

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 972 701**

51 Int. Cl.:

H04N 19/597 (2014.01)

H04N 19/70 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **11.03.2020 PCT/US2020/022136**

87 Fecha y número de publicación internacional: **17.09.2020 WO20185922**

96 Fecha de presentación y número de la solicitud europea: **11.03.2020 E 20770118 (6)**

97 Fecha y número de publicación de la concesión europea: **27.12.2023 EP 3925212**

54 Título: **Imágenes con tipos de unidad NAL mezclados**

30 Prioridad:

11.03.2019 US 201962816749 P

10.04.2019 US 201962832132 P

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
14.06.2024

73 Titular/es:

**HUAWEI TECHNOLOGIES CO., LTD. (100.0%)
Huawei Administration Building, Bantian,
Longgang District, Shenzhen,
Guangdong 518129, CN**

72 Inventor/es:

**WANG, YE-KUI y
HENDRY, FNU**

74 Agente/Representante:

PONS ARIÑO, Ángel

ES 2 972 701 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Imágenes con tipos de unidad NAL mezclados

5 Referencia cruzada a solicitudes relacionadas

Esta solicitud de patente reivindica el beneficio de la Solicitud de Patente Provisional de EE. UU., N.º 62/816,749, presentada el 11 de Marzo de 2019, por Ye-Kui Wang, et al., y titulada "Support Of Mixed NAL Unit Types Within One Picture In Video Coding," y de la Solicitud de Patente Provisional de EE. UU., N.º 62/832,132, presentada el 10 de Abril de 2019, por Ye-Kui Wang, et al., y titulada "Support Of Mixed NAL Unit Types Within One Picture In Video Coding," que aquí se incorporan como referencia.

Campo técnico

15 La presente descripción generalmente está relacionada con la codificación de video, y específicamente está relacionada con la codificación de sub-imágenes de imágenes en la codificación de video.

Antecedentes

20 La cantidad de datos de video necesarios para mostrar incluso un video relativamente corto puede ser sustancial, lo cual puede dar como resultado dificultades cuando los datos han de ser transmitidos por secuencias o comunicados de otra manera a través de una red de comunicaciones con capacidad de ancho de banda limitado. Por lo tanto, los datos de video generalmente son comprimidos antes de ser comunicados a través de las redes de telecomunicaciones por módems. El tamaño de un video también podría ser un problema cuando el video es almacenado en un dispositivo de almacenamiento debido a que los recursos de memoria pueden ser limitados. Dispositivos de compresión de video con frecuencia utilizan software y/o hardware en el origen para codificar los datos de video antes de la transmisión o almacenamiento, disminuyendo así la cantidad de datos necesarios para representar imágenes de video digitales. Los datos comprimidos son entonces recibidos en el destino por un dispositivo de descompresión de video que decodifica los datos de video. Con recursos de red limitados y demandas cada vez mayores de calidad de video superior, son deseables técnicas de compresión y descompresión mejoradas que mejoren la relación de compresión con poco a ningún sacrificio en la calidad de las imágenes.

35 Los documentos EP3906685A1 y WO2020/142704A1 describen que `sps_mixed_tile_groups_in_pic_flag` igual a 1 especifica que puede haber imágenes en el CVS que tienen tanto grupos de mosaico IRAP como grupos que no son de mosaico IRAP. `sps_mixed_tile_groups_in_pic_flag` igual a 0 especifica que cada imagen en el CVS tiene o bien grupos de mosaico IRAP solamente o grupos de mosaico no-IRAP solamente.

40 HANNUKSELA (NOKIA) M M: "AHG12/AHG17: On merging of MCTSs for viewport-dependent streaming" propone que las subimágenes de la misma imagen codificada se permite que tengan diferentes valores de `nal_unit_type` para que el diseño de VVC pueda permitirse fusionar MCTS originados a partir de una imagen de acceso aleatorio y MCTS originados a partir de una imagen de acceso no aleatorio en la misma imagen codificada sin dificultad sustancial.

45 HANNUKSELA (NOKIA) M M ET AL: "AHG12: On grouping of tiles" describe la división y decodificación de imágenes basadas en subimágenes, y las subimágenes se tratan como imágenes en el proceso de decodificación. Y además describe un indicador `multiple_subpics_enabled_flag` para indicar que una imagen contiene exactamente una subimagen, o que una imagen puede contener más de una sub-imagen.

50 HANNUKSELA (NOKIA) M M ET AL: "Use cases and proposed design choices for adaptive resolution changing (ARC)" propone que el diseño de VVC debe permitir la fusión de una subimagen que se origina en una imagen de acceso aleatorio y otra subimagen que se origina en una imagen de acceso no aleatorio en la misma imagen codificada que es conforme a VVC.

55 Compendio

En una realización, la descripción incluye un método según la reivindicación 1 implementado en un decodificador.

60 Una imagen puede ser dividida en múltiples sub-imágenes. Dichas sub-imágenes pueden ser codificadas en sub-secuencias de bits separadas, las cuales entonces pueden ser fusionadas en una corriente de bits para transmisión a un decodificador. Por ejemplo, se pueden emplear sub-imágenes para aplicaciones de realidad virtual (VR). Como un ejemplo específico, un usuario solamente puede ver una porción de una imagen de VR en cualquier momento. Por consiguiente, diferentes sub-imágenes pueden ser transmitidas a diferentes resoluciones de manera que pueda ser asignado más ancho de banda a sub-imágenes que probablemente sean mostradas y sub-imágenes que tengan poca probabilidad de ser mostradas pueden ser comprimidas

para incrementar la eficiencia de la codificación. Además, las transmisiones de video pueden ser codificadas utilizando imágenes de punto de acceso intra-aleatorio (IRAP). Una imagen IRAP es codificada según intra-predicción y puede ser decodificada sin referencia a otras imágenes. Las imágenes no-IRAP pueden ser codificadas según inter-predicción y pueden ser decodificadas haciendo referencia a otras imágenes. Las imágenes no-IRAP están significativamente más condensadas que las imágenes IRAP. Sin embargo, una secuencia de video debe comenzar la decodificación con una imagen IRAP debido a que la imagen IRAP contiene suficientes datos para ser decodificados sin hacer referencia a otras imágenes. Las imágenes IRAP pueden ser utilizadas en sub-imágenes, y pueden permitir cambios de resolución dinámica. Por consiguiente, un sistema de video puede transmitir más imágenes IRAP para sub-imágenes que tienen más probabilidad de ser vistas (por ejemplo, basado en la ventana de visualización actual de usuarios) y una menor cantidad de imágenes IRAP para sub-imágenes que tienen pocas probabilidades de ser vistas a fin de aumentar adicionalmente la eficiencia de la codificación. Sin embargo, sub-imágenes son parte de la misma imagen. Por consiguiente, este esquema puede tener como resultado una imagen que contiene tanto una sub-imagen IRAP como una sub-imagen no-IRAP. Algunos sistemas de video no están equipados para manejar una imagen mezclada con regiones tanto IRAP como no-IRAP. La presente descripción incluye un indicador que indica si una imagen es mezclada y, por lo tanto, contiene componentes IRAP y no-IRAP. Basado en este indicador, el decodificador puede tratar diferentes sub-imágenes de manera diferente cuando se decodifica para decodificar y mostrar de manera apropiada la imagen/sub-imágenes. Este indicador puede ser almacenado en un PPS y puede ser referido como un `mixed_nalu_types_in_pic_flag`. Debido a esto, los mecanismos descritos permiten la implementación de funcionalidad adicional. Además, los mecanismos descritos permiten cambios de resolución dinámica cuando se emplean secuencias de bits de sub-imagen. Por lo tanto, los mecanismos descritos permiten que secuencias de bits de sub-imagen de resolución inferior sean transmitidas cuando se transmite video de VR sin afectar significativamente a la experiencia del usuario. Debido a esto, los mecanismos descritos aumentan la eficiencia de la codificación, y por lo tanto reducen el uso de recursos de red, recursos de memoria, y/o recursos de procesamiento en el codificador y el decodificador.

Opcionalmente, en cualquiera de los aspectos precedentes, se proporciona otra implementación del aspecto, en donde la corriente de bits incluye un conjunto de parámetros de imagen (PPS) incluyendo el indicador.

Opcionalmente, en cualquiera de los aspectos precedentes, se proporciona otra implementación del aspecto, en donde el primer valor de tipo de unidad NAL es igual a Actualización de Decodificación Instantánea (IDR) con imagen principal decodificable de acceso aleatorio (IDR_W_RADL), IDR sin imágenes principales (IDR_N_LP), o tipo de unidad NAL de acceso aleatorio limpio (CRA), (CRA_NUT).

Opcionalmente, en cualquiera de los aspectos precedentes, se proporciona otra implementación del aspecto, en donde el segundo valor de tipo de unidad NAL es igual al tipo de unidad NAL de imagen final (TRAIL_NUT), el tipo de unidad NAL de imagen principal decodificable de acceso aleatorio (RADL_NUT), o el tipo de unidad NAL de imagen principal omitida de acceso aleatorio (RASL), (RASL_NUT).

Opcionalmente, en cualquiera de los aspectos precedentes, se proporciona otra implementación del aspecto, en donde el indicador es un `mixed_nalu_types_in_pic_flag`.

Opcionalmente, en cualquiera de los aspectos precedentes, se proporciona otra implementación del aspecto, en donde el `mixed_nalu_types_in_pic_flag` es igual a uno cuando se especifica que la imagen que se refiere al PPS tiene más de una de las unidades NAL VCL y las unidades NAL VCL no tienen el mismo valor de tipo de unidad NAL (`nal_unit_type`), y en donde `mixed_nalu_types_in_pic_flag` es igual a cero cuando se especifica que la imagen que hace referencia al PPS tiene una o más de las unidades NAL VCL y las unidades NAL VCL tienen el mismo valor de `nal_unit_type`.

En una realización, la descripción incluye un método según la reivindicación 7 implementado en un codificador.

Una imagen puede ser dividida en múltiples sub-imágenes. Dichas sub-imágenes pueden ser codificadas en sub-secuencias de bits separadas, las cuales entonces pueden ser fusionadas en una corriente de bits para transmisión a un decodificador. Por ejemplo, se pueden emplear sub-imágenes para aplicaciones de realidad virtual (VR). Como un ejemplo específico, un usuario solamente puede ver una porción de una imagen de VR en cualquier momento. Por consiguiente, diferentes sub-imágenes pueden ser transmitidas a diferentes resoluciones de manera que pueda ser asignado más ancho de banda a sub-imágenes que tienen probabilidad de ser mostradas y sub-imágenes que tienen poca probabilidad de ser mostradas puedan ser comprimidas para incrementar la eficiencia de la codificación. Además, pueden ser codificadas secuencias de video utilizando imágenes de punto de acceso intra-aleatorio (IRAP). Una imagen IRAP es codificada según intra-predicción y puede ser decodificada sin hacer referencia a otras imágenes. Las imágenes no-IRAP pueden ser codificadas según inter-predicción y pueden ser decodificadas haciendo referencia a otras imágenes. Las imágenes no-IRAP están significativamente más condensadas que las imágenes IRAP. Sin embargo, una secuencia de video debe comenzar la decodificación con una imagen IRAP debido a que la

imagen IRAP contiene suficientes datos que han de ser decodificados sin hacer referencia a otras imágenes. Las imágenes IRAP pueden ser utilizadas en sub-imágenes, y pueden permitir cambios de resolución dinámica. Por consiguiente, un sistema de video puede transmitir más imágenes IRAP para sub-imágenes que tienen más probabilidad de ser vistas (por ejemplo, basado en la ventana de visualización actual de usuario) y menos imágenes IRAP para sub-imágenes que tienen pocas probabilidades de ser vistas para aumentar adicionalmente la eficiencia de la codificación. Sin embargo, las sub-imágenes son parte de la misma imagen. Por consiguiente, este esquema puede dar como resultado una imagen que contiene tanto una sub-imagen IRAP como una sub-imagen no-IRAP. Algunos sistemas de video no están equipados para manejar una imagen mezclada con regiones IRAP y no-IRAP. La presente descripción incluye un indicador que indica si una imagen es mezclada y, por lo tanto, contiene componentes IRAP y no-IRAP. Basándose en este indicador, el decodificador puede tratar diferentes sub-imágenes de manera diferente cuando se decodifican para decodificar y mostrar de manera apropiada la imagen/sub-imágenes. Este indicador puede ser almacenado en un PPS y puede ser denominado como un `mixed_nalu_types_in_pic_flag`. Como tal, los mecanismos descritos permiten la implementación de funcionalidad adicional. Además, los mecanismos descritos permiten cambios de resolución dinámica cuando se emplean corrientes de bits de sub-imagen. Por lo tanto, los mecanismos descritos permiten que corrientes de bits de sub-imagen de resolución inferior sean transmitidas cuando se transmite video de VR sin afectar significativamente a la experiencia del usuario. Como tal, los mecanismos descritos aumentan la eficiencia de la codificación, y por lo tanto reducen el uso de recursos de red, recursos de memoria, y/o recursos de procesamiento en el codificador y el decodificador.

Opcionalmente, en cualquiera de los aspectos precedentes, se proporciona otra implementación del aspecto, que comprende además la codificación de un PPS en la corriente de bits, en donde el indicador es codificado en el PPS.

Opcionalmente, en cualquiera de los aspectos precedentes, se proporciona otra implementación del aspecto, en donde el primer valor de tipo de unidad NAL es igual a `IDR_W_RADL`, `IDR_N_LP`, o `CRA_NUT`.

Opcionalmente, en cualquiera de los aspectos precedentes, se proporciona otra implementación del aspecto, en donde el segundo valor de tipo de unidad NAL es igual a `TRAIL_NUT`, `RADL_NUT`, o `RASL_NUT`.

Opcionalmente, en cualquiera de los aspectos precedentes, se proporciona otra implementación del aspecto, en donde el indicador es un `mixed_nalu_types_in_pic_flag`.

Opcionalmente, en cualquiera de los aspectos precedentes, se proporciona otra implementación del aspecto, en donde el `mixed_nalu_types_in_pic_flag` es igual a uno cuando se especifica que la imagen que hace referencia al PPS tiene más de una de las unidades NAL VCL y las unidades NAL VCL no tienen el mismo valor de `nal_unit_type`, y en donde `mixed_nalu_types_in_pic_flag` es igual a cero cuando se especifica que la imagen que hace referencia al PPS tiene una o más de las unidades NAL VCL y las unidades NAL VCL tienen el mismo valor de `nal_unit_type`.

En una realización, la descripción incluye un dispositivo de codificación de video que comprende: un procesador, un receptor acoplado al procesador, una memoria acoplada al procesador, y un transmisor acoplado al procesador, en donde el procesador, el receptor, la memoria y el transmisor están configurados para realizar el método de cualquiera de los aspectos precedentes.

Estas y otras características se entenderán de manera más clara a partir de la siguiente descripción detallada tomada en combinación con los dibujos adjuntos y las reivindicaciones.

Breve descripción de los dibujos

Para una comprensión más completa de esta descripción, a continuación se hace referencia a la siguiente breve descripción, tomada en conexión con los dibujos adjuntos y la descripción detallada, en donde números de referencia similares representan partes similares.

La figura 1 es un diagrama de flujo de un método de codificación ejemplar de una señal de video.

La figura 2 es un diagrama esquemático de un sistema de codificación y decodificación (códec) ejemplar para codificación de video.

La figura 3 es un diagrama esquemático que ilustra un codificador de video ejemplar.

La figura 4 es un diagrama esquemático que ilustra un decodificador de video ejemplar.

La figura 5 es un diagrama esquemático que ilustra una secuencia de video codificado ejemplar.

La figura 6 es un diagrama esquemático que ilustra una pluralidad de corrientes de video de sub-imagen

divididas a partir de una corriente de video de imagen de realidad virtual (VR).

La figura 7 es un diagrama esquemático que ilustra un ejemplo de corriente de bits que contiene imágenes con tipos de unidad de Capa de Abstracción de Red (NAL) mezclados.

5

La figura 8 es un diagrama esquemático de un dispositivo de codificación de video ejemplar.

La figura 9 es un diagrama de flujo de un método de codificación ejemplar de una secuencia de video que contiene una imagen con tipos de unidad NAL mezclados en una corriente de bits.

10

La figura 10 es un diagrama de flujo de un método de decodificación ejemplar de una secuencia de video que contiene una imagen con tipos de unidad NAL mezclados de una corriente de bits.

15

La figura 11 es un diagrama esquemático de un sistema ejemplar para codificar una secuencia de video que contiene una imagen con tipos de unidad NAL mezclados en una corriente de bits.

Descripción detallada

20

Debe entenderse al principio que aunque a continuación se proporciona una implementación ilustrativa de una o más realizaciones, los sistemas y/o métodos descritos pueden ser implementados utilizando cualquier número de técnicas, ya sean actualmente conocidas o existentes. La descripción no debe limitarse de ninguna manera a las implementaciones ilustrativas, dibujos y técnicas ilustradas a continuación, incluyendo los diseños e implementaciones ejemplares, ilustrados y descritos en la presente memoria, pero pueden ser modificados dentro del alcance de las reivindicaciones adjuntas.

25

En la presente memoria se utilizan los siguientes acrónimos, Secuencia de video Codificado (CVS), Memoria Tampón de Imágenes Decodificadas (DPB), Actualización de Decodificación Instantánea (IDR), Punto de Acceso Intra-Aleatorio (IRAP), Bit Menos Significativo (LSB), Bit Más Significativo (MSB), Capa de Abstracción de Red (NAL), Cómputo de Orden de Imagen (POC), Carga Útil de Secuencia de Bytes Sin Procesar (RBSP), Conjunto de Parámetros de Secuencia (SPS), y Borrador de Trabajo (WD).

30

35

Se pueden emplear muchas técnicas de compresión de video para reducir el tamaño de archivos de video con pérdida mínima de datos. Por ejemplo, las técnicas de compresión de video pueden incluir realizar predicción espacial (por ejemplo, intra-imagen) y/o predicción temporal (por ejemplo, inter-imagen) para reducir o eliminar redundancia de datos en secuencias de video. Para codificación de video basada en bloques, un segmento de video (por ejemplo, una imagen de video o una porción de una imagen de video) puede ser dividido en bloques de video, los cuales también pueden ser denominados como bloques de árbol, bloques de árbol de codificación (CTB), unidades de árbol de codificación (CTU), unidades de codificación (CU), y/o nodos de codificación. Los bloques de video en un segmento intra-codificado (I) de una imagen son codificados utilizando predicción espacial con respecto a muestras de referencia en bloques contiguos en la misma imagen. Bloques de video en un segmento inter-codificado de predicción unidireccional (P) o de predicción bidireccional (B) de una imagen pueden ser codificados empleando predicción espacial con respecto a muestras de referencia en bloques contiguos en la misma imagen o predicción temporal con respecto a muestras de referencia en otras imágenes de referencia. Las imágenes pueden ser referidas como fotogramas y/o imágenes, y las imágenes de referencia pueden ser referidas como fotogramas de referencia y/o imágenes de referencia. La predicción espacial o temporal tiene como resultado un bloque predictivo que representa un bloque de imagen. Datos residuales representan diferencias de pixel entre el bloque de imagen original y el bloque predictivo. Por consiguiente, un bloque inter-codificado es codificado según un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo y los datos residuales que indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intra-codificado es codificado según un modo de intra-codificación y los datos residuales. Para compresión adicional, los datos residuales pueden ser transformados desde el dominio de pixel a un dominio de transformada. Estos dan como resultado coeficientes de transformada residuales, los cuales pueden ser cuantificados. Los coeficientes de transformada cuantificados inicialmente pueden ser dispuestos en una matriz bidimensional. Los coeficientes de transformada cuantificados pueden ser escaneados para producir un vector uni-dimensional de coeficientes de transformada. La codificación entrópica puede ser aplicada para lograr incluso más compresión. Dichas técnicas de compresión de video se analizan con mayor detalle a continuación.

50

55

60

Para garantizar que un video codificado pueda ser decodificado con precisión, el video es codificado y decodificado según normas de codificación de video correspondientes. Las normas de codificación de video incluyen el Sector de Estandarización (ITU-T) H.261 de la Unión Internacional de Telecomunicaciones (ITU), Grupo de Expertos de Imágenes en Movimiento (MPEG)-1 Parte 2, de la Organización para Normalización/Comisión Electrotécnica Internacional (ISO/IEC), ITU-T H.262 o ISO/IEC MPEG-2 Parte 2, ITU-T H.263, ISO/IEC MPEG-4 Parte 2, Codificación de Video Avanzada (AVC), también conocida como ITU-T H.264 o ISO/IEC MPEG-4 Parte 10, y Codificación de Video de Alta Eficiencia (HEVC), también conocida

65

- como ITU-T H.265 o MPEG-H Parte 2. La AVC incluye extensiones tales como Codificación de Video Escalable (SVC), Codificación de Video Multivisión (MVC) y Codificación de Video Multivisión más Profunda (MVC+D), y AVC (3D-AVC) tridimensional (3D). La HEVC incluye extensiones tales como HEVC Escalable (SHVC), HEVC Multivisión (MV-HEVC), y 3D HEVC (3D-HEVC). El equipo conjunto de expertos de video (JVET) de ITU-T e ISO/IEC ha comenzado a desarrollar una norma de codificación de video denominada como Codificación de Video Versátil (VVC). La VVC está incluida en un Borrador de Trabajo (WD), el cual incluye JVET-M1001-v6 que proporciona una descripción de algoritmo, una descripción del lado del codificador del VVC WD, y software de referencia.
- 5
- 10 Los sistemas de codificación de video pueden codificar video empleando imágenes IRAP e imágenes no-IRAP. Las imágenes IRAP son imágenes codificadas de acuerdo con intra-predicción que sirven como puntos de acceso aleatorio para una secuencia de video. En intra-predicción, los bloques de una imagen son codificados por referencia a otros bloques en la misma imagen. Esto es en contraste a imágenes no-IRAP que emplean inter-predicción. En inter-predicción, los bloques de una imagen actual son codificados por referencia a otros bloques en una imagen de referencia que es diferente de la imagen actual. Debido a que una imagen IRAP es codificada sin referencia a otras imágenes, la imagen IRAP puede ser decodificada sin primero decodificar ninguna otra imagen. Por consiguiente, un decodificador puede comenzar a decodificar una secuencia de video en cualquier imagen IRAP. En contraste, una imagen no-IRAP es codificada en referencia a otras imágenes, y por lo tanto un decodificador generalmente es incapaz de comenzar a decodificar una secuencia de video en una imagen no-IRAP. Las imágenes IRAP también actualizan la DPB. Esto se debe a que la imagen IRAP es un punto de inicio para una CVS, e imágenes en la CVS no se refieren a imágenes en la CVS anterior. Como tal, las imágenes IRAP también pueden detener los errores de codificación relacionados con la inter-predicción debido a que dichos errores no se pueden propagar a través de la imagen IRAP. Sin embargo, las imágenes IRAP son significativamente más grandes que las imágenes no-IRAP desde un punto de vista de tamaño de datos. Debido a esto, una secuencia de video generalmente incluye muchas imágenes no-IRAP con un número más pequeño imágenes IRAP entremezcladas para equilibrar la eficiencia de la codificación con funcionalidad. Por ejemplo, una CVS de sesenta fotogramas puede incluir una imagen IRAP y cincuenta y nueve imágenes no-IRAP.
- 15
- 20
- 25
- 30 En algunos casos, se pueden emplear sistemas de codificación de video para codificar video de realidad virtual (VR), el cual también puede ser denominado como video de 360 grados. Un video de VR puede incluir una esfera de contenido de video mostrada como si el usuario estuviera en el centro de la esfera. Solamente una porción de la esfera, denominada como un puerto de visualización, es mostrada al usuario. Por ejemplo, el usuario puede emplear un dispositivo de visualización montado en la cabeza (HMD) que selecciona y presenta un puerto de visualización de la esfera basado en el movimiento de cabeza del usuario. Esto proporciona la impresión de estar físicamente presente en un espacio virtual tal como es representado por el video. Para lograr este resultado, cada imagen de la secuencia de video incluye una esfera completa de datos de video en un instante correspondiente en tiempo. Sin embargo, solamente una pequeña porción (por ejemplo, un solo puerto de visualización) de la imagen es mostrada al usuario. El resto de la imagen es desechado sin ser representado. La imagen completa es transmitida generalmente de manera que un puerto de visualización diferente puede ser dinámicamente seleccionado y desplegado en respuesta al movimiento de cabeza de los usuarios. Este enfoque puede dar como resultado tamaños de archivo de video muy grandes.
- 35
- 40
- 45 Para mejorar la eficiencia de la codificación, algunos sistemas dividen las imágenes en sub-imágenes. Una sub-imagen es una región espacial definida de una imagen. Cada sub-imagen contiene un puerto de visualización correspondiente de la imagen. El video puede ser codificado a dos o más resoluciones. Cada resolución es codificada en una sub-corriente de bits diferente. Cuando un usuario transmite el video de VR, el sistema de codificación puede fusionar las sub-corrientes de bits en una corriente de bits para transmisión con base en el puerto de visualización actual en uso por el usuario. Específicamente, el puerto de visualización actual se obtiene a partir de la sub-corriente de bits de alta resolución y los puertos de visualización que no están siendo vistos se obtienen de las corrientes de bits de baja resolución. De esta manera, la calidad de video más alta es mostrada al usuario y la calidad de video inferior es desechada. En el caso en que el usuario selecciona un nuevo puerto de visualización, el video de resolución inferior es presentado al usuario. El decodificador puede solicitar que el nuevo puerto de visualización reciba el video de resolución superior. El codificador entonces puede alterar el proceso de fusión consecuentemente. Una vez que se alcanza una imagen IRAP, el decodificador puede comenzar a decodificar la secuencia de video de resolución superior en el nuevo puerto de visualización. Este enfoque incrementa significativamente la compresión de video sin impactar negativamente la experiencia de visualización del usuario.
- 50
- 55
- 60 Una preocupación con el enfoque antes mencionado es que el período de tiempo necesario para cambiar las resoluciones se basa en el período de tiempo hasta que se alcanza una imagen IRAP. Esto se debe a que el decodificador es incapaz de comenzar a decodificar una secuencia de video diferente en una imagen no-IRAP como se describió antes. Un enfoque para reducir dicha latencia es incluir más imágenes IRAP. Sin embargo, esto da como resultado un incremento en el tamaño de archivo. A fin de equilibrar la funcionalidad con la eficiencia de la codificación, diferentes puertos de visualización/sub-imágenes pueden incluir imágenes
- 65

IRAP a diferentes frecuencias. Por ejemplo, puertos de visualización que tienen más probabilidad de ser vistos pueden tener más imágenes IRAP que otros puertos de visualización. Por ejemplo, en un contexto de baloncesto, los puertos de visualización relacionados con las canastas y/o el centro de la cancha pueden incluir imágenes IRAP a una mayor frecuencia que los puertos de visualización que observan los soportes o el techo debido a que dichos puertos de visualización tienen menos probabilidad de ser vistos por el usuario.

Este enfoque conduce a otros problemas. Específicamente, las sub-imágenes que contienen los puertos de visualización son parte de una sola imagen. Cuando diferentes sub-imágenes tienen imágenes IRAP a diferentes frecuencias, algunas de las imágenes incluyen tanto sub-imágenes IRAP como sub-imágenes no-IRAP. Este es un problema debido a que las imágenes son almacenadas en una corriente de bits empleando unidades NAL. Una unidad NAL es una unidad de almacenamiento que contiene un conjunto de parámetros o un segmento de una imagen y una cabecera de segmento correspondiente. Una unidad de acceso es una unidad que contiene una imagen completa. Como tal, una unidad de acceso contiene todas las unidades NAL relacionadas con la imagen. Las unidades NAL también contienen un tipo que indica el tipo de imagen que incluye el segmento. En algunos sistemas de video, se requiere que todas las unidades NAL relacionadas con una sola imagen (por ejemplo, incluidas en la misma unidad de acceso) tengan el mismo tipo. Como tal, el mecanismo de almacenamiento de unidad NAL puede dejar de operar correctamente cuando una imagen incluye tanto sub-imágenes IRAP como sub-imágenes no-IRAP.

Aquí se describen mecanismos para ajustar el esquema de almacenamiento NAL para soportar imágenes que incluyen tanto sub-imágenes IRAP como sub-imágenes no-IRAP. Esto a su vez permite video de VR que incluye diferentes frecuencias de sub-imagen IRAP para diferentes puertos de visualización. La invención utiliza un indicador que indica si una imagen es mezclada. En particular, el indicador indica si la imagen contiene tanto sub-imágenes IRAP como sub-imágenes no-IRAP. Basado en este indicador, el decodificador puede tratar diferentes tipos de sub-imágenes de manera diferente cuando decodifica a fin de decodificar y mostrar de manera apropiada la imagen/sub-imágenes. Este indicador puede ser almacenado en un conjunto de parámetros de imagen (PPS) y puede ser denominado como un `mixed_nalu_types_in_pic_flag`.

La invención implica un indicador que indica si una imagen está mezclada. En particular, el indicador indica si la imagen contiene tanto sub-imágenes IRAP como no-IRAP. Además, el indicador restringe la imagen de manera que la imagen mezclada contiene exactamente dos tipos de unidad NAL incluyendo un tipo IRAP y un tipo no-IRAP. Por ejemplo, la imagen puede contener unidades NAL IRAP que incluyen uno y solamente uno de actualizar decodificación instantánea (IDR) con imagen principal decodificable de acceso aleatorio (IDR_W_RADL), IDR sin imágenes principales (IDR_N_LP), o un tipo (CRA_NUT) de unidad NAL de acceso aleatorio limpio (CRA). Además, la imagen puede contener unidades NAL no-IRAP incluyendo uno y solamente uno de un tipo de unidad NAL de imagen final (TRAIL_NUT), un tipo de unidad NAL de imagen principal decodificable de acceso aleatorio (RADL_NUT), o un tipo de unidad NAL (RASL) de imagen principal omitida de acceso aleatorio (RASL_NUT). Basado en este indicador, el decodificador puede tratar diferentes sub-imágenes de manera diferente cuando decodifica para decodificar y mostrar de manera apropiada la imagen/sub-imágenes. Este indicador puede ser almacenado en un PPS y se puede denominar como un `mixed_nalu_types_in_pic_flag`.

La figura 1 es un diagrama de flujo de un método operativo ejemplar 100 de codificación de una señal de video. Específicamente, una señal de video es codificada en un codificador. El proceso de codificación comprime la señal de video empleando varios mecanismos para reducir el tamaño del archivo de video. Un tamaño de archivo más pequeño permite que el archivo de video comprimido sea transmitido hacia un usuario, al tiempo que se reduce la sobrecarga de ancho de banda asociada. El decodificador entonces decodifica el archivo de video comprimido para reconstruir la señal de video original para mostrar a un usuario final. El proceso de decodificación generalmente refleja el proceso de codificación para permitir que el decodificador reconstruya de manera consistente la señal de video.

En la etapa 101, la señal de video es introducida en el codificador. Por ejemplo, la señal de video puede ser un archivo de video no comprimido almacenado en memoria. Como otro ejemplo, el archivo de video puede ser capturado por un dispositivo de captura de video, tal como una cámara de video, y codificado para soportar la transmisión en vivo del video. El archivo de video puede incluir tanto un componente de audio como un componente de video. El componente de video contiene una serie de fotogramas de imagen que, cuando son vistos en una secuencia, da la impresión visual de movimiento. Los fotogramas contienen píxeles que son expresados en términos de luz, denominados en la presente memoria como componentes de luminancia (o muestras de luminancia), y color, que son denominados como componentes de crominancia (o muestras de color). En algunos ejemplos, los fotogramas también pueden contener valores de profundidad para soportar visión tridimensional.

En la etapa 103, el video es dividido en bloques. La división incluye subdividir los píxeles en cada fotograma en bloques cuadrados y/o rectangulares para compresión. Por ejemplo, en Codificación de Video de Alta Eficiencia (HEVC) (también conocida como H.265 y MPEG-H Parte 2) el fotograma puede primero ser dividido en unidades de árbol de codificación (CTU), que son bloques de un tamaño predefinido (por ejemplo,

sesenta y cuatro píxeles por sesenta y cuatro píxeles). Las CTU contienen tanto muestras de luminancia como de crominancia. Los árboles de codificación pueden ser empleados para dividir las CTU en bloques y después subdividir de modo recurrente los bloques hasta que se logran las configuraciones que soportan codificación adicional. Por ejemplo, componentes de luminancia de un fotograma pueden ser subdivididos hasta que los bloques individuales contienen valores de iluminación relativamente homogéneos. Además, componentes de crominancia de un fotograma pueden ser subdivididos hasta que los bloques individuales contienen valores de color relativamente homogéneos. Por consiguiente, los mecanismos de división varían dependiendo del contenido de los fotogramas de video.

En la etapa 105, se emplean varios mecanismos de compresión para comprimir los bloques de imagen divididos en la etapa 103. Por ejemplo, se puede emplear inter-predicción y/o intra-predicción. La Inter-predicción está diseñada para beneficiarse del hecho de que los objetos en una escena común tienden a aparecer en fotogramas sucesivos. Por consiguiente, un bloque que representa un objeto en un fotograma de referencia no necesita describirse repetidamente en fotogramas adyacentes. Específicamente, un objeto, tal como una mesa, puede permanecer en una posición constante sobre múltiples fotogramas. Por lo tanto, la mesa se describe una vez y los fotogramas adyacentes pueden hacer referencia nuevamente al fotograma de referencia. Se pueden emplear mecanismos de emparejamiento de patrón para emparejar objetos sobre múltiples fotogramas. Además, objetos en movimiento pueden ser representados a través de múltiples fotogramas, por ejemplo debido al movimiento de los objetos o al movimiento de la cámara. Como un ejemplo particular, un video puede mostrar un automóvil que se mueve por toda la pantalla sobre múltiples fotogramas. Se pueden emplear vectores de movimiento para describir dicho movimiento. Un vector de movimiento es un vector bidimensional que proporciona un desplazamiento de las coordenadas de un objeto en un fotograma a las coordenadas del objeto en un fotograma de referencia. Debido a esto, la inter-predicción puede codificar un bloque de imagen en un fotograma actual como un conjunto de vectores de movimiento indicando un desplazamiento de un bloque correspondiente en un fotograma de referencia.

La intra-predicción codifica bloques en un fotograma común. La intra-predicción se beneficia del hecho de que componentes de luminancia y crominancia tienden a agruparse en un fotograma. Por ejemplo, un parche de verde en una porción de un árbol tiende a quedar colocado junto a parches similares de verde. La intra-predicción emplea múltiples modos de predicción direccional (por ejemplo, treinta y tres en HEVC), un modo plano, y un modo de corriente continua (CC). Los modos direccionales indican que un bloque actual es similar/el mismo que las muestras de un bloque contiguo en una dirección correspondiente. El modo plano indica que una serie de bloques a lo largo de una fila/columna (por ejemplo, un plano) pueden ser interpolados con base en bloques contiguos en los bordes de la fila. El modo plano, en efecto, indica una transición suave de luz/color a través de una fila/columna empleando una inclinación relativamente constante en valores cambiantes. El modo CC es empleado para suavización de límite e indica que un bloque es similar/el mismo que un valor promedio asociado con muestras de todos los bloques contiguos asociados con las direcciones angulares de los modos de predicción direccionales. Por consiguiente, bloques de intra-predicción pueden representar bloques de imagen como varios valores de modo de predicción de relación en lugar de los valores reales. Además, bloques de inter-predicción pueden representar bloques de imagen como valores de vector de movimiento en lugar de los valores reales. En cualquier caso, los bloques de predicción pueden no representar exactamente los bloques de imagen en algunos casos. Cualesquiera diferencias son almacenadas en bloques residuales. Las transformadas pueden ser aplicadas a los bloques residuales para comprimir más el archivo.

En la etapa 107, se pueden aplicar varias técnicas de filtración. En HEVC, los filtros se aplican según un esquema de filtración en bucle. La predicción basada en bloques antes analizada puede dar como resultado la creación de imágenes de bloqueo en el decodificador. Además, el esquema de predicción basada en bloques puede codificar un bloque y después puede reconstruir el bloque codificado para uso posterior como un bloque de referencia. El esquema de filtración en bucle iterativamente aplica filtros de supresión de ruido, filtros de desbloqueo, filtros de bucle adaptativo, y filtros de compensación adaptativa de muestra (SAO) a los bloques/fotogramas. Estos filtros mitigan dichos artefactos de bloqueo de manera que el archivo codificado puede ser reconstruido con precisión. Además, estos filtros mitigan los artefactos en los bloques de referencia reconstruidos de manera que los artefactos tienen menos probabilidades de crear artefactos adicionales en bloques subsiguientes que son codificados basados en los bloques de referencia reconstruidos.

Una vez que la señal de video ha sido dividida, comprimida y filtrada, los datos resultantes son codificados en una corriente de bits en la etapa 109. La corriente de bits incluye los datos antes analizados así como cualesquiera datos de señalización deseados para soportar la correcta reconstrucción de la señal de video en el decodificador. Por ejemplo, dichos datos pueden incluir datos de división, datos de predicción, bloques residuales, y varios indicadores que proporcionan instrucciones de codificación al decodificador. La corriente de bits puede ser almacenada en memoria para transmisión a un decodificador bajo solicitud. La corriente de bits también puede ser transmitida y/o difundida a una pluralidad de decodificadores. La creación de la corriente de bits es un proceso iterativo. Por consiguiente, las etapas 101, 103, 105, 107, y 109 pueden ocurrir de manera continua y/o simultánea sobre muchos fotogramas y bloques. El orden mostrado en la

figura 1 es presentado para claridad y facilidad de análisis, y no pretende limitar el proceso de codificación de video a un orden particular.

5 El decodificador recibe la corriente de bits y comienza el proceso de decodificación en la etapa 111. Específicamente, el decodificador emplea un esquema de decodificación entrópica para convertir la corriente de bits en datos de sintaxis y video correspondientes. El decodificador emplea los datos de sintaxis de la corriente de bits para determinar las divisiones para los fotogramas en la etapa 111. La división debería emparejarse con los resultados de la división de bloque en la etapa 103. Ahora se describe la codificación/decodificación entrópica, tal como se emplea en la etapa 111. El codificador realiza muchas elecciones durante el proceso de compresión, tal como la selección de esquemas de división de bloque entre 10 varias posibles elecciones con base en el posicionamiento espacial de valores en la imagen o imágenes de entrada. La señalización de las elecciones exactas puede emplear un número grande de bins. Tal como aquí se utiliza, un bin es un valor binario que es tratado como una variable (por ejemplo, un valor de bit que puede variar dependiendo del contexto). La codificación entrópica permite al codificador desechar cualesquiera 15 opciones que claramente no sean viables para un caso particular, dejando un conjunto de opciones permisibles. A cada opción permisible entonces se le asigna una palabra código. La longitud de la palabras código se basa en el número de opciones permitidas (por ejemplo, un bin para dos opciones, dos bins para tres a cuatro opciones, etc.) El codificador codifica luego la palabra código para la opción seleccionada. Este esquema reduce el tamaño de las palabras código debido a que las palabras código son tan grandes como 20 se desee para indicar de manera única la selección de entre un pequeño sub-conjunto de opciones permisibles en oposición a indicar de manera única la selección de entre un conjunto potencialmente grande de todas las posibles opciones. El decodificador entonces decodifica la selección determinando el conjunto de opciones permisibles en una manera similar al codificador. Al determinar el conjunto de opciones permisibles, el decodificador puede leer la palabra código y puede determinar la selección realizada por el 25 codificador.

En la etapa 113, el decodificador realiza decodificación de bloques. Específicamente, el decodificador emplea transformadas inversas para generar bloques residuales. Después el decodificador emplea los bloques residuales y los bloques de predicción correspondientes para reconstruir los bloques de imagen según la 30 división. Los bloques de predicción pueden incluir tanto bloques de intra-predicción como bloques de inter-predicción tal como son generados en el codificador en la etapa 105. Los bloques de imagen reconstruidos entonces son posicionados en fotogramas de una señal de video reconstruida según los datos de división determinados en la etapa 111. La sintaxis para la etapa 113 también puede ser señalizada en la corriente de bits mediante codificación entrópica tal como se analizó antes.

35 En la etapa 115, la filtración es realizada en los fotogramas de la señal de video reconstruida en una manera similar a la etapa 107 en el codificador. Por ejemplo, filtros de supresión de ruido, filtros de desbloqueo, filtros de bucle adaptativo, y filtros SAO pueden ser aplicados a los fotogramas para eliminar artefactos de bloqueo. Una vez que los fotogramas son filtrados, la señal de video puede ser emitida a un dispositivo de 40 visualización en la etapa 117 para visualización por un usuario final.

La figura 2 es un diagrama esquemático de un sistema 200 ejemplar de codificación y decodificación (códec) para codificación de video. Específicamente, el sistema 200 de códec proporciona funcionalidad para soportar la implementación del método operativo 100. El sistema 200 de códec es generalizado para 45 representar componentes empleados tanto en un codificador como en un decodificador. El sistema 200 de códec recibe y divide una señal de video tal como se analizó con respecto a las etapas 101 y 103 en el método operativo 100, lo cual da como resultado una señal 201 de video dividida. El sistema 200 de códec entonces comprime la señal 201 de video dividida en una corriente de bits codificada cuando actúa como un codificador como se analizó con respecto a las etapas 105, 107, y 109 en el método 100. Cuando actúa como un decodificador, el sistema 200 de códec genera una señal de video de salida a partir de la corriente de bits 50 como se analizó con respecto a las etapas 111, 113, 115, y 117 en el método operativo 100. El sistema 200 de códec incluye un componente 211 de control de codificador general, un componente 213 de cuantificación y escalado de transformada, un componente 215 de estimación de intra-imagen, un componente 217 de predicción de intra-imagen, un componente 219 de compensación de movimiento, un componente 221 de estimación de movimiento, un componente 229 de transformada inversa y escalado, un componente 227 de análisis de control de filtro, un componente 225 de filtros en bucle, un componente 223 de memoria tampón de imágenes decodificadas, y un componente 231 de codificación aritmética binaria de contexto adaptativo (CABAC) y formateo de cabecera. Dichos componentes son acoplados como se muestra. En la figura 2, 55 líneas negras indican el movimiento de datos que han de ser codificados/decodificados mientras que líneas de trazos indican el movimiento de datos de control que controlan la operación de otros componentes. Los componentes del sistema 200 de códec pueden estar todos presentes en el codificador. El decodificador puede incluir un subconjunto de los componentes del sistema 200 de códec. Por ejemplo, el decodificador puede incluir el componente 217 de predicción de intra-imagen, el componente 219 de compensación de movimiento, el componente 229 de transformada inversa y escalado, el componente 225 de filtros en bucle, y 60 el componente 223 de memoria tampón de imágenes decodificadas. A continuación se describen estos componentes.

La señal 201 de video dividida es una secuencia de video capturada que ha sido dividida en bloques de pixeles por un árbol de codificación. Un árbol de codificación emplea varios modos de división para subdividir un bloque de pixeles en bloques más pequeños de pixeles. Estos bloques pueden luego ser subdivididos adicionalmente en bloques más pequeños. Los bloques pueden ser denominados como nodos en el árbol de codificación. Los nodos padres más grandes son divididos en nodos hijos más pequeños. El número de veces que un nodo es subdividido se denomina como la profundidad del nodo/árbol de codificación. Los bloques divididos pueden ser incluidos en unidades de codificación (CU) en algunos casos. Por ejemplo, una CU puede ser una sub-porción de una CTU que contiene un bloque de luminancia, bloque o bloques de crominancia de diferencia roja (Cr), y bloque o bloques de crominancia de diferencia azul (Cb) junto con instrucciones de sintaxis correspondientes para la CU. Los modos de división pueden incluir un árbol binario (BT), árbol ternario (TT), y un árbol cuaternario (QT) empleados para dividir un nodo en dos, tres o cuatro nodos hijos, respectivamente, de diversas formas dependiendo de los modos de división empleados. La señal 201 de video dividida es reenviada al componente 211 de control de codificador general, al componente 213 de cuantificación y escalado de transformada, al componente 215 de estimación de intra-imagen, al componente 227 de análisis de control de filtro, y al componente 221 de estimación de movimiento para compresión.

El componente 211 de control de codificador general está configurado para tomar decisiones relacionadas con la codificación de las imágenes de la secuencia de video en la corriente de bits según restricciones de aplicación. Por ejemplo, el componente 211 de control de codificador general gestiona la optimización del tamaño de tasa de bits/corriente de bits versus la calidad de la reconstrucción. Dichas decisiones pueden tomarse basándose en el espacio de almacenamiento/disponibilidad de ancho de banda así como las solicitudes de resolución de imagen. El componente 211 de control de codificador general también gestiona la utilización de memoria tampón en virtud de la velocidad de transmisión para mitigar los problemas de insuficiencia y exceso de memoria tampón. Para gestionar estos problemas, el componente 211 de control de codificador general gestiona la división, predicción, y filtración por los otros componentes. Por ejemplo, el componente 211 de control de codificador general puede aumentar dinámicamente la complejidad de la compresión para aumentar la resolución y aumentar el uso de ancho de banda o reducir la complejidad de la compresión para disminuir la resolución y el uso de ancho de banda. Por lo tanto, el componente 211 de control de codificador general controla los otros componentes del sistema 200 de códec para equilibrar la calidad de reconstrucción de señal de video con las cuestiones relacionadas con la de tasa de bits. El componente 211 de control de codificador general crea datos de control, que controlan la operación de los otros componentes. Los datos de control también son reenviados al componente 231 CABAC y de formateo de cabecera para ser codificados en la corriente de bits con la finalidad de señalar parámetros para decodificación en el decodificador.

La señal 201 de video dividida también es enviada al componente 221 de estimación de movimiento y al componente 219 de compensación de movimiento para inter-predicción. Un fotograma o segmento de la señal 201 de video dividida puede ser dividido en múltiples bloques de video. El componente 221 de estimación de movimiento y el componente 219 de compensación de movimiento realizan codificación inter-predictiva del bloque de video recibido con relación a uno o más bloques en uno o más fotogramas de referencia para proporcionar predicción temporal. El sistema 200 de códec puede realizar múltiples etapas de codificación, por ejemplo, para seleccionar un modo de codificación apropiado para cada bloque de datos de video.

El componente 221 de estimación de movimiento y el componente 219 de compensación de movimiento pueden estar muy integrados, pero se ilustran por separado con propósitos conceptuales. La estimación de movimiento, realizada por el componente 221 de estimación de movimiento, es el proceso de generar vectores de movimiento, que estiman el movimiento para bloques de video. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de un objeto codificado con relación a un bloque predictivo. Un bloque predictivo es un bloque que resulta estar estrechamente emparejado con el bloque que ha de ser codificado, en términos de diferencia de pixel. Un bloque predictivo también puede ser denominado como un bloque de referencia. Dicha diferencia de pixel puede ser determinada por la suma de diferencia absoluta (SAD), suma de diferencia cuadrada (SSD), u otras métricas de diferencia. El HEVC emplea varios objetos codificados incluyendo una CTU, bloques de árbol de codificación (CTB), y unas CU. Por ejemplo, una CTU puede ser dividida en CTB, los cuales entonces pueden ser divididos en CB para inclusión en las CU. Una CU puede ser codificada como una unidad de predicción (PU) que contiene datos de predicción y/o una unidad de transformada (TU) que contiene datos residuales transformados para la CU. El componente 221 de estimación de movimiento genera vectores de movimiento, utilizando las PU, y las TU, un análisis de distorsión de tasa como parte de un proceso de optimización de distorsión de tasa. Por ejemplo, el componente 221 de estimación de movimiento puede determinar múltiples bloques de referencia, múltiples vectores de movimiento, etc., para un bloque/fotograma actual, y puede seleccionar los bloques de referencia, vectores de movimiento, etc., que tienen las mejores características de distorsión de tasa. Las mejores características de distorsión de tasa equilibran tanto la reconstrucción de la calidad de video (por ejemplo, cantidad de pérdida de datos por compresión) con la eficiencia de codificación (por ejemplo, tamaño

de la codificación final).

En algunos ejemplos, el sistema 200 de códec puede calcular valores para posiciones de pixel de sub-número entero de imágenes de referencia almacenadas en el componente 223 de memoria tampón de imágenes decodificadas. Por ejemplo, el sistema 200 de códec de video puede interpolar valores de posiciones de pixel de un-cuarto, posiciones de pixel de un-octavo, u otras posiciones de pixel fraccionarias de la imagen de referencia. Por lo tanto, el componente 221 de estimación de movimiento puede realizar búsqueda de movimiento con relación a las posiciones de pixel completas y posiciones de pixel fraccionarias y emitir un vector de movimiento con precisión de pixel fraccionaria. El componente 221 de estimación de movimiento calcula un vector de movimiento para una PU de un bloque de video en un segmento inter-codificado comparando la posición de la PU con la posición de un bloque predictivo de una imagen de referencia. El componente 221 de estimación de movimiento emite el vector de movimiento calculado como datos de movimiento al componente 231 de CABAC y de formateo de cabecera para codificación y movimiento al componente 219 de compensación de movimiento.

La compensación de movimiento, realizada por el componente 219 de compensación de movimiento, puede implicar la extracción o generación del bloque predictivo basándose en el vector de movimiento determinado por el componente 221 de estimación de movimiento. De nuevo, el componente 221 de estimación de movimiento y el componente 219 de compensación de movimiento pueden estar funcionalmente integrados, en algunos ejemplos. Después de recibir el vector de movimiento para la PU del bloque de video actual, el componente 219 de compensación de movimiento puede localizar el bloque predictivo al cual apunta el vector de movimiento. Un bloque de video residual es entonces formado sustrayendo valores de pixel del bloque predictivo a partir de los valores de pixel del bloque de video actual que está siendo codificado, formando valores de diferencia de pixel. En general, el componente 221 de estimación de movimiento realiza estimación de movimiento con relación a los componentes de luminancia, y el componente 219 de compensación de movimiento utiliza vectores de movimiento calculados basándose en los componentes de luminancia para componentes de crominancia y componentes de luminancia. El bloque predictivo y el bloque residual son reenviados al componente 213 de cuantificación y escalado de transformada.

La señal 201 de video dividida también es enviada al componente 215 de estimación de intra-imagen y al componente 217 de predicción de intra-imagen. Al igual que con el componente 221 de estimación de movimiento y el componente 219 de compensación de movimiento, el componente 215 de estimación de intra-imagen y el componente 217 de predicción de intra-imagen pueden estar muy integrados, pero se ilustran por separado con propósitos conceptuales. El componente 215 de estimación de intra-imagen y el componente 217 de predicción de intra-imagen intra-predicen un bloque actual con relación a bloques en un fotograma actual, como una alternativa a la inter-predicción ejecutada por el componente 221 de estimación de movimiento y el componente 219 de compensación de movimiento entre fotogramas, como se describió antes. En particular, el componente 215 de estimación de intra-imagen determina un modo de intra-predicción a utilizar para codificar un bloque actual. En algunos ejemplos, el componente 215 de estimación de intra-imagen selecciona un modo de intra-predicción apropiado para codificar un bloque actual a partir de múltiples modos de intra-predicción probados. Los modos de intra-predicción seleccionados son entonces reenviados al componente 231 de CABAC y de formateo de cabecera para codificación.

Por ejemplo, el componente 215 de estimación de intra-imagen calcula valores de distorsión de tasa utilizando un análisis de distorsión de tasa para los diversos modos de intra-predicción probados, y selecciona el modo de intra-predicción que tenga las mejores características de distorsión de tasa entre los modos probados. El análisis de distorsión de tasa generalmente determina una cantidad de distorsión (o error) entre un bloque codificado y un bloque no codificado original que fue codificado para producir el bloque codificado, así como una tasa de bits (por ejemplo, un número de bits) utilizada para producir el bloque codificado. El componente 215 de estimación de intra-imagen calcula relaciones de las distorsiones y tasas para los diversos bloques codificados a fin de determinar qué modo de intra-predicción exhibe el mejor valor de distorsión de tasa para el bloque. Además, el componente 215 de estimación de intra-imagen puede ser configurado para codificar bloques profundos de un mapa de profundidad utilizando un modo de modelado de profundidad (DMM) basado en optimización de distorsión de tasa (RDO).

El componente 217 de predicción de intra-imagen puede generar un bloque residual a partir del bloque predictivo basándose en los modos de intra-predicción seleccionados determinados por el componente 215 de estimación de intra-imagen cuando se implementa en un codificador o puede leer el bloque residual de la corriente de bits cuando se implementa en un decodificador. El bloque residual incluye la diferencia en valores entre el bloque predictivo y el bloque original, representada como una matriz. El bloque residual entonces es reenviado al componente 213 de cuantificación y escalado de transformada. El componente 215 de estimación de intra-imagen y el componente 217 de predicción de intra-imagen pueden operar sobre componentes de luminancia y crominancia.

El componente 213 de cuantificación y escalado de transformada está configurado para comprimir adicionalmente el bloque residual. El componente 213 de cuantificación y escalado de transformada aplica

una transformada, tal como una transformada discreta del coseno (DCT), una transformada discreta de seno (DST), o una transformada conceptualmente similar, al bloque residual, produciendo un bloque de video que comprende valores de coeficientes de transformada residuales. También se podrían utilizar transformadas de ondícula, transformadas de número entero, transformadas de sub-banda u otros tipos de transformadas. La transformada puede convertir la información residual de un dominio de valor de pixel en un dominio de transformada, tal como un dominio de frecuencia. El componente 213 de cuantificación y escalado de transformada también está configurado para escalar la información residual transformada, por ejemplo basado en frecuencia. Dicho escalado implica aplicar un factor de escala a la información residual de manera que se cuantifique diferente información de frecuencia a diferentes granularidades, lo cual puede afectar a la calidad visual final del video reconstruido. El componente 213 de cuantificación y escalado de transformada también está configurado para cuantificar los coeficientes de transformada para reducir adicionalmente la tasa de bits. El proceso de cuantificación puede reducir la profundidad de bits asociada con algunos o todos los coeficientes. El grado de cuantificación puede ser modificado al ajustar un parámetro de cuantificación. En algunos ejemplos, el componente 213 de cuantificación y escalado de transformada puede luego realizar un escaneo de la matriz incluyendo los coeficientes de transformada cuantificados. Los coeficientes de transformada cuantificados son reenviados al componente 231 de CABAC y de formateo de cabecera para ser codificados en la corriente de bits.

El componente 229 de transformada inversa y escalado aplica una operación inversa a la del componente 213 de cuantificación y escalado de transformada para soportar estimación de movimiento. El componente 229 de transformada inversa y escalado aplica escalado, transformación y/o cuantificación inversa para reconstruir el bloque residual en el dominio de pixel, por ejemplo, para uso posterior como un bloque de referencia que puede resultar un bloque predictivo para otro bloque actual. El componente 221 de estimación de movimiento y/o el componente 219 de compensación de movimiento pueden calcular un bloque de referencia añadiendo el bloque residual de nuevo a un bloque predictivo correspondiente para uso en estimación de movimiento de un bloque/fotograma final. Se aplican filtros a los bloques de referencia reconstruidos para mitigar artefactos creados durante el escalado, la cuantificación y la transformación. Dichos artefactos de otra manera podrían ocasionar una predicción imprecisa (y crear artefactos adicionales) cuando se predicen bloques subsiguientes.

El componente 227 de análisis de control de filtro y el componente 225 de filtros en bucle aplican los filtros a los bloques residuales y/o a los bloques de imagen reconstruidos. Por ejemplo, el bloque residual transformado del componente 229 de transformada inversa y escalado se puede combinar con un bloque de predicción correspondiente del componente 217 de predicción de intra-imagen y/o del componente 219 de compensación de movimiento para reconstruir el bloque de imagen original. Los filtros pueden entonces ser aplicados al bloque de imagen reconstruido. En algunos ejemplos, los filtros en su lugar pueden ser aplicados a los bloques residuales. Al igual que con otros componentes en la figura 2, el componente 227 de análisis de control de filtro y el componente 225 de filtros en bucle están muy integrados y pueden ser implementados juntos, pero se muestran por separado con propósitos conceptuales. Los filtros aplicados a los bloques de referencia reconstruidos son aplicados a regiones espaciales particulares e incluyen múltiples parámetros para ajustar cómo se aplican dichos filtros. El componente 227 de análisis de control de filtro analiza los bloques de referencia reconstruidos para determinar en dónde deberían aplicarse dichos filtros y establece parámetros correspondientes. Dichos datos son reenviados al componente 231 de CABAC y de formateo de cabecera como datos de control de filtro para codificación. El componente 225 de filtros en bucle aplica dichos filtros basado en los datos de control de filtro. Los filtros pueden incluir un filtro de desbloqueo, un filtro de supresión de ruido, un filtro SAO, y un filtro de bucle adaptativo. Dichos filtros pueden ser aplicados en el dominio de pixel/espacial (por ejemplo, en un bloque de pixeles reconstruido) o en el dominio de frecuencia, dependiendo del ejemplo.

Cuando opera como un codificador, el bloque de imagen reconstruido filtrado, el bloque residual, y/o el bloque de predicción son almacenados en el componente 223 de memoria tampón de imágenes decodificadas para uso posterior en estimación de movimiento como se analizó antes. Cuando opera como un decodificador, el componente 223 de memoria tampón de imágenes decodificadas almacena y reenvía los bloques reconstruidos y filtrados hacia un dispositivo de visualización como parte de una señal de video de salida. El componente 223 de memoria tampón de imágenes decodificadas puede ser cualquier dispositivo de memoria capaz de almacenar bloques de predicción, bloques residuales, y/o bloques de imagen reconstruidos.

El componente 231 de CABAC y de formateo de cabecera recibe los datos desde los diversos componentes del sistema 200 de códec y codifica dichos datos en una corriente de bits codificada para transmisión hacia un decodificador. Específicamente, el componente 231 de CABAC y de formateo de cabecera genera varias cabeceras para codificar datos de control, tal como datos de control generales y datos de control de filtro. Además, datos de predicción, incluyendo datos de intra-predicción y movimiento, así como datos residuales en la forma de datos de coeficiente de transformada cuantificados son codificados en la corriente de bits. La corriente de bits final incluye toda la información deseada por el decodificador para reconstruir la señal 201 de video dividida original. Dicha información también puede incluir tablas de índice de modo de intra-

predicción (también referidas como tablas de mapeo de palabra código), definiciones de contextos de codificación para diversos bloques, indicaciones de modos de intra-predicción más probables, una indicación de información de división, etc. Dichos datos pueden ser codificados empleando codificación entrópica. Por ejemplo, la información puede ser codificada empleando codificación de longitud variable adaptativa de contexto (CAVLC), CABAC, codificación aritmética binaria adaptativa de contexto basada en sintaxis (SBAC), codificación entrópica de división por intervalo de probabilidad (PIPE), u otra técnica de codificación entrópica. Después de la codificación entrópica, la corriente de bits codificada puede ser transmitida a otro dispositivo (por ejemplo, un decodificador de video) o se puede archivar para posterior transmisión o recuperación.

La figura 3 es un diagrama de bloques que ilustra un codificador ejemplar 300 de video. El codificador 300 de video puede ser empleado para implementar las funciones de codificación del sistema 200 de códec y/o implementar las etapas 101, 103, 105, 107, y/o 109 del método operativo 100. El codificador 300 divide una señal de video de entrada, dando como resultado una señal 301 de video dividida, la cual es sustancialmente similar a la señal 201 de video dividida. La señal 301 de video dividida es luego comprimida y codificada en una corriente de bits por componentes del codificador 300.

Específicamente, la señal 301 de video dividida es reenviada a un componente 317 de predicción de intra-imagen para intra-predicción. El componente 317 de predicción de intra-imagen puede ser sustancialmente similar al componente 215 de estimación de intra-imagen y al componente 217 de predicción de intra-imagen. La señal 301 de video dividida también es reenviada a un componente 321 de compensación de movimiento para inter-predicción basada en bloques de referencia en un componente 323 de memoria tampón de imágenes decodificadas. El componente 321 de compensación de movimiento puede ser sustancialmente similar al componente 221 de estimación de movimiento y al componente 219 de compensación de movimiento. Los bloques de predicción y los bloques residuales del componente 317 de predicción de intra-imagen y del componente 321 de compensación de movimiento son reenviados a un componente 313 de transformada y cuantificación para transformar y cuantificar los bloques residuales. El componente 313 de transformada y cuantificación puede ser sustancialmente similar al componente 213 de cuantificación y escalado de transformada. Los bloques residuales y los bloques de predicción transformados y cuantificados correspondientes (junto con datos de control asociados) son reenviados a un componente 331 de codificación entrópica para codificación en una corriente de bits. El componente 331 de codificación entrópica puede ser sustancialmente similar al componente 231 de CABAC y de formateo de cabecera.

Los bloques residuales y/o los bloques de predicción transformados y cuantificados correspondientes también son reenviados desde el componente 313 de transformada y cuantificación a un componente 329 de transformada y cuantificación inversa para reconstrucción en bloques de referencia para uso por el componente 321 de compensación de movimiento. El componente 329 de transformada y cuantificación inversa puede ser sustancialmente similar al componente 229 de transformada inversa y escalado. Filtros en bucle en un componente 325 de filtros en bucle también son aplicados a los bloques residuales y/o bloques de referencia reconstruidos, dependiendo del ejemplo. El componente 325 de filtros en bucle puede ser sustancialmente similar al componente 227 de análisis de control de filtro y al componente 225 de filtros en bucle. El componente 325 de filtros en bucle puede incluir múltiples filtros como se analizó con respecto al componente 225 de filtros en bucle. Los bloques filtrados entonces son almacenados en un componente 323 de memoria tampón de imágenes decodificadas para uso como bloques de referencia por el componente 321 de compensación de movimiento. El componente 323 de memoria tampón de imágenes decodificadas puede ser sustancialmente similar al componente 323 de memoria tampón de imágenes decodificadas.

La figura 4 es un diagrama de bloques que ilustra un decodificador ejemplar 400 de video. El decodificador 400 de video puede ser empleado para implementar las funciones de decodificación del sistema 200 de códec y/o implementar las etapas 111, 113, 115, y/o 117 del método operativo 100. El decodificador 400 recibe una corriente de bits, por ejemplo, procedente de un codificador 300, y genera una señal de video de salida reconstruida basada en la corriente de bits para mostrar a un usuario final.

La corriente de bits es recibida por un componente 433 de decodificación entrópica. El componente 433 de decodificación entrópica está configurado para implementar un esquema de decodificación entrópica, tal como codificación CAVLC, CABAC, SBAC, PIPE, u otras técnicas de codificación entrópica. Por ejemplo, el componente 433 de decodificación entrópica puede emplear información de cabecera para proporcionar un contexto para interpretar datos adicionales codificados como palabras código en la corriente de bits. La información decodificada incluye cualquier información deseada para decodificar la señal de video, tal como datos de control general, datos de control de filtro, información de división, datos de movimiento, datos de predicción, y coeficientes de transformada cuantificados a partir de bloques residuales. Los coeficientes de transformada cuantificados son reenviados a un componente 429 de transformada y cuantificación inversa para reconstrucción en bloques residuales. El componente 429 de transformada y cuantificación inversa puede ser similar al componente 329 de transformada y cuantificación inversa.

Los bloques residuales y/o los bloques de predicción reconstruidos son reenviados al componente 417 de

predicción de intra-imagen para reconstrucción en bloques de imagen basándose en operaciones de intra-predicción. El componente 417 de predicción de intra-imagen puede ser similar al componente 215 de estimación de intra-imagen y a un componente 217 de predicción de intra-imagen. Específicamente, el componente 417 de predicción de intra-imagen emplea modos de predicción para localizar un bloque de referencia en el fotograma y aplica un bloque residual al resultado para reconstruir bloques de imagen intra-predichos. Los bloques de imagen intra-predichos reconstruidos y/o los bloques residuales y los datos de inter-predicción correspondientes son reenviados a un componente 423 de memoria tampón de imágenes decodificadas mediante un componente 425 de filtros en bucle, el cual puede ser sustancialmente similar al componente 223 de memoria tampón de imágenes decodificadas y al componente 225 de filtros en bucle, respectivamente. El componente 425 de filtros en bucle filtra los bloques de imagen reconstruidos, los bloques residuales y/o los bloques de predicción, y dicha información es almacenada en el componente 423 de memoria tampón de imágenes decodificadas. Los bloques de imagen reconstruidos del componente 423 de memoria tampón de imágenes decodificadas son reenviados a un componente 421 de compensación de movimiento para inter-predicción. El componente 421 de compensación de movimiento puede ser sustancialmente similar al componente 221 de estimación de movimiento y/o al componente 219 de compensación de movimiento. Específicamente, el componente 421 de compensación de movimiento emplea vectores de movimiento de un bloque de referencia para generar un bloque de predicción y aplica un bloque residual al resultado para reconstruir un bloque de imagen. Los bloques reconstruidos resultantes también pueden ser reenviados a través del componente 425 de filtros en bucle al componente 423 de memoria tampón de imágenes decodificadas. El componente 423 de memoria tampón de imágenes decodificadas continúa almacenando bloques de imagen reconstruidos adicionales, que pueden ser reconstruidos en fotogramas mediante la información de división. Dichos fotogramas también pueden ser colocados en una secuencia. La secuencia es emitida hacia un dispositivo de visualización como una señal de video de salida reconstruida.

La figura 5 es un diagrama esquemático que ilustra una CVS ejemplar 500. Por ejemplo, la CVS 500 puede ser codificada por un codificador, tal como un sistema 200 de códec y/o codificador 300, según el método 100. Además, la CVS 500 puede ser decodificada por un decodificador, tal como el sistema 200 de códec y/o decodificador 400. La CVS 500 incluye imágenes codificadas en un orden 508 de decodificación. Un orden 508 de decodificación es un orden en el cual las imágenes son posicionadas en una corriente de bits. Las imágenes de la CVS 500 son luego emitidas en un orden 510 de presentación. Un orden 510 de presentaciones un orden en el cual las imágenes deberían ser mostradas por un decodificador para hacer que el video resultante sea mostrado de manera apropiada. Por ejemplo, las imágenes de la CVS 500 generalmente pueden ser posicionadas en el orden 510 de presentación. Sin embargo, algunas imágenes pueden ser movidas a diferentes ubicaciones para aumentar la eficiencia de la codificación, por ejemplo colocando imágenes similares en estrecha proximidad para soportar la inter-predicción. El movimiento de dichas imágenes de esta manera da como resultado el orden 508 de decodificación. En el ejemplo mostrado, las imágenes son indexadas en el orden 508 de decodificación de cero a cuatro. En el orden 510 de presentación, las imágenes en índice dos e índice tres han sido movidas en frente de la imagen en el índice cero.

La CVS 500 incluye una imagen IRAP 502. Una imagen IRAP 502 es una imagen codificada según intra-predicción que sirve como un punto de acceso aleatorio para la CVS 500. Específicamente, bloques de la imagen IRAP 502 son codificados por referencia a otros bloques de imagen IRAP 502. Debido a que la imagen IRAP 502 es codificada sin referencia a otras imágenes, la imagen IRAP 502 puede ser decodificada sin primero decodificar ninguna otra imagen. Por consiguiente, un decodificador puede comenzar a decodificar la CVS 500 en la imagen IRAP 502. Además, la imagen IRAP 502 puede hacer que el DPB sea refrescado. Por ejemplo, imágenes presentadas después de una imagen IRAP 502 no pueden basarse en imágenes principales a la imagen IRAP 502 (por ejemplo, índice de imagen cero) para inter-predicción. Como tal, la memoria tampón de imágenes puede ser actualizada una vez que la imagen IRAP 502 es decodificada. Esto tiene el efecto de detener cualesquiera errores de codificación relacionados con la inter-predicción debido a que dichos errores no pueden propagarse a través de la imagen IRAP 502. La imagen IRAP 502 puede incluir varios tipos de imágenes. Por ejemplo, una imagen IRAP puede ser codificada como una IDR o una CRA. Una IDR es una imagen intra-codificada que comienza una nueva CVS 500 y actualiza la memoria tampón de imágenes. Una CRA es una imagen intra-codificada que actúa como un punto de acceso aleatorio sin comenzar una nueva CVS 500 o actualizar la memoria tampón de imágenes. De esta manera, imágenes principales 504 asociadas con una CRA pueden hacer referencia a imágenes principales a la CRA, mientras que imágenes principales 504 asociadas con una IDR no pueden hacer referencia a imágenes principales a la IDR.

La CVS 500 también incluye varias imágenes no-IRAP. Estas incluyen imágenes principales 504 e imágenes finales 506. Una imagen principal 504 es una imagen posicionada después de la imagen IRAP 502 en orden 508 de decodificación, pero posicionada antes que la imagen IRAP 502 en orden 510 de presentación. Las imágenes finales 506 son colocadas después de la imagen IRAP 502 en el orden 508 de decodificación y en el orden 510 de presentación. Las imágenes principales 504 y las imágenes finales 506 son codificadas según inter-predicción. Las imágenes finales 506 son codificadas con referencia a la imagen IRAP 502 o

imágenes posicionadas después de la imagen IRAP 502. Por lo tanto, las imágenes finales 506 siempre pueden ser decodificadas una vez que la imagen IRAP 502 es decodificada. Las imágenes principales 504 pueden incluir imágenes principales omitidas de acceso aleatorio (RASL) y principales decodificables de acceso aleatorio (RADL). Una imagen RASL es codificada por referencia a imágenes principales a la imagen IRAP 502, pero codificada en una posición después de la imagen IRAP 502. Debido a que las imágenes RASL se basan en imágenes principales, la imagen RASL no puede ser decodificada cuando el decodificador comienza a decodificar en la imagen IRAP 502. Por consiguiente, las imágenes RASL son omitidas y no decodificadas cuando la imagen IRAP 502 es utilizada como un punto de acceso aleatorio. Sin embargo, las imágenes RASL son decodificadas y mostradas cuando el decodificador utiliza una imagen IRAP principal (antes del índice cero y no mostrada) como el punto de acceso aleatorio. Las imágenes RADL son codificadas con referencia a la imagen IRAP 502 y/o a las imágenes que siguen a la imagen IRAP 502, pero son posicionadas antes de la imagen IRAP 502 en el orden 510 de presentación. Debido a que las imágenes RADL no se basan en imágenes previas a la imagen IRAP 502, las imágenes RADL pueden ser decodificadas y mostradas cuando la imagen IRAP 502 es el punto de acceso aleatorio.

Las imágenes procedentes de la CVS 500 pueden ser almacenadas, cada una, en una unidad de acceso. Además, las imágenes pueden ser divididas en segmentos, y los segmentos pueden ser incluidos en unidades NAL. Una unidad NAL es una unidad de almacenamiento que contiene un conjunto de parámetros o un segmento de una imagen y una cabecera de segmento correspondiente. A las unidades NAL se les asignan tipos para indicar al decodificador el tipo de datos contenidos en la unidad NAL. Por ejemplo, segmentos de una imagen IRAP 502 pueden ser contenidos en una IDR con unidad NAL RADL (IDR_W_RADL), una unidad IDR sin unidad NAL de imágenes principales (IDR_N_LP), una unidad CRA NAL etc. La unidad NAL IDR_W_RADL indica que la imagen IRAP 502 es una imagen IDR que está asociada con una imagen principal RADL 504. La unidad NAL IDR_N_LP indica que la imagen IRAP 502 es una imagen IDR que no está asociada con ninguna imagen principal 504. La unidad NAL CRA indica que la imagen IRAP 502 es una imagen CRA que puede estar asociada con imágenes principales 504. Los segmentos de imágenes no-IRAP también pueden ser colocados en unidades NAL. Por ejemplo, los segmentos de las imágenes finales 506 pueden ser colocados en un tipo de unidad NAL de imagen final (TRAIL_NUT), lo cual indica que las imágenes finales 506 son imágenes codificadas de inter-predicción. Los segmentos de las imágenes principales 504 pueden ser incluidos en un tipo de unidad RASL NAL (RASL_NUT) y/o un tipo de unidad NAL RADL (RADL_NUT), lo cual puede indicar que las imágenes correspondientes son imágenes principales 504 codificadas de inter-predicción de los tipos correspondientes. Al señalar los segmentos de las imágenes en unidades NAL correspondientes, el decodificador fácilmente puede determinar los mecanismos de decodificación apropiados para aplicar a cada imagen/segmento.

La figura 6 es un diagrama esquemático que ilustra una pluralidad de corrientes 601, 602, y 603 de video de sub-imagen divididas a partir de una corriente 600 de video de imagen de VR. Por ejemplo, cada una de las corrientes 601- 603 de video de sub-imagen y/o la corriente 600 de video de imagen de VR pueden ser codificadas en una CVS 500. Por consiguiente, las corrientes 601- 603 de video de sub-imagen y/o la corriente 600 de video de imagen de VR pueden ser codificadas por un codificador, tal como el sistema 200 de códec y/o codificador 300, según el método 100. Además, las corrientes 601-603 de video de sub-imagen y/o la corriente 600 de video de imagen de VR pueden ser decodificadas por un decodificador, tal como el sistema 200 de códec y/o decodificador 400.

Una corriente 600 de video de imágenes de VR incluye una pluralidad de imágenes presentadas a lo largo del tiempo. Específicamente, la VR opera codificando una esfera de contenido de video, la cual puede ser mostrada como si el usuario estuviese en el centro de la esfera. Cada imagen incluye la esfera completa. Mientras tanto, solamente una porción de la imagen, conocida como un puerto de visualización, es mostrada al usuario. Por ejemplo, el usuario puede emplear un dispositivo de visualización montado en la cabeza (HMD) que selecciona y muestra un puerto de visualización de la esfera con base en el movimiento de cabeza del usuario. Esto proporciona la impresión de estar físicamente presente en un espacio virtual tal como se muestra mediante el video. A fin de lograr este resultado, cada imagen de la secuencia de video incluye una esfera completa de datos de video en un instante en tiempo correspondiente. Sin embargo, solamente una pequeña porción (por ejemplo, un solo puerto de visualización) de la imagen es mostrada al usuario. El resto de la imagen es desechado sin ser representado. Generalmente se transmite toda la imagen de manera que un puerto de visualización diferente puede ser dinámicamente seleccionado y mostrado en respuesta al movimiento de cabeza de los usuarios.

En el ejemplo mostrado, las imágenes de la corriente 600 de video de imágenes de VR pueden ser, cada una, sub-divididas en sub-imágenes basadas en puertos de visualización disponibles. Por consiguiente, cada imagen y sub-imagen correspondiente incluye una posición temporal (por ejemplo, orden de imagen) como parte de la presentación temporal. Corrientes 601-603 de video de sub-imagen son creadas cuando la sub-división es aplicada consistentemente a lo largo del tiempo. Dicha sub-división consistente crea corrientes 601-603 de video de sub-imágenes donde cada corriente contiene un conjunto de sub-imágenes de un tamaño, forma y posición espacial predeterminados con relación a imágenes correspondientes en la corriente 600 de video de imagen de VR. Además, el conjunto de sub-imágenes en una corriente 601-603 de video de

sub-imágenes varía en posición temporal a lo largo del tiempo de presentación. Como tal, las sub-imágenes de las corrientes 601-603 de video de sub-imagen pueden ser alineadas en el dominio de tiempo basándose en la posición temporal. Después las sub-imágenes de las corrientes 601-603 de video de sub-imágenes en cada posición temporal pueden ser fusionadas en el dominio espacial basado en la posición espacial predefinida para reconstruir la corriente 600 de video de imagen de VR para mostrarla. Específicamente, las corrientes 601-603 de video de sub-imágenes pueden ser codificadas en sub-corrientes de bits separadas. Cuando dichas sub-corrientes de bits son fusionadas juntas, estas tienen como resultado una corriente de bits que incluye todo el conjunto de imágenes a lo largo del tiempo. La corriente de bits resultante puede ser transmitida hacia el decodificador para decodificación y presentación basada en el puerto de visualización actualmente seleccionado del usuario.

Uno de los problemas con el video de VR es que todas las corrientes 601-603 de video de sub-imágenes pueden ser transmitidas a un usuario a una alta calidad (por ejemplo, alta resolución). Esto permite que el decodificador seleccione dinámicamente el puerto de visualización actual del usuario y muestre las sub-imágenes de las corrientes 601-603 de video de sub-imágenes correspondientes en tiempo real. Sin embargo, el usuario solamente puede ver un solo puerto de visualización, por ejemplo a partir de la corriente 601 de video de sub-imágenes, mientras que las corrientes 602-603 de video de sub-imágenes son desechadas. Debido a esto, la transmisión de corrientes 602-603 de video de sub-imágenes a una alta calidad puede desperdiciar una cantidad importante de ancho de banda. A fin de mejorar la eficiencia de la codificación, el video de VR puede ser codificado en una pluralidad de corrientes 600 de video donde cada corriente 600 de video es codificada a una calidad/resolución diferente. De esta manera, el decodificador puede transmitir una solicitud para una corriente 601 de video de sub-imágenes actual. En respuesta, el codificador (o un segmentador intermedio u otro servidor de contenido) puede seleccionar la corriente 601 de video de sub-imágenes de calidad superior a partir de la corriente 600 de video de calidad superior y las corrientes 602-603 de video de sub-imágenes de calidad inferior a partir de la corriente 600 de video de calidad inferior. El codificador entonces puede fusionar dichas sub-corrientes de bits juntas en una corriente de bits codificada completa para transmisión al decodificador. De esta manera, el decodificador recibe una serie de imágenes donde el puerto de visualización actual es de calidad superior y los otros puertos de visualización son de calidad inferior. Además, las sub-imágenes de calidad más alta generalmente son mostradas al usuario (movimiento de cabeza ausente) y las sub-imágenes de calidad inferior generalmente son desechadas, lo cual equilibra la funcionalidad con la eficiencia de la codificación.

En caso de que el usuario se gire para ver desde la corriente 601 de video de sub-imágenes a la corriente 602 de video de sub-imágenes, el decodificador solicita que la nueva corriente 602 de video de sub-imágenes actual sea transmitida a la calidad superior. El codificador entonces puede alterar el mecanismo de fusión consecuentemente. Como se observó antes, un decodificador solo puede comenzar a decodificar una nueva CVS 500 en una imagen IRAP 502. Por consiguiente, la corriente 602 de video de sub-imágenes es mostrada a la calidad inferior hasta que se alcanza una imagen/sub-imagen IRAP. La imagen IRAP entonces puede ser decodificada a la calidad superior para comenzar la decodificación de la versión de calidad superior de la corriente 602 de video de sub-imágenes. Este enfoque aumenta significativamente la compresión de video sin impactar negativamente la experiencia de visualización del usuario.

Una preocupación con el enfoque antes mencionado es que el período de tiempo necesario para cambiar las resoluciones se basa en el período de tiempo hasta que se alcanza una imagen IRAP en la corriente de video. Esto se debe a que el decodificador no puede comenzar a decodificar una versión diferente de corriente 602 de video de sub-imagen en una imagen no-IRAP. Un enfoque para reducir dicha latencia es incluir más imágenes IRAP. Sin embargo, esto da como resultado un aumento del tamaño de archivo. A fin de equilibrar la funcionalidad con la eficiencia de la codificación, diferentes puertos de visualización/corrientes 601-603 de video de sub-imágenes pueden incluir imágenes IRAP a diferentes frecuencias. Por ejemplo, puertos de visualización/corrientes 601-603 de video de sub-imágenes que tienen más probabilidad de ser vistos pueden tener más imágenes IRAP que otros puertos de visualización/corrientes 601-603 de video de sub-imágenes. Por ejemplo, en un contexto de baloncesto, los puertos de visualización/corrientes 601-603 de video de sub-imágenes relacionados con las canastas y/o con la cancha central pueden incluir imágenes IRAP a una mayor frecuencia que los puertos de visualización/corrientes 601-603 de video de sub-imágenes que observan los soportes o el techo debido a que esos puertos de visualización/corrientes 601-603 de video de sub-imágenes tienen menos probabilidad de ser vistos por el usuario.

Este enfoque conduce a problemas adicionales. Específicamente, las sub-imágenes de las corrientes 601-603 de video de sub-imágenes que comparten un POC son parte de una sola imagen. Como se ha observado antes, segmentos procedentes de una imagen están incluidos en una unidad NAL basada en el tipo de imagen. En algunos sistemas de codificación de video, todas las unidades NAL relacionadas con una sola imagen están restringidas a incluir el mismo tipo de unidad NAL. Cuando diferentes corrientes 601-603 de video de sub-imágenes tienen imágenes IRAP a diferentes frecuencias, algunas de las imágenes incluyen tanto sub-imágenes IRAP como sub-imágenes no-IRAP. Esto infringe la restricción de que cada única imagen debería emplear solamente unidades NAL del mismo tipo.

La presente descripción aborda este problema eliminando la restricción de que todas las unidades NAL para segmentos en una imagen empleen el mismo tipo de unidad NAL. Por ejemplo, una imagen es incluida en una unidad de acceso. Al eliminar esta restricción, una unidad de acceso puede incluir tanto tipos de unidad NAL IRAP como tipos de unidad NAL no-IRAP. Además, un indicador es codificado para indicar cuando una imagen/unidad de acceso incluye una mezcla de tipos de unidad NAL IRAP y tipos de unidad NAL no-IRAP. En algunos ejemplos, el indicador es un indicador de tipos de unidad NAL de imágenes mezcladas (mixed_nalu_types_in_pic_flag). Además, se aplica una restricción para requerir que una sola imagen mezclada/unidad de acceso pueda contener solamente un tipo de unidad NAL IRAP y un tipo de unidad NAL no-IRAP. Esto impide que ocurran mezclas no intencionadas de tipos de unidad NAL. Si dichas mezclas estuviesen permitidas, el decodificador tendría que ser diseñado para gestionar dichas mezclas. Esto aumentaría innecesariamente la complejidad requerida del hardware sin proporcionar un beneficio adicional al proceso de codificación. Por ejemplo, la imagen mezclada puede incluir un tipo de unidad NAL IRAP seleccionado de entre IDR_W_RADL, IDR_N_LP, o CRA_NUT. Además, la imagen mezclada puede incluir un tipo de unidad NAL no-IRAP seleccionada de entre TRAIL_NUT, RADL_NUT, y RASL_NUT. Implementaciones ejemplares de este esquema se analizan con más detalle a continuación.

La figura 7 es un diagrama esquemático que ilustra un ejemplo de la corriente 700 de bits que contiene imágenes con tipos de unidad NAL mezclados. Por ejemplo, la corriente 700 de bits puede ser generada por un sistema 200 de códec y/o un codificador 300 para decodificación por un sistema 200 de códec y/o un decodificador 400 según el método 100. Además, la corriente 700 de bits puede incluir la corriente 600 de video de imagen de VR fusionada de entre múltiples corrientes 601-603 de video de sub-imagen a una pluralidad de resoluciones de video donde cada corriente de video de sub-imagen contiene una CVS 500 en una posición espacial diferente.

La corriente 700 de bits incluye un conjunto de parámetros de secuencia (SPS) 710, una pluralidad de conjuntos de parámetros de imagen (PPS) 711, una pluralidad de cabeceras 715 de segmento, y datos 720 de imagen. Un SPS 710 contiene datos de secuencia comunes a todas las imágenes en la secuencia de video contenida en la corriente 700 de bits. Dichos datos pueden incluir dimensionamiento de imagen, profundidad de bit, parámetros de herramientas de codificación, restricciones de tasa de bits, etc. El PPS 711 contiene parámetros que se aplican a una imagen completa. Por lo tanto, cada imagen en la secuencia de video se puede referir a un PPS 711. Se debería observar que, aunque cada imagen se refiere a un PPS 711, un solo PPS 711 puede contener datos para múltiples imágenes en algunos ejemplos. Por ejemplo, múltiples imágenes similares pueden ser codificadas según parámetros similares. En dicho caso, un solo PPS 711 puede contener datos para dichas imágenes similares. El PPS 711 puede indicar herramientas de codificación disponibles para segmentos en imágenes correspondientes, parámetros de cuantificación, compensaciones, etc. La cabecera 715 de segmento contiene parámetros que son específicos para cada segmento en una imagen. Por lo tanto, puede haber una cabecera 715 de segmento por segmento en la secuencia de video. La cabecera 715 de segmento puede contener información de tipo de segmento, recuentos de orden de imagen (POC), listas de imágenes de referencia, pesos de predicción, puntos de entrada de mosaico, parámetros de desbloqueo, etc. Se debería observar que una cabecera 715 de segmento también puede ser denominada como una cabecera de grupo de mosaicos en algunos contextos.

Los datos 720 de imagen contienen datos de video codificados según inter-predicción y/o intra-predicción así como datos residuales transformados y cuantificados correspondientes. Por ejemplo, una secuencia de video incluye una pluralidad de imágenes 721 codificadas como datos 720 de imagen. Una imagen 721 es un solo fotograma de una secuencia de video y por lo tanto generalmente es mostrada como una sola unidad cuando se muestra la secuencia de video. Sin embargo, se pueden mostrar sub-imágenes 723 para implementar algunas tecnologías tales como realidad virtual. Las imágenes 721 hacen referencia, cada una, a un PPS 711. Las imágenes 721 pueden ser divididas en sub-imágenes 723, mosaicos, y/o segmentos. Una sub-imagen 723 es una región espacial de una imagen 721 que es aplicada de manera consistente sobre una secuencia de video codificada. Por consiguiente, una sub-imagen 723 puede ser mostrada por un HMD en un contexto de VR. Además, una sub-imagen 723 con un POC especificado se puede obtener de una corriente 601-603 de video de sub-imagen a una resolución correspondiente. Una sub-imagen 723 puede hacer referencia al SPS 710. En algunos sistemas, los segmentos 725 son referidos como grupos de mosaicos que contienen mosaicos. Los segmentos 725 y/o grupos de mosaicos se refieren a una cabecera 715 de segmento. Un segmento 725 puede ser definido como un número entero de mosaicos completos o un número entero de filas de CTU completas consecutivas dentro de un mosaico de una imagen 721 que están exclusivamente contenidas en una sola unidad NAL. Por lo tanto, los segmentos 725 además son divididos en CTU y/o CTB. Los CTU/CTB son divididos adicionalmente en bloques de codificación basados en árboles de codificación. Los bloques de codificación entonces pueden ser codificados/decodificados según mecanismos de predicción.

Los conjuntos de parámetros y/o segmentos 725 son codificados en unidades NAL. Una unidad NAL puede ser definida como una estructura de sintaxis que contiene una indicación del tipo de datos a seguir y los bytes que contienen esos datos en la forma de un RBSP intercalado según sea necesario con bytes de prevención de emulación. De manera más específica, una unidad NAL es una unidad de almacenamiento que contiene

un conjunto de parámetros o un segmento 725 de una imagen 721 y una cabecera 715 de segmento correspondiente. Específicamente, unidades NAL VCL 740 son unidades NAL que contienen un segmento 725 de una imagen 721 y una cabecera 715 de segmento correspondiente. Además, unidades NAL no-VCL 730 contienen conjuntos de parámetros, tales como el SPS 710 y el PPS 711. Se pueden emplear varios tipos de unidades NAL. Por ejemplo, el SPS 710 y el PPS 711 pueden ser incluidos en un tipo de unidad NAL SPS (SPS_NUT) 731 y un tipo de unidad NAL PPS (PPS_NUT) 732, respectivamente, que son ambas unidades NAL no-VCL 730.

Como se ha observado antes, imágenes IRAP, tales como la imagen IRAP 502, pueden estar contenidas en unidades NAL IRAP 745. Imágenes no-IRAP, tales como las imágenes principales 504 y las imágenes finales 506, pueden estar incluidas en unidades NAL no-IRAP 749. Específicamente, una unidad NAL IRAP 745 es cualquier unidad NAL que contiene un segmento 725 tomado de una imagen o sub-imagen IRAP. Una unidad NAL no-IRAP 749 es cualquier unidad NAL que contiene un segmento 725 tomado de cualquier imagen que no sea una imagen o sub-imagen IRAP (por ejemplo, imágenes principales o imágenes finales). Unidades NAL IRAP 745 y unidades NAL no-IRAP 749 son ambas unidades NAL VCL 740 debido a que ambas contienen datos de segmento. En una realización ejemplar, una unidad NAL IRAP 745 puede incluir segmentos 725 procedentes de una imagen IDR sin imágenes principales o un IDR asociado con imágenes RADL en una unidad NAL IDR_N_LP 741 o una unidad NAL IDR_W_RADL 742, respectivamente. Además, una unidad NAL IRAP 745 puede incluir segmentos 725 de una imagen CRA en una CRA_NUT 743. En una realización ejemplar, una unidad NAL no-IRAP 749 puede incluir segmentos 725 procedentes de una imagen RASL, una imagen RADL, o una imagen final, en una RASL_NUT 746, una RADL_NUT 747, o una TRAIL_NUT 748, respectivamente. En una realización ejemplar, una lista complete de posibles unidades NAL se muestra a continuación como clasificada por tipo de unidad NAL.

nal_unit_type	Nombre de nal_unit_type	Contenido de unidad NAL y estructura de sintaxis Rbsp	Clase de tipo de unidad NAL
0	TRAIL_NUT	Segmento codificado de una imagen final slice_layer_rbsp()	VCL
1	STSA_NUT	Segmento codificado de una imagen STSA slice_layer_rbsp()	VCL
2	RADL_NUT	Segmento codificado de una imagen RADL slice_layer_rbsp()	VCL
3	RASL_NUT	Segmento codificado de una imagen RASL slice_layer_rbsp()	VCL
4..6	RSV_VCL_4.. RSV_VCL_6	Tipos de unidad NAL VCL no-IRAP reservados	VCL
7	IDR_W_RADL	Segmento codificado de una imagen IDR	VCL
8	IDR_N_LP	slice_layer_rbsp()	
9	CRA_NUT	Segmento codificado de una imagen CRA slice_layer_rbsp()	VCL
10	GDR_NUT	Segmento codificado de una imagen GDR slice_layer_rbsp()	VCL
11	RSV_IRAP_11	Tipos de unidad NAL VCL IRAP reservados	VCL
12	RSV_IRAP_12		
13	DCI_NUT	Información de capacidad de decodificación decoding_capability_information_rbsp()	no-VCL
14	VPS_NUT	Conjunto de parámetros de video video_parameter_set_rbsp()	no-VCL
15	SPS_NUT	Conjunto de parámetros de secuencia seq_parameter_set_rbsp()	no-VCL
16	PPS_NUT	Conjunto de parámetros de imagen pic_parameter_set_rbsp()	no-VCL
17	PREFIX_APS_NUT	Conjunto de parámetros de adaptación	no-VCL
18	SUFFIX_APS_NUT	adaptation_parameter_set_rbsp()	
19	PH_NUT	Cabecera de imagen picture_header_rbsp()	no-VCL
20	AUD_NUT	Delimitador de AU access_unit_delimiter_rbsp()	no-VCL
21	EOS_NUT	Fin de secuencia end_of_seq_rbsp()	no-VCL
22	EOB_NUT	Fin de corriente de bits end_of_bitstream_rbsp()	no-VCL
23	PREFIX_SET_NUT	Información de mejora complementaria	no-VCL
24	SUFFIX_SEI_NUT	sei_rbsp()	
25	FD_NUT	Datos de relleno filler_data_rbsp()	no-VCL
26	RSV_NVCL_26	Tipos de unidad NAL no-VCL reservados	no-VCL
27	RSV_NVCL_27		

28..31	UNSPEC_28.. UNSPEC_31	Tipos de unidad NAL no-VCL no especificados	no-VCL
--------	--------------------------	---	--------

5 Como se ha observado antes, una corriente de video de VR puede incluir sub-imágenes 723 con imágenes IRAP a diferentes frecuencias. Esto permite que se empleen menos imágenes IRAP para regiones espaciales que un usuario tiene pocas probabilidades de observar y que se empleen más imágenes IRAP para regiones espaciales que un usuario tiene probabilidades de ver a menudo. De esta manera, las regiones espaciales a las que el usuario tiene probabilidad de volver con regularidad pueden ser rápidamente ajustadas a una resolución superior. Cuando este enfoque da como resultado una imagen 721 que incluye tanto unidades NAL IRAP 745 como unidades NAL no-IRAP 749, la imagen 721 es denominada como una imagen mezclada. Esta condición puede ser señalizada por tipos de unidad NAL mezclados en el indicador de imagen (mixed_nalu_types_in_pic_flag) 727. El mixed_nalu_types_in_pic_flag 727 puede ser establecido en el PPS 711. Además, el mixed_nalu_types_in_pic_flag 727 puede ser establecido igual a uno cuando se especifica que cada imagen 721 que hace referencia al PPS 711 tiene más de una unidad NAL VCL 740 y las unidades NAL VCL 740 no tienen el mismo valor de tipo de unidad NAL (nal_unit_type). Además, el mixed_nalu_types_in_pic_flag 727 puede ser establecido igual a cero cuando cada imagen 721 que hace referencia al PPS 711 tiene una o más unidades NAL VCL 740 y las unidades NAL VCL 740 de cada imagen 721 que hace referencia al PPS 711 tienen el mismo valor de nal_unit_type.

20 Además, se puede emplear una restricción de manera que las unidades NAL VCL 740 de una o más de las sub-imágenes 723 de la imagen 721 tienen todas un primer valor particular de tipo de unidad NAL y otras unidades NAL VCL 740 en la imagen 721 tienen un segundo valor particular diferente de tipo de unidad NAL cuando se establece el mixed_nalu_types_in_pic_flag 727. En particular, la restricción requiere que una imagen mezclada 721 contenga un solo tipo de unidad NAL IRAP 745 y un solo tipo de unidad NAL no-IRAP 749. Por ejemplo, la imagen 721 puede incluir una o más unidades NAL IDR_N_LP 741, una o más unidades NAL IDR_w_RADL 742, o una o más CRA_NUT 743, pero no cualquier combinación de dichas unidades NAL IRAP 745. Además, la imagen 721 puede incluir una o más RASL_NUT 746, una o más RADL_NUT 747, o una o más TRAIL_NUT 748, pero no cualquier combinación de dichas unidades NAL IRAP 745.

30 En una implementación ejemplar, se emplean tipos de imagen para definir procesos de decodificación. Dichos procesos incluyen derivación de identificación de imagen, por ejemplo por recuento de orden de imagen (POC), marcado de estatus de imágenes de referencia en la memoria tampón de imágenes decodificadas (DPB), salida de imágenes desde la DPB, etc. Una imagen puede ser identificada por tipo basada en el tipo de unidad NAL que contiene toda la imagen codificada o una sub-porción de la misma. En algunos sistemas de codificación de video, tipos de imagen pueden incluir imágenes de actualizar decodificación instantánea de imágenes (IDR) e imágenes no-IDR. En otros sistemas de codificación de video, tipos de imagen pueden incluir imágenes finales, imágenes de acceso de sub-capa temporal (TSA), imágenes de acceso de sub-capa temporal escalonada (STSA), imágenes principales decodificables de acceso aleatorio (RADL), imágenes principales omitidas de acceso aleatorio (RASL), imágenes de acceso de enlace roto (BLA), imágenes de acceso aleatorio instantáneo, e imágenes de acceso aleatorio limpio. Dichos tipos de imagen pueden ser adicionalmente diferenciados basándose en si la imagen es una imagen referenciada de sub-capa o una imagen no-referenciada de sub-capa. Una imagen BLA puede ser diferenciada adicionalmente como BLA con imagen principal, BLA con imagen RADL, y BLA sin imagen principal. Una imagen IDR puede ser diferenciada adicionalmente como una imagen IDR con RADL y una IDR sin imagen principal.

45 Dichos tipos de imagen pueden ser empleados para implementar distintas funciones relacionadas con video. Por ejemplo, imágenes IDR, BLA, y/o CRA pueden ser empleadas para implementar una imagen IRAP. Una imagen IRAP puede proporcionar las siguientes funcionalidades/beneficios. La presencia de una imagen IRAP puede indicar que un proceso de decodificación puede ser iniciado desde esa imagen. Esta funcionalidad permite la implementación de una característica de acceso aleatorio en la cual un proceso de decodificación comienza en una posición designada en una corriente de bits siempre y cuando una imagen IRAP esté presente en esa posición. Dicha posición no es necesaria al comienzo de una corriente de bits. La presencia de una imagen IRAP también actualiza un proceso de decodificación de manera que imágenes codificadas que comienzan en la imagen IRAP, excluyendo imágenes RASL, son codificadas sin ninguna referencia a imágenes posicionadas antes de la imagen IRAP. Por consiguiente, una imagen IRAP colocada en una corriente de bits detiene la propagación de errores de decodificación. Por lo tanto, errores de decodificación de imágenes codificadas posicionadas antes de la imagen IRAP no se pueden propagar a través de la imagen IRAP y hacia dentro de las imágenes que siguen a la imagen IRAP en orden de decodificación.

60 Las imágenes IRAP proporcionan varias funcionalidades, pero crean una penalidad para una eficiencia de compresión. Por consiguiente, la presencia de una imagen IRAP puede provocar una subida en la tasa de bits. Esta penalidad a la eficiencia de compresión tiene varias causas. Por ejemplo, una imagen IRAP es una imagen intra-predicha que es representada significativamente por más bits que las imágenes inter-predichas

utilizadas como imágenes no-IRAP. Además, la presencia de una imagen IRAP rompe la predicción temporal utilizada en inter-predicción. Específicamente, la imagen IRAP actualiza el proceso de decodificación eliminando imágenes de referencia principales procedentes de la DPB. La eliminación de imágenes previas de referencia reduce la disponibilidad de imágenes de referencia para usar en la codificación de imágenes que siguen a la imagen IRAP en orden de decodificación, y por lo tanto reduce la eficiencia de este proceso.

Las imágenes IDR pueden emplear diferentes procesos de señalización y derivación de otros tipos de imagen IRAP. Por ejemplo, procesos de señalización y derivación relacionados con IDR pueden establecer la parte del bit más significativo (MSB) del POC a cero en lugar de derivar el MSB desde una imagen clave previa. Además, una cabecera de segmento de una imagen IDR no puede contener información utilizada para ayudar en la gestión de imagen de referencia. Mientras tanto, otros tipos de imagen, tal como CRA, final, TSA, etc., pueden contener información de imagen de referencia tal como un conjunto de imágenes de referencia (RPS) o una lista de imágenes de referencia, que pueden ser empleadas para implementar el proceso de marcado de imágenes de referencia. El proceso de marcado de imágenes de referencia es el proceso de determinar el estatus de imágenes de referencia en la DPB ya sea utilizada para referencia o no utilizada para referencia. Para imágenes IDR, dicha información no puede ser señalizada debido a que la presencia de IDR indica que el proceso de decodificación simplemente marcará todas las imágenes de referencia en la DPB como no utilizadas para referencia.

Además de los tipos de imagen, también se emplea la identificación de imagen por POC para múltiples propósitos, tal como para gestión de uso de imágenes de referencia en inter-predicción, para emisión de imágenes desde la DPB, para escalado de vectores de movimiento, para predicción ponderada, etc. Por ejemplo, en algunos sistemas de codificación de video, las imágenes en la DPB pueden ser marcadas como utilizadas para referencia a corto plazo, utilizadas para referencia a largo plazo, o no utilizadas para referencia. Una vez que una imagen ha sido marcada como no utilizada para referencia, la imagen ya no puede ser utilizada para predicción. Cuando dicha imagen ya no es necesaria para emitir la imagen puede ser eliminada de la DPB. En otros sistemas de codificación de video, las imágenes de referencia pueden ser marcadas como de corto plazo y largo plazo. Una imagen de referencia puede ser marcada como no utilizada para referencia cuando la imagen ya no es necesaria para referencia de predicción. La conversión entre estos estatus puede ser controlada mediante el proceso de marcación de imágenes de referencia decodificadas. Un proceso de ventana de deslizamiento implícito y/o un proceso de operación de control de gestión de memoria (MMCO) explícito pueden ser empleados como mecanismos de marcación de imagen de referencia decodificada. El proceso de ventana deslizante marca una imagen de referencia de corto plazo como no utilizada para referencia cuando el número de fotogramas de referencia es igual a un número máximo especificado denotado como `max_num_ref_frames` en el SPS. Las imágenes de referencia de corto plazo pueden ser almacenadas en una manera de primera-en-entrar, primera-en-salir de manera que las imágenes de corto plazo más recientemente decodificadas son retenidas en la DPB. El proceso MMCO explícito puede incluir múltiples comandos MMCO. Un comando MMCO puede marcar una o más imágenes de referencia de corto plazo o largo plazo como no utilizadas para referencia, puede marcar todas las imágenes como no utilizadas para referencia, o puede marcar la imagen de referencia actual o una imagen de referencia de corto plazo existente como de largo plazo y asignar un índice de referencia de largo plazo a esa imagen de referencia de largo plazo.

En algunos sistemas de codificación de video, las operaciones de marcación de imagen de referencia así como los procesos para la emisión y eliminación de imágenes de la DPB son realizadas después de que una imagen ha sido decodificada. Otros sistemas de codificación de video emplean un RPS para gestión de imagen de referencia. La diferencia más fundamental entre el mecanismo RPS y el proceso de ventana deslizante/MMCO es que para cada segmento particular el RPS proporciona un conjunto completo de las imágenes de referencia que son utilizadas por la imagen actual o cualquier imagen subsiguiente. Por lo tanto, un conjunto completo de todas las imágenes que deberían ser mantenidas en la DPB para uso por la imagen actual o futura es señalizado en el RPS. Esto es diferente del esquema de ventana deslizante/MMCO donde únicamente se señalizan cambios relativos a la DPB. Con el mecanismo RPS, no se requiere información de imágenes previas en orden de decodificación para mantener el estatus correcto de las imágenes de referencia en la DPB. El orden de decodificación de imágenes y las operaciones de la DPB son alterados en algunos sistemas de codificación de video para explotar las ventajas del RPS y mejorar la resiliencia al error. En algunos sistemas de codificación de video, operaciones de memoria tampón y marcación de imagen incluyendo tanto emisión como eliminación de imágenes decodificadas de la DPB pueden ser aplicadas después que una imagen actual ha sido decodificada. En otros sistemas de codificación de video, el RPS es primero decodificado de una cabecera de segmento de la imagen actual, y después se pueden aplicar operaciones de memoria tampón y marcación de imagen antes de decodificar la imagen actual.

En VVC, el enfoque de gestión de imagen de referencia se puede resumir de la siguiente manera. Dos listas de imágenes de referencia, indicadas lista 0 y lista 1, son directamente señalizadas y derivadas. No están basadas en RPS o el proceso de ventana deslizante más MMCO como se analizó antes. El marcado de imagen de referencia está directamente basado en listas 0 y 1 de imágenes de referencia utilizando tanto entradas activas como inactivas en las listas de imágenes de referencia, mientras que solamente pueden ser

utilizadas entradas activas como índices de referencia en inter-predicción de las CTU. Información para derivación de las dos listas de imágenes de referencia es señalizada por elementos de sintaxis y estructuras de sintaxis en el SPS, el PPS, y la cabecera de segmento. Estructuras RPL predefinidas son señalizadas en el SPS para uso haciendo referencia en la cabecera de segmento. Las dos listas de imágenes de referencia son generadas para todos los tipos de segmentos incluyendo segmentos de inter-predicción (B) direccionales, segmentos de inter-predicción (P) unidireccionales, y segmentos de intra-predicción (I). Las dos listas de imágenes de referencia pueden ser construidas sin utilizar un proceso de inicialización de lista de imágenes de referencia o un proceso de modificación de lista de imágenes de referencia. Las imágenes de referencia de largo plazo (LTRP) son identificadas por POC LSB. Los ciclos POC MSB delta pueden ser señalizados para LTRP conforme a lo determinado por una base imagen por imagen.

A fin de codificar una imagen de video, la imagen es dividida primero, y las divisiones son codificadas en una corriente de bits. Hay disponibles varios esquemas de división de imagen. Por ejemplo, una imagen puede ser dividida en segmentos regulares, segmentos dependientes, mosaicos, y/o según un Procesamiento Paralelo de Frente de onda (WPP). Por simplicidad, HEVC restringe los codificadores de manera que solamente segmentos regulares, segmentos dependientes, mosaicos, WPP, y combinaciones de los mismos pueden ser utilizados cuando se divide un segmento en grupos de CTB para codificación de video. Dicha división puede ser aplicada para soportar emparejamiento de tamaño de Unidad de Transferencia Máxima (MTU), procesamiento paralelo, y retraso reducido de extremo-a-extremo. La MTU indica la cantidad máxima de datos que puede ser transmitida en un solo paquete. Si una carga útil de paquete excede de la MTU, esa carga útil es dividida en dos paquetes a través de un proceso llamado fragmentación.

Un segmento regular, también denominado simplemente como un segmento, es una porción dividida de una imagen que puede ser reconstruida independientemente de otros segmentos regulares dentro de la misma imagen, no obstante algunas interdependencias debido a las operaciones de filtración de bucle. Cada segmento regular es encapsulado en su propia unidad de Capa de Abstracción de Red (NAL) para transmisión. Además, la predicción en-imagen (predicción intra muestra, predicción de información de movimiento, predicción de modo de codificación) y la dependencia de codificación entrópica a través de límites de segmento pueden ser inhabilitadas para soportar la reconstrucción independiente. Dicha reconstrucción independiente soporta paralelización. Por ejemplo, paralelización basada en segmento regular emplea comunicación mínima inter-procesador o inter-núcleo. Sin embargo, debido a que cada segmento regular es independiente, cada segmento está asociado con una cabecera de segmento separada. El uso de segmentos regulares puede incurrir en una sobrecarga de codificación sustancial debido al costo en bits de la cabecera de segmento para cada segmento y debido a la falta de predicción a través de los límites de segmento. Además, se pueden emplear segmentos regulares para soportar el emparejamiento para requerimientos de tamaño de MTU. Específicamente, debido a que un segmento regular es encapsulado en una unidad NAL separada y puede ser independientemente codificado, cada segmento regular debería ser más pequeño que la MTU en esquemas MTU para evitar la rotura del segmento en múltiples paquetes. Debido a esto, el objetivo de la paralelización y el objetivo del emparejamiento de tamaño de MTU pueden plantear demandas contradictorias para una disposición de segmento en una imagen.

Segmentos dependientes son similares a segmentos regulares, pero tienen cabeceras de segmento acortadas y permiten la división de los límites de bloque de árbol de imagen sin romper la predicción en-imagen. Por consiguiente, segmentos dependientes permiten que un segmento regular sea fragmentado en múltiples unidades NAL, lo cual proporciona un retraso reducido de extremo-a-extremo al permitir que una parte de un segmento regular sea enviada antes que se complete la codificación de todo el segmento regular.

Las imágenes pueden ser divididas en grupos/segmentos de mosaicos y mosaicos. Un mosaico es una secuencia de CTU que cubren una región rectangular de una imagen. Un grupo/segmento de mosaicos contiene varios mosaicos de una imagen. El modo de grupo de mosaicos de escaneo de trama y el modo de grupo de mosaicos rectangular pueden ser empleados para crear mosaicos. En el modo de grupo de mosaicos de escaneo de trama, un grupo de mosaicos contiene una secuencia de mosaicos en escaneo de trama de mosaicos de una imagen. En el modo de grupo de mosaicos rectangular, un grupo de mosaicos contiene varios mosaicos de una imagen que colectivamente forman una región rectangular de la imagen. Los mosaicos dentro de un grupo de mosaicos rectangular están en el orden de escaneo de trama de mosaico del grupo de mosaicos. Por ejemplo, un mosaico puede ser una porción dividida de una imagen creada por límites horizontales y verticales que crean columnas y filas de mosaicos. Los mosaicos pueden ser codificados en un orden de escaneo de trama (de derecha a izquierda y de arriba a abajo). El orden de escaneo de los CTB es local dentro de un mosaico. Por consiguiente, los CTB en un primer mosaico son codificados en el orden de escaneo de trama, antes de proseguir a los CTB en el siguiente mosaico. Similar a los segmentos regulares, los mosaicos rompen las dependencias de predicción en-imagen así como las dependencias de decodificación entrópica. Sin embargo, los mosaicos pueden no ser incluidos en unidades NAL individuales, y por lo tanto los mosaicos pueden no ser utilizados para emparejamiento de tamaño de MTU. Cada mosaico puede ser procesado por un procesador/núcleo, y la comunicación inter-procesador/inter-núcleo empleada para predicción en-imagen entre unidades de procesamiento que decodifican mosaicos vecinos puede estar limitada a transmitir una cabecera de segmento compartida

(cuando mosaicos adyacentes están en el mismo segmento), y realizar filtración de bucle relacionada con el hecho de compartir muestras reconstruidas y metadatos. Cuando se incluye más de un mosaico en un segmento, el desplazamiento de bytes de punto de entrada para cada mosaico distinto al primer desplazamiento de punto de entrada en el segmento puede ser señalado en la cabecera de segmento.

5 Para cada segmento y mosaico, se debería cumplir al menos una de las siguientes condiciones: 1) todos los bloques de árbol codificados en un segmento pertenecen al mismo mosaico; y 2) todos los bloques de árbol codificados en un mosaico pertenecen al mismo segmento.

10 En WPP, la imagen es dividida en filas únicas de los CTB. Mecanismos de decodificación entrópica y predicción pueden utilizar datos de los CTB en otras filas. Se hace posible un procesamiento paralelo a través de decodificación paralela de filas de CTB. Por ejemplo, una fila actual puede ser decodificada en paralelo con una fila precedente. Sin embargo, la decodificación de la fila actual está retrasada del proceso de decodificación de las filas precedentes por dos CTB. Este retraso asegura que los datos relacionados con el CTB de encima y el CTB de encima y a la derecha del CTB actual en la fila actual estén disponibles antes de que el CTB actual sea codificado. Este enfoque aparece como un frente de onda cuando es representado gráficamente. Este inicio escalonado permite la paralelización con hasta tantos procesadores/núcleos como filas de CTB contiene la imagen. Debido a que está permitida la predicción en-imagen entre filas de bloques de árbol contiguas dentro de una imagen, la comunicación inter-procesador/inter-núcleo para permitir la predicción en-imagen puede ser sustancial. La división WPP considera los tamaños de unidad NAL. Por lo tanto, WPP no soporta el emparejamiento de tamaño MTU. Sin embargo, pueden ser utilizados segmentos regulares junto con WPP, con cierta sobrecarga de codificación, para implementar el emparejamiento de tamaño de MTU según se desee. Finalmente, un segmento de frente de onda puede contener exactamente una fila de CTB. Además, cuando se emplea WPP y cuando un segmento se inicia dentro de una fila de CTB, el segmento debería finalizar en la misma fila de CTB.

25 Los mosaicos también pueden incluir conjuntos de mosaicos de movimiento restringido. Un conjunto de mosaicos de movimiento restringido (MCTS) es un conjunto de mosaicos diseñado de manera que vectores de movimiento asociados están restringidos a apuntar a ubicaciones de muestra completa dentro del MCTS y a ubicaciones de muestra fraccionaria que solamente requieren ubicaciones de muestra completa dentro del MCTS para interpolación. Además, no está permitido el uso de candidatos de vector de movimiento para predicción de vector de movimiento temporal derivados de bloques fuera del MCTS. De esta manera, cada MCTS puede ser decodificado independientemente sin la existencia de mosaicos no incluidos en el MCTS. Se pueden utilizar mensajes de información de mejoras complementarias (SEI) de MCTS temporales para indicar la existencia de MCTS en la corriente de bits y señalar los MCTS. El mensaje SEI de MCTS proporciona información complementaria que puede ser utilizada en la extracción de sub-corriente de bits de MCTS (especificada como parte de la semántica del mensaje SEI) para generar una corriente de bits de conformación para un conjunto MCTS. La información incluye un número de conjuntos de información de extracción, definiendo cada uno varios conjuntos MCTS y conteniendo bytes de carga útil de secuencia de bytes sin procesar (RBSP) del conjunto de parámetros de video (VPS) de sustitución, conjuntos de parámetros de secuencia (SPS), y conjuntos de parámetros de imagen (PPS) que han de ser utilizados durante el proceso de extracción de sub-corriente de bits de MCTS. Cuando se extrae una sub-corriente de bits según el proceso de extracción de sub-corriente de bits de MCTS, los conjuntos de parámetros (VPS, SPS, y PPS) pueden ser reescritos o reemplazados, y las cabeceras de segmento pueden ser actualizadas debido a que uno o todos los elementos de sintaxis relacionados con la dirección de segmento (incluyendo `first_slice_segment_in_pic_flag` y `slice_segment_address`) pueden emplear diferentes valores en la sub-corriente de bits extraída.

50 Aplicaciones de VR, también denominadas como aplicaciones de video de 360 grados, pueden mostrar solamente parte de una esfera completa y consecuentemente solo un sub-conjunto de una imagen completa. Un mecanismo de entrega de 360 que depende del puerto de visualización sobre transmisión adaptativa dinámica sobre protocolo de transferencia de hipertexto (DASH) puede ser empleado para reducir la tasa de bits y soportar la entrega de video de 360 grados mediante mecanismos de transmisión. Este mecanismo divide la esfera/imagen proyectada en múltiples MCTS, por ejemplo empleando la proyección de mapa de cubos (CMP). Dos o más corrientes de bits pueden ser codificadas con diferentes resoluciones o calidades espaciales. Cuando se entregan datos al decodificador, los MCTS procedentes de una corriente de bits de resolución/calidad superior son transmitidos para que el puerto de visualización sea mostrado (por ejemplo, el puerto de visualización frontal). Los MCTS de corrientes de bits de resolución/calidad inferior son transmitidos para los otros puertos de visualización. Estos MCTS son empaquetados en una cierta forma y después son enviados al receptor para ser decodificados. La expectativa es que el puerto de visualización visto por el usuario esté representado por un MCTS de resolución/calidad alta para crear una experiencia de visualización positiva. Cuando el usuario gira la cabeza para ver otro puerto de visualización (por ejemplo, el puerto de visualización izquierdo o derecho), el contenido presentado viene del puerto de visualización de resolución/calidad inferior para un periodo corto mientras el sistema está extrayendo los MCTS de resolución/calidad alta para el nuevo puerto de visualización. Cuando el usuario gira la cabeza para ver otro puerto de visualización, existe un retraso entre el tiempo en que el usuario gira la cabeza y el momento en que se observa una representación de resolución/calidad superior del puerto de visualización. Este retraso

depende de cómo de rápido pueda extraer el sistema los MCTS de resolución/calidad superior para ese puerto de visualización, lo cual a su vez depende del periodo IRAP. El periodo IRAP es el intervalo entre las ocurrencias de dos IRAP. Este retraso se refiere al periodo IRAP debido a que los MCTS del nuevo puerto de visualización solamente se pueden decodificar comenzando desde una imagen IRAP.

5

Por ejemplo, si el periodo IRAP es codificado cada segundo entonces se aplica lo siguiente. El mejor del caso para el retraso es el mismo que el retraso del viaje de ida y vuelta de red si el usuario gira la cabeza para ver el nuevo puerto de visualización justo antes que el sistema comience a extraer el nuevo segmento/periodo IRAP. En este escenario, el sistema podría solicitar los MCTS de resolución/calidad superior para el nuevo puerto de visualización de inmediato, y así el único retraso es el retraso del viaje de ida y vuelta de red, que es el retraso de la solicitud de extracción más el tiempo de transmisión de los MCTS solicitados, asumiendo que el retraso mínimo de almacenamiento en memoria tampón puede ser establecido aproximadamente a cero y el retraso de sensor es pequeño y despreciable. El retraso de ida y vuelta de red puede ser aproximadamente doscientos milisegundos, por ejemplo. El escenario del peor caso para el retraso es el periodo IRAP + retraso de viaje de ida y vuelta de red si el usuario gira la cabeza para ver el nuevo puerto de visualización justo después que el sistema haya hecho ya una solicitud para el siguiente segmento. Las corrientes de bits pueden ser codificadas con imágenes IRAP más frecuentes de manera que el periodo IRAP es más corto para mejorar el peor escenario del caso anterior ya que esto reduce el retraso global. Sin embargo, este enfoque aumenta las exigencias de ancho de banda a medida que disminuye la eficiencia de compresión.

10

15

20

En una implementación ejemplar, se permite que sub-imágenes de la misma imagen decodificada contengan diferentes valores de `nal_unit_type`. Este mecanismo es descrito a continuación. Las imágenes pueden ser divididas en sub-imágenes. Una sub-imagen es un conjunto rectangular de grupos/segmentos de mosaicos que comienza con un grupo de mosaicos que tiene `tile_group_address` igual a cero. Cada sub-imagen se puede referir a un PPS correspondiente y por lo tanto puede tener una división de mosaico separada. La existencia de sub-imágenes puede ser indicada en el PPS. Cada sub-imagen es tratada como una imagen en el proceso de decodificación. La filtración en-bucle a través de los límites de sub-imagen siempre puede ser inhabilitada. La anchura y altura de sub-imagen pueden ser especificadas en unidades de tamaños de CTU de luminancia. La ubicación de una sub-imagen en una imagen no puede ser señalizada, pero puede ser derivada utilizando la siguiente regla. La sub-imagen toma la siguiente ubicación sin ocupar en el orden de escaneo de trama de CTU dentro de una imagen que es lo bastante grande para contener la sub-imagen dentro de los límites de imagen. Las imágenes de referencia para decodificar cada sub-imagen son generadas extrayendo el área colocada con la sub-imagen actual de las imágenes de referencia en la memoria tampón de imágenes decodificadas. El área extraída es una sub-imagen decodificada, y por lo tanto la inter-predicción tiene lugar entre sub-imágenes del mismo tamaño y la misma ubicación dentro de la imagen. En dicho caso, permitir diferentes valores de `nal_unit_type` dentro de una imagen codificada permite que sub-imágenes que se originan a partir de una imagen de acceso aleatorio y sub-imágenes que se originan a partir de una imagen de acceso no-aleatorio sean fusionadas en la misma imagen codificada sin dificultad sustancial (por ejemplo, sin modificaciones a nivel de VCL). Dicho beneficio también se mantiene para codificación basada en MCTS.

25

30

35

40

Permitir diferentes valores de `nal_unit_type` dentro de una imagen codificada puede ser beneficioso en otros escenarios. Por ejemplo, un usuario puede ver algunas áreas de un contenido de video de 360 grados con más frecuencia que otras áreas. Para crear una mejor compensación entre la eficiencia de la codificación y la latencia de conmutación del puerto de visualización de calidad promedio comparable en entrega de video de 360 grados que depende del puerto de visualización en MCTS/basada en sub-imagen, pueden ser codificadas imágenes IRAP más frecuentes para áreas más comúnmente vistas que las otras áreas. La latencia de conmutación del puerto de visualización de calidad comparable es la latencia esperada por el usuario cuando se conmuta desde un primer puerto de visualización a un segundo puerto de visualización hasta que la calidad de presentación del segundo puerto de visualización alcanza una calidad de presentación comparable con la del primer puerto de visualización.

45

50

Otra implementación emplea las siguientes soluciones para soportar tipos de unidad NAL mezclados dentro de una imagen que incluye derivación de POC y gestión de imagen de referencia. Un indicador (`sps_mixed_tile_groups_in_pic_flag`) está presente en un conjunto de parámetros que es referido directamente o indirectamente por grupos de mosaicos para especificar si puede haber o no imágenes con sub-imágenes IRAP y no-IRAP mezcladas. Para una unidad NAL que contiene un grupo de mosaicos IDR, un indicador (`poc_msb_reset_flag`) está presente en una cabecera de grupo de mosaicos correspondiente para especificar si POC MSB es o no reestablecido en la derivación de POC para la imagen. Se define una variable denominada `PicRefreshFlag` y se asocia con una imagen. Este indicador especifica si la derivación de POC y el estado DPB deben ser actualizados cuando se decodifica la imagen. El valor de `PicRefreshFlag` es derivado de la siguiente manera. Si el grupo de mosaicos actual está incluido en la primera unidad de acceso en la corriente de bits, `PicRefreshFlag` es establecido igual a uno. De otra manera, si el grupo de mosaicos actual es un grupo de mosaicos IDR, `PicRefreshFlag` es establecido igual a `sps_mixed_tile_groups_in_pic_flag?poc_msb_reset_flag` : 1. De otra manera si el grupo de mosaicos actual

60

65

es un grupo de mosaicos CRA, se aplica lo siguiente. Si la unidad de acceso actual es la primera unidad de acceso de una secuencia codificada, PicRefreshFlag se establece igual a uno. La unidad de acceso actual es la primera unidad de acceso de una secuencia codificada cuando la unidad de acceso sigue inmediatamente un extremo de la unidad NAL de secuencia o la variable asociada HandleCraAsFirstPicInCvsFlag es establecida igual a uno. De otra manera, PicRefreshFlag es establecido igual a cero (por ejemplo, el grupo de mosaicos actual no pertenece a la primera unidad de acceso en la corriente de bits y no es un grupo de mosaicos IRAP).

Cuando PicRefreshFlag es igual a uno, el valor de POC MSB (PicOrderCntMsb) es restablecido para que sea igual a cero durante la derivación de POC para la imagen. La información empleada para gestión de imágenes de referencia tal como el conjunto de imágenes de referencia (RPS) o la lista de imágenes de referencia (RPL) es señalizada en la cabecera del grupo/segmento de mosaicos independientemente de un tipo de unidad NAL correspondiente. Las listas de imágenes de referencia son construidas al comienzo de la decodificación de cada grupo de mosaicos independientemente del tipo de unidad NAL. Las listas de imágenes de referencia pueden incluir RefPicList[0] y RefPicList[1] para un enfoque RPL, RefPicList0[] y RefPicList1[] para un enfoque RPS, o listas similares que contengan imágenes de referencia para operaciones de inter-predicción para una imagen. Cuando PicRefreshFlag es igual a uno, durante el proceso de marcación de imágenes de referencia, todas las imágenes de referencia en el DPB son marcadas como no utilizadas para referencia.

Dichas implementaciones están asociadas con algunos problemas. Por ejemplo, cuando el mezclado de valores de nal_unit_type dentro de una imagen no está permitido y cuando la derivación respecto a si una imagen es una imagen IRAP y la derivación de la variable NoRasOutputFlag se describen a nivel de imagen, el decodificador puede realizar estas derivaciones después de recibir la primera unidad NAL VCL de cualquier imagen. Sin embargo, debido al soporte de tipos de unidad NAL mezclados dentro de una imagen, el decodificador tendría que esperar a la llegada de otras unidades NAL VCL de una imagen antes de realizar las derivaciones anteriores. En el peor caso, el decodificador tendría que esperar a la llegada de la última unidad NAL VCL de una imagen. Además, dichos sistemas pueden señalar un indicador en las cabeceras de grupos de mosaico de unidades IDR NAL para especificar si el POC MSB es o no restablecido en la derivación POC para la imagen. Este mecanismo tiene los siguientes problemas. El caso de tipos de unidad NAL CRA y tipos de unidad NAL no-IRAP mezclados no sería soportado por este mecanismo. Además, la señalización de esta información en la cabecera del grupo/segmento de mosaicos de la unidad NAL VCL requeriría que el valor fuese cambiado durante la extracción o fusión de corriente de bits cuando un cambio al estatus respecto a si las unidades NAL IRAP (IDR o CRA) son mezcladas con unidades NAL no-IRAP en una imagen. Dicha reescritura de cabeceras de segmento ocurriría siempre que un usuario solicita el video, y por lo tanto requiere recursos de hardware significativos. Además, algunas otras mezclas de diferentes tipos de unidad NAL dentro de una imagen diferente de la mezcla de un tipo de unidad NAL IRAP particular y un tipo de unidad NAL no-IRAP particular están permitidas. Dicha flexibilidad no proporciona soporte para casos de uso práctico al mismo tiempo que complican el diseño del códec, lo cual innecesariamente aumenta la complejidad del decodificador y por lo tanto incrementa los costes de implementación asociados.

En general, esta descripción escribe técnicas para soportar el acceso aleatorio basado en sub-imagen o MCTS en codificación de video. De manera más específica, esta descripción describe diseños mejorados para soporte de tipos de unidad NAL mezclados dentro de una imagen, lo cual se emplea para soportar acceso aleatorio basado en sub-imagen o MCTS. La descripción de las técnicas se basa en la norma VVC, pero también se aplica a otras especificaciones de códec de video/medios.

A fin de resolver los problemas anteriores, se han descrito las siguientes implementaciones ejemplares. Dichas implementaciones pueden ser aplicadas individualmente o en combinación. En un ejemplo, cada imagen está asociada con una indicación respecto a si la imagen contiene valores mezclados de nal_unit_type. Esta indicación es señalizada en el PPS. Esta indicación soporta la determinación de si se restablece el POC MSB y/o si se restablece la DPB marcando todas las imágenes de referencia como no utilizadas para referencia. Cuando la indicación es señalizada en el PPS, un cambio del valor en el PPS puede ser conducido durante la fusión o extracción separada. Sin embargo esto es aceptable ya que los PPS son reescritos y reemplazados por otros mecanismos durante dicha extracción o fusión de corriente de bits.

Alternativamente, esta indicación puede ser señalizada en la cabecera del grupo de mosaicos pero se requiere que sea la misma para todos los grupos de mosaicos de una imagen. Sin embargo, en este caso, puede ser necesario que el valor sea cambiado durante la extracción de sub-corriente de bits de MCTS/secuencias de sub-imagen. Alternativamente, esta indicación puede ser señalizada en la cabecera de unidad NAL pero se requiere que sea la misma para todos los grupos de mosaicos de una imagen. Sin embargo, en este caso, el valor puede ser necesario que sea cambiado durante la extracción de sub-corriente de bits de MCTS/secuencias de sub-imagen. Alternativamente, esta indicación puede ser señalizada definiendo dichos tipos de unidad NAL VCL adicionales que, cuando son utilizados para una imagen, todas las unidades NAL VCL de la imagen deberían tener el mismo valor de tipo de unidad NAL. Sin embargo, en este caso, puede ser necesario que el valor de tipo de unidad NAL de las unidades NAL VCL

- 5 sea cambiado durante la extracción de sub-corriente de bits de MCTS/secuencias de sub-imagen. Alternativamente, esta indicación puede ser señalizada definiendo dichos tipos de unidad NAL VCL IRAP adicionales que, cuando son utilizados para una imagen, todas las unidades NAL VCL de la imagen deberían tener el mismo valor de tipo de unidad NAL. Sin embargo, en este caso, el valor de tipo de unidad NAL de las unidades NAL VCL puede ser necesario que sea cambiado durante la extracción de sub-corriente de bits de MCTS/secuencias de sub-imagen. Alternativamente, cada imagen que tiene al menos una unidad NAL VCL con cualquiera de los tipos de unidad NAL IRAP puede estar asociada con una indicación de si la imagen contiene valores de tipo de unidad NAL mezclados.
- 10 Además, se aplica una restricción de manera que el mezclado de valores de `nal_unit_type` dentro de una imagen está permitido de una manera limitada al permitir únicamente tipos de unidad NAL IRAP y no-IRAP mezclados. Para cualquier imagen particular, o bien todas las unidades NAL VCL tienen el mismo tipo de unidad NAL o bien algunas unidades NAL VCL tienen un tipo de unidad NAL IRAP particular y el resto tienen un tipo de unidad NAL no-IRAP particular. En otras palabras, unidades NAL VCL de cualquier imagen particular no pueden tener más de un tipo de unidad NAL IRAP y no pueden tener más de un tipo de unidad NAL no-IRAP. Una imagen puede ser considerada como una imagen IRAP solamente si la imagen no contiene valores mezclados de `nal_unit_type` y las unidades NAL VCL tienen un tipo de unidad NAL IRAP. Para cualquier unidad NAL IRAP (incluyendo IDR) que no pertenece a una imagen IRAP, el POC MSB no puede ser reestablecido. Para cualquier unidad NAL IRAP (incluyendo IDR) que no pertenece a una imagen IRAP, la DPB no es restablecida, y por lo tanto no se realiza el marcado de todas las imágenes de referencia como no utilizadas para referencia. El `TemporalId` puede ser establecido igual a cero para una imagen si al menos una unidad NAL VCL de la imagen es una unidad NAL IRAP.
- 20 Lo siguiente es una implementación específica de uno o más de los aspectos antes descritos. Una imagen IRAP puede ser definida como una imagen codificada para la cual el valor de `mixed_nalu_types_in_pic_flag` es igual a cero y cada unidad NAL VCL tiene un `nal_unit_type` del orden de `IDR_W_RADL` a `RSV_IRAP_VCL13`, inclusivo. Lo siguiente son ejemplos de sintaxis y semántica de PPS.

Pic_parameter_set_rbsp() {	Descriptor
pps_pic_parameter_set_id	ue(v)
pps_seq_parameter_set_id	ue(v)
mixed_nalu_types_in_pic_flag	ue(v)
single_tile_in_pic_flag	u(1)
...	

- 30 El `mixed_nalu_types_in_pic_flag` es establecido igual a cero para especificar que cada imagen que se refiere al PPS tiene múltiples unidades NAL VCL y estas unidades NAL no tienen el mismo valor de `nal_unit_type`. El `mixed_nalu_types_in_pic_flag` es establecido igual a uno para especificar que las unidades NAL VCL de cada imagen que se refieren al PPS tienen el mismo valor de `nal_unit_type`.
- 35 Ejemplo de sintaxis de cabecera de grupo/segmento de mosaicos es como siguen.

tile_group_header() {	Descriptor
tile_group_pic_parameter_set_id	ue(v)
si(rect_tile_group_flag NumTilesInPic > 1)	
tile_group_address	u(v)
si(!rect_tile_group_flag && !single_tile_per_tile_group_flag)	
num_tiles_in_tile_group_minus1	ue(v)
tile_group_type	ue(v)
tile_group_pic_order_cnt_lsb	u(v)
para(i = 0; i < 2; i++) {	
si(num_ref_pic_lists_in_sps[i] > 0 && (i = 0 (i = 1 && rpl1_idx_present_flag)))	
ref_pic_list_sps_flag[i]	u(1)
si(ref_pic_list_sps_flag[i]) {	
si(num_ref_pic_lists_in_sps[i] > 1 && (i = 0 (i = 1 && rpl1_idx_present_flag)))	
ref_pic_list_idx[i]	u(v)
} además	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i])	

para(j = 0; j < NumLtrpEntries[i][RplIdx[i]]; j++) {	
delta_poc_msb_present_flag[i][j]	u(1)
si(delta_poc_msb_present_flag[i][j])	
delta_poc_msb_cycle_lt[i][j]	ue(v)
}	
}	
si(tile_group_type == P tile_group_type == B) {	
num_ref_idx_active_override_flag	u(1)
si(num_ref_idx_active_override_flag)	
para(i = 0; i < (tile_group_type == B ? 2 : 1); i++)	
si(num_ref_entries[i][RplIdx[i]] > 1)	
num_ref_idx_active_minus1[i]	ue(v)
}	
...	

La semántica de cabecera de unidad NAL ejemplar es la siguiente. Para unidades NAL VCL de cualquier imagen particular, se satisfará cualquiera de las siguientes dos condiciones. Todas las unidades NAL VCL tienen el mismo valor de `nal_unit_type`. Algunas de las unidades NAL VCL tienen un valor de tipo de unidad NAL IRAP particular (es decir, un valor de `nal_unit_type` del orden de `IDR_W_RADL` a `RSV_IRAP_VCL13`, inclusive), mientras que todas las otras unidades NAL VCL tienen un tipo de unidad NAL VCL no-IRAP particular (es decir, un valor de `nal_unit_type` del orden de `TRAIL_NUT` a `RSV_VCL_7`, inclusive, o del orden de `RSV_VCL14` a `RSV_VCL15`, inclusive). El `nuh_temporal_id_plus1` minus 1 especifica un identificador temporal para la unidad NAL. El valor de `nuh_temporal_id_plus1` no debe ser igual cero.

La variable `TemporalId` es derivada de la siguiente manera:

$$\text{TemporalId} = \text{nuh_temporal_id_plus1} - 1 \quad (7-1)$$

Cuando `nal_unit_type` es del orden de `IDR_W_RADL` a `RSV_IRAP_VCL13`, inclusive, para una unidad NAL VCL de una imagen, independientemente del valor `nal_unit_type` de otras unidades NAL VCL de la imagen, `TemporalId` será igual a cero para todas las unidades NAL VCL de la imagen. El valor de `TemporalId` será el mismo para todas las unidades NAL VCL de una unidad de acceso. El valor de `TemporalId` de una imagen codificada o una unidad de acceso es el valor del `TemporalId` de las unidades NAL VCL de la imagen codificada o de la unidad de acceso.

Un ejemplo de proceso de decodificación para una imagen codificada es como sigue. El proceso de decodificación opera de la siguiente manera para la imagen actual `CurrPic`. La decodificación de unidades NAL es especificada en la presente memoria. Los siguientes procesos de decodificación utilizan elementos de sintaxis en la capa de cabecera de grupo de mosaicos y por encima. Variables y funciones relacionadas con el recuento de orden de imagen son derivadas conforme a lo especificado en la presente memoria. Esto es consultado solamente para el primer grupo/segmento de mosaicos de una imagen. Al inicio del proceso de decodificación para cada grupo/segmento de mosaicos, el proceso de decodificación para construcción de listas de imágenes de referencia es consultado para derivación de lista 0 de imágenes de referencia (`RefPicList[0]`) y lista 1 de imágenes de referencia (`RefPicList[1]`). Si la imagen actual es una imagen IDR, entonces el proceso de decodificación para la construcción de listas de imágenes de referencia puede ser consultado para propósito de comprobación de conformación de corriente de bits, pero puede no ser necesario para decodificar la imagen o imágenes actuales que siguen a la imagen actual en orden de decodificación.

Un proceso de decodificación para la construcción de listas de imágenes de referencia es el siguiente. Este proceso es consultado al comienzo del proceso de decodificación para cada grupo de mosaicos. Las imágenes de referencia son direccionadas a través de índices de referencia. Un índice de referencia es un índice en una lista de imágenes de referencia. Cuando se decodifica un grupo I de mosaicos, ninguna lista de imágenes de referencia es utilizada en la decodificación de los datos del grupo de mosaicos. Cuando se decodifica un grupo P de mosaicos, solamente se utiliza la lista de imágenes de referencia cero (`RefPicList[0]`) en la decodificación de los datos del grupo de mosaicos. Cuando se decodifica un grupo B de mosaicos, ambas, la lista cero de imágenes de referencia y la lista uno de imágenes de referencia (`RefPicList[1]`) son utilizadas en la decodificación de los datos del grupo de mosaicos. Al comienzo del proceso de decodificación para cada grupo de mosaicos, las listas de imágenes de referencia `RefPicList[0]` y `RefPicList[1]` son derivadas. Las listas de imágenes de referencia son utilizadas en el marcado de imágenes de referencia o en la decodificación de los datos del grupo de mosaicos. Para cualquier grupo de mosaicos de una imagen IDR o un grupo I de mosaicos de una imagen no-IDR, `RefPicList[0]` y la `RefPicList[1]` pueden ser derivadas para propósito de comprobación de conformación de corriente de bits, pero su derivación no es necesaria para decodificar la imagen o imágenes actuales que siguen a la imagen actual en el orden de decodificación. Para un grupo P de mosaicos, la `RefPicList[1]` puede ser derivada para

propósito de comprobación de conformación de corriente de bits, pero la derivación no es necesaria para decodificar la imagen o imágenes actuales que siguen a la imagen actual en orden de decodificación.

5 La figura 8 es un diagrama esquemático de un ejemplo de dispositivo 800 de codificación de video. El dispositivo 800 de codificación de video es conveniente para implementar los ejemplos/realizaciones descritas como se han descrito en la presente memoria. El dispositivo 800 de codificación de video comprende puertos 820 aguas abajo, puertos 850 aguas arriba, y/o unidades 810 de transceptor (Tx/Rx), incluyendo transmisores y/o receptores para comunicar datos aguas arriba y/o aguas abajo sobre una red. El dispositivo 800 de codificación de video también incluye un procesador 830 que incluye una unidad lógica y/o
10 una unidad de procesamiento central (CPU) para procesar los datos y una memoria 832 para almacenar los datos. El dispositivo 800 de codificación de video también puede comprender componentes eléctricos, óptico-eléctricos (OE), componentes electro-ópticos (EO), y/o componentes de comunicación inalámbrica acoplados a los puertos 850 de aguas arriba y/o puertos 820 de aguas abajo para comunicación de datos a través de redes de comunicación eléctricas, ópticas, o inalámbricas. El dispositivo 800 de codificación de video también puede incluir dispositivos 860 de entrada y/o salida (I/O) para comunicar datos hacia y desde un usuario. Los dispositivos 860 de I/O pueden incluir dispositivos de salida tales como un dispositivo de visualización para mostrar datos de video, altavoces para emitir datos de audio, etc. Los dispositivos 860 de I/O también pueden incluir dispositivos de entrada, tales como un teclado, ratón, bola de desplazamiento, etc., y/o interfaces correspondientes para interactuar con dichos dispositivos de salida.

20 El procesador 830 es implementado por hardware y software. El procesador 830 puede ser implementado como uno o más chips de CPU, núcleos (por ejemplo, como un procesador multi-núcleo), matrices de puertas lógicas programables en campo (FPGA), circuitos integrados de aplicación específica (ASIC), y procesadores de señal digital (DSP). El procesador 830 está en comunicación con los puertos 820 de aguas abajo, Tx/Rx 810, los puertos 850 de aguas arriba, y la memoria 832. El procesador 830 comprende un módulo 814 de codificación. El módulo 814 de codificación implementa las realizaciones descritas en la presente memoria, tal como los métodos 100, 900, y 1000, que pueden emplear una CVS 500, una corriente 600 de video de imagen de VR, y/o una corriente 700 de bits. El módulo 814 de codificación también puede implementar cualquier otro método/mecanismo aquí descritos. Además, el módulo 814 de codificación puede implementar un sistema 200 de códec, un codificador 300, y/o un decodificador 400. Por ejemplo, el módulo 814 de codificación puede establecer un indicador en un PPS para indicar cuando una imagen contiene unidades NAL IRAP y no-IRAP y restringir dichas imágenes para contener solamente un único tipo de unidad NAL IRAP y un único tipo de unidad NAL no-IRAP. Por lo tanto, el módulo 814 de codificación hace que el dispositivo 800 de codificación de video proporcione funcionalidad adicional y/o eficiencia de codificación
35 cuando se codifican datos de video. Debido a esto, el módulo 814 de codificación mejora la funcionalidad del dispositivo 800 de codificación de video así como problemas de direcciones que son específicos de las técnicas de codificación de video. Además, el módulo 814 de codificación efectúa una transformación del dispositivo 800 de codificación de video a un estado diferente. Alternativamente, el módulo 814 de codificación puede ser implementado como instrucciones almacenadas en la memoria 832 y ejecutadas por el procesador 830 (por ejemplo, como un producto de programa informático almacenado en un medio no transitorio).

45 La memoria 832 comprende uno o más tipos de memoria tales como discos, unidades de cinta, unidades de estado sólido, memoria de solo lectura (ROM), memoria de acceso aleatorio (RAM), memoria flash, memoria ternaria de contenido direccionable (TCAM), memoria estática de acceso aleatorio (SRAM), etc. La memoria 832 puede ser utilizada como un dispositivo de almacenamiento de datos de sobreflujo, para almacenar programas cuando dichos programas son seleccionados para ejecución, y almacenar instrucciones y datos que son leídos durante la ejecución del programa.

50 La figura 9 es un gráfico de flujo de un método ejemplar 900 de codificación de una secuencia de video, tal como CVS 500, que contiene una imagen con tipos de unidad NAL mezclados en una corriente de bits, tal como una corriente 700 de bits que incluye la corriente 600 de video de imagen de VR fusionada de múltiples corrientes 601-603 de video de sub-imagen a una pluralidad de resoluciones de video. El método 900 puede ser empleado por un codificador, tal como un sistema 200 de códec, un codificador 300, y/o un dispositivo
55 800 de codificación de video cuando se ejecuta el método 100.

El método 900 puede comenzar cuando un codificador recibe una secuencia de video que incluye una pluralidad de imágenes, tales como imágenes de VR, y determina codificar esa secuencia de video en una corriente de bits, por ejemplo basándose en la entrada de usuario. En la etapa 901, el codificador determina si una imagen actual contiene una pluralidad de sub-imágenes de diferentes tipos. Dichos tipos pueden incluir al menos un segmento de la imagen que contiene una porción de una sub-imagen IRAP y al menos un segmento de la imagen que contiene una porción de una sub-imagen NAL no-IRAP. En la etapa 903, el codificador codifica las sub-imágenes de la imagen en una pluralidad de unidades NAL VCL en una corriente de bits. Dichas unidades NAL VCL pueden incluir una o más unidades NAL IRAP y una o más unidades NAL no-IRAP. Por ejemplo, la etapa de codificación puede incluir la fusión de sub-corrientes de bits de diferentes resoluciones en una sola corriente de bits para comunicación a un decodificador.

En la etapa 905, el codificador codifica un PPS en la corriente de bits y codifica un indicador en el PPS en la corriente de bits. Como ejemplo específico, la codificación del PPS puede incluir que se altere un PPS previamente codificado para incluir un valor de indicador, por ejemplo en respuesta a una fusión de sub-corrientes de bits. El indicador puede ser establecido a un primer valor cuando un valor de tipo de unidad NAL es el mismo para todas las unidades NAL VCL asociadas con la imagen. El indicador también puede ser establecido a un segundo valor cuando un primer valor de tipo de unidad NAL para unidades NAL VCL que contienen una o más de las sub-imágenes de la imagen es diferente de un segundo valor de tipo de unidad NAL para unidades NAL VCL que contienen una o más de las sub-imágenes de la imagen. Por ejemplo, el primer valor de tipo de unidad NAL puede indicar que la imagen contiene una sub-imagen IRAP, y el segundo valor de tipo de unidad NAL puede indicar que la imagen también contiene una sub-imagen no-IRAP. Además, el primer valor de tipo de unidad NAL puede ser igual a uno de IDR_W_RADL, IDR_N_LP, o CRA_NUT. Además, el segundo valor de tipo de unidad NAL puede ser igual a uno de TRAIL_NUT, RADL_NUT, o RASL_NUT. Como ejemplo específico, el indicador puede ser un `mixed_nalu_type_in_pic_flag`. En un ejemplo específico, el `mixed_nalu_type_in_pic_flag` puede ser establecido igual a uno para especificar que cada imagen que se refiere al PPS que contiene el indicador tiene más de una unidad VCL NAL. Además, el indicador especifica que las unidades NAL VCL asociadas con la imagen correspondiente no tienen el mismo valor de tipo de unidad NAL (`nal_unit_type`). En otro ejemplo específico, el `mixed_nalu_type_in_pic_flag` puede ser establecido igual a cero para especificar que cada imagen que se refiere al PPS que contiene el indicador tiene una o más unidades NAL VCL. Además, el indicador especifica que todas las unidades NAL VCL de la imagen correspondiente tienen el mismo valor de `nal_unit_type`.

En la etapa 907, el codificador puede almacenar la corriente de bits para comunicación hacia un decodificador.

La figura 10 es un diagrama de flujo de un método ejemplar 1000 de decodificación de una secuencia de video, tal como CVS 500, que contiene una imagen con tipos de unidad NAL mezclados procedentes de una corriente de bits, tal como una corriente 700 de bits que incluye corriente 600 de video de imagen de VR fusionada de múltiples corrientes 601-603 de video de sub-imagen a una pluralidad de resoluciones de video. El método 1000 puede ser empleado por un decodificador, tal como un sistema 200 de códec, un decodificador 400, y/o un dispositivo 800 de codificación de video cuando se realiza el método 100.

El método 1000 puede comenzar cuando un decodificador comienza a recibir una corriente de bits de datos codificados que representan una secuencia de video, por ejemplo como un resultado del método 900. En la etapa 1001, el decodificador recibe una corriente de bits. La corriente de bits comprende una pluralidad de sub-imágenes asociadas con una imagen y un indicador. Como un ejemplo específico, la corriente de bits puede comprender un PPS que incluye el indicador. Además, las sub-imágenes están contenidas en una pluralidad de unidades NAL VCL. Por ejemplo, segmentos asociados con las sub-imágenes están incluidos en las unidades NAL VCL.

En la etapa 1003, el decodificador determina que un valor de tipo de unidad NAL es el mismo para todas las unidades NAL VCL asociadas con la imagen cuando el indicador es establecido a un primer valor. Además, el decodificador determina que un primer valor de tipo de unidad NAL para unidades NAL VCL que contienen una o más de las sub-imágenes de la imagen es diferente de un segundo valor de tipo de unidad NAL para unidades NAL VCL que contienen una o más de las sub-imágenes de la imagen cuando el indicador es establecido a un segundo valor. Por ejemplo, el primer valor de tipo de unidad NAL puede indicar que la imagen contiene una sub-imagen IRAP, y el segundo valor de tipo de unidad NAL puede indicar que la imagen también contiene una sub-imagen no-IRAP. Además, el primer valor de tipo de unidad NAL puede ser igual a uno de IDR_W_RADL, IDR_N_LP, o CRA_NUT. Además, el segundo valor de tipo de unidad NAL puede ser igual a uno de TRAIL_NUT, RADL_NUT, o RASL_NUT. Como un ejemplo específico, el indicador puede ser un `mixed_nalu_type_in_pic_flag`. El `mixed_nalu_type_in_pic_flag` puede ser establecido igual a uno cuando se especifica que cada imagen que hace referencia al PPS tiene más de una unidad NAL VCL y las unidades NAL VCL no tienen el mismo valor de tipo de unidad NAL (`nal_unit_type`). También, el `mixed_nalu_type_in_pic_flag` puede ser establecido igual a cero cuando cada imagen que hace referencia al PPS tiene una o más unidades NAL VCL y las unidades NAL VCL de cada imagen que hace referencia al PPS tienen el mismo valor de `nal_unit_type`.

En la etapa 1005, el decodificador puede decodificar una o más de las sub-imágenes basándose en los valores de tipo de unidad NAL. El decodificador también puede reenviar una o más de las sub-imágenes para mostrar como parte de una secuencia de video decodificada en la etapa 1007.

La figura 11 es un diagrama esquemático de un sistema ejemplar 1100 para codificar una secuencia de video, tal como la CVS 500, que contiene una imagen con tipos de unidad NAL mezclados en una corriente de bits, tal como una corriente 700 de bits que incluye la corriente 600 de video de imagen de VR fusionada de múltiples corrientes 601-603 de video de sub-imagen en una pluralidad de resoluciones de video. El sistema 1100 puede ser implementado por un codificador y un decodificador tal como un sistema 200 de

códec, un codificador 300, un decodificador 400, y/o un dispositivo 800 de codificación de video. Además, se puede emplear el sistema 1100 cuando se implementa el método 100, 900, y/o 1000.

5 El sistema 1100 incluye un codificador 1102 de video. El codificador 1102 de video comprende un módulo 1101 de determinación para determinar si una imagen contiene una pluralidad de sub-imágenes de diferentes tipos. El codificador 1102 de video además comprende un módulo 1103 de codificación para codificar las sub-imágenes de la imagen en una pluralidad de unidades NAL VCL en una corriente de bits. El módulo 1103 de codificación además es para codificar en la corriente de bits un indicador establecido a un primer valor cuando un valor de tipo de unidad NAL es el mismo para todas las unidades NAL VCL asociadas con la imagen y establecido a un segundo valor cuando un primer valor de tipo de unidad NAL para unidades NAL VCL que contienen una o más de las sub-imágenes de la imagen es diferente de un segundo valor de tipo de unidad NAL para unidades NAL VCL que contienen una o más de las sub-imágenes de la imagen. El codificador 1102 de video además comprende un módulo 1105 de almacenamiento para almacenar la corriente de bits para comunicación hacia un decodificador. El codificador 1102 de video además comprende un módulo 1107 de transmisión para transmitir la corriente de bits hacia el decodificador 1110 de video. El codificador 1102 de video además puede ser configurado para realizar cualquiera de las etapas del método 900.

20 El sistema 1100 también incluye un decodificador 1110 de video. El decodificador 1110 de video comprende un módulo 1111 de recepción para recibir una corriente de bits que comprende una pluralidad de sub-imágenes asociadas con una imagen y un indicador, en donde las sub-imágenes están contenidas en una pluralidad de unidades NAL VCL. El decodificador 1110 de video además comprende un módulo 1113 de determinación para determinar que un valor de tipo de unidad NAL es el mismo para todas las unidades NAL VCL asociadas con la imagen cuando el indicador es establecido a un primer valor. El módulo 1113 de determinación además es para determinar que un primer valor de tipo de unidad NAL para unidades NAL VCL que contienen una o más de las sub-imágenes de la imagen es diferente de un segundo valor de tipo de unidad NAL para unidades NAL VCL que contienen una o más de las sub-imágenes de la imagen cuando el indicador es establecido a un segundo valor. El decodificador 1110 de video además comprende un módulo 1115 de decodificación para decodificar una o más de las sub-imágenes con base en los valores de tipo de unidad NAL. El decodificador 1110 de video además comprende un módulo 1117 de reenvío para reenviar una o más de las sub-imágenes para mostrar como parte de una secuencia de video decodificada. El decodificador 1110 de video además puede ser configurado para realizar cualquiera de las etapas del método 1000.

35 Un primer componente está directamente acoplado a un segundo componente cuando no hay componentes intermedios, excepto por una línea, una traza, u otro medio entre el primer componente y el segundo componente. El primer componente es indirectamente acoplado al segundo componente cuando hay componentes intermedios diferentes a una línea, una traza, u otro medio entre el primer componente y el segundo componente. El término "acoplado" y sus variantes incluyen tanto acoplado directamente como acoplado indirectamente. El uso del término "aproximadamente" significa un intervalo que incluye $\pm 10\%$ del número subsiguiente a menos que se indique de otra manera.

45 También se debería entender que las etapas de los métodos ejemplares establecidos en la presente memoria no requieren ser ejecutadas necesariamente en el orden descrito, y el orden de las etapas de dichos métodos debería entenderse como simplemente ejemplar. De igual manera, pueden incluirse etapas adicionales en dichos métodos, y algunas etapas se pueden omitir o combinar, en métodos consistentes con varias realizaciones de la presente descripción.

50 Aunque se han proporcionado varias realizaciones en la presente descripción, se puede entender que los sistemas y métodos descritos podrían ser incorporados de muchas otras formas específicas sin apartarse del alcance de la presente descripción. Los presentes ejemplos han de ser considerados como ilustrativos y no restrictivos, y la intención es no estar limitados a los detalles proporcionados en la presente memoria. Por ejemplo, los diversos elementos o componentes se pueden combinar o integrar en otro sistema o algunas características se pueden omitir o no implementar.

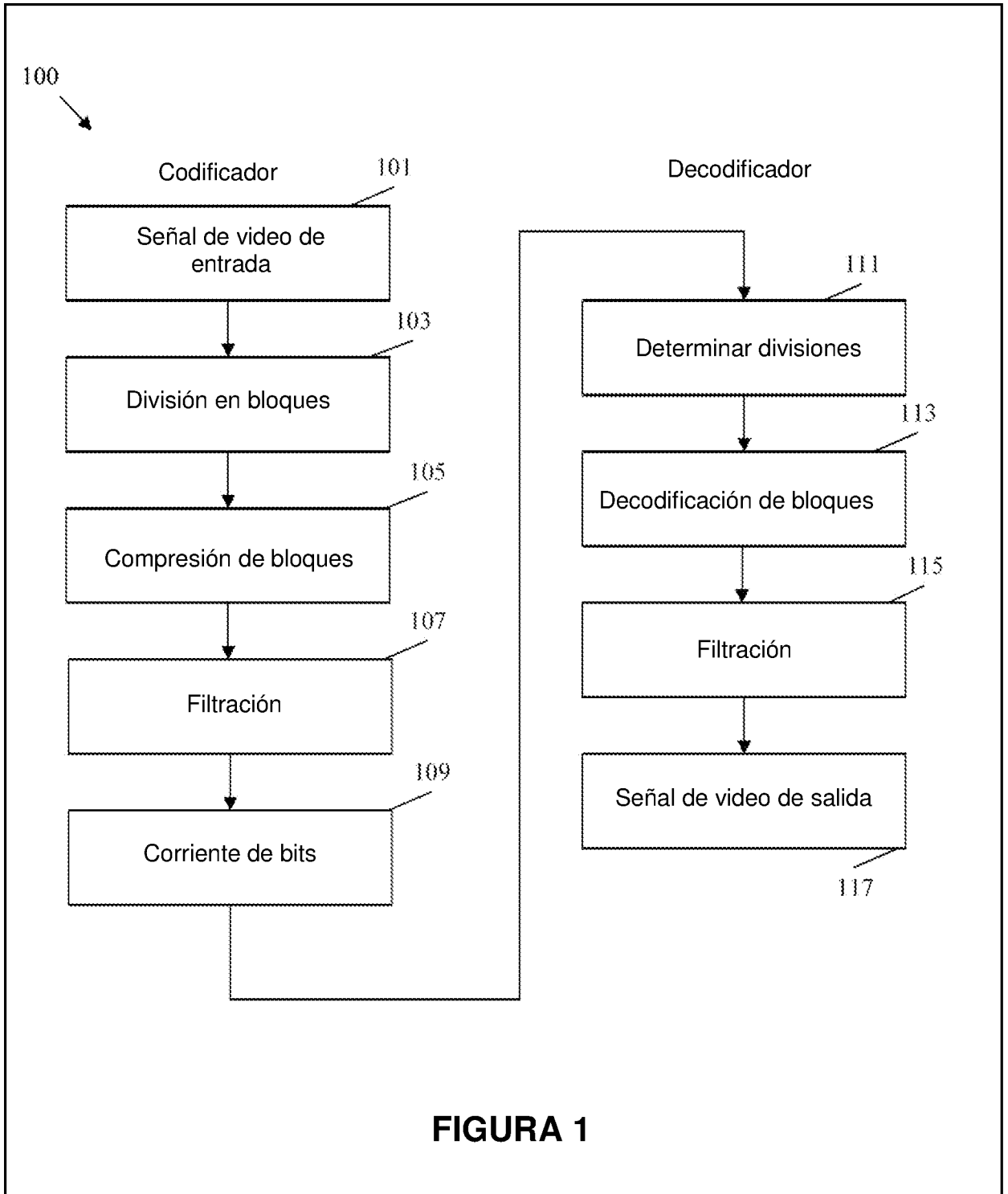
55 Además, técnicas, sistemas, subsistemas, y métodos descritos e ilustrados en las diversas realizaciones como discretos o separados se pueden combinar o integrar con otros sistemas, componentes, técnicas o métodos sin apartarse del alcance de la presente descripción. Otros ejemplos de cambios, sustituciones y alteraciones pueden ser determinados por un experto en la técnica y pueden realizarse sin apartarse del espíritu y alcance descritos en la presente memoria.

60

REIVINDICACIONES

1. Un método implementado en un decodificador, comprendiendo el método:
 - 5 recibir, (1001), mediante un receptor del decodificador, una corriente de bits que comprende una pluralidad de sub-imágenes asociadas con una imagen y un indicador, en donde las sub-imágenes están contenidas en unidades de capa de abstracción de red, NAL, de capa de codificación de video, VCL;
 - determinar, (1003), mediante un procesador del decodificador, que todas las unidades NAL VCL asociadas con la imagen tienen el mismo tipo de unidad NAL cuando el indicador es establecido sobre un primer valor;
 - 10 determinar, (1003), mediante el procesador, que unidades NAL VCL de las sub-imágenes de imagen tienen todas un valor particular de un tipo de unidad NAL, IRAP, y que todas las otras unidades NAL VCL de la imagen tienen un valor particular de un tipo de unidad NAL no-IRAP cuando el indicador es establecido sobre un segundo valor; y
 - 15 decodificar (1005), mediante el procesador, una o más de las sub-imágenes sobre la base del valor particular del tipo de unidad NAL IRAP o del valor particular del tipo de unidad NAL no-IRAP.
- 20 2. El método según la reivindicación 1, en donde la corriente de bits incluye un conjunto de parámetros de imagen, PPS, incluyendo el indicador.
 3. El método según cualquiera de las reivindicaciones 1-2, en donde el valor particular del tipo de unidad NAL IRAP es igual a una Actualización de Decodificación Instantánea, IDR, con imagen principal decodificable de acceso aleatorio, IDR_W_RADL, una IDR sin imágenes principales, IDR_N_LP, o un tipo de
 - 25 unidad NAL de acceso aleatorio limpio, CRA, CRA_NUT.
 4. El método según cualquiera de las reivindicaciones 1-3, en donde el valor particular del tipo de unidad NAL no-IRAP es igual al tipo de unidad NAL de imagen final, TRAIL_NUT, tipo de unidad NAL de imagen principal decodificable de acceso aleatorio, RADL_NUT, o tipo de unidad NAL de imagen principal omitida de acceso aleatorio, RASL, RASL_NUT.
 - 30 5. El método según cualquiera de las reivindicaciones 1-4, en donde el indicador es mixed_nalu_types_in_pic_flag.
 - 35 6. El método según la reivindicación 5, en donde mixed_nalu_types_in_pic_flag es igual a uno cuando se especifica que la imagen que hace referencia al PPS tiene más de una de las unidades NAL de VCL y las unidades NAL de VCL no tienen el mismo valor de tipo de unidad NAL, nal_unit_type, y en donde mixed_nalu_types_in_pic_flag es igual a cero cuando se especifica que la imagen que hace referencia al PPS tiene una o más de las unidades NAL de VCL y las unidades NAL de VCL tienen el mismo valor de
 - 40 nal_unit_type.
7. Un método implementado en un codificador, comprendiendo el método:
 - 45 determinar (901), mediante un procesador del codificador, si una imagen contiene una pluralidad de sub-imágenes de diferentes tipos;
 - codificar (903), mediante el procesador, las sub-imágenes de la imagen en una pluralidad de unidades de capa de abstracción de red, NAL, de capa de codificación de video, VCL, en una corriente de bits;
 - 50 codificar (905) en la corriente de bits, mediante el procesador, un indicador establecido a un primer valor cuando todas las unidades NAL de VCL asociadas con la imagen tienen un mismo tipo de unidad NAL y establecido a un segundo valor cuando las unidades NAL de VCL de una o más de las sub-imágenes de la imagen tienen todas un valor particular de un tipo de unidad NAL, IRAP, y todas las otras unidades NAL de
 - 55 VCL de la imagen tienen un valor particular de un tipo de unidad NAL no-IRAP; y
 - almacenar (907), mediante una memoria acoplada al procesador, la corriente de bits para comunicación hacia un decodificador.
- 60 8. El método según la reivindicación 7, que además comprende codificar un conjunto de parámetros de imagen, PPS, en la corriente de bits, en donde el indicador es codificado en el PPS.
 9. El método según cualquiera de las reivindicaciones 7-8, en donde el valor particular del tipo de unidad NAL IRAP es igual a una Actualización de Decodificación Instantánea, IDR, con imagen principal decodificable de acceso aleatorio, IDR_W_RADL, una IDR sin imágenes principales, IDR_N_LP, o un tipo de
 - 65 unidad NAL de acceso aleatorio limpio, CRA, CRA_NUT.

10. El método según cualquiera de las reivindicaciones 7-9, en donde el valor particular del tipo de unidad NAL no- IRAP es igual al tipo de unidad NAL de imagen final TRAIL_NUT, tipo de unidad NAL de imagen principal decodificable de acceso aleatorio, RADL_NUT, o tipo de unidad NAL de imagen principal omitida de acceso aleatorio, RASL, RASL_NUT.
- 5
11. El método según cualquiera de las reivindicaciones 7-10, en donde el indicador es mixed_nalu_types_in_pic_flag.
- 10
12. El método según cualquiera de las reivindicaciones 7-11, en donde mixed_nalu_types_in_pic_flag es igual a uno cuando se especifica que la imagen que hace referencia al PPS tiene más de una de las unidades NAL de VCL y las unidades NAL de VCL no tienen el mismo valor de tipo de unidad NAL, nal_unit_type, y en donde mixed_nalu_types_in_pic_flag es igual a cero cuando se especifica que la imagen que hace referencia al PPS tiene una o más de las unidades NAL de VCL y las unidades NAL de VCL tienen el mismo valor de nal_unit_type.
- 15
13. Un dispositivo de codificación de video, que comprende:
un procesador (830), un receptor (810) acoplado al procesador (830), y una memoria (832) acoplada al procesador (830), en donde el procesador (830), el receptor (810), y la memoria (832) están configurados para realizar el método según cualquiera de las reivindicaciones 1-6.
- 20
14. Un dispositivo de codificación de video, que comprende:
un procesador (830), una memoria (832) acoplada al procesador (830), y un transmisor (810) acoplado al procesador (830), en donde el procesador (830), la memoria (832) y el transmisor (810) están configurados para realizar el método según cualquiera de las reivindicaciones 7-12.
- 25



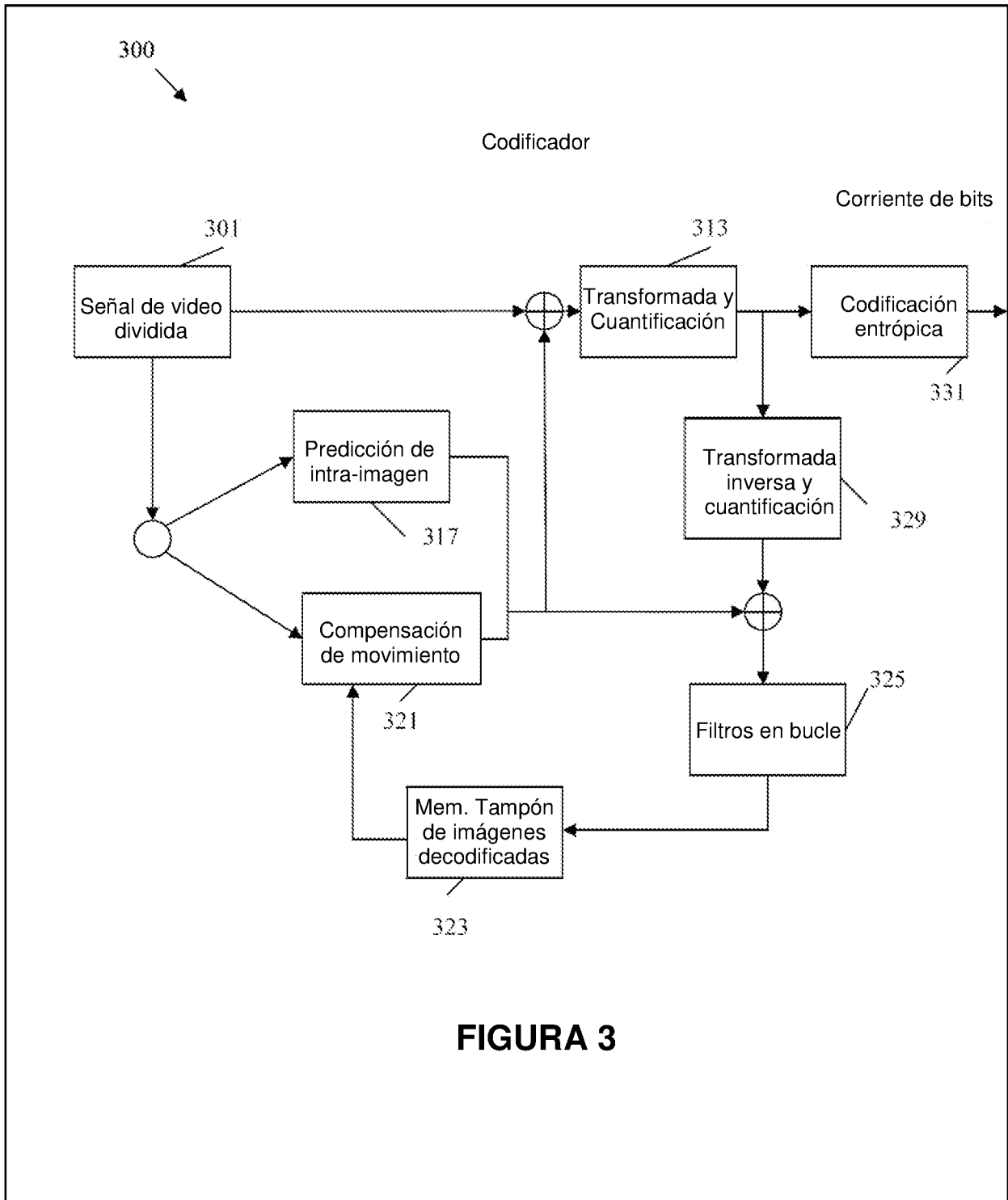


FIGURA 3

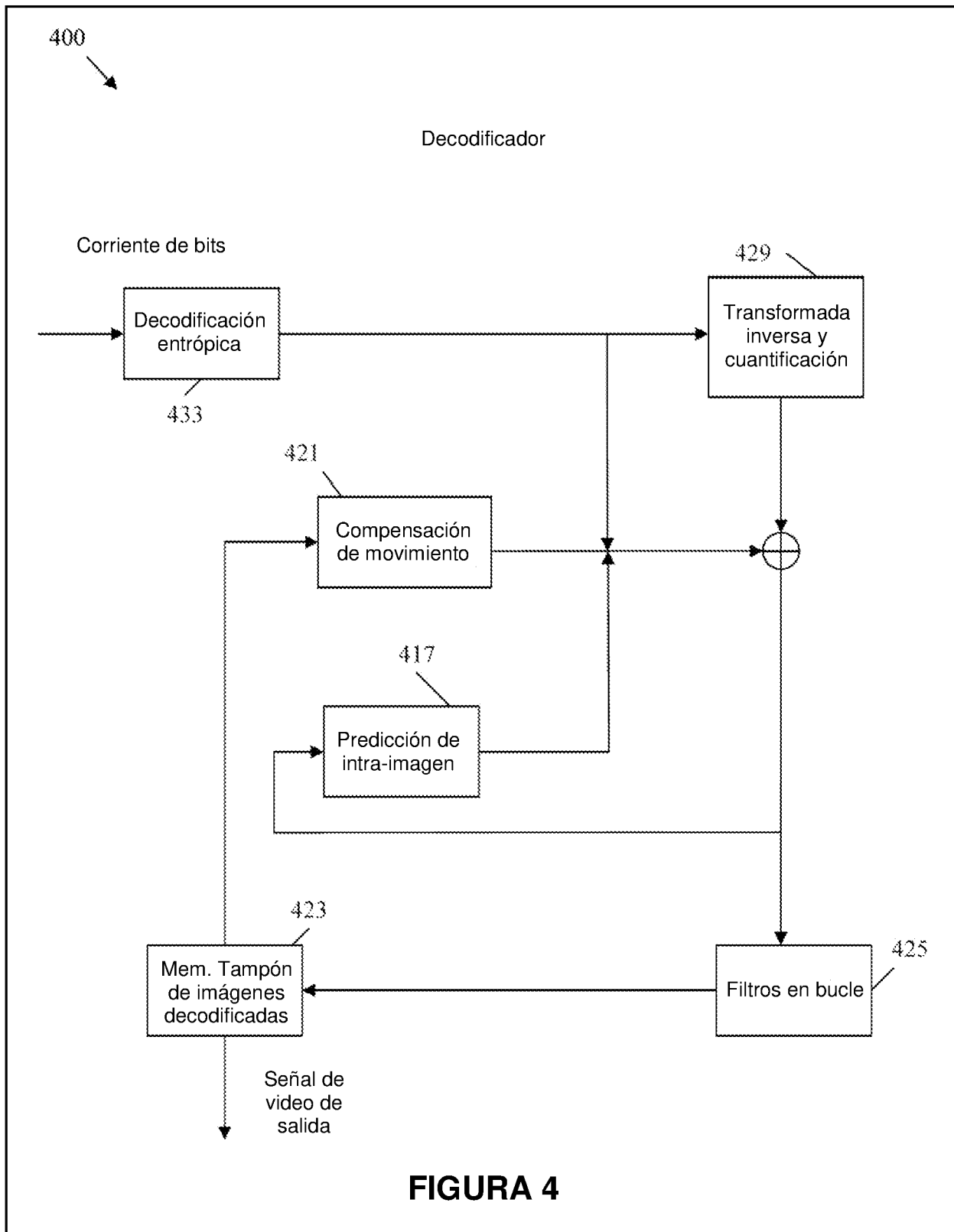
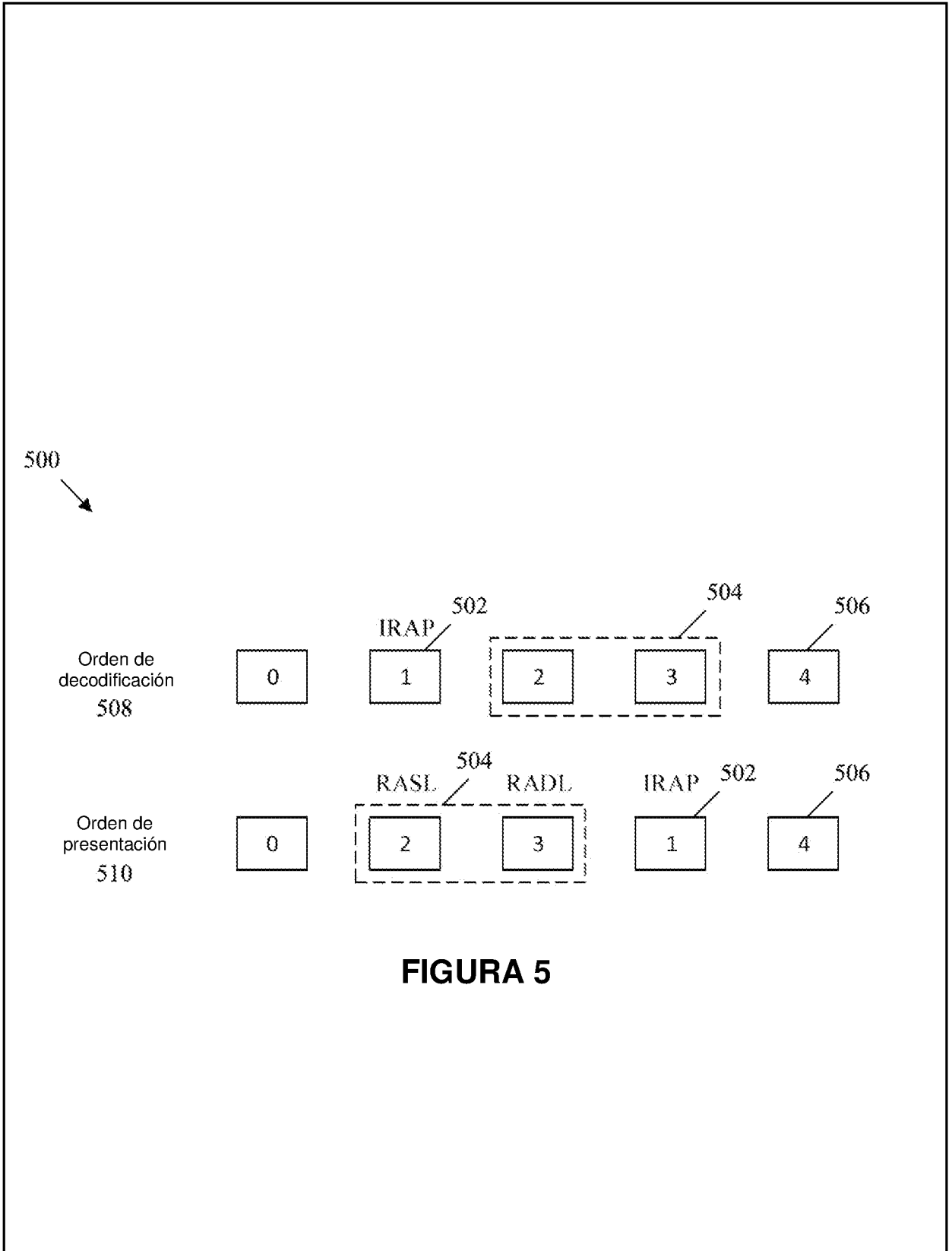
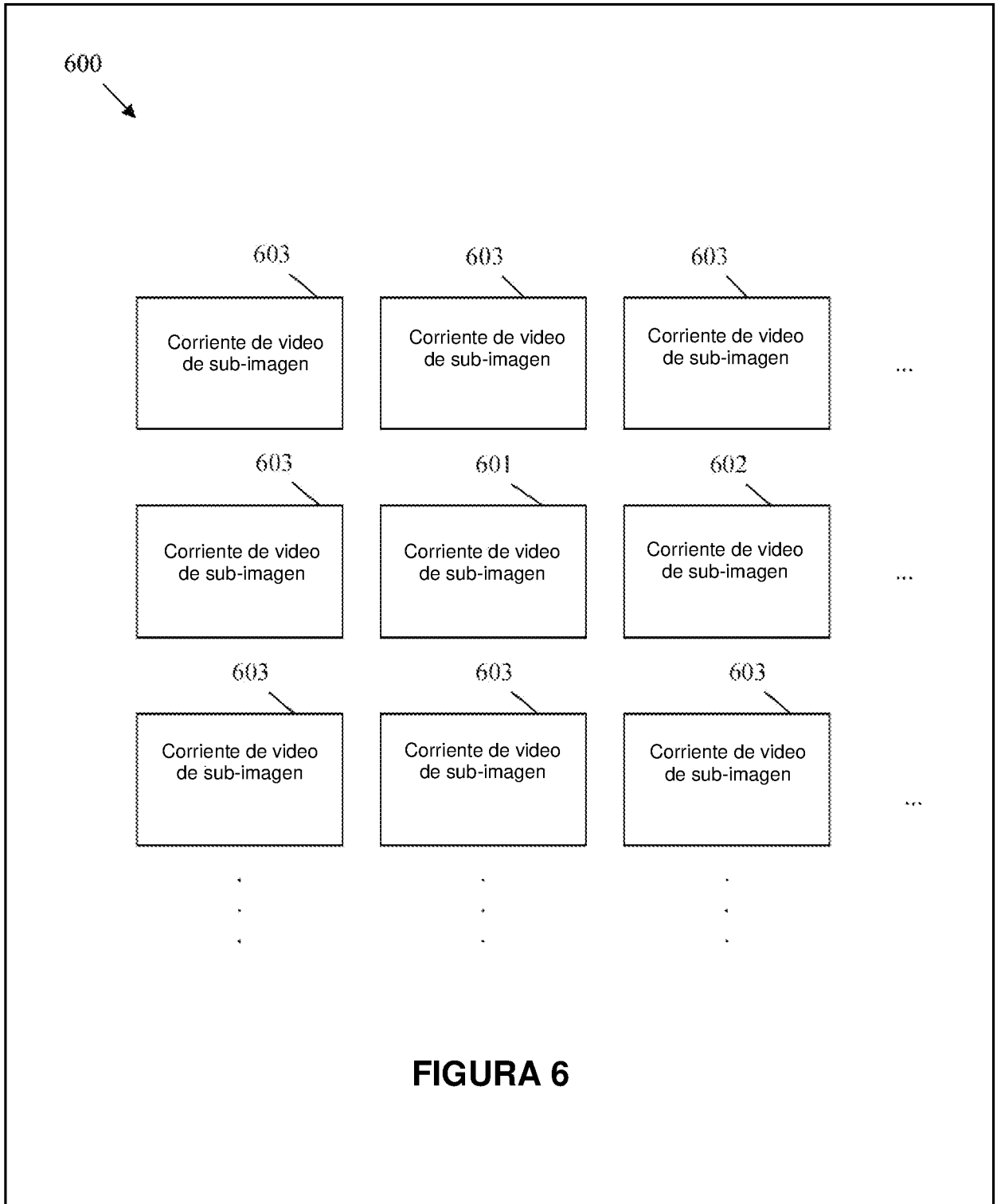


FIGURA 4





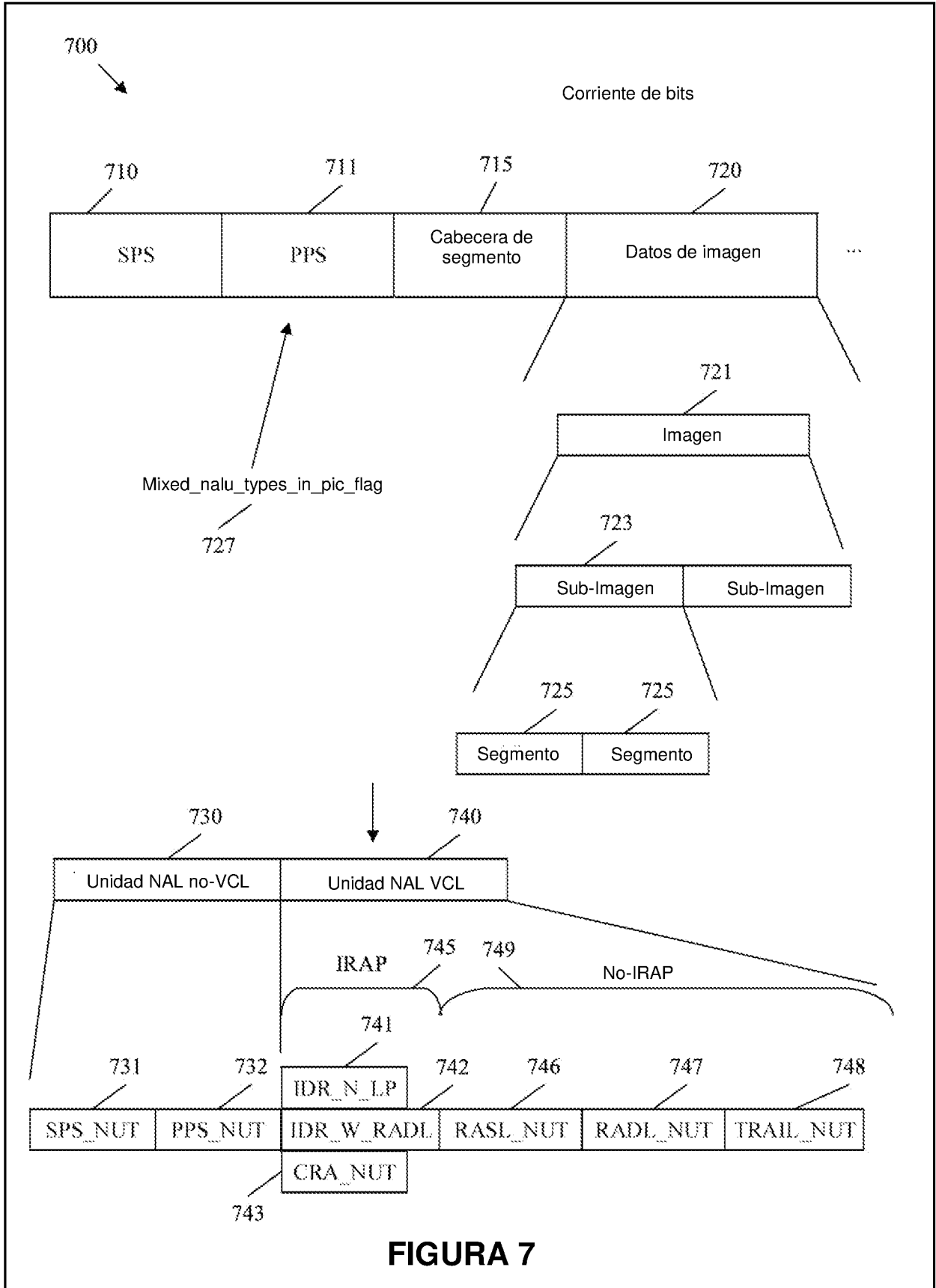


FIGURA 7

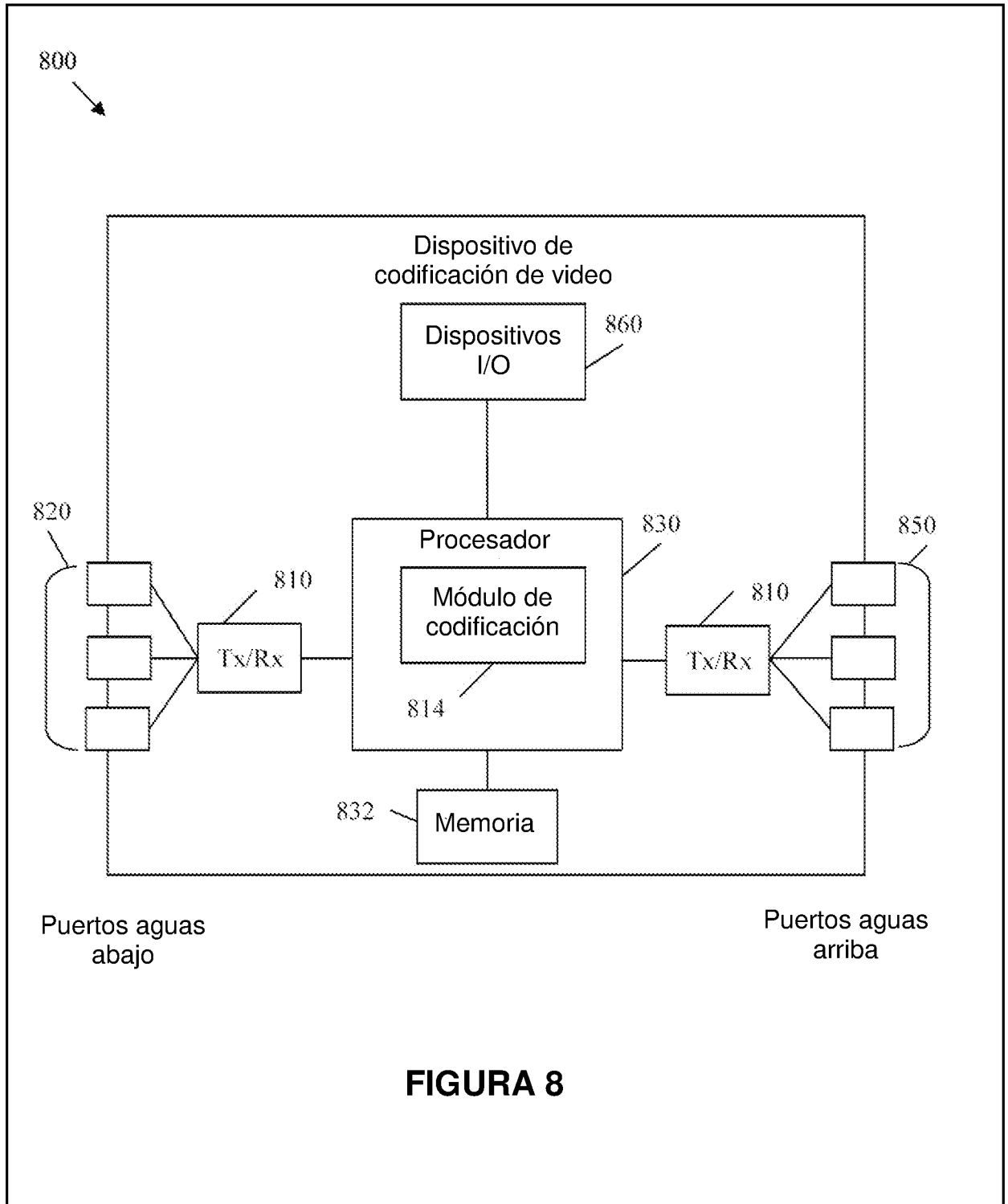


FIGURA 8

