фільтр, ніж попередня комбінація. Друге кодове слово, однак, не має потреби ідентифікувати новий фільтр. Замість цього новий фільтр може бути визначений на основі порядку передачі для класів і порядку, при якому передаються коефіцієнти фільтра. При використанні порядку передачі зліва-направо, знизу-вгору, описаного вище для Фіг. 4B як приклад, кодові слова могли б передаватися відповідно: 421 (0)=>422 (0)=>423 (1)=>424 (0)=>425 (0)=>426 (0)=>427 (0)=>428 (1)=>429 (0)=>430 (0)=>431 (0)=>432 (1)=>433 (0)=>434 (0)=>435 (0), з числом в круглих дужках, що представляє кодове слово для цієї комбінації. У цьому прикладі комбінації 421-422 були б приведені у відповідність до першого фільтра, комбінації 423-427 - другого фільтра, комбінації 428-431 - третього фільтра, а комбінації 432-435 - четвертого фільтра. Коефіцієнти для першого фільтра, другого фільтра, третього фільтра і четвертого фільтра можуть відповідати порядку, при якому набори коефіцієнтів фільтра сигналізуються, де сигналізований перший набір коефіцієнтів фільтра відповідає першому фільтру, сигналізований другий набір коефіцієнтів фільтра відповідає другому фільтру тощо. Визначення порядку для передачі наборів коефіцієнтів фільтра розглядається нижче більш детально.

Різні методики, описані в цьому розкритті для сигналізації відповідності фільтрів комбінаціям діапазонів, не є взаємовиключними альтернативами, а навпаки, можуть використовуватися спільно один з одним. Наприклад, в деяких варіантах здійснення визначені комбінації могли б сигналізуватися з використанням першої методики, тоді як інші комбінації сигналізуються з використанням другої методики. Як один приклад, де одна з комбінацій, яка спільно використовує той самий діапазон для Метрики 1, або комбінації, яка спільно використовує той самий діапазон для Метрики 2, ще не була декодована (наприклад, комбінації 402, 403, 404, 405, 409 та 413), модуль 349 фільтра може використовувати першу методику. Там, де і комбінація, яка спільно використовує той самий діапазон для Метрики 1, і комбінація, яка спільно використовує той самий діапазон для Метрики 2, була декодована (наприклад,

комбінації 406, 407, 408, 410, 411, 412, 414, 415 та 416), могла б використовуватися друга методика. Додатково кодові слова, використані для будь-якого з першого, другого і третього кодових слів, описаних вище, можуть бути будь-якими з кодових слів фіксованої довжини, змінних кодових слів або контекстно-адаптивних змінних кодових слів.

Додатково до генерування інформації, що дозволяє декодеру реконструювати відповідність фільтрів комбінаціям діапазонів, модуль 349 фільтра також генерує інформацію, що дозволяє декодеру реконструювати самі фільтри. Реконструкція фільтрів включає в себе реконструкцію коефіцієнтів фільтра цих фільтрів. Як буде описано більш детально нижче, модуль 349 фільтра може використовувати методики різницевого кодування, щоб сигналізувати коефіцієнти фільтра. Щоб використовувати методики різницевого кодування, модуль 349 фільтра визначає порядок, в якому сигналізувати набори коефіцієнтів фільтра.

Як частина визначення порядку модуль 349 фільтра визначає ідентифікацію комбінації (ID), яка представляє порядкове значення для кожної комбінації діапазонів. При використанні фіг. 4А як прикладу комбінаціям можна було б призначити ID комбінації, які представляють порядкові значення в порядку зліва-направо, зверху-вниз, в цьому випадку комбінація 401 набула б першого порядкового значення, комбінація 402 набула б другого порядкового значення, а комбінації, що залишилися, набули б порядкових значень в порядку 403=>404=>405=>406=>407=>408=>409 =>410=>411=>412=>413=>414=>415=>416. Модуль 349 фільтра також може призначити ID комбінації, використовуючи порядок зверху-вниз, зигзагоподібно, де комбінації одержали б ID комбінації з порядковими значеннями в порядку 401=>402=>403=>404=>408=>407=> 406=>405=>409=>410=>411=>412=>416=>415=>414=>413. Модуль 349 фільтра також може призначити ID комбінації, використовуючи порядок зверху-вниз, зліва-направо, де комбінації одержали б ID комбінації з порядковими значеннями в порядку 401=>405=>409=>413=>414=>410=>406=>402=>403=>407=>411=>415=>416=>412=>408=>404. Як можна представити, це тільки деякі з множини порядків, які могли б використовуватися. Крім того, будь-який з описаних порядків, може бути і від найнижчого до найвищого, і від найвищого до найнижчого.

Після того як модуль 349 фільтра визначив відповідність фільтрів комбінаціям діапазонів, модуль 349 фільтра може ідентифікувати групування комбінацій діапазонів, які приведені у відповідність до одного і того самого фільтра. З використанням фіг. 4А як прикладу групування були б наступними.

Група Фільтра 1: комбінації 413, 414 та 415

Група Фільтра 2: комбінації 409, 410

Група Фільтра 3: комбінації 411 та 412

Група Фільтра 4: комбінація 416

Група Фільтра 5: комбінації 401 та 405

Група Фільтра 6: комбінації 402 та 406

Група Фільтра 7: комбінації 403 та 407

Група Фільтра 8: комбінації 404 та 408.

Модуль 349 фільтра потім може призначити кожній групі ID групи, і ID групи може представляти порядкове значення. ID групи може призначатися групам на основі порядкових значень, зв'язаних з комбінаціями, які містять групу. Наприклад, група, яка має комбінацію з найнижчим відповідним порядковим значенням, на основі ID комбінації могла б одержати ID групи з найнижчим порядковим значенням. З груп, що залишилися, група, що залишилася, група, яка має комбінацію з найнижчим відповідним порядковим значенням, могла б одержати ID групи з наступним найнижчим порядковим значенням. Цей процес може повторюватися доти, доки всім групам не буду призначений ID групи. У деяких варіантах здійснення ID групи може призначатися на основі комбінацій з найвищими відповідними порядковими значеннями, ніж найнижчими. У деяких варіантах здійснення група, яка має комбінацію з найнижчим відповідним порядковим значенням, на основі ID комбінації могла б одержати ID групи з найвищим порядковим значенням або навпаки.

Знову варто згадати, що використовуючи фіг. 4А як приклад і допускаючи, що комбінаціям 401-416 привласнені ID комбінації з порядковими значеннями в порядку зліва-направо, зверху-вниз, модуль 349 фільтра може призначати ID групи групам фільтрів, як показано нижче в Таблиці 1.

Таблиця 1

Ім'я групи	Комбінації в групі	Комбінація з найнижчим порядковим значенням	ID групи
Група Фільтра 1	413, 414, 415	413	7
Група Фільтра 2	409, 410	409	5
Група Фільтра 3	411, 412	411	6
Група Фільтра 4	416	416	8
Група Фільтра 5	401, 405	401	1
Група Фільтра 6	402, 406	402	2
Група Фільтра 7	403, 407	403	3
Група Фільтра 8	404, 408	404	4

У прикладі з фіг. 4А, показаному в Таблиці 1, модуль 349 фільтра призначає Групі Фільтра 5 ID групи з найнижчим порядковим значенням, оскільки Група Фільтра 5 включає в себе комбінацію діапазонів з найнижчим порядковим значенням (тобто комбінацію 401). Модуль 349 фільтра призначає Групі Фільтра 6 ID групи з другим найнижчим порядковим значенням, оскільки з груп фільтрів, що залишилися, (тобто усіх груп за винятком Групи Фільтра 5) Група Фільтра 6 включає в себе комбінацію діапазонів з другим найнижчим порядковим значенням (тобто комбінацію 402). Модуль 349 фільтра призначає Групі Фільтра 7 ID групи з третім найнижчим порядковим значенням, оскільки з груп фільтрів, що залишилися, (тобто усіх груп за винятком Групи Фільтра 5 та Групи Фільтра 6) Група Фільтра 7 включає в себе комбінацію діапазонів з найнижчим порядковим значенням (тобто комбінацію 403). Модуль 349 фільтра призначає Групі Фільтра 8 ID групи з четвертим найнижчим порядковим значенням, оскільки з груп фільтрів, що залишилися, (тобто усіх груп за винятком Групи Фільтра 5, Групи Фільтра 6 та Групи Фільтра 7) Група Фільтра 8 включає в себе комбінацію діапазонів з четвертим найнижчим порядковим значенням (тобто комбінацію 404). Модуль 349 фільтра призначає Групі Фільтра 2 ID групи з п'ятим найнижчим порядковим значенням, оскільки з груп фільтрів, що залишилися, (тобто усіх груп за винятком Групи Фільтра 5, Групи Фільтра 6, Групи Фільтра 7 та Групи Фільтра 8) Група Фільтра 2 включає в себе комбінацію діапазонів з найнижчим порядковим значенням (тобто комбінацію 409). Модуль 349 фільтра призначає Групі Фільтра 3 ID групи з шостим найнижчим порядковим значенням, оскільки з груп фільтрів, що залишилися, (тобто усіх груп за винятком Групи Фільтра 5, Групи Фільтра 6, Групи Фільтра 7, Групи Фільтра 8 та Групи Фільтра 2) Група Фільтра 3 включає в себе комбінацію діапазонів з найнижчим порядковим значенням (тобто комбінацію 411). Модуль 349 фільтра призначає Групі Фільтра 1 ID групи з сьомим найнижчим порядковим значенням, оскільки з груп фільтрів, що залишилися, (тобто усіх груп за винятком Групи Фільтра 5, Групи Фільтра 6, Групи Фільтра 7, Групи Фільтра 8, Групи Фільтра 2 та Групи Фільтра 3) Група Фільтра 1 включає в себе комбінацію діапазонів з найнижчим порядковим значенням (тобто комбінацію 413). Нарешті, модуль 349 фільтра призначає Групі Фільтра 4, останній групі фільтрів, що залишилися, ID групи з найвищим порядковим значенням (8 в конкретному прикладі).

На основі ID групи фільтрів модуль 349 фільтра визначає порядок, в якому сигналізувати коефіцієнти фільтра вибраного фільтра. Знову, при використанні прикладу з фіг. 4А і Таблиці 1 модуль 349 фільтра спочатку сигналізує коефіцієнт для Фільтра 5, потім коефіцієнт для Фільтра 6, після цього коефіцієнт для Фільтра 7, потім коефіцієнт для Фільтра 8, після цього коефіцієнт для Фільтра 2, потім коефіцієнт для Фільтра 3, після цього коефіцієнт для Фільтра 1 і, нарешті, коефіцієнт для Фільтра 4. Використовуючи методики різницевого кодування, як описується в даному розкритті, модуль 349 фільтра може кодувати коефіцієнти для Фільтра 6 як інформацію різниці відносно коефіцієнтів фільтра від Фільтра 5, кодувати коефіцієнти для Фільтра 7 як інформацію різниці відносно коефіцієнтів фільтра від Фільтра 6 і так далі, на основі впорядкування ID групи.

Приведення у відповідність двох або більше метрик для введення до фільтрів може реалізовуватися декількома способами. Наприклад, в деяких варіантах здійснення кожне введення може мати унікальний набір фільтрів, тоді як в деяких варіантах здійснення введення

спільно використовує загальний набір фільтрів. Додатково в деяких варіантах здійснення дві або більше метрик для кожного введення могли б використовуватися, щоб ідентифікувати конкретний фільтр для кожного введення. В інших варіантах здійснення, однак, дві або більше метрик для єдиного введення могли б використовуватися, щоб ідентифікувати фільтри для всього введення. У ще інших варіантах здійснення дві або більше метрик для першого введення могли б використовуватися, щоб ідентифікувати фільтр для другого, іншого введення.

Відповідно до цього варіанта здійснення модуль 349 фільтра може реалізовувати методики кодування відносно інформації фільтра, що може скоротити кількість даних, необхідних, щоб кодувати і переносити інформацію фільтра від кодера 350 до іншого пристрою. Знову зазначимо, що для кожного кадру або вирізки модуль 349 фільтра може визначати або вибирати один або більше наборів коефіцієнтів фільтра, що підлягають застосуванню до пікселів різних CU для цього кадру або вирізки. Модуль 349 фільтра застосовує коефіцієнти фільтра з метою відфільтрувати відеоблоки реконструйованих відеокадрів, що зберігаються в пам'яті 334, які можуть використовуватися для кодування з прогнозуванням, сумісного з внутрішньоконтурною фільтрацією. Модуль 349 фільтра може кодувати коефіцієнти фільтра як інформацію фільтра, яка пересилається до елемента 346 ентропійного кодування для включення в кодований бітовий потік.

Додатково методики цього розкриття можуть використовувати той факт, що деякі з коефіцієнтів фільтра, визначених або вибраних FSU 353, можуть бути дуже схожими на інші коефіцієнти фільтра, що застосовуються відносно пікселів CU іншого кадру або вирізки. Один і той самий тип фільтра може застосовуватися для різних кадрів і вирізок (наприклад, однакова підтримка фільтра), але фільтри можуть бути різними з точки зору значень коефіцієнтів фільтра, зв'язаних з різними індексами підтримки фільтра. Відповідно, з метою зменшити кількість даних, необхідних для перенесення таких коефіцієнтів фільтра, модуль 349 фільтра може піддавати кодуванню з прогнозуванням один або більше коефіцієнтів фільтра, що підлягають використанню для фільтрації на основі коефіцієнтів фільтра іншого CU, потенційно використовуючи схожість між коефіцієнтами фільтра. У деяких випадках, однак, може бути більш бажаним варіантом кодувати коефіцієнти фільтра напряму, наприклад, без використання якого-небудь прогнозування. Різні методики, такі як методики, які введуть в дію використання метрики активності, щоб визначити, коли кодувати коефіцієнти фільтра, використовуючи методики кодування з прогнозуванням, або коли кодувати коефіцієнти фільтра напряму, без якого-небудь кодування з прогнозуванням, можуть використовуватися для ефективного повідомлення коефіцієнтів фільтра декодеру. Додатково симетрія також може накладатися так, що піднабір коефіцієнтів (наприклад, 5, -2, 10), відомий декодеру, може використовуватися, щоб визначити повний набір коефіцієнтів (наприклад, 5, -2, 10, 10, -2, 5). Симетрія може накладатися як для сценарію прямого кодування, так і для сценарію кодування з прогнозуванням.

Як було описано вище, відеокодер 350 являє собою приклад відеокодера, виконаного з можливістю визначати першу метрику для групи пікселів у межах блока пікселів, визначати другу метрику для групи пікселів, визначати фільтр на основі першої метрики і другої метрики та генерувати відфільтроване зображення шляхом застосування фільтра до групи пікселів. Відеокодер 350 також являє собою приклад відеокодера, виконаного з можливістю визначати першу метрику для блока пікселів, при цьому перша метрика визначається на основі порівняння піднабору пікселів у блоці з іншими пікселями в блоці; визначати другу метрику для блока пікселів; визначати фільтр на основі першої метрики і другої метрики; і генерувати відфільтроване зображення шляхом застосування фільтра до блока пікселів.

Як було описано вище, відеокодер 350 також являє собою приклад відеокодера, виконаного з можливістю визначати відповідність комбінацій діапазонів фільтрам, при цьому комбінація діапазонів містить діапазон для першої метрики і діапазон для другої метрики, при цьому кожна комбінація діапазонів має унікальну ідентифікацію комбінації діапазонів (ID), при цьому кожна унікальна комбінація діапазонів ID відповідає порядковому значенню для комбінації діапазонів; призначати унікальний ID групи групам комбінацій діапазонів на основі порядкових значень для комбінацій діапазонів, при цьому кожний унікальний ID групи відповідає порядковому значенню для групи; і кодувати групи коефіцієнтів фільтра, що відповідають фільтрам, на основі унікальних ID групи. Відеокодер 350 може кодувати набори коефіцієнтів фільтра шляхом сигналізації наборів коефіцієнтів фільтра в кодованому бітовому потоці в порядку, який вибирається на основі порядкових значень унікальних ID групи. Відеокодер 350 може сигналізувати набори коефіцієнтів фільтра з використанням методик різницевого кодування.

Як було описано вище, відеокодер 350 також являє собою приклад відеокодера, виконаного з можливістю визначати відповідність комбінацій діапазонів фільтрам, при цьому комбінація діапазонів містить діапазон значень для першої метрики і діапазон значень для другої метрики;



генерувати перше кодове слово, якщо поточна комбінація діапазонів приведена у відповідність до того самого фільтра, що і попередня комбінація діапазонів, яка містить той самий діапазон значень для першої метрики; генерувати друге кодове слово, якщо поточна комбінація діапазонів приведена у відповідність до того самого фільтра, що і попередня комбінація діапазонів, яка містить той самий діапазон значень для другої метрики; і генерувати третє кодове слово, якщо поточна комбінація діапазонів приведена у відповідність до іншого фільтра, ніж попередня комбінація діапазонів, яка містить той самий діапазон значень для першої метрики і попередню комбінацію діапазонів, яка містить той самий діапазон значень для другої метрики. Відеокoder 350 також являє собою приклад відеокodера, виконаного з можливістю визначати відповідність комбінацій діапазонів фільтрам, при цьому комбінація діапазонів містить діапазон для першої метрики і діапазон для другої метрики; генерувати перше кодове слово, якщо поточна комбінація діапазонів приведена у відповідність до того самого фільтра, що і попередня комбінація діапазонів; і генерувати друге кодове слово, якщо поточна комбінація діапазонів приведена у відповідність до іншого фільтра, ніж попередня комбінація діапазонів, при цьому друге кодове слово ідентифікує фільтр, приведений у відповідність до поточної комбінації діапазонів.

Фіг. 5 являє собою блок-схему, яка ілюструє приклад відеокodера 560, який декодує відеопослідовність, яка кодована способом, описаним тут. Прийнята відеопослідовність може містити кодований набір кадрів зображення, набір вирізок кадру, звичайно кодовану групу зображень (GOP) або широке різноманіття типів серій відеоблоків, які включають в себе кодовані відеоблоки і синтаксис, щоб визначити, як декодувати такі відеоблоки.

Відеокoder 560 включає в себе модуль 552 ентропійного декодування, який здійснює функцію оберненого декодування відносно кодування, виконаного елементом 346 ентропійного кодування з фіг. 3. Зокрема, модуль 552 ентропійного декодування може здійснювати декодування CAVLC та CABAC або будь-який інший тип ентропійного декодування, що використовується відеокoderом 350. Ентропійно декодовані відеоблоки в одновимірному перетвореному в послідовну форму форматі можуть бути піддані оберненому скануванню, щоб перетворити один або більше одновимірних векторів коефіцієнтів зворотно в формат двовимірного блока. Кількість і розмір векторів, як і порядок сканування, визначений для відеоблоків, можуть визначати, як реконструюється двовимірний блок. Ентропійно декодований синтаксис прогнозування може бути відправлений від модуля 552 ентропійного декодування до модуля 554 прогнозування, і інформація ентропійно декодованого фільтра може бути відправлена від модуля 552 ентропійного декодування до модуля 559 фільтра.

Відеокoder 560 також включає в себе модуль 554 прогнозування, модуль 556 оберненого квантування, модуль 558 оберненого перетворення, запам'ятовуючий пристрій і суматор 564. Додатково до цього відеокoder 560 також включає в себе фільтр 557 видалення блочності, який фільтрує виведення суматора 564. Сумісний з цим розкриттям, модуль 559 фільтра може приймати ентропійно декодовану інформацію фільтра, яка включає в себе один або більше фільтрів, що підлягають застосуванню до одного або більше введень. Хоча це і не показано на фіг. 5, фільтр 557 видалення блочності також може приймати ентропійно декодовану інформацію фільтра, яка включає в себе один або більше фільтрів, що підлягають застосуванню.

Фільтри, що застосовуються модулем 559 фільтра, можуть визначатися наборами коефіцієнтів фільтра. Модуль 559 фільтра може бути виконаний з можливістю генерувати набори коефіцієнтів фільтра на основі інформації фільтра, прийнятої від модуля 552 ентропійного декодування. Інформація фільтра може включати в себе синтаксис опису фільтра, який ідентифікує максимальну кількість фільтрів у наборі фільтрів і/або форму фільтрів у наборі фільтрів, наприклад. Синтаксис опису фільтра може бути увімкнений в заголовок серії відеоблоків, наприклад, заголовок LCU, заголовок кадру, заголовок вирізки, заголовок GOP, заголовок послідовності тощо. В інших прикладах синтаксис опису фільтра може бути увімкнений в нижній колонтитул або іншу структуру даних. На основі синтаксису опису фільтра модуль 559 фільтра може реконструювати набір фільтрів, використаний в кодєрі.

Інформація фільтра також може включати в себе додатковий синтаксис сигналізування, який сигналізує декодеру спосіб кодування, використаний для будь-якого заданого набору коефіцієнтів. У деяких варіантах здійснення інформація фільтра може, наприклад, також включати в себе діапазони для двох або більше метрик, для яких повинен використовуватися будь-який заданий набір коефіцієнтів. Після декодування фільтрів модуль 559 фільтра може фільтрувати значення пікселів декодованих відеоблоків на основі одного або більше наборів коефіцієнтів фільтра і синтаксис сигналізування, який включає в себе діапазони, для яких повинні використовуватися різні набори коефіцієнтів фільтра.

Модуль 559 фільтра може приймати в бітовому потоці один або більше синтаксичних компонентів, які вказують набір фільтрів для кожного кадру або вирізки, як і відповідність фільтрів двом або більше метрикам. Наприклад, якщо кодер використовує відповідність діапазонів для метрик до фільтрів, що показано на фіг. 4А, кодер буде або сигналізувати цю відповідність, або передавати дані, щоб дозволити модулю 559 фільтра реконструювати цю відповідність. Незалежно від того, чи чітко повідомляється ця відповідність, модуль 559 фільтра може підтримувати ту саму відповідність фільтрів комбінаціям діапазонів, що і використовувалося кодером.

Як було згадано вище, модуль 559 фільтра генерує відповідність на основі інформації фільтра, сигналізованої в бітовому потоці. На основі цієї відповідності модуль 559 фільтра може визначати групи і призначати ID групи таким саме чином, як описувалося вище відносно модуля 349 фільтра. Використовуючи ці ID групи, модуль 559 фільтра може зв'язувати прийняті коефіцієнти фільтра з...

Для кожного CU в межах кадру або вирізки модуль 559 фільтра може обчислювати одну або більше метрик, зв'язаних з декодованими пікселями CU для множинних введень (тобто PI, EI, pRI та RI), щоб визначити, який фільтр (які фільтри) з набору (наборів) застосовувати до кожного введення. Як альтернатива, модуль 559 фільтра може обчислювати одну або більше метрик для єдиного введення, такого як pRI або RI. Модуль 559 фільтра визначає, який фільтр застосовувати, на основі метрик, визначених для конкретного пікселя або групи пікселів. При використанні значення Лапласіана із зміненою сумою і напрямку як прикладів для Метрики 1 та Метрики 2 і при використанні відповідності, показаної на фіг. 4А, як приклад, якщо модуль 559 фільтра визначає, що піксель або група пікселів має значення Лапласіана із зміненою сумою в діапазоні 1-2 і напрямком, відповідно до Діапазону 2-3, модуль 559 фільтра може застосовувати Фільтр 2 до цього пікселя або групи пікселів. Якщо модуль 559 фільтра визначає, що піксель або група пікселів має значення Лапласіана із зміненою сумою в Діапазоні 1-4 і напрямком, відповідно до Діапазону 2-2, модуль 559 фільтра може застосовувати Фільтр 6 до цього пікселя або групи пікселів і так далі. Фільтр може, загалом, допускати будь-який тип форми або розподілу підтримки фільтра. Підтримка фільтра стосується форми фільтра щодо заданого пікселя, що піддається фільтрації, і коефіцієнти фільтра можуть визначати зважування, що застосовується до значень сусідніх пікселів відповідно до підтримки фільтра. Відповідно до методик цього розкриття дані синтаксису можуть бути включені в бітовий потік, щоб сигналізувати декодеру, як були кодовані фільтри (наприклад, як були кодовані коефіцієнти фільтра), як і діапазони метрики активності, для яких повинні використовуватися різні фільтри.

Для кожного CU в межах кадру або вирізки модуль 559 фільтра може обчислювати одну або більше метрик, зв'язаних з декодованими пікселями CU для множинних введень (тобто PI, EI, pRI та RI), щоб визначити, який фільтр (які фільтри) з набору (наборів) застосовувати до кожного введення. Як альтернатива, модуль 559 фільтра може обчислювати одну або більше метрик для єдиного введення, такого як pRI або RI. Як альтернатива, модуль 559 фільтра може обчислювати одну або більше метрик для єдиного введення, такого як pRI або RI. Модуль 559 фільтра визначає, який фільтр застосовувати, на основі метрик, визначених для конкретного пікселя або групи пікселів. При використанні значення Лапласіана із зміненою сумою і напрямку як прикладів для Метрики 1 та Метрики 2 і при використанні відповідності, показаної на фіг. 4А, як приклад, якщо модуль 559 фільтра визначає, що піксель або група пікселів має значення Лапласіана із зміненою сумою в діапазоні 1-2 і напрямком, відповідно до Діапазону 2-3, модуль 559 фільтра може застосовувати Фільтр 2 до цього пікселя або групи пікселів. Якщо модуль 559 фільтра визначає, що піксель або група пікселів має значення Лапласіана із зміненою сумою в Діапазоні 1-4 і напрямком, відповідно до Діапазону 2-2, модуль 559 фільтра може застосовувати Фільтр 6 до цього пікселя або групи пікселів і так далі. Фільтр може, загалом, допускати будь-який тип форми або розподілу підтримки фільтра. Підтримка фільтра стосується форми фільтра щодо заданого пікселя, що піддається фільтрації, і коефіцієнти фільтра можуть визначати зважування, що застосовується до значень сусідніх пікселів відповідно до підтримки фільтра. Відповідно до методик цього розкриття дані синтаксису можуть бути включені в бітовий потік, щоб сигналізувати декодеру, як були кодовані фільтри (наприклад, як були кодовані коефіцієнти фільтра), а також діапазони метрики активності, для яких повинні використовуватися різні фільтри.

Модуль 554 прогнозування приймає синтаксис прогнозування (такий як вектори руху) від модуля 552 ентропійного кодування. Використовуючи синтаксис прогнозування, модуль 554 прогнозування генерує блоки прогнозування, які були використані, щоб кодувати відеоблоки. Модуль 556 оберненого квантування здійснює обернене квантування, а модуль 558 оберненого перетворення здійснює обернене перетворення, щоб змінити коефіцієнти залишкових

відеоблоків зворотно в ділянку пікселя. Суматор 564 комбінує кожний блок прогнозування з відповідним залишковим блоком, виведеним модулем 558 оберненого перетворення, щоб реконструювати відеоблок.

Модуль 559 фільтра генерує коефіцієнти фільтра, що підлягають застосуванню для кожного введення CU, і потім застосовує такі коефіцієнти фільтра, щоб відфільтрувати реконструйовані відеоблоки цього CU. Фільтрація, наприклад, може містити додаткову деблокуючу фільтрацію, яка згладжує краї і/або усуває артефакти, зв'язані з відеоблоками, фільтрацію із зниженням рівня шуму, щоб зменшити шум квантування, або будь-який інший тип фільтрації, який може поліпшити якість кодування. Відфільтровані відеоблоки акумулюються в пам'яті 562, щоб реконструювати декодовані кадри (або інші декодовані елементи) відеоінформації. Декодовані елементи можуть бути виведені від відеодекодера 560 для представлення користувачу, але також можуть зберігатися для використання при подальшому декодуванні з прогнозуванням.

У ділянці кодування відео прийнято застосовувати фільтрацію в кодері та декодері з метою поліпшити якість декодованого відеосигналу. Фільтрація може застосовуватися за допомогою пост-фільтра, в цьому випадку відфільтрований кадр не використовується для прогнозування майбутніх кадрів. Як альтернатива, фільтрація може застосовуватися "внутрішньоконтурно", в цьому випадку відфільтрований кадр може використовуватися для прогнозування майбутніх кадрів. Бажаний фільтр може бути створений шляхом мінімізації помилки між оригінальним сигналом і декодованим відфільтрованим сигналом. Звичайно така фільтрація ґрунтувалася на застосуванні одного або більше фільтрів до реконструйованого зображення. Наприклад, деблокуючий фільтр може застосовуватися до реконструйованого зображення до того, як це зображення зберігається в запам'ятовуючому пристрої, або деблокуючий фільтр і один додатковий фільтр могли б застосовуватися до реконструйованого зображення до того, як це зображення зберігається в запам'ятовуючому пристрої.

Подібно до квантування коефіцієнтів перетворення, коефіцієнти фільтра  $h(k, l)$ , де  $k = -K, \dots, K$ , та  $l = -L, \dots, L$ , також можуть бути квантовані.  $K$  та  $L$  можуть являти собою цілі значення. Коефіцієнти фільтра  $h(k, l)$  можуть бути квантовані як:

$$f(k, l) = \text{round}(\text{normFact} \cdot h(k, l))$$

де  $\text{normFact}$  - фактор нормалізації та  $\text{round}$  - операція округлення, що виконується, щоб досягнути квантування до бажаної бітової глибини. Квантування коефіцієнтів фільтра може здійснюватися модулем 349 фільтра з фіг. 3 під час кодування, а деквантування або обернене квантування може здійснюватися відносно декодованих коефіцієнтів фільтра модулем 559 фільтра з фіг. 5. Фільтр  $h(k, l)$  призначений для того, щоб загалом представляти будь-який фільтр. Наприклад, фільтр  $h(k, l)$  може застосовуватися до будь-якого з множинних введень. У деяких випадках множинні введення, зв'язані з відеоблоком, будуть використовувати різні фільтри, в цьому випадку множинні фільтри, подібні до  $h(k, l)$ , можуть бути квантовані і деквантовані, як описується вище.

Квантовані коефіцієнти фільтра кодуються і відправляються від пристрою-джерела, зв'язаного з кодером 350, до пристрою-адресата, зв'язаного з декодером 560, як частина кодованого бітового потоку. У прикладі, представленому вище, значення  $\text{normFact}$  звичайно дорівнює  $2^n$ , хоча інші значення також могли б бути використані. Більші значення  $\text{normFact}$  приводять до більш точного квантування, так що квантовані коефіцієнти фільтра  $f(k, l)$  забезпечують більш високу ефективність. Однак більші значення  $\text{normFact}$  можуть створити коефіцієнти  $f(k, l)$ , які вимагають більшої кількості бітів для сигналізування декодеру.

У декодері 560 декодовані коефіцієнти фільтра  $f(k, l)$  можуть застосовуватися до прийнятного введення. Наприклад, якщо декодовані коефіцієнти фільтра підлягають використанню у відношенні  $RI$ , коефіцієнти фільтра можуть застосовуватися до пост-деблокованого реконструйованого зображення  $RI(i, j)$ ,

де  $i=0, \dots, M$  та  $j=0, \dots, N$  таким чином:

$$\tilde{RI}(i, j) = \sum_{k=-K}^K \sum_{l=-L}^L f(k, l) RI(i+k, j+l) / \sum_{k=-K}^K \sum_{l=-L}^L f(k, l)$$

Змінні  $M$ ,  $N$ ,  $K$  та  $L$  можуть являти собою цілі числа.  $K$  та  $L$  можуть визначати блок пікселів, який захоплює два вимірювання від  $-K$  до  $K$  і від  $-L$  до  $L$ . Фільтри, що застосовуються до інших введень, можуть застосовуватися аналогічним чином.

Методики цього розкриття можуть підвищувати ефективність пост-фільтра або внутрішньоконтурного фільтра, а також можуть скорочувати кількість бітів, необхідних, щоб сигналізувати коефіцієнти фільтра  $f(k, l)$ . У деяких випадках деяка кількість різних пост-фільтрів або внутрішньоциклічних фільтрів сигналізується декодеру для кожної серії відеоблоків, наприклад, для кожного кадру, вирізки, частини кадру, групи кадрів (GOP) тощо. Для кожного

фільтра додаткова інформація включена в бітовий потік, щоб ідентифікувати CU, макроблоки і/або пікселі, для яких повинен застосовуватися заданий фільтр.

Кадри можуть бути ідентифіковані за номером кадру і/або типом кадру (наприклад, I-кадри, P-кадри або B-кадри). I-кадри стосуються внутрішніх кадрів, які зазнали внутрішнього прогнозування. P-кадри стосуються прогнозованих кадрів, які містять відеоблоки, прогнозовані на основі списку даних (наприклад, один попередній кадр). B-кадри стосуються двонаправлених прогнозованих кадрів, які прогнозовані на основі двох списків даних (наприклад, попередній і наступний кадр). Макроблоки можуть бути ідентифіковані за типами макроблока і/або значеннями параметра діапазону квантування (QP), що використовуються для реконструкції макроблока.

Коефіцієнти фільтра  $f(k, l)$  для будь-якого введення можуть бути кодовані з використанням прогнозування від коефіцієнтів, сигналізованих для попередніх CU. Для кожного введення CU  $m$  (наприклад, кожний кадр, вирізка або GOP) кодер може кодувати і передавати набір  $M$ -фільтрів:  $g_i^m$ , де  $i=0, \dots, M-1$ .

Для кожного фільтра бітовий потік також може кодуватися, щоб ідентифікувати комбінацію діапазонів для двох або більше метрик, для яких повинен використовуватися фільтр.

Коефіцієнти фільтра можуть бути прогнозовані з використанням реконструйованих коефіцієнтів фільтра, використаних в попередньому CU. Попередні коефіцієнти фільтра можуть бути представлені як:

$$f_i^n, \text{ де } i=0, \dots, N-1,$$

У цьому випадку кількість CU  $n$  може використовуватися, щоб ідентифікувати один або більше фільтрів, використаних для прогнозування поточних фільтрів, і кількість  $n$  може бути відправлена до декодера як частина кодованого бітового потоку. Додатково до цього інформація може кодуватися і передаватися до декодера, щоб ідентифікувати комбінації діапазонів для двох або більше метрик, для яких використовується кодування з прогнозуванням.

Амплітуда коефіцієнтів фільтра  $g(k, l)$  залежить від значень  $k$  та  $l$ . Звичайно коефіцієнт з найбільшою амплітудою - це коефіцієнт  $g(0,0)$ . Інші коефіцієнти, які, як очікується, мають великі амплітуди, є коефіцієнтами, для яких значення  $k$  або  $l$  дорівнює 0. Це явище може використовуватися, щоб додатково скоротити кількість бітів, необхідних для сигналізування коефіцієнтів. Індексні значення  $k$  та  $l$  можуть визначати місця розташування в межах відомої підтримки фільтра.

Коефіцієнти:

$$g_i^m(k, l), i=0, \dots, M-1$$

для кожного кадру  $m$  можуть кодуватися з використанням параметризованих кодів змінної довжини, таких як код Голомба (Golomb) або експонентний код Голомба (exp-Golomb), визначених відповідно до параметра  $p$ . Шляхом зміни значення параметра  $p$ , які визначає параметризовані коди змінної довжини, ці коди можуть використовуватися, щоб ефективно представити широкий діапазон розподілу джерел. Розподіл коефіцієнтів  $g(k, l)$  (тобто імовірність того, що вони будуть мати великі або маленькі значення) залежить від значень  $k$  та  $l$ . Отже, щоб збільшити ефективність кодування, для кожного кадру  $m$  значення параметра  $p$  передається для кожної пари  $(k, l)$ . Параметр  $p$  може використовуватися для параметризованого кодування змінної довжини, коли коефіцієнти кодування:

$$g_i^m(k, l), \text{ де } k=-K, \dots, K, l=-L, \dots, L.$$

Як було описано вище, відеодекодер 560 являє собою приклад відеодекодера, виконаного з можливістю визначати першу метрику для групи пікселів у межах блока пікселів, визначати другу метрику для групи пікселів, визначати фільтр на основі першої метрики і другої метрики та генерувати відфільтроване зображення шляхом застосування фільтра до групи пікселів. Відеодекодер 560 також являє собою приклад відеодекодера, виконаного з можливістю визначати першу метрику для блока пікселів, при цьому перша метрика визначається на основі порівняння піднабору пікселів у блоці з іншими пікселями в блоці; визначати другу метрику для блока пікселів; визначати фільтр на основі першої метрики і другої метрики; і генерувати відфільтроване зображення шляхом застосування фільтра до блока пікселів.

Як було описано вище, відеодекодер 560 також являє собою приклад відеодекодера, виконаного з можливістю визначати відповідність комбінацій діапазонів фільтрам, при цьому комбінація діапазонів містить діапазон для першої метрики і діапазон для другої метрики, при цьому кожна комбінація діапазонів має унікальну ідентифікацію комбінації діапазонів (ID), при цьому кожна унікальна комбінація діапазонів ID відповідає порядковому значенню для комбінації діапазонів; призначати унікальний ID групи групам комбінацій діапазонів на основі порядкових значень для комбінацій діапазонів, при цьому кожний унікальний ID групи відповідає

порядковому значенню для групи; і кодувати набори коефіцієнтів фільтра, що відповідають фільтрам, на основі унікальних ID групи. Відеодекодер 560 може кодувати набори коефіцієнтів фільтра шляхом генерування наборів коефіцієнтів фільтра на основі інформації, прийнятої в кодованому бітовому потоці. Відеодекодер 560 може генерувати набори коефіцієнтів фільтра з використанням методик різницевого кодування.

Відеодекодер 560 також являє собою приклад відеодекодера, виконаного з можливістю приводити у відповідність першу комбінацію діапазонів до першого фільтра, при цьому перша комбінація діапазонів містить перший діапазон значень для першої метрики і перший діапазон значень для другої метрики; приводити у відповідність другу комбінацію діапазонів до другого фільтра, при цьому друга комбінація діапазонів містить другий діапазон значень для першої метрики і другий діапазон значень для другої метрики; приводити у відповідність поточну комбінацію діапазонів до фільтра, при цьому поточна комбінація діапазонів містить перший діапазон значень для першої метрики і другий діапазон значень для другої метрики. Приведення у відповідність поточної комбінації діапазонів фільтру може включати в себе приведення у відповідність поточної комбінації діапазонів до першого фільтра у відповідь на прийом першого кодового слова, при цьому перше кодове слово показує, що поточна комбінація діапазонів приведена у відповідність до того самого фільтра, що і перша комбінація діапазонів; приведення у відповідність поточної комбінації діапазонів до другого фільтра у відповідь на прийом другого кодового слова, при цьому друге кодове слово показує, що поточна комбінація діапазонів приведена у відповідність до того самого фільтра, що і друга комбінація діапазонів; і приведення у відповідність поточної комбінації діапазонів до третього фільтра у відповідь на прийом третього кодового слова, при цьому третє кодове слово показує цей третій фільтр. Відеодекодер 560 також являє собою приклад відеодекодера, виконаного з можливістю генерувати відповідність комбінацій діапазонів фільтрам, при цьому комбінація діапазонів містить діапазон для першої метрики і діапазон для другої метрики; приводити у відповідність поточну комбінацію діапазонів до того самого фільтра, що і для попередньої комбінації діапазонів, у відповідь на прийом першого кодового слова, яке сигналізує, що поточна комбінація діапазонів приведена у відповідність до того самого фільтра, що і попередня комбінація діапазонів; і приводити у відповідність поточну комбінацію діапазонів до фільтра, ідентифікованого другим кодовим словом, у відповідь на прийом другого кодового слова, яке сигналізує, що поточна комбінація діапазонів приведена у відповідність до іншого фільтра, ніж попередня комбінація діапазонів.

Як було представлено вище, декілька різних типів метрик можуть використовуватися разом з методиками мультиметричної фільтрації, яка описується в цьому розкритті. Деякі з цих метрик - це метрики активності, які визначають кількість активності, зв'язану з одним або більше блоками пікселів у межах відеоданих. Метрики активності можуть містити метрики дисперсії, що показують дисперсію пікселя в межах набору пікселів. Як буде описано, деяка з цих метрик активності - зв'язана з напрямком. Наприклад, метрика горизонтальної активності визначає кількість активності вздовж горизонтальної осі, метрика вертикальної активності визначає кількість активності вздовж вертикальної осі, метрика діагональної активності визначає кількість активності вздовж діагональної осі тощо.

Деякі метрики активності - не зв'язані з напрямком. Наприклад, значення Лапласіана із зміненою сумою є метрикою активності, основою на двомірному вікні пікселів, які оточують поточний піксель або поточну групу пікселів. Для поточного пікселя (i, j) значення Лапласіана із зміненою сумою може обчислюватися таким чином:

$$\text{var}(i, j) = \sum_{k=-K}^K \sum_{l=-L}^L |2R(i+k, j+l) - R(i+k-1, j+l) - R(i+k+1, j+l)| + \\ |2R(i+k, j+l) - R(i+k, j+l-1) - R(i+k, j+l+1)| \quad (1),$$

де k представляє значення складання піксельних значень від -K до K та l представляють значення складання від -L до L для двомірного вікна, яке проходить від -K до K і від -L до L, при цьому i та j представляють координати пікселя даних пікселя,  $R(i, j)$  представляє значення даного пікселя в координатах i та j, а  $\text{var}(i, j)$  - це метрика активності (тобто значення Лапласіана із зміненою сумою).

Методики цього розкриття також можуть застосовуватися з використанням метрик, зв'язаних з напрямком, для горизонтальної активності, вертикальної активності та діагональної активності. Рівняння 2 та 3 демонструють приклади того, як горизонтальна активність і вертикальна активність можуть обчислюватися для поточного пікселя (x, y) шляхом порівняння значення пікселя (Rec), такого як інтенсивність, поточного пікселя з піксельним значенням сусідніх пікселів.

$$\text{Hor\_act}(x,y) = R(2*\text{Rec}[x][y] - \text{Rec}[x+1][y] - \text{Rec}[x-1][y]) \quad (2)$$

$$\text{Ver\_act}(x,y) = R(2*\text{Rec}[x][y] - \text{Rec}[x][y+1] - \text{Rec}[x][y-1]) \quad (3)$$

Як показано в рівнянні 2, при визначенні горизонтальної активності поточний піксель (x, y) може порівнюватися з лівим сусіднім (x-1, y) і правим сусіднім (x+1, y). Як показано в рівнянні 3, при визначенні вертикальної активності поточний піксель (x, y) може порівнюватися з верхнім сусіднім (x, y+1) і нижнім сусіднім (x, y-1).

Рівняння 4 та 5 демонструють приклади того, як обчислюється діагональна активність для поточного пікселя (x, y), шляхом порівняння піксельного значення (Rec) поточного пікселя з піксельним значенням сусідніх пікселів.

$$45\text{deg\_act}(x,y) = R(2*\text{Rec}[x][y] - \text{Rec}[x+1][y+1] - \text{Rec}[x-1][y-1]) \quad (4)$$

$$135\text{deg\_act}(x,y) = R(2*\text{Rec}[x][y] - \text{Rec}[x-1][y+1] - \text{Rec}[x+1][y-1]) \quad (5)$$

Як показано в рівнянні 4, діагональна активність може обчислюватися, наприклад, в напрямку 45 градусів шляхом порівняння поточного пікселя (x, y) із сусіднім верхнім правим (x+1, y+1) і сусіднім нижнім лівим (x-1, y-1). Як показано в рівнянні 5, діагональна активність може обчислюватися, наприклад, в напрямку 135 градусів шляхом порівняння поточного пікселя (x, y) із сусіднім верхнім лівим (x-1, y+1) і сусіднім нижнім правим (x+1, y-1).

Представлені вище рівняння 2-5 ілюструють, як горизонтальна активність, вертикальна активність та діагональна активність можуть визначатися на попиксельній основі, але в деяких варіантах здійснення горизонтальна активність, вертикальна активність і діагональна активність можуть визначатися на погруповій основі, де група пікселів - блок пікселів  $2 \times 2$ ,  $4 \times 4$  або  $M \times N$ . У такому варіанті здійснення горизонтальна активність, наприклад, може визначатися шляхом порівняння піксельних значень з поточної групи зі значеннями пікселів з лівої групи і правої групи, аналогічно тому, як описується в рівнянні 2; і вертикальна активність може визначатися шляхом порівняння поточної групи пікселів з верхньою групою і нижньою групою, аналогічно тому, як описується в рівнянні 3. Подобним чином, 45-градусна діагональна активність може визначатися шляхом порівняння поточної групи пікселів з верхньою правою сусідньою групою і лівою нижньою сусідньою групою, аналогічно тому, як описується в рівнянні 4, а 135-градусна діагональна активність може визначатися шляхом порівняння поточної групи пікселів з верхньою лівою сусідньою групою і правою нижньою сусідньою групою, аналогічно тому, як описується в рівнянні 5.

У деяких варіантах здійснення горизонтальна активність, вертикальна активність, 45-градусна діагональна активність та 135-градусна діагональна активність можуть визначатися шляхом порівняння поточного пікселя або групи пікселів із сусідніми пікселями або групами пікселів тільки в одному напрямку. Наприклад, замість визначення горизонтальної активності на основі порівняння поточного пікселя з лівим сусіднім і правим сусіднім, горизонтальна активність може визначатися на основі тільки лівого сусіда або тільки правого сусіда. Додатково, в деяких варіантах здійснення, горизонтальна активність, вертикальна активність, 45-градусна діагональна активність та 135-градусна діагональна активність можуть визначатися з використанням середніх або зважених середніх з ділянок сусідніх пікселів замість окремих сусідніх пікселів або окремих груп пікселів.

Значення, одержані з рівнянь 2-5, можуть бути розділені на кінцеве число діапазонів, як то 2, 4, 8 або будь-яке інше кінцеве число, і кожному діапазону може призначатися ідентифікація діапазону. З посиланням знову на фіг. 4А, наприклад, Діапазон 1-1, Діапазон 1-2, Діапазон 2-1 тощо - це все приклади ідентифікацій діапазону. Як один приклад значення горизонтальної активності можуть бути розділені на чотири діапазони, і діапазонам можуть призначатися ID Діапазону 1-1, Діапазону 1-2, Діапазону 1-3 та Діапазону 1-4. Горизонтальні порогові значення (тобто  $\text{ThH}_1, \dots, \text{ThH}_{P-1}$ ) можуть визначати, де починаються і закінчуються діапазони. Таблиця 2, представлена нижче, демонструє загальний випадок того, як горизонтальні ID могли б призначатися Р діапазонам.

Таблиця 2

Індекс метрики активності

Стан $\text{Hor\_act\_B}$	Горизонтальний ID
$\text{Hor\_act\_B} < \text{ThH}_1$	Діапазон 2-1
$\text{ThH}_1 \leq \text{Hor\_act\_B} < \text{ThH}_2$	Діапазон 2-2
...	...
$\text{ThH}_{P-1} \leq \text{Hor\_act\_B}$	Діапазон 2-P

При використанні прикладу з Таблиці 2, якщо поточний піксель має значення горизонтальної активності більше  $ThH_1$ , але менше  $ThH_2$ , то поточний піксель знаходиться в діапазоні 2-2 для метрики 2. Поточні пікселі можуть призначатися вертикальним діапазоном з вертикальними ID, 45-градусним діагональним діапазоном - з 45-градусними діагональними ID і 135-градусним діагональним діапазоном - з 135-градусними діагональними ID, подібно до того, як описується вище в Таблиці 2 для горизонтальних діапазонів і горизонтальних ID.

Будь-яка з горизонтальної активності, вертикальної активності, 45-градусної діагональної активності та 135-градусної діагональної активності може використовуватися як метрика відповідно до методик фільтрації мультиметричного фільтра, що описуються в цьому розкритті. Наприклад, повертаючись до фіг. 4А, Метрика 1 могла б бути вимірюванням для вертикальної активності, а Метрика 2 могла бути вимірюванням для горизонтальної активності. У такому прикладі елемент фільтра, такий як модуль 349 фільтра з фіг. 4А або модуль 559 фільтра з фіг. 5, може визначати фільтр для пікселя або групи пікселів на основі горизонтальної активності пікселя або групи пікселів і вертикальної активності пікселя або групи пікселів. Якщо, наприклад, поточний піксель має метрику горизонтальної активності, яка попадає в діапазон 2-3, і метрику вертикальної активності, яка попадає в діапазон 1-3, то елемент фільтра фільтрує піксель, використовуючи Фільтр 4. Подібним чином, комбінації 45-градусної діагональної активності та 135-градусної діагональної активності, 45-градусної діагональної активності та горизонтальної активності, 45-градусної діагональної активності та вертикальної активності, 135-градусної діагональної активності та горизонтальної активності або 135-градусної діагональної активності та вертикальної активності також можуть використовуватися блоком фільтра для вибору фільтра для пікселя або групи пікселів. У деяких варіантах здійснення три або всі чотири з горизонтальної активності, вертикальної активності, 45-градусної діагональної активності та 135-градусної діагональної активності можуть використовуватися елементом фільтра для вибору фільтра пікселя або групи пікселів.

У варіантах здійснення, описаних вище, горизонтальна активність, вертикальна активність, 45-градусна діагональна активність та 135-градусна діагональна активність - всі можуть використовуватися як метрики, як Метрика 1 і/або Метрика 2 на фіг. 4А, наприклад. У деяких варіантах здійснення, однак горизонтальна активність, вертикальна активність, 45-градусна діагональна активність та 135-градусна діагональна активність самі по собі можуть і не бути метриками, але замість цього можуть використовуватися як проміжні визначення для визначення загальної метрики напрямку. Метрика напрямку загалом описує, в якому напрямку (відсутність напрямку, горизонтальний, вертикальний, 45-градусний діагональний або 135-градусний діагональний) пікселі більше усього змінюються.

В одному прикладі, з використанням тільки горизонтальної активності та вертикальної активності, як описується в рівняннях 2 та 3, напрямком для пікселя міг би визначатися на основі наступних умов:

Напрямок 1=горизонтальний, якщо  $Hor\_activity > k1 * Ver\_activity$

Напрямок 2=вертикальний, якщо  $Ver\_activity > k2 * Hor\_activity$

Напрямок 0=відсутність напрямку, інакше.

Константи,  $k1$  та  $k2$ , можуть вибиратися так, що напрямком тільки вважається напрямком 1 або напрямком 2, якщо горизонтальна активність значно більше, ніж вертикальна активність, або вертикальна активність значно більше, ніж горизонтальна активність. Якщо горизонтальна активність і вертикальна активність рівні або приблизно рівні, то напрямком - це напрямком 0. Напрямок 1 загалом вказує, що піксельні значення більше змінюються в горизонтальному напрямку, ніж у вертикальному напрямку, а напрямком 2 вказує, що піксельні значення більше змінюються у вертикальному напрямку, ніж в горизонтальному напрямку. Напрямок 0 вказує, що зміни піксельних значень в горизонтальному напрямку приблизно дорівнюють змінам піксельних значень у вертикальному напрямку.

Визначена метрика напрямку (наприклад, напрямком 0, напрямком 1, напрямком 2) може використовуватися як метрика для методик мультиметричної фільтрації, описаних в цьому розкритті. При використанні прикладу з фіг. 4А знову Метрика 1 могла б бути метрикою дисперсії, як то значення Лапласіана із зміненою сумою, тоді як Метрика 2 могла б бути визначенням напрямку, як описано вище. Як описується з посиланням на фіг. 4А, кожний з напрямку 1, напрямку 2 і напрямку 0 може бути зв'язаний з діапазоном Метрики 2, хоча напрямком 1, напрямком 2 і напрямком 0 представляють кінцеві визначення замість спектра значень.

Додатково до використання тільки горизонтальної активності та вертикальної активності, як описано вище, методики цього розкриття також включають в себе використання 45-градусної

діагональної активності та 135-градусної діагональної активності, як описується в рівняннях 4 та 5, щоб визначити напрямку на основі наступних умов:

Напрямок =1, якщо  $45deg\_activity > k1 * 135deg\_activity$

Напрямок =2, якщо  $135deg\_activity > k2 * 45deg\_activity$

5 Напрямок =0 - інакше.

Визначення напрямку на основі 45-градусної діагональної активності та 135-градусної діагональної активності можуть використовуватися як метрика з іншою метрикою, як то значення Лапласіана із зміненою сумою, як описано вище.

Додатково метрика напрямку може також визначатися на основі наступних умов:

10 Напрямок =1, якщо  $45deg\_activity > k1 * 135deg\_activity, k2 * Hor\_activity \vee k3 * Ver\_activity$

Напрямок =2, якщо  $135deg\_activity > k4 * 45deg\_activity, k5 * Hor\_activity \vee k6 * Ver\_activity$

Напрямок =3, якщо  $Hor\_activity > k7 * Ver\_activity, k8 * 135deg\_activity \vee k9 * 45deg\_activity$

Напрямок =4, якщо  $Ver\_activity > k10 * Hor\_activity, k11 * 135deg\_activity \vee k12 * 45deg\_activity$

Напрямок =0, інакше.

15 Як описано вище,  $k1-k12$  - константи, вибрані для визначення того, наскільки більше одна з горизонтальної активності, вертикальної активності, 45-градусної діагональної активності та 135-градусної діагональної активності потребує порівняння з іншими для того, щоб був вибраний визначений напрямок. Визначення напрямку на основі горизонтальної активності, вертикальної активності, 45-градусної діагональної активності та 135-градусної діагональної активності можуть використовуватися як метрика з іншою метрикою, як то значення Лапласіана із зміненою сумою, як описується вище.

Інша метрика, яка може використовуватися з методиками цього розкриття, включає в себе метрику краю. Метрика краю загалом визначає кількість активності, яка могла б показувати наявність краю в блоці пікселів. Край може з'явитися, наприклад, в блоці пікселів, якщо цей блок пікселів містить межу об'єкта в межах зображення. Один приклад виявлення краю включає в себе використання чотирьох пікселів, сусідніх з поточним пікселем (наприклад, лівий, правий, верхній, нижній) або використання восьми пікселів, сусідніх з поточним пікселем (лівий, правий, верхній, нижній, верхній правий, верхній лівий, нижній правий, нижній лівий). Додатково визначення типу краю може включати в себе використання двох сусідніх пікселів, таких як

30 верхній і нижній, лівий і правий, верхній лівий і нижній правий або верхній правий і нижній лівий. Псевдокод, представлений нижче, демонструє приклади того, як інформація краю може обчислюватися для поточного пікселя ( $x, y$ ) шляхом порівняння піксельного значення ( $Rec$ ), такого як інтенсивність, поточного пікселя з піксельними значеннями тих сусідніх пікселів (тобто 4/8 пікселів).

35 Змінна `EdgeType` (ТипКраю) ініціюється на 0. Кожен раз, коли твердження істинне, змінна `EdgeType` або збільшується на 1 (як показано в псевдокоді `EdgeType ++`), або зменшується на 1 (як показано в псевдокоді `EdgeType --`). `Rec[x][y]` стосується значення пікселя, такого як інтенсивність пікселя, пікселя, розташованого на ( $x, y$ ). Перше групування тверджень "якщо" ("if") - для порівняння поточного пікселя з верхнім, нижнім, лівим і правим сусіднім. Друге групування тверджень "якщо" - для порівняння поточного пікселя з верхнім лівим, верхнім правим, нижнім лівим і нижнім правим сусідами. Методики цього розкриття можуть реалізовуватися з використанням або групи, або обох груп.

`EdgeType=0;`

`if (Rec[x][y]>Rec[x-1][y]) EdgeType ++;`

45 `if (Rec[x][y]<Rec[x-1][y]) EdgeType --;`

`if (Rec[x][y]>Rec[x+1][y]) EdgeType ++;`

`if (Rec[x][y]<Rec[x+1][y]) EdgeType --;`

`if (Rec[x][y]>Rec[x][y-1]) EdgeType ++;`

`if (Rec[x][y]<Rec[x][y-1]) EdgeType --;`

50 `if (Rec[x][y]>Rec[x][y+1]) EdgeType ++;`

`if (Rec[x][y]<Rec[x][y+1]) EdgeType --;`

`if (Rec[x][y]>Rec[x-1][y-1]) EdgeType ++;`

`if (Rec[x][y]<Rec[x-1][y-1]) EdgeType --;`

`if (Rec[x][y]>Rec[x+1][y-1]) EdgeType ++;`

55 `if (Rec[x][y]<Rec[x+1][y-1]) EdgeType --;`

`if (Rec[x][y]>Rec[x-1][y+1]) EdgeType ++;`

`if (Rec[x][y]<Rec[x-1][y+1]) EdgeType --;`

`if (Rec[x][y]>Rec[x+1][y+1]) EdgeType ++;`

`if (Rec[x][y]<Rec[x+1][y+1]) EdgeType --;`



Якщо поточний піксель - локальний максимум, значення пікселя цього пікселя буде більше, ніж у всіх його сусідів, і буде мати тип краю 4 при використанні чотирьох сусідів або тип краю 8 при використанні восьми сусідів. Якщо поточний піксель - локальний мінімум, значення пікселя цього пікселя буде менше, ніж у всіх його сусідів, і буде мати тип краю -4 при використанні чотирьох сусідів або тип краю -8 при використанні восьми сусідів. Так, використання зразкових методик, описаних вище, для визначення типу краю між -4 та 4 або -8 та 8 може використовуватися і при визначенні фільтра. Значення, визначені для типу краю (тобто значення -4 та 4 або значення -8 та 8), можуть приводитися у відповідність до діапазонів метрики, такої як Метрика 1 або Метрика 2 з фіг. 4А. В деяких варіантах здійснення абсолютні значення визначення типу краю можуть приводитися у відповідність до діапазонів так, що тип краю -3 та 3, наприклад, приводився б у відповідність до одного і того самого фільтра.

Обчислення різних метрик, описаних в цьому розкритті, направлені тільки на те, щоб бути прикладами, але не вичерпними. Наприклад, метрики можуть визначатися на основі вікон або рядків пікселів, які включають в себе більше сусідніх пікселів, ніж описується в цьому розкритті.

Додатково в деяких варіантах здійснення метрики, описані в цьому розкритті, можуть обчислюватися з використанням підвибірки пікселів у конкретному рядку або вікні. Наприклад, щоб обчислити метрику активності блока для блока пікселів  $4 \times 4$ , метрики для активності та напрямку можуть обчислюватися таким чином:

Метрика Напряму

$Ver\_act(i, j) = \text{abs}(X(i, j) - X(i, j-1) - X(i, j+1))$

$Hor\_act(i, j) = \text{abs}(X(i, j) - X(i-1, j) - X(i+1, j))$

$H_B = \sum_{i=0,2} \sum_{j=0,2} Hor\_act(i, j)$

$V_B = \sum_{i=0,2} \sum_{j=0,2} Ver\_act(i, j)$

Напрямок = 0,  $1(H_B > k_1 \cdot V_B)$ ,  $2(V_B > k_2 \cdot H_B)$

Метрика Активності

$L_B = H_B + V_B$

5 класів (0, 1, 2, 3, 4)

Метрика

Комбінація Активності та Напряму (наприклад, 15 або 16 комбінацій, як було пояснено вище в прикладі з Фіг. 4В).

$Hor\_act(i, j)$  загалом стосується горизонтальної активності поточного пікселя (i, j), і  $Ver\_act(i, j)$  загалом стосується вертикальної активності поточного пікселя (i, j).  $X(i, j)$  загалом стосується значення пікселя (i, j).  $H_B$  загалом стосується горизонтальної активності блока  $4 \times 4$ , який в цьому прикладі визначається на основі суми горизонтальної активності для пікселів (0,0), (0,2), (2,0) та (2,2).  $V_B$  стосується вертикальної активності блока  $4 \times 4$ , який в цьому прикладі визначається на основі суми вертикальної активності для пікселів (0,0), (0,2), (2,0) та (2,2). «<<1" являє собою операцію множення на два. Як пояснюється вище, на основі значень  $H_B$  та  $V_B$  може бути визначений напрямок. З використанням прикладу, представленого вище, якщо значення  $H_B$  більше, ніж k разів, значення  $V_B$ , то напрямок може бути визначений як напрямок 1 (тобто горизонтальний), що може відповідати більш горизонтальній активності, ніж вертикальній активності. Якщо значення  $V_B$  більше, ніж k разів, значення  $H_B$ , то напрямок може бути визначений як напрямок 2 (тобто вертикальний), що може відповідати більш вертикальній активності, ніж горизонтальній активності. В іншому випадку напрямок може бути визначений як напрямок 0 (тобто відсутність напрямку), що означає, що ні горизонтальна, ні вертикальна активність не є домінантною. Мітки для різних напрямків і співвідношень, що використовуються, щоб визначити напрямки, що складають тільки один приклад, як і інші мітки і співвідношення, також можуть використовуватися.

Активність ( $L_B$ ) для блока  $4 \times 4$  може визначатися як сума горизонтальної та вертикальної активності. Значення  $L_B$  може класифікуватися в діапазон, як описується вище. Цей конкретний приклад демонструє п'ять діапазонів, хоча більша або менша кількість діапазонів може подібним чином використовуватися. Як описано вище, фільтр може бути вибраний на основі двомірної відповідності активності та напрямку фільтрам, як описується з посиланням на фіг. 4А та фіг. 4В, або активність і напрямок можуть комбінуватися в єдину метрику, і ця єдина метрика може використовуватися для вибору фільтра.

Фіг. 6А представляє блок пікселів  $4 \times 4$ . При використанні методик підвибірки, описаних вище, тільки чотири з шістнадцяти пікселів використовуються. Чотири пікселі - піксель (0,0), який позначається як піксель 601, піксель (2,0), який позначається як піксель 602, піксель (0,2), який позначається як піксель 603, і піксель (2,2), який позначається як піксель 604. Горизонтальна активність пікселя 601 (є  $hor\_act(0,0)$ ), наприклад, визначається на основі лівого сусіднього пікселя і правого сусіднього пікселя. Правий сусідній піксель позначається як піксель

605. Лівий сусідній піксель розташовується в іншому блоці, ніж блок  $4 \times 4$ , і не показаний на фіг. 6А. Вертикальна активність пікселя 602 (тобто  $vert\_act(2,0)$ ), наприклад, визначається на основі верхнього сусіднього пікселя і нижнього сусіднього пікселя. Нижній сусідній піксель позначається як піксель 606, і верхній сусідній піксель розташовується в іншому блоці, ніж блок  $4 \times 4$ , і не показаний на фіг. 6А.

Загалом, використовуючи ті самі методики, що і описані вище, метрика активності блока може також обчислюватися з використанням різних піднаборів пікселів таким чином:

Метрика Напрямку

$Ver\_act(i, j) = \text{abs}(X(i, j) - X(i, j-1) - X(i, j+1))$

$Hor\_act(i, j) = \text{abs}(X(i, j) - X(i-1, j) - X(i+1, j))$

$H_B = \sum_{i=1,2} \sum_{j=1,2} H(i, j)$

$V_B = \sum_{i=1,2} \sum_{j=1,2} V(i, j)$

Напрямок = 0, 1  $1(H > k_1 * V)$ ,  $2(V > k_2 * H)$

Метрика Активності

$L_B = H_B + V_B$

5 класів (0, 1, 2, 3, 4)

Метрика

Комбінація Активності та Напрямку (наприклад, 15 або 16 комбінацій, як було пояснено вище в прикладі з Фіг. 4В)

Цей інший піднабір пікселів для обчислення  $H_B$  та  $V_B$  включає в себе пікселі (1,1), (2,1), (1,2) та (2,2), показані на фіг. 6В як пікселі 611, 612, 613 та 614 відповідно. Як можна побачити з фіг. 6В, всі з верхніх сусідніх пікселів, нижніх сусідніх пікселів, правих сусідніх пікселів і лівих сусідніх пікселів для пікселів 611, 612, 613 та 614 розташовуються в межах блока  $4 \times 4$ . У прикладі з фіг. 6В пікселі 611, 612, 613 та 614 всі розташовуються всередині блока, на протилежному розташуванню на межі блока. В інших варіантах здійснення додаткові різні піднабори пікселів можуть бути вибрані. Наприклад, піднабори можуть бути вибрані так, що сусідні пікселі зверху і знизу для пікселів піднабору - в межах блока  $4 \times 4$ , але деякі сусідні пікселі зліва і праворуч - в сусідніх блоках. Піднабори також можуть вибиратися так, що сусідні пікселі зліва і праворуч для пікселів піднабору - в межах блока  $4 \times 4$ , але деякі сусідні пікселі зверху і знизу - в сусідніх блоках.

Загалом, використовуючи ті самі методики, що і описані вище, метрика активності блока може також обчислюватися з використанням піднабору з восьми пікселів таким чином:

Метрика Напрямку

$Ver\_act(i, j) = \text{abs}(X(i, j) - X(i, j-1) - X(i, j+1))$

$Hor\_act(i, j) = \text{abs}(X(i, j) - X(i-1, j) - X(i+1, j))$

$H_B = \sum_{i=0,1,2,3} \sum_{j=1,2} H(i, j)$

$V_B = \sum_{i=0,1,2,3} \sum_{j=1,2} V(i, j)$

Напрямок = 0, 1  $1(H > k_1 * V)$ ,  $2(V > k_2 * H)$

Метрика Активності

$L_B = H_B + V_B$

5 класів (0, 1, 2, 3, 4)

Метрика

Комбінація Активності та Напрямку (наприклад, 15 або 16 комбінацій, як було пояснено вище в прикладі з Фіг. 4В)

Цей інший піднабір пікселів для обчислення  $H_B$  та  $V_B$  включає в себе пікселі (0,1), (1,1), (2,1), (3,1), (0,2), (1,2), (2,2) та (3,2), показані на фіг. 6С як пікселі 621, 622, 623 та 624, 625, 626, 627 та 628 відповідно. Як можна побачити з фіг. 6С, всі з верхніх сусідніх пікселів і нижніх сусідніх пікселів 621, 622, 623 та 624, 625, 626, 627 та 628 розташовуються в межах блока  $4 \times 4$ , хоча пікселі 621 та 625 мають ліві сусідні пікселі в лівому сусідньому блоці кожний, а пікселі 624 та 628 мають праві сусідні пікселі в правому сусідньому блоці кожний. Цей конкретний вибір пікселів може зменшити складність кодера і/або декодера шляхом уникнення потреби в лінійному буфері для зберігання піксельних значень сусідніх блоків зверху і/або знизу. Внаслідок порядку растрового сканування зліва-направо, зверху-вниз лінійні буфери для піксельних значень сусідніх блоків зверху і знизу часто повинні зберігати піксельні значення для всього верхнього або нижнього рядка, який у випадку 1080Р відео, наприклад, міг б становити 1920 пікселів. Лінійні буфери для сусідніх блоків зліва і праворуч, однак, часто повинні зберігати тільки значення для одного LCU або пари LCU, що могло б бути тільки 64 або 128 пікселів, наприклад. Так, лінійні буфери для піксельних значень для сусідніх блоків зверху і знизу можуть потребувати того, щоб бути значно більше, ніж лінійні буфери, що використовуються для піксельних значень сусідніх блоків зліва і праворуч. Вибір пікселів, показаний на фіг. 6С, може

дозволити уникнути використання лінійних буферів для піксельних значень верхнього і нижнього сусіднього блока, таким чином знижуючи складність кодування.

Приклади з фіг. 6А - фіг. 6С - тільки представлені методики цього розкриття. Передбачається, що ці методики можуть бути розширені для блоків, відмінних від тільки  $4 \times 4$ , і що різні піднабори пікселів можуть бути вибрані.

При обчисленні метрики активності блока, замість оригінальних пікселів можуть використовуватися квантовані пікселі (тобто  $X(i, j) > N$ ), щоб знизити складність операцій, таких як операції додавання. Додатково обчислення можуть бути на основі абсолютної різниці замість основаних на значенні Лапласіана. Наприклад, при обчисленні  $Hor\_act(i, j)$  або  $Ver\_act(i, j)$  абсолютні різниці можуть використовуватися замість значень Лапласіана таким чином:

Метрика Напрямку

$Ver\_act(i, j) = \text{abs}(X(i, j) - X(i, j-1))$

$Hor\_act(i, j) = \text{abs}(X(i, j) - X(i-1, j))$

$H_B = \sum_{i=0,1,2} \sum_{j=0,1,2} H(i, j)$

$V_B = \sum_{i=0,1,2} \sum_{j=0,1,2} V(i, j)$

Напрямок = 0, 1 ( $H > 2V$ ), 2 ( $V > 2H$ )

Метрика Активності

$L_B = H_B + V_B$

5 класів (0, 1, 2, 3, 4)

Метрика

Активність + Напрямок (наприклад, 15 або 16 комбінацій, як було пояснено вище в прикладі з Фіг. 4В)

Це розкриття описало методики підвибірки з посиланням на обмежену групу конкретних метрик. Припускається, що, однак, ці методики підвибірки загалом застосовні до інших метрик, таких як інші метрики, розглянуті в цьому розкритті, які можуть використовуватися з метою визначення фільтра. Додатково, хоча методики підвибірки цього розкриття були описані з посиланням на блоки пікселів  $4 \times 4$ , методики також можуть бути застосовні до блоків інших розмірів.

Фіг. 7 являє собою блок-схему послідовності операцій, що ілюструє методику кодування відео, сумісну з даним розкриттям. Методики, описані на фіг. 7, можуть реалізовуватися елементом фільтра відеокодера або відеодекодера, як то модуль 349 фільтра відеокодера 350 або модуль 559 фільтра відеодекодера 560. Елемент фільтра визначає першу метрику для групи пікселів у межах блока пікселів (710). Перша метрика може, наприклад, бути метрикою активності, такою як значення Лапласіана із зміненою сумою, або перша метрика може бути метрикою напрямку. Перша метрика може визначатися, наприклад, на основі порівняння набору пікселів у блоці або на основі піднабору пікселів у блоці з іншими пікселями в блоці. Елемент фільтра додатково визначає другу метрику для блока (720). Друга метрика може, наприклад, бути метрикою напрямку, яка визначається на основі порівняння вимірювання горизонтальної активності з вимірюванням вертикальної активності. На основі першої метрики і другої метрики модуль фільтра визначає фільтр (730). Модуль фільтра генерує відфільтроване зображення шляхом застосування фільтра до блока (740). Як розглянуто вище, в деяких варіантах здійснення блок може бути блоком пікселів  $2 \times 2$ ,  $4 \times 4$  або  $M \times N$ , що використовується для визначення першої метрики або другої метрики. У деяких варіантах здійснення перша метрика може бути метрикою горизонтальної активності, тоді як друга метрика - метрика вертикальної активності, або перша метрика може бути метрикою краю, тоді як друга метрика - метрика напрямку.

Фіг. 8А являє собою блок-схему послідовності операцій, що ілюструють методики кодування відео відповідно до даного розкриття. Методики, описані на фіг. 8А, можуть реалізовуватися модулем фільтра відеодекодера, як то модуль 559 фільтра відеодекодера 560. Модуль 559 фільтра приводить у відповідність першу комбінацію діапазону до першого фільтра (810А). Перша комбінація діапазону - комбінація першого діапазону значень для першої метрики і першого діапазону значень для другої метрики. Перша метрика може, наприклад, бути значенням Лапласіана із зміненою сумою, а друга метрика може бути метрикою напрямку, хоча інші метрики також можуть використовуватися. Модуль 559 фільтра приводить у відповідність другу комбінацію діапазону до другого фільтра (820А). Друга комбінація діапазону - комбінація другого діапазону значень для першої метрики і другого діапазону значень для другої метрики. Модуль 559 фільтра потім встановлює відповідність поточної комбінації діапазонів фільтру на основі прийнятого кодового слова. Поточна комбінація діапазонів включає в себе перший діапазон значень першої метрики і другий діапазон значень другої метрики. Якщо кодове слово - це перше кодове слово (830А, так), модуль 559 фільтра встановлює відповідність поточної

комбінації діапазонів першому фільтру (840A). Перше кодове слово вказує, що поточна комбінація діапазонів приведена у відповідність до того самого фільтра, що і перша комбінація діапазонів. Якщо кодове слово - це друге кодове слово (830A, ні), модуль 559 фільтра встановлює відповідність поточної комбінації діапазонів другому фільтру (860A). Друге кодове слово вказує, що поточна комбінація діапазонів приведена у відповідність до того самого фільтра, що і друга комбінація діапазонів. Якщо кодове слово - це ні перше кодове слово, ні друге кодове слово (850A, ні), то модуль 559 фільтра приводить поточну комбінацію діапазонів у відповідність до третього фільтра (870A). Якщо у відповідь на прийом третього кодового слова, при цьому третє кодове слово ідентифікує той третій фільтр. У прикладі з фіг. 8A перше кодове слово і друге кодове слово - кожне з них - можуть включати в себе менше бітів, ніж третє кодове слово.

Фіг. 8B являє собою блок-схему послідовності операцій, що ілюструють методики кодування відео відповідно до даного розкриття. Методики, описані на фіг. 8B, можуть реалізовуватися модулем фільтра відеодекодера, як то модуль 559 фільтра відеодекодера 560. Модуль 559 фільтра генерує відповідність комбінації діапазонів фільтрам (810B). Кожна комбінація діапазонів, наприклад, може включати в себе діапазон для першої метрики і діапазон для другої метрики. У відповідь на прийом першого кодового слова, яке сигналізує, що поточна комбінація діапазонів приводиться у відповідність до того самого фільтра, що і попередня комбінація діапазонів (820B, так), модуль 559 фільтра встановлює відповідність поточної комбінації діапазонів тому самому фільтру, що і для попередньої комбінації діапазонів (830B). У відповідь на прийом другого кодового слова, яке повідомляє, що поточна комбінація діапазонів приводиться у відповідність до іншого фільтра, ніж попередня комбінація діапазонів (820B, ні), модуль 559 фільтра встановлює відповідність поточної комбінації діапазонів новому фільтру (840B). Як описано вище, поточна комбінація діапазонів може визначатися на основі відомого порядку передачі. У деяких прикладах новий фільтр може бути ідентифікований на основі другого кодового слова, тоді як в інших прикладах новий фільтр може визначатися на основі порядку, в якому сигналізуються коефіцієнти фільтра.

Фіг. 9A являє собою блок-схему послідовності операцій, що ілюструють методики кодування відео відповідно до даного розкриття. Методики, описані на фіг. 9A, можуть реалізовуватися блоком фільтра відеодекодера, як то модуль 349 фільтра відеодекодера 350. Модуль 349 фільтра визначає відповідність комбінації діапазонів фільтрам (910A). Кожна комбінація діапазонів включає в себе діапазон значень для першої метрики і діапазон значень для другої метрики. Для поточної комбінації діапазонів, якщо поточна комбінація діапазонів приведена у відповідність до того самого фільтра, що і попередня комбінація діапазонів, яка містить той самий діапазон значень для першої метрики (920A, так), то модуль 349 фільтра генерує перше кодове слово (930A). Якщо поточна комбінація діапазонів приведена у відповідність до того самого фільтра, що і попередня комбінація діапазонів, яка містить той самий діапазон значень для другої метрики (940A, так), то модуль 349 фільтра генерує друге кодове слово (950A). Якщо поточна комбінація діапазонів не приведена у відповідність ні до попередньої комбінації діапазонів, яка містить той самий діапазон значень для першої метрики, ні до попередньої комбінації діапазонів, яка містить той самий діапазон значень для другої метрики (950A, ні), то модуль 349 фільтра генерує третє кодове слово (960A). Третє кодове слово може ідентифікувати фільтр, приведений у відповідність до поточної комбінації діапазонів.

Фіг. 9B являє собою блок-схему послідовності операцій, що ілюструють методики кодування відео відповідно до даного розкриття. Методики, описані на фіг. 9B, можуть реалізовуватися модулем фільтра відеодекодера, як то модуль 349 фільтра відеодекодера 350. Модуль 349 фільтра визначає відповідність комбінації діапазонів фільтрам (910B). Кожна комбінація діапазонів може, наприклад, включати в себе діапазон для першої метрики і діапазон для другої метрики. Коли поточна комбінація діапазонів, що піддається кодуванню, має той самий фільтр, що і раніше кодована комбінація діапазонів (920B, так), модуль 349 фільтра може генерувати перше кодове слово, щоб сигналізувати, що поточна комбінація діапазонів приведена у відповідність до того самого фільтра, що і попередня комбінація діапазонів (930B). Коли поточна комбінація діапазонів, що піддається кодуванню, не має того самого фільтра, що і раніше кодована комбінація діапазонів (920B, ні), модуль 349 фільтра може генерувати друге кодове слово (940B). Друге кодове слово може ідентифікувати фільтр, приведений у відповідність до поточної комбінації діапазонів. Як описано вище, поточна комбінація діапазонів може визначатися на основі відомого порядку передачі. У прикладі з фіг. 9B перше кодове слово може включати в себе менше бітів, ніж друге кодове слово.

У прикладах з Фіг. 8А та фіг. 8В і Фіг. 9А та фіг. 9В терміни "перше кодове слово", "друге кодове слово" і "третє кодове слово" використовуються, щоб визначити різницю між різними кодовими словами і не мається на увазі послідовне впорядкування кодових слів.

Фіг. 10 являє собою блок-схему послідовності операцій, що ілюструє методики кодування відео відповідно до даного розкриття. Методики, описані на фіг. 10, можуть реалізовуватися модулем фільтра відеокодера, як то модуль 349 фільтра відеокодера 350, або модулем фільтра відеодекодера, як то модуль 559 фільтра. Модуль фільтра визначає відповідність комбінацій діапазонів фільтрам (1010). Комбінації діапазонів включають в себе діапазон для першої метрики і діапазон для другої метрики. Модуль фільтра визначає унікальну ідентифікацію комбінації діапазонів (ID) для кожної комбінації діапазонів (1020). Унікальні ID комбінації діапазонів відповідають порядковим значенням. Модуль фільтра призначає перший унікальний ID групи першій групі комбінацій діапазонів на основі порядкового значення ID комбінації діапазонів щонайменше однієї комбінації діапазонів у першій групі комбінацій діапазонів (1030). Групи комбінацій діапазонів включають в себе комбінації діапазонів, приведені у відповідність до одного і того самого фільтра, унікальні ID групи відповідають набору порядкових значень. Модуль фільтра кодує перший набір коефіцієнтів фільтра, що відповідає одному і тому самому фільтру, на основі порядкового значення першого унікального ID фільтра (1040). У випадку відеокодера кодування першого набору коефіцієнтів фільтра може включати в себе, наприклад, сигналізування коефіцієнтів фільтра в кодованому бітовому потоці з використанням методик різницевого кодування. У випадку відеодекодера кодування першого набору коефіцієнтів фільтра може включати в себе реконструкцію коефіцієнтів фільтра на основі інформації, прийнятої в кодованому бітовому потоці.

Фіг. 11 являє собою блок-схему послідовності операцій, що ілюструє методики кодування відео відповідно до даного розкриття. Методики, описані на фіг. 11, можуть реалізовуватися блоком фільтра відеокодера, як то модуль 349 фільтра відеокодера 350, або модулем фільтра відеодекодера, як то модуль 559 фільтра. Модуль фільтра визначає відповідність комбінацій діапазонів фільтрам (1110). Комбінації діапазонів включають в себе діапазон для першої метрики і діапазон для другої метрики. Кожна комбінація діапазонів може мати унікальну ідентифікацію комбінації діапазонів (ID), і кожний унікальний ID комбінації діапазонів може відповідати порядковому значенню для цієї комбінації діапазонів. Модуль фільтра може призначати унікальний ID групи кожній групі комбінацій діапазонів (1120). Модуль фільтра може призначати унікальні ID групи, наприклад, на основі порядкових значень комбінацій діапазонів. Група комбінацій діапазонів може включати в себе комбінації діапазонів, приведені у відповідність до загального фільтра, і унікальні ID групи можуть відповідати набору порядкових значень. Модуль фільтра може кодувати набори коефіцієнтів фільтра для фільтрів на основі унікальних ID групи (1140).

У прикладі з фіг. 11 модуль фільтра може призначати унікальні ID групи, призначаючи, наприклад, унікальний ID групи, що відповідає найнижчому порядковому значенню з унікальних ID групи, групі комбінацій діапазонів, яка містить комбінацію діапазонів з ID комбінації діапазонів, що відповідає найнижчому порядковому значенню з ID комбінації діапазонів. В іншому прикладі модуль фільтра може призначати унікальний ID групи, що відповідає найвищому порядковому значенню з унікальних ID групи, групі комбінацій діапазонів, яка містить комбінацію діапазонів з ID комбінації діапазонів, що відповідає найвищому порядковому значенню з ID комбінації діапазонів.

У випадках, коли модуль фільтра є частиною відеодекодера, модуль фільтра може кодувати набори коефіцієнтів фільтра шляхом генерування наборів коефіцієнтів фільтра на основі інформації, прийнятої в кодованому бітовому потоці. Модуль фільтра може, наприклад, генерувати набори коефіцієнтів фільтра, використовуючи методики різницевого кодування. У випадках, коли модуль фільтра є частиною відеокодера, модуль фільтра може кодувати набори коефіцієнтів фільтра шляхом сигналізування наборів коефіцієнтів фільтра в кодованому бітовому потоці в порядку, вибраному на основі порядкових значень унікальних ID групи. Модуль фільтра може, наприклад, сигналізувати набори коефіцієнтів фільтра, використовуючи методики різницевого кодування.

Вищевикладене розкриття було спрощене до деякої міри з метою передати деталі. Наприклад, розкриття загалом описує набори фільтрів, які сигналізуються на покадровій основі або на основі "вирізка за вирізкою", але набори фільтрів також можуть сигналізуватися на основі "послідовність за послідовністю", на основі "група зображень за групою зображень", на основі "група вирізок за групою вирізок", на основі "CU за CU", на основі "LCU за LCU" або іншій подібній основі. Загалом, фільтри можуть сигналізуватися для будь-якого групування одного або більше CU. Додатково, у варіанті здійснення, може бути множина фільтрів на кожне

введення на кожний CU, множина коефіцієнтів на кожний фільтр і множина різних рівнів дисперсії з кожним з фільтрів, що визначаються для різного діапазону дисперсії. Наприклад, в деяких випадках може бути шістнадцять або більше фільтрів, визначених для кожного введення CU, і шістнадцять різних діапазонів дисперсії, що відповідають кожному фільтру. Додатково, коли це розкриття описує передачу інформації фільтра, не повинне допускатися, що вся інформація фільтра передається на одному і тому самому рівні кодування. Наприклад, в деяких варіантах здійснення деяка інформація фільтра, така як синтаксис опису фільтра, може сигналізуватися на покадровій основі або на основі вирізка за вирізкою, тоді як інша інформація фільтра, така як коефіцієнти фільтра, сигналізується на основі LCU-за-LCU. Синтаксис та інші рівні ієрархії кодування, як то рівень послідовності, GOP-рівень або інші рівні, можуть також визначатися для перенесення деякої або всієї подібної інформації фільтра.

Кожний з фільтрів для кожного введення може включати в себе багато коефіцієнтів. В одному прикладі фільтри містять двомірні фільтри з 81 різними коефіцієнтами, визначеними для підтримки фільтра, який охоплює два вимірювання. Однак кількість коефіцієнтів фільтра, які сигналізуються для кожного фільтра, може бути менше 81 в деяких випадках. Симетрія коефіцієнтів, наприклад, може встановлюватися так, що коефіцієнти фільтра в одному вимірюванні або квадранті можуть відповідати оберненим або симетричним значенням відносно коефіцієнтів в інших вимірюваннях або квадрантах. Симетрія коефіцієнтів може допускати вираження 81 різних коефіцієнтів меншою кількістю коефіцієнтів, в цьому випадку кодер і декодер можуть допускати, що обернені або дзеркальні значення коефіцієнтів визначають інші коефіцієнти. Наприклад, коефіцієнти (5, -2, 10, 10, -2, 5) можуть кодуватися і сигналізуватися як піднабір коефіцієнтів (5, -2, 10). У цьому випадку декодер може знати, що ці три коефіцієнти визначають великий симетричний набір коефіцієнтів (5, -2, 10, 10, -2, 5).

Методики цього розкриття можуть реалізовуватися в широкому різноманітті пристроїв та апаратів, включаючи бездротову радіотелефону гарнітуру та інтегральну схему (IC) або набір IC (тобто набір чіпів). Будь-які компоненти, модулі або елементи були описані з метою виділити функціональні аспекти і не вимагають обов'язкової реалізації різними елементами апаратного забезпечення.

Відповідно, методики, описані тут, можуть реалізовуватися в апаратному забезпеченні, програмному забезпеченні, програмно-апаратному забезпеченні або будь-якій їх комбінації. При застосуванні в апаратному забезпеченні будь-які ознаки, описані як модулі, елементи або компоненти, можуть бути виконані разом в інтегральному логічному пристрої або окремо як окремі логічні пристрої, що мають можливість взаємодіяти. При застосуванні в програмному забезпеченні методики можуть бути реалізовані щонайменше частково машинозчитуваним носієм, що містить команди, які при виконанні в процесорі виконують один або більше способів, описаних вище. Машинозчитуваний носій може містити машинозчитуваний носій даних і може складати частину комп'ютерного програмного продукту, який може включати в себе пакувальні матеріали. Машинозчитуваний носій даних може містити оперативний запам'ятовуючий пристрій (RAM), такий як синхронний динамічний оперативний запам'ятовуючий пристрій (SDRAM), постійний запам'ятовуючий пристрій (ROM), енергонезалежний оперативний запам'ятовуючий пристрій (NVRAM), електрично стираєний програмований постійний запам'ятовуючий пристрій (EEPROM), FLASH-пам'ять, магнітні та оптичні носії даних тощо. Методики додатково або як альтернатива можуть бути реалізовані щонайменше, частково, машинозчитуваним засобом зв'язку, який переносить або повідомляє код у вигляді команд або структур даних і доступ до якого, зчитування і/або виконання якого може здійснюватися з комп'ютера.

Код може виконуватися одним або більше процесорами, такими як один або більше цифрових процесорів сигналів (DSP), універсальних мікропроцесорів, спеціалізованих інтегральних схем (ASIC), програмованих користувачем вентиляльних матриць (FPGA) або інших еквівалентів інтегральних або дискретних логічних схем. Відповідно, термін "процесор", як використовується тут, може відноситися до будь-якої з вищезазначених структур або будь-якої іншої структури, прийнятної для реалізації методик, описаних тут. Крім того, в деяких аспектах функціональність, описана тут, може бути передбачена в спеціалізованих модулях програмного забезпечення або модулях апаратного забезпечення, сконфігурованих для кодування і декодування, або вбудована в комбінований відеокодек. Також методики могли б в повній мірі бути реалізовані в одній або більше схемах або логічних елементах.

Різні аспекти розкриття були описані. Ці та інші аспекти знаходяться в межах обсягу наступної формули винаходу.

Посилальні позиції

110 система кодування і декодування відео

	112 пристрій-джерело
	115 канал зв'язку
	116 пристрій-адресат
	120 відеоджерело
5	122 відеокодер
	123 модулятор/демодулятор (модем)
	124 передавач
	126 приймач
	127 модем
10	128 відеодекодер
	130 пристрій відображення
	132 пристрій зберігання даних
	250 квадродерево
	252 кореневий вузол
15	254 вузол
	256 вузли-потомки
	258 кінцеві вершини
	272 LCU
	274, 276, 278 суб-CU
20	332 елемент прогнозування
	334 запам'ятовуючий пристрій
	338 елемент перетворення
	340 елемент квантування
	342 елемент оберненого квантування
25	344 елемент оберненого перетворення
	346 елемент ентропійного кодування
	347 фільтр видалення блочності
	348, 351 суматор
	349 елемент адаптивного фільтра
30	350 відеокодер
	351 суматор
	353 елемент вибору фільтра
	401-435 комбінації діапазонів
35	552 модуль ентропійного декодування
	554 модуль прогнозування
	556 модуль оберненого квантування
	557 фільтр видалення блочності
	558 модуль оберненого перетворення
	559 модуль фільтра
40	560 відеодекодер
	562 пам'ять
	564 суматор

#### ФОРМУЛА ВИНАХОДУ

45	1. Спосіб кодування/декодування відео, що включає:
	визначення напрямку щонайменше одного пікселя з блока пікселів,
	визначення метрики для згаданого щонайменше одного пікселя зі згаданого блока пікселів, при
50	цьому метрика визначається на основі порівняння піксельних значень для піднабору пікселів в
	цьому блоці зі згаданим щонайменше одним пікселем, при цьому блок пікселів являє собою
	блок пікселів $M \times N$ , і при цьому щонайменше одне з $M$ і $N$ становить 2 або більше;
	визначення, на основі згаданої метрики і згаданого напрямку, фільтра для згаданого
	щонайменше одного пікселя зі згаданого блока пікселів; і
55	генерування відфільтрованого зображення шляхом застосування фільтра до згаданого
	щонайменше одного пікселя зі згаданого блока пікселів.
	2. Спосіб за п. 1, в якому напрямок містить одне з горизонтального напрямку, вертикального
	напрямку, напрямку 45 градусів і напрямку 135 градусів.
	3. Спосіб за п. 1, в якому визначення напрямку включає порівняння вимірювання горизонтальної
60	активності з вимірюванням вертикальної активності, і причому горизонтальна активність
	визначається на основі порівняння піксельного значення щонайменше одного пікселя у

згаданому піднаборі з піксельним значенням лівого сусіднього пікселя і піксельним значенням правого сусіднього пікселя, і при цьому лівий сусідній піксель і правий сусідній піксель, обидва, знаходяться у згаданому блоці пікселів.

4. Спосіб за п. 3, в якому вертикальна активність визначається на основі порівняння піксельного значення щонайменше одного пікселя у згаданому піднаборі з піксельним значенням верхнього сусіднього пікселя і піксельним значенням нижнього сусіднього пікселя, при цьому верхній сусідній піксель і нижній сусідній піксель, обидва, знаходяться у згаданому блоці пікселів.

5. Спосіб за п. 1, в якому піднабір пікселів містить пікселі не на межі блока  $M \times N$ .

6. Спосіб за п. 1, в якому блок пікселів являє собою блок пікселів  $4 \times 4$ , що включає в себе шістнадцять пікселів, і при цьому піднабір пікселів включає в себе чотири пікселі.

7. Спосіб за п. 1, в якому визначення метрики включає:

порівняння згаданого щонайменше одного пікселя з першим сусіднім пікселем зі згаданого піднабору пікселів,

у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя більше піксельного значення першого сусіднього пікселя, збільшення змінної метрики краю; і

у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя менше піксельного значення першого сусіднього пікселя, зменшення змінної метрики краю.

8. Спосіб за п. 7, в якому визначення метрики включає:

порівняння щонайменше одного пікселя з другим сусіднім пікселем зі згаданого піднабору пікселів,

у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя більше піксельного значення другого сусіднього пікселя, збільшення змінної метрики краю; і

у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя менше піксельного значення другого сусіднього пікселя, зменшення змінної метрики краю.

9. Спосіб за п. 8, в якому перший сусідній піксель і другий сусідній піксель містять одне з:

лівого сусіднього пікселя і правого сусіднього пікселя,

верхнього сусіднього пікселя і нижнього сусіднього пікселя,

нижнього лівого сусіднього пікселя і верхнього правого сусіднього пікселя, і

верхнього лівого сусіднього пікселя і нижнього правого сусіднього пікселя.

10. Спосіб за п. 1, в якому спосіб виконується пристроєм кодування/декодування відео, що містить відеокoder і камеру, яка сконфігурована захоплювати відеодані.

11. Спосіб за п. 1, в якому спосіб виконується пристроєм кодування/декодування відео, що містить відеодекодер і дисплей, який сконфігурований відображати декодовані відеодані.

12. Пристрій кодування/декодування відео, що містить:

модуль фільтра, виконаний з можливістю:

визначати напрямок щонайменше одного пікселя з блока пікселів,

визначати метрику для згаданого щонайменше одного пікселя зі згаданого блока пікселів, при цьому метрика визначається на основі порівняння піксельних значень піднабору пікселів у згаданому блоці зі згаданим щонайменше одним пікселем, і при цьому блок пікселів являє

собою блок пікселів  $M \times N$ , і при цьому щонайменше одне з  $M$  і  $N$  становить 2 або більше;

визначати фільтр для згаданого щонайменше одного пікселя зі згаданого блока пікселів на основі згаданої метрики і згаданого напрямку; і

генерувати відфільтроване зображення шляхом застосування фільтра до згаданого щонайменше одного пікселя згаданого блока пікселів;

запам'ятовуючий пристрій, виконаний з можливістю зберігати відфільтрований результат модуля фільтра.

13. Пристрій за п. 12, в якому напрямок містить одне з горизонтального напрямку, вертикального напрямку, напрямку 45 градусів або напрямку 135 градусів.

14. Пристрій за п. 12, в якому модуль фільтра визначає напрямок за допомогою порівняння вимірювання горизонтальної активності з вимірюванням вертикальної активності, і причому

горизонтальна активність визначається на основі порівняння піксельного значення щонайменше одного пікселя у згаданому піднаборі з піксельним значенням лівого сусіднього пікселя і

піксельним значенням правого сусіднього пікселя, при цьому лівий сусідній піксель і правий сусідній піксель, обидва, знаходяться в блоці пікселів.

15. Пристрій за п. 14, в якому вертикальна активність визначається на основі порівняння піксельного значення щонайменше одного пікселя у згаданому піднаборі з піксельним значенням

верхнього сусіднього пікселя і піксельним значенням нижнього сусіднього пікселя, при цьому верхній сусідній піксель і нижній сусідній піксель, обидва, знаходяться у згаданому блоці пікселів.

16. Пристрій за п. 12, в якому піднабір пікселів містить пікселі не на межі блока  $M \times N$ .



17. Пристрій за п. 12, в якому блок пікселів являє собою блок пікселів  $4 \times 4$ , що включає в себе шістнадцять пікселів, і при цьому піднабір пікселів включає в себе чотири пікселі.

18. Пристрій за п. 12, в якому модуль фільтра додатково виконаний з можливістю визначати метрику за допомогою:

5 порівняння щонайменше одного пікселя з першим сусіднім пікселем зі згаданого піднабору пікселів,

у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя більше піксельного значення першого сусіднього пікселя, збільшення змінної метрики краю; і

10 у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя менше піксельного значення першого сусіднього пікселя, зменшення змінної метрики краю.

19. Пристрій за п. 18, в якому модуль фільтра додатково виконаний з можливістю визначати метрику за допомогою:

порівняння щонайменше одного пікселя з другим сусіднім пікселем з піднабору пікселів,

15 у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя більше піксельного значення другого сусіднього пікселя, збільшення змінної метрики краю; і

у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя менше піксельного значення другого сусіднього пікселя, зменшення змінної метрики краю.

20. Пристрій за п. 12, в якому:

перший сусідній піксель і другий сусідній піксель містять одне з:

20 лівого сусіднього пікселя і правого сусіднього пікселя,

верхнього сусіднього пікселя і нижнього сусіднього пікселя,

нижнього лівого сусіднього пікселя і верхнього правого сусіднього пікселя, і

верхнього лівого сусіднього пікселя і нижнього правого сусіднього пікселя.

21. Пристрій за п. 12, в якому пристрій кодування/декодування відео містить відеокодер і камеру, яка сконфігурована захоплювати відеодані.

22. Пристрій за п. 12, в якому пристрій кодування/декодування відео містить відеокодер і дисплей, який сконфігурований відображати декодовані відеодані.

23. Апарат для кодування/декодування відео, що містить:

засіб для визначення напрямку щонайменше одного пікселя з блока пікселів,

30 засіб для визначення першої метрики для згаданого щонайменше одного пікселя зі згаданого блока пікселів, при цьому метрика визначається на основі порівняння піксельних значень для піднабору пікселів в цьому блоці зі згаданим щонайменше одним пікселем, і при цьому блок пікселів являє собою блок пікселів  $M \times N$ , і при цьому щонайменше одне з  $M$  і  $N$  становить 2 або більше;

35 засіб для визначення фільтра для згаданого щонайменше одного пікселя з блока пікселів на основі згаданої метрики і згаданого напрямку; і

засіб для генерування відфільтрованого зображення шляхом застосування фільтра до згаданого щонайменше одного пікселя зі згаданого блока пікселів.

40 24. Апарат за п. 23, в якому напрямок містить одне з горизонтального напрямку, вертикального напрямку, напрямку 45 градусів або напрямку 135 градусів.

25. Апарат за п. 23, в якому засіб для визначення метрики містить засіб для порівняння вимірювання горизонтальної активності з вимірюванням вертикальної активності, і причому засіб для порівняння вимірювання горизонтальної активності з вимірюванням вертикальної активності містить засіб для порівняння піксельного значення щонайменше одного пікселя в піднаборі з піксельним значенням лівого сусіднього пікселя і піксельним значенням правого сусіднього пікселя, при цьому лівий сусідній піксель і правий сусідній піксель, обидва, знаходяться в блоці пікселів.

26. Апарат за п. 25, в якому засіб для порівняння вимірювання горизонтальної активності з вимірюванням вертикальної активності додатково містить засіб для порівняння піксельного значення щонайменше одного пікселя в піднаборі з піксельним значенням верхнього сусіднього пікселя і піксельним значенням нижнього сусіднього пікселя, і при цьому верхній сусідній піксель і нижній сусідній піксель, обидва, знаходяться в блоці пікселів.

27. Апарат за п. 23, в якому піднабір пікселів містить пікселі не на межі блока  $M \times N$ .

55 28. Апарат за п. 23, в якому блок пікселів являє собою блок пікселів  $4 \times 4$ , що включає в себе шістнадцять пікселів, і при цьому піднабір пікселів включає в себе чотири пікселі.

29. Апарат за п. 23, в якому засіб для визначення метрики містить:

засіб для порівняння щонайменше одного пікселя з першим сусіднім пікселем зі згаданого піднабору пікселів,

60 засіб для збільшення змінної метрики краю у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя більше піксельного значення першого сусіднього пікселя; і

засіб для зменшення змінної метрики краю у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя менше піксельного значення першого сусіднього пікселя.

30. Апарат за п. 29, в якому засіб для визначення метрики додатково містить:

засіб для порівняння щонайменше одного пікселя з другим сусіднім пікселем зі згаданого піднабору пікселів,

засіб для збільшення змінної метрики краю у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя більше піксельного значення другого сусіднього пікселя; і

засіб для зменшення змінної метрики краю у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя менше піксельного значення другого сусіднього пікселя.

31. Апарат за п. 30, в якому перший сусідній піксель і другий сусідній піксель містять одне з: лівого сусіднього пікселя і правого сусіднього пікселя, верхнього сусіднього пікселя і нижнього сусіднього пікселя, нижнього лівого сусіднього пікселя і верхнього правого сусіднього пікселя, або

верхнього лівого сусіднього пікселя і нижнього правого сусіднього пікселя.

32. Апарат за п. 23, в якому апарат містить відеокодер.

33. Апарат за п. 23, в якому апарат містить відеодекодер.

34. Машиночитаний носій даних, в якому зберігаються команди, які при виконанні змушують один або більше процесорів:

визначати напрямок щонайменше одного пікселя з блока пікселів,

визначати метрику для згаданого блока пікселів, при цьому метрика визначається на основі порівняння піксельних значень для згаданого піднабору пікселів у згаданому блоці зі згаданим щонайменше одним пікселем, і при цьому блок пікселів являє собою блок пікселів  $M \times N$ , і при цьому щонайменше одне з  $M$  і  $N$  становить 2 або більше;

визначати фільтр для згаданого щонайменше одного пікселя зі згаданого блока пікселів на основі згаданої метрики і згаданого напрямку; і

генерувати відфільтроване зображення шляхом застосування фільтра до згаданого щонайменше одного пікселя зі згаданого блока пікселів.

35. Машиночитаний носій даних за п. 34, в якому напрямок містить одне з горизонтального напрямку, вертикального напрямку, напрямку 45 градусів і напрямку 135 градусів.

36. Машиночитаний носій даних за п. 34, в якому згадані один або більше процесорів визначають напрямок за допомогою порівняння вимірювання горизонтальної активності з вимірюванням вертикальної активності, і визначають горизонтальну активність на основі порівняння піксельного значення щонайменше одного пікселя у згаданому піднаборі з піксельним значенням лівого сусіднього пікселя і піксельним значенням правого сусіднього пікселя, при цьому лівий сусідній піксель і правий сусідній піксель, обидва, знаходяться у згаданому блоці пікселів.

37. Машиночитаний носій даних за п. 36, в якому згадані один або більше процесорів визначають вертикальну активність на основі порівняння піксельного значення щонайменше одного пікселя у згаданому піднаборі з піксельним значенням верхнього сусіднього пікселя і піксельним значенням нижнього сусіднього пікселя, і при цьому верхній сусідній піксель і нижній сусідній піксель, обидва, знаходяться у згаданому блоці пікселів.

38. Машиночитаний носій даних за п. 34, в якому піднабір пікселів містить пікселі не на межі блока  $M \times N$ .

39. Машиночитаний носій даних за п. 34, в якому блок пікселів являє собою блок пікселів  $4 \times 4$ , що включає в себе шістнадцять пікселів, і при цьому піднабір пікселів включає в себе чотири пікселі.

40. Машиночитаний носій даних за п. 34, в якому зберігаються додаткові команди, які при виконанні змушують згадані один або більше процесорів визначати метрику за допомогою:

порівняння щонайменше одного пікселя з першим сусіднім пікселем зі згаданого піднабору пікселів,

у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя більше піксельного значення першого сусіднього пікселя, збільшення змінної метрики краю; і

у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя менше піксельного значення першого сусіднього пікселя, зменшення змінної метрики краю.

41. Машиночитаний носій даних за п. 40, в якому зберігаються додаткові команди, які при виконанні змушують згадані один або більше процесорів визначати метрику за допомогою: порівняння щонайменше одного пікселя з другим сусіднім пікселем зі згаданого піднабору пікселів,

у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя більше піксельного значення другого сусіднього пікселя, збільшення змінної метрики краю; і

у відповідь на те, що піксельне значення згаданого щонайменше одного пікселя менше піксельного значення другого сусіднього пікселя, зменшення змінної метрики краю.

42. Машиночитаний носій даних за п. 41, в якому перший сусідній піксель і другий сусідній піксель містять одне з:

5 лівого сусіднього пікселя і правого сусіднього пікселя, верхнього сусіднього пікселя і нижнього сусіднього пікселя, нижнього лівого сусіднього пікселя і верхнього правого сусіднього пікселя, або

верхнього лівого сусіднього пікселя і нижнього правого сусіднього пікселя.

10 43. Машиночитаний носій даних за п. 34, в якому один або більше процесорів здійснюють кодування відео.

44. Машиночитаний носій даних за п. 34, в якому один або більше процесорів здійснюють декодування відео.

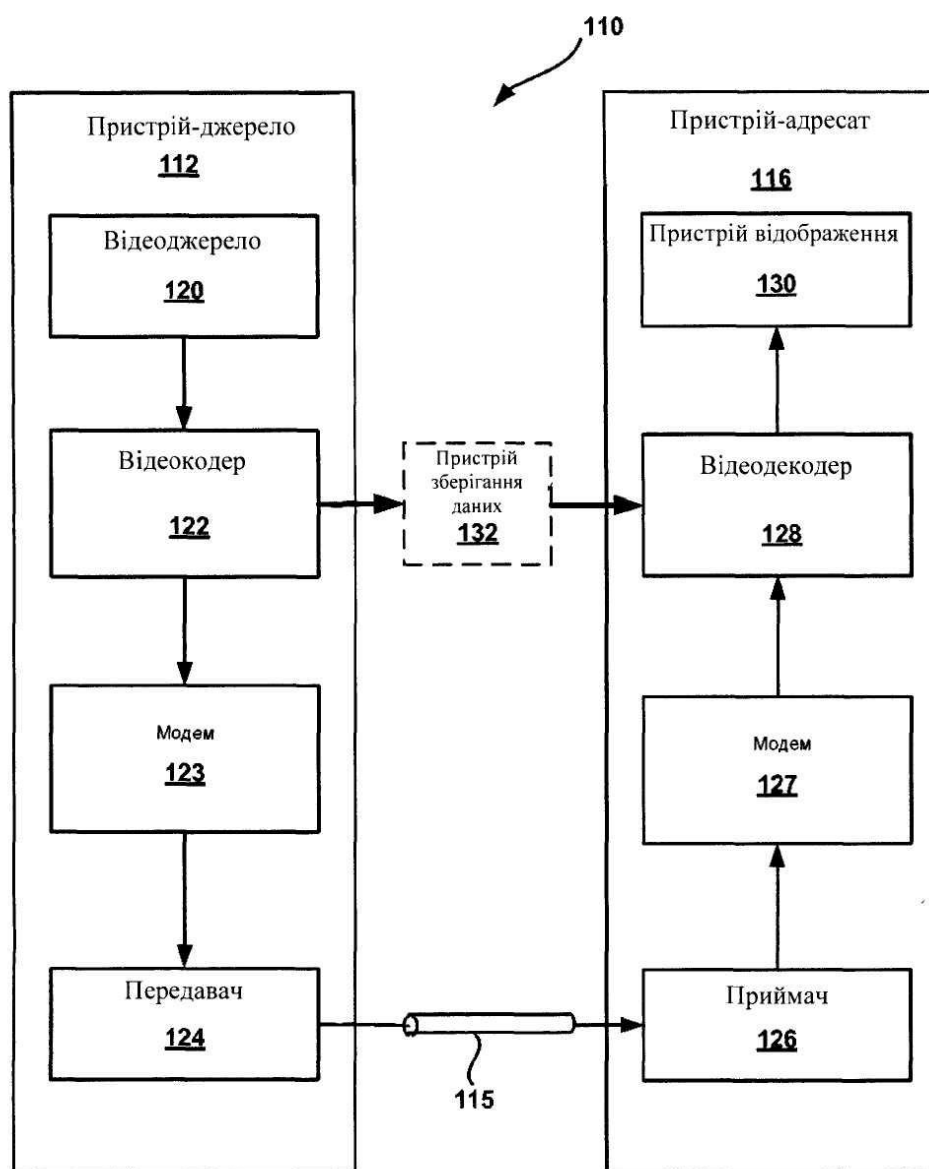
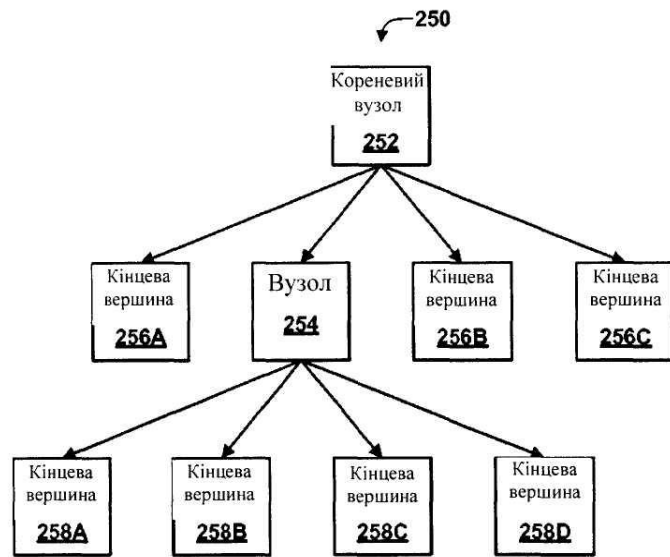
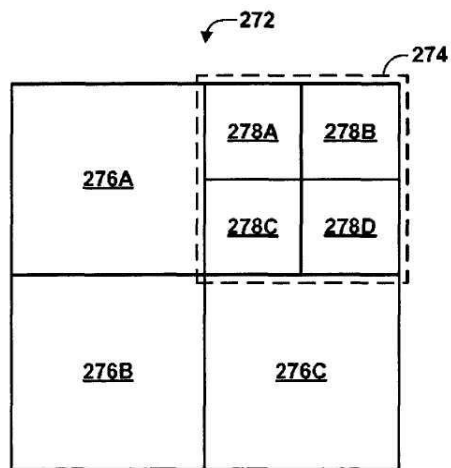


Fig. 1



Фіг. 2А



Фіг. 2В

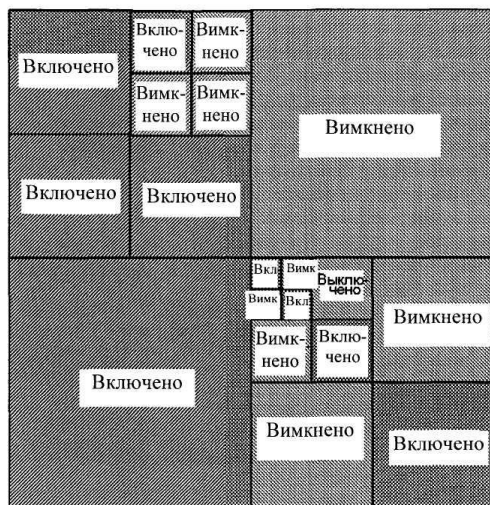
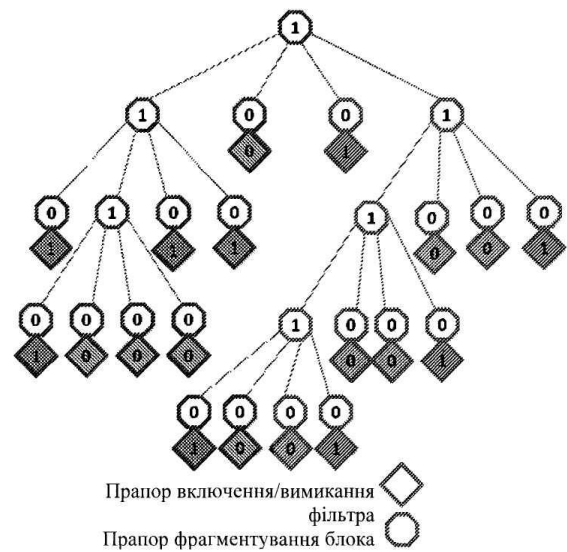
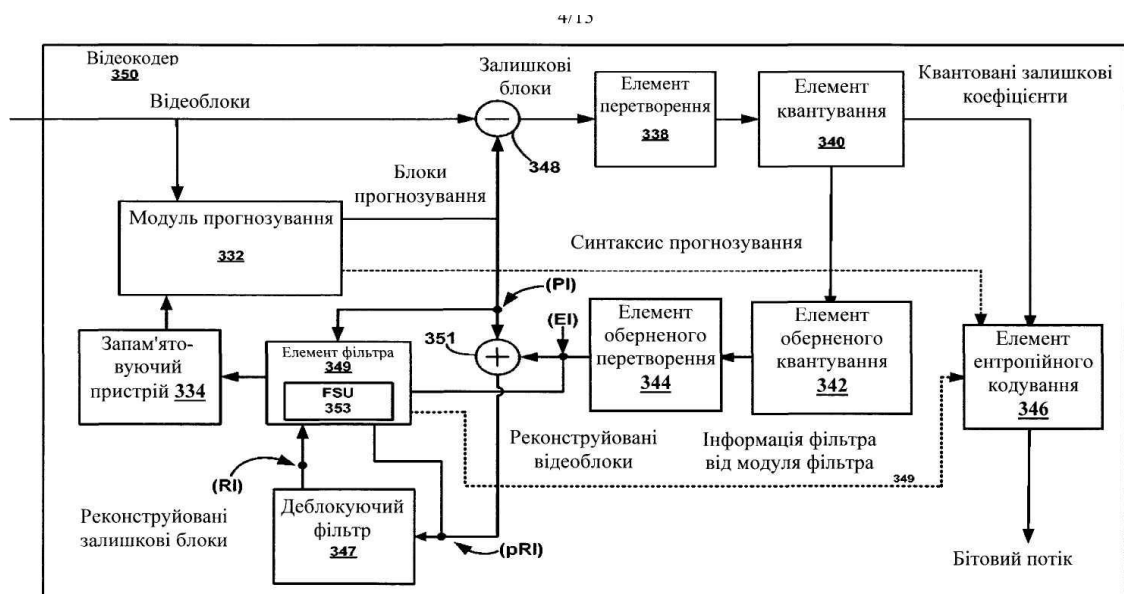


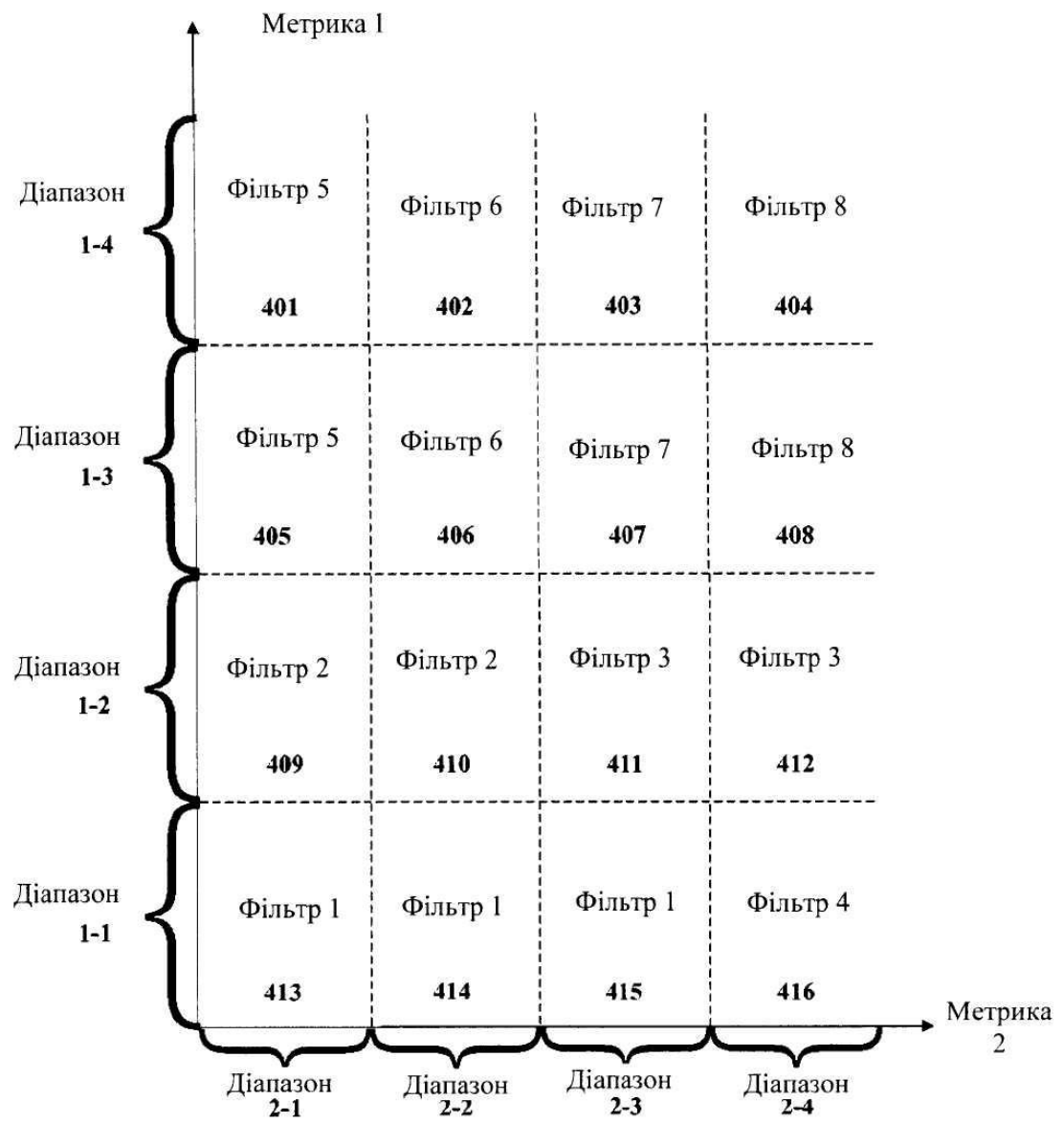
Fig. 2C



Фиг. 2D

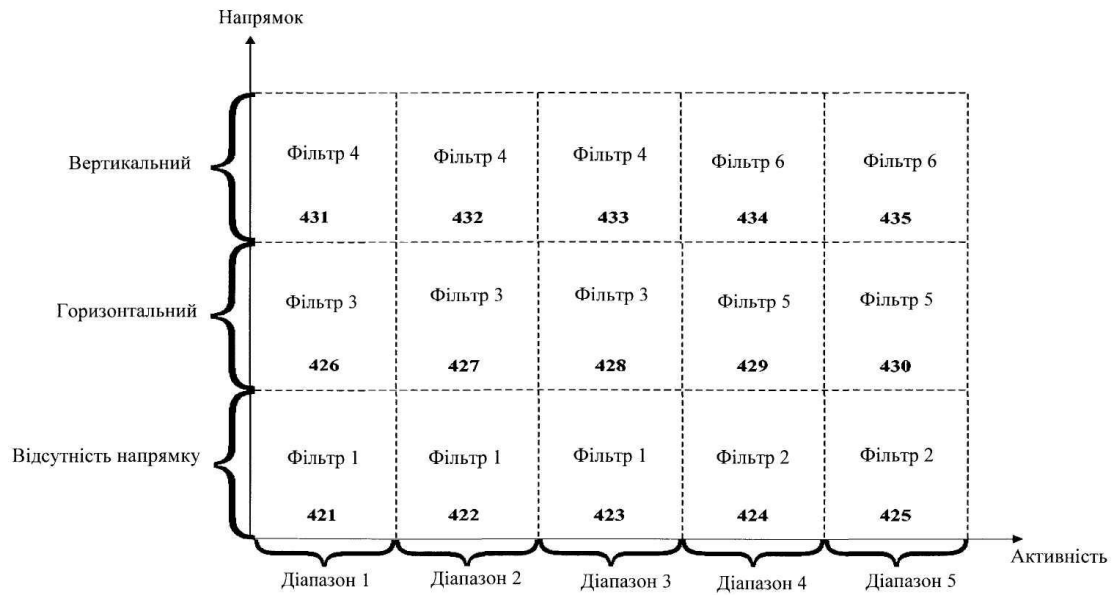


Фиг. 3

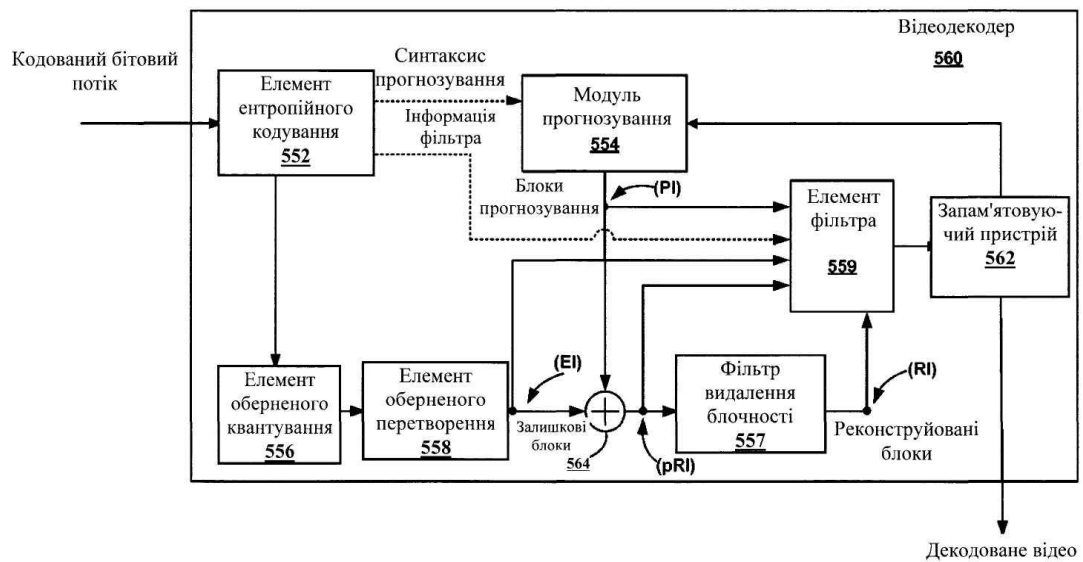


Фіг. 4А

6/15



Фіг. 4В



Фіг. 5

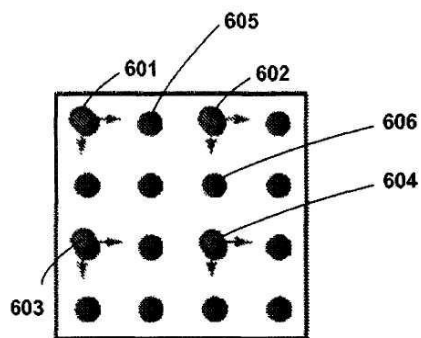


Fig. 6A

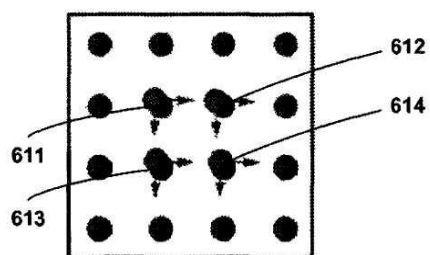


Fig. 6B

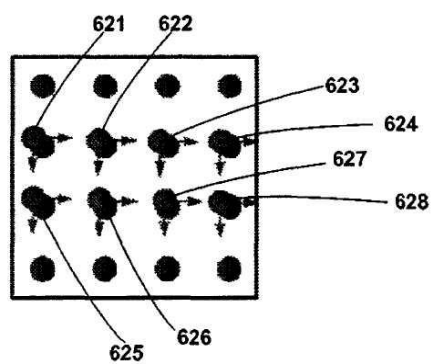
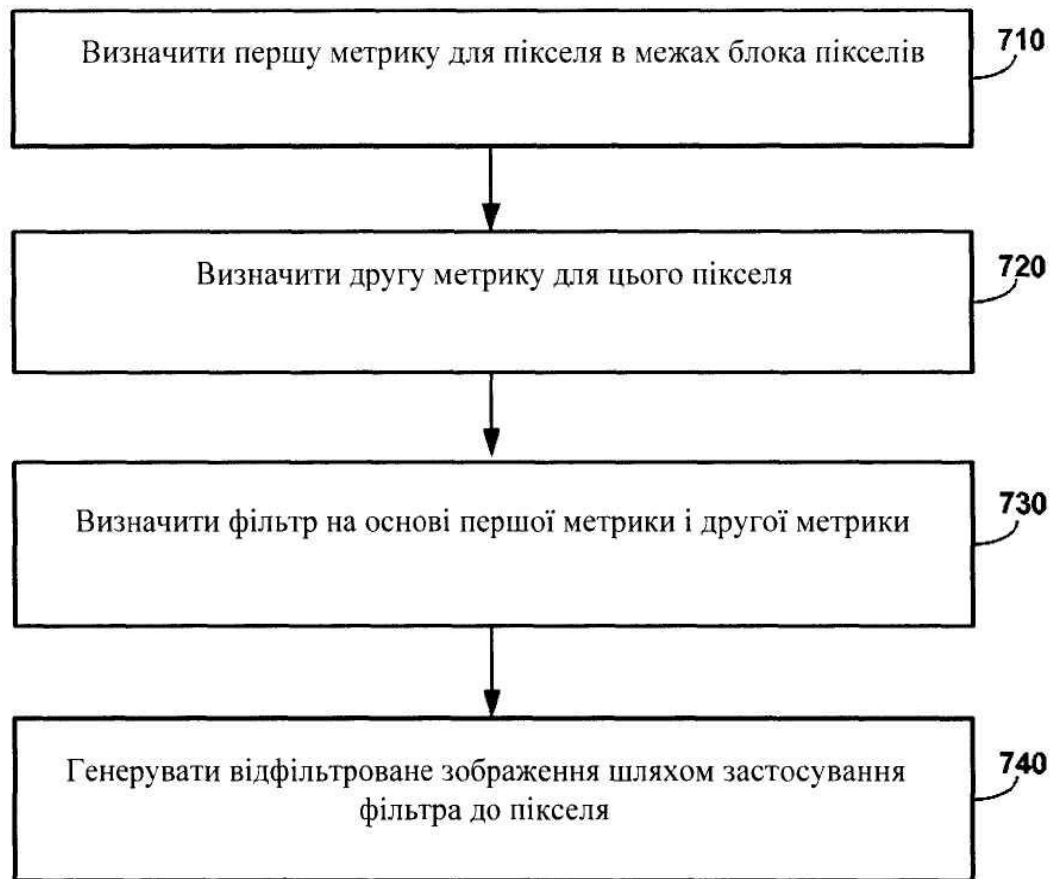
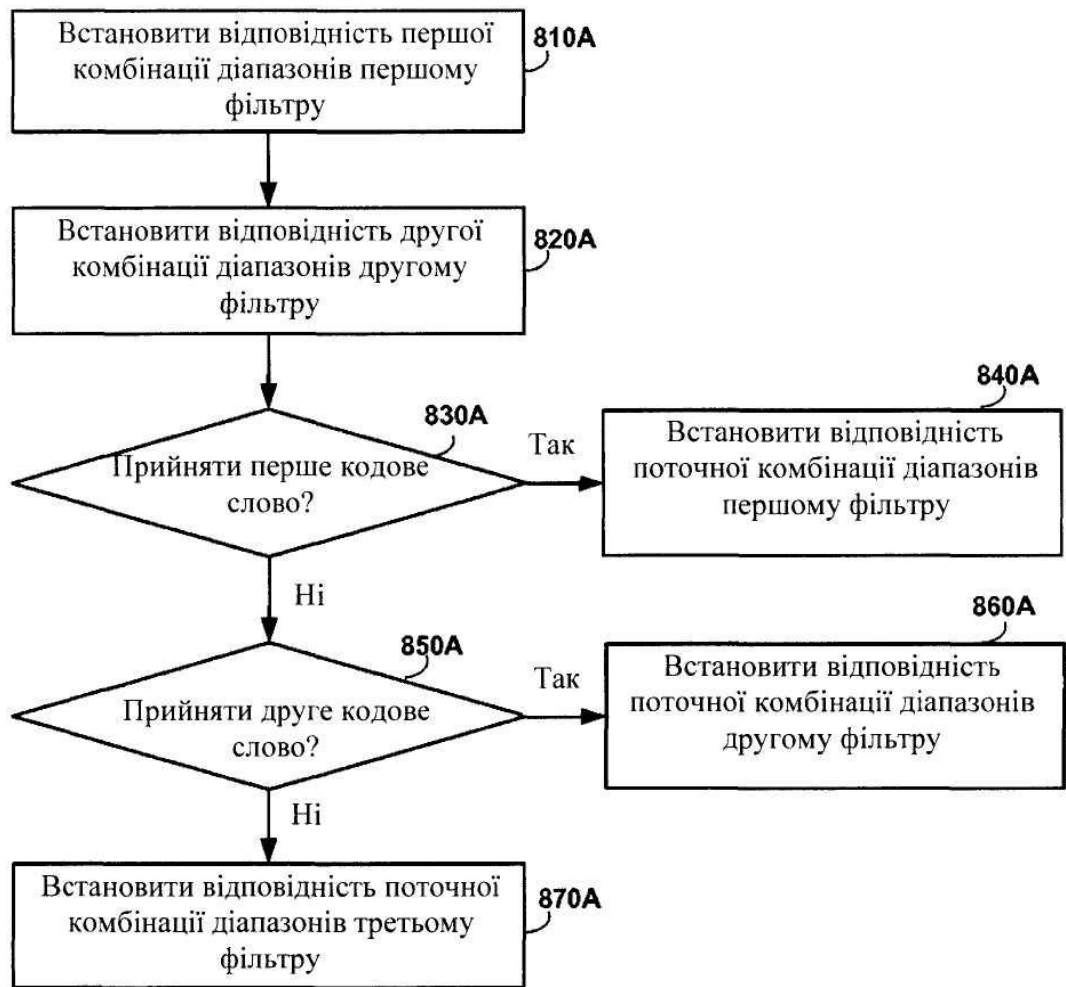


Fig. 6C

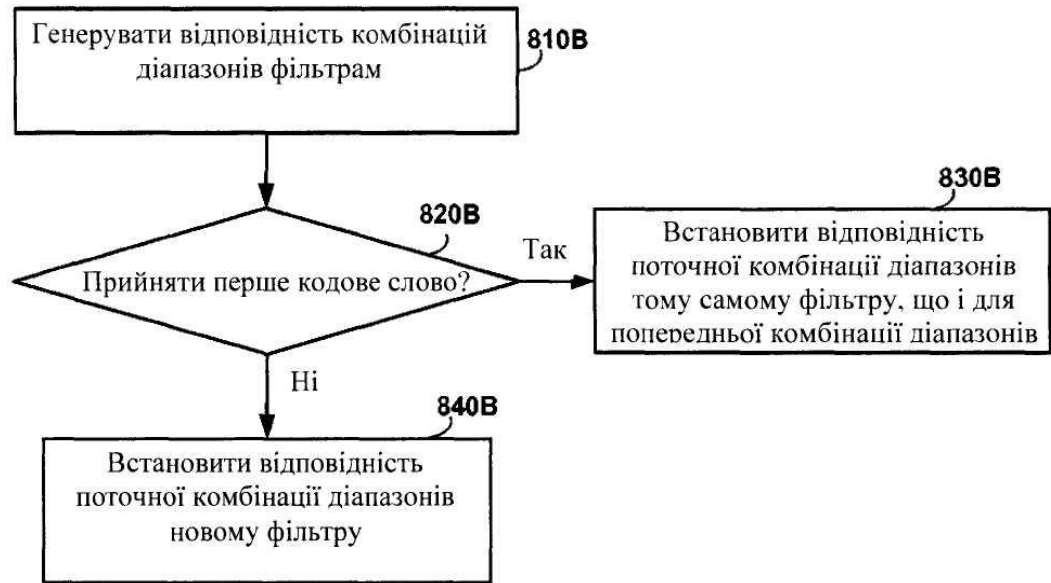




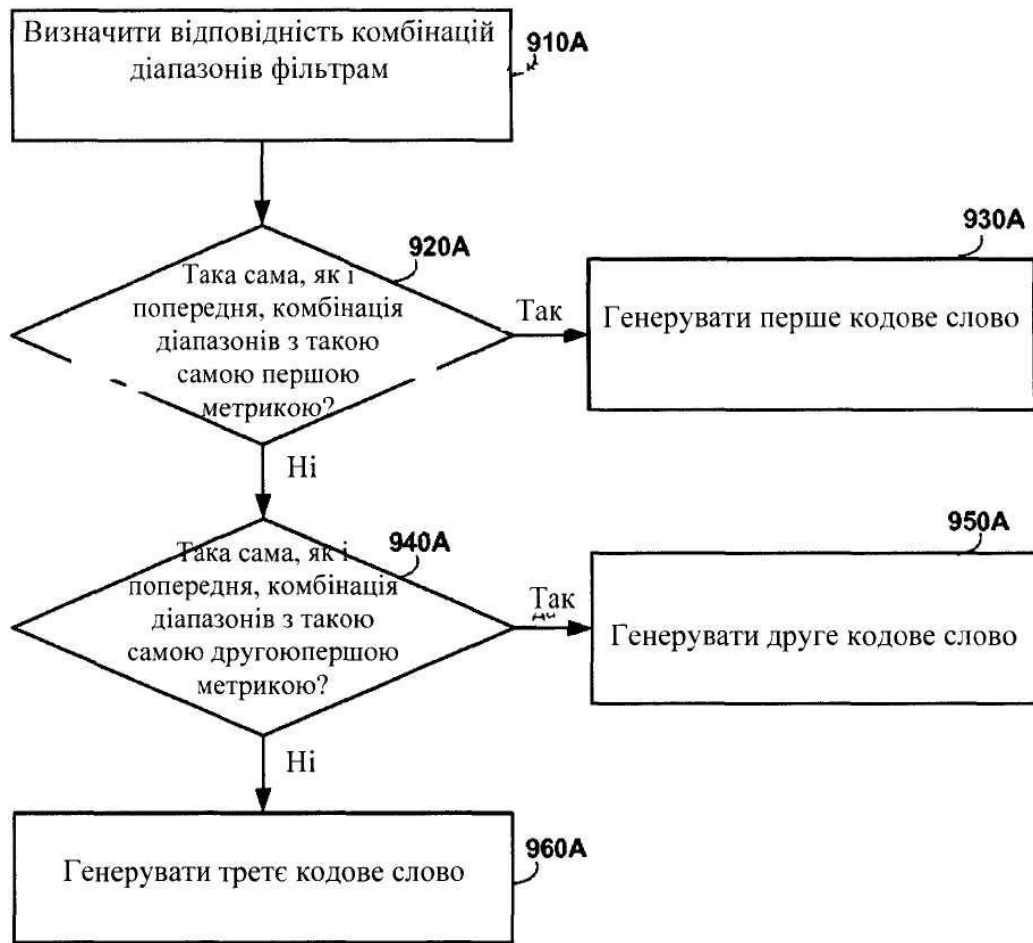
Фіг. 7



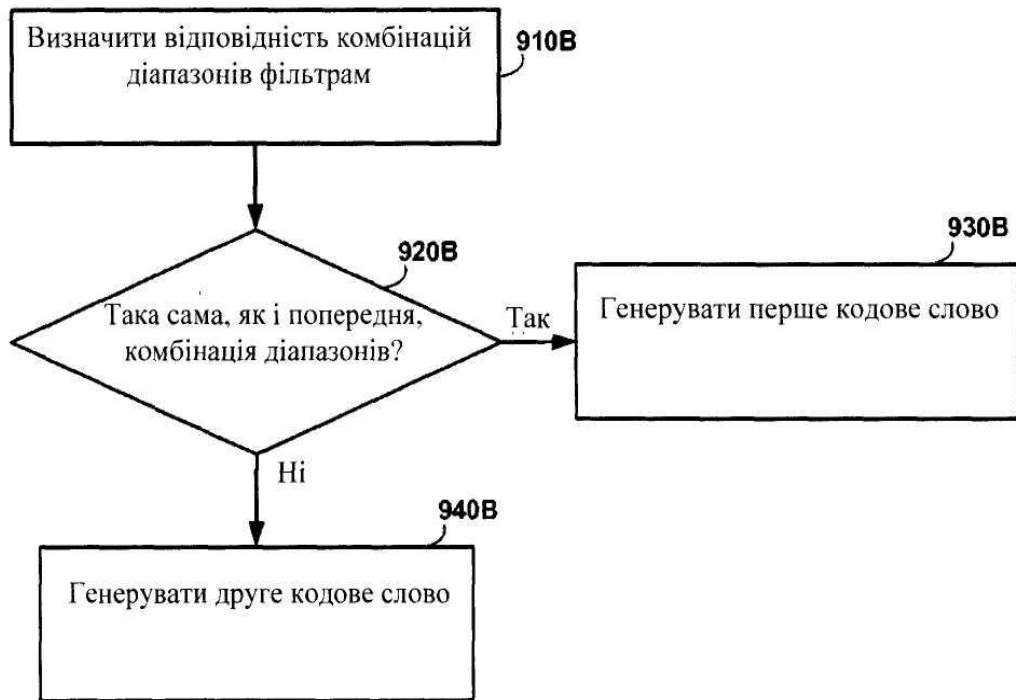
Фіг. 8А



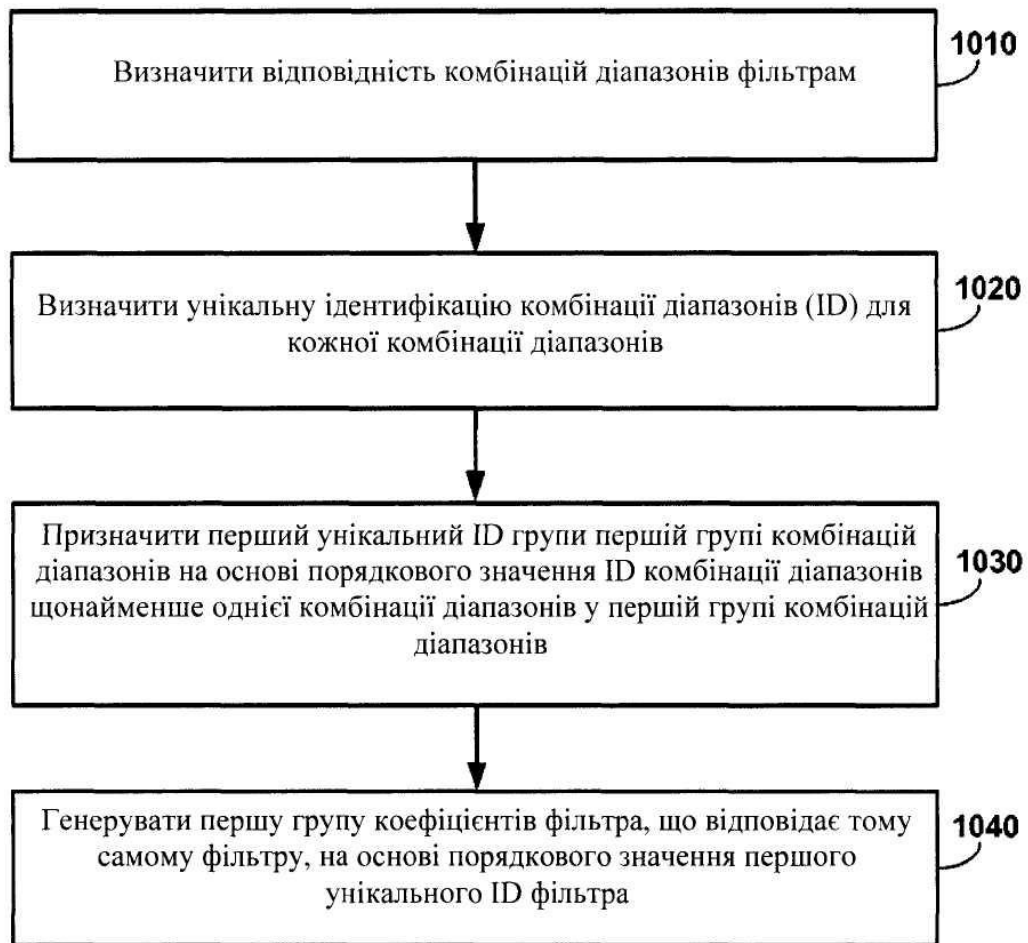
Фіг. 8В



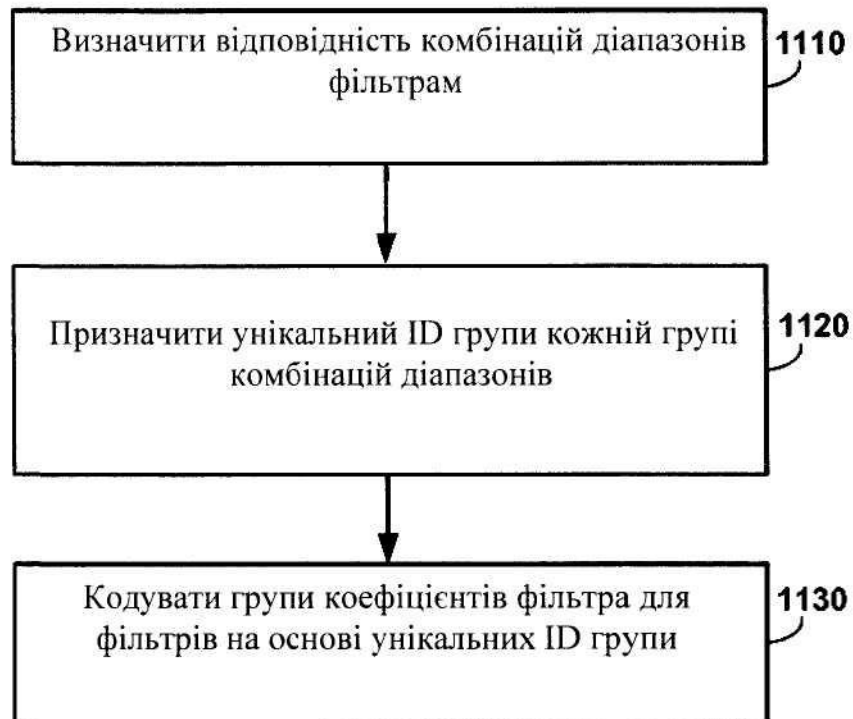
Фіг. 9А



Фіг. 9В



Фіг. 10



Фіг. 11