

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 663 444**

51 Int. Cl.:

H04N 19/70 (2014.01)

H04N 19/46 (2014.01)

H04N 19/31 (2014.01)

H04N 19/573 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **08.07.2013 PCT/US2013/049614**

87 Fecha y número de publicación internacional: **16.01.2014 WO14011570**

96 Fecha de presentación y número de la solicitud europea: **08.07.2013 E 13737978 (0)**

97 Fecha y número de publicación de la concesión europea: **03.01.2018 EP 2873244**

54 Título: **Información de temporización de codificación para codificación de vídeo**

30 Prioridad:

10.07.2012 US 201261670066 P
13.03.2013 US 201313802045

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
12.04.2018

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)
5775 Morehouse Drive
San Diego, CA 92121, US

72 Inventor/es:

WANG, YE-KUI

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 663 444 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

Información de temporización de codificación para codificación de vídeo

5 CAMPO TÉCNICO

Esta divulgación se refiere, en general, al procesamiento de datos de vídeo y, más en particular, a imágenes de acceso aleatorio utilizadas en datos de vídeo.

10 ANTECEDENTES

Las capacidades del vídeo digital pueden incorporarse en una amplia gama de dispositivos, incluyendo televisores digitales, sistemas de difusión directa digital, sistemas de difusión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, ordenadores de tableta, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos celulares o de radio por satélite, los denominados teléfonos "inteligentes", dispositivos de videoconferencia, dispositivos de transmisión de flujos de vídeo y similares. Los dispositivos de vídeo digital implementan técnicas de codificación de vídeo, tales como las descritas en las normas definidas por ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262, ISO/IEC MPEG-2 Visual, ITU-T H.263, ITU-T H.264/MPEG-4, Parte 10, Codificación Avanzada de Vídeo (AVC) y la norma de Codificación de Vídeo de Alta Eficacia (HEVC), actualmente en desarrollo, y las extensiones de tales normas. Los dispositivos de vídeo pueden transmitir, recibir, codificar, decodificar y/o almacenar información de vídeo digital más eficazmente, implementando tales técnicas de codificación de vídeo.

Las técnicas de codificación de vídeo incluyen la predicción espacial (intra-imagen) y/o la predicción temporal (entre imágenes) para reducir o eliminar la redundancia intrínseca en las secuencias de vídeo. Para la codificación de vídeo predictiva basada en bloques, un fragmento de vídeo (por ejemplo, una trama de vídeo o una parte de una trama de vídeo) puede dividirse en bloques de vídeo, que también pueden denominarse macro-bloques, bloques arbolados, unidades arboladas de codificación (CTU), bloques arbolados de codificación (CTB), unidades de codificación (CU) y/o nodos de codificación. Los bloques de vídeo en un fragmento intra-codificado (I) de una imagen son codificados usando la predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen. Los bloques de vídeo en un fragmento inter-codificado (P o B) de una imagen pueden usar la predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen, o la predicción temporal con respecto a muestras de referencia en otras imágenes de referencia. Las imágenes pueden denominarse tramas, y las imágenes de referencia pueden denominarse tramas de referencia.

La predicción espacial o temporal da como resultado un bloque predictivo para un bloque a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original a codificar y el bloque predictivo. Los píxeles también pueden denominarse elementos de imagen, pels o muestras. Un bloque inter-codificado se codifica 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 se codifica según una modalidad de intra-codificación y los datos residuales. Para una mayor compresión, los datos residuales pueden transformarse desde el dominio de píxeles a un dominio de transformación, dando como resultado coeficientes de transformación residuales, los cuales pueden cuantizarse posteriormente. Los coeficientes de transformación cuantizados, inicialmente dispuestos en una formación bidimensional, pueden explorarse con el fin de producir un vector unidimensional de coeficientes de transformación, y puede aplicarse codificación por entropía para lograr aún más compresión.

50 SUMARIO

El alcance de la protección está definido en las reivindicaciones adjuntas, a las cuales se debería hacer referencia ahora. En general, esta divulgación describe técnicas para el procesamiento de datos de vídeo. En particular, esta divulgación describe técnicas que se pueden utilizar para reducir los retardos en las aplicaciones de vídeo, tales como aplicaciones de conversación, proporcionar mejoras en el acceso aleatorio de una secuencia de vídeo codificado y proporcionar información para el contenido de vídeo que sea de velocidad de imágenes fija y preste soporte a la ajustabilidad temporal a escala.

En un ejemplo, un procedimiento de presentación de datos de vídeo incluye la determinación de un valor entero para los datos de vídeo, la determinación de un valor de diferencia entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen, en el que el valor de la diferencia es igual al valor entero multiplicado por un valor de pulso de reloj, y la presentación de la primera imagen y la segunda imagen en función del valor de la diferencia determinada.

En otro ejemplo, un dispositivo para la presentación de datos de vídeo incluye un procesador configurado para determinar un valor entero para los datos de vídeo, determinar un valor de diferencia entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen, en el que el valor de la

diferencia es igual al valor entero multiplicado por un valor de pulso de reloj, y presentar la primera imagen y la segunda imagen en función del valor de la diferencia determinada.

5 En otro ejemplo, un dispositivo para la presentación de datos de vídeo incluye medios para determinar un valor entero para los datos de vídeo, medios para determinar un valor de diferencia entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen, en el que el valor de la diferencia es igual al valor entero multiplicado por un valor de pulso de reloj, y medios para la presentación de la primera imagen y la segunda imagen según el valor de la diferencia determinada.

10 En otro ejemplo, un medio de almacenamiento legible por ordenador tiene almacenadas en el mismo instrucciones que, al ejecutarse, hacen que un procesador determine un valor entero para los datos de vídeo, determine un valor de diferencia entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen, en donde el valor de la diferencia es igual al valor entero multiplicado por un valor de pulso de reloj, y presente la primera imagen y la segunda imagen según el valor de la diferencia determinada.

15 En otro ejemplo, un procedimiento de generación de un flujo de bits que incluye datos de vídeo incluye la generación de datos que indican si una diferencia entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen es o no un múltiplo entero de un valor de pulso de reloj y, cuando los datos indican que la diferencia es el múltiplo entero del valor de pulso de reloj, la generación de datos representativos del múltiplo entero.

20 En otro ejemplo, un dispositivo para generar un flujo de bits que incluye datos de vídeo incluye un procesador configurado para generar datos que indican si una diferencia entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen es o no un múltiplo entero de un valor de pulso de reloj y, cuando los datos indican que la diferencia es el múltiplo entero del valor de pulso de reloj, generar datos representativos del múltiplo entero.

25 En otro ejemplo, un dispositivo para generar un flujo de bits que incluye datos de vídeo incluye medios para generar datos que indican si una diferencia entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen es o no un múltiplo entero de un valor de pulso de reloj, y medios para generar, cuando los datos indican que la diferencia es el múltiplo entero del valor de pulso de reloj, datos representativos del múltiplo entero.

30 En otro ejemplo, un medio de almacenamiento legible por ordenador tiene almacenadas en el mismo instrucciones que, al ejecutarse, hacen que un procesador genere datos que indican si una diferencia entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen es un múltiplo entero de un valor de pulso de reloj; y, cuando los datos indican que la diferencia es el múltiplo entero del valor de pulso de reloj, genere datos representativos del múltiplo entero.

35 Los detalles de uno o más ejemplos se exponen en los dibujos adjuntos y en la siguiente descripción. Otras características, objetivos y ventajas resultarán evidentes a partir de la descripción y los dibujos, y a partir de las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

45 La FIG. 1 es un diagrama conceptual que ilustra una secuencia de vídeo codificada según las técnicas predictivas de codificación de vídeo.

50 La FIG. 2 es un diagrama conceptual que ilustra un ejemplo de una secuencia de vídeo codificada.

La FIG. 3 es un diagrama de bloques que ilustra un sistema ejemplar de codificación y decodificación de vídeo que puede utilizar las técnicas descritas en esta divulgación.

55 La FIG. 4 es un diagrama de bloques que ilustra una unidad ejemplar de encapsulación que puede implementar las técnicas descritas en esta divulgación.

La FIG. 5 es un diagrama de flujo que ilustra un ejemplo de generación de unidades de NAL de VCL según las técnicas de esta divulgación.

60 La FIG. 6 es un diagrama de flujo que ilustra un ejemplo de generación de unidades de NAL no de VCL, según las técnicas de esta divulgación.

65 La FIG. 7 es un diagrama de flujo que ilustra un ejemplo de señalización de un valor de delta de tiempo de presentación.

La FIG. 8 es un diagrama de bloques que ilustra un codificador ejemplar de vídeo que puede implementar las

técnicas descritas en esta divulgación.

La FIG. 9 es un diagrama de flujo que ilustra un ejemplo de determinación de un valor de delta de tiempo de presentación.

La FIG. 10 es un diagrama de bloques que ilustra un decodificador ejemplar de vídeo que puede implementar las técnicas descritas en esta divulgación.

DESCRIPCIÓN DETALLADA

Esta divulgación describe diversos diseños mejorados de codificación de vídeo. En particular, esta divulgación describe técnicas que se pueden utilizar para reducir los retardos en las aplicaciones de vídeo, tales como aplicaciones de conversación, y proporcionar mejoras en el acceso aleatorio de una secuencia de vídeo codificada.

Los dispositivos de vídeo digital implementan técnicas de compresión de vídeo para codificar y decodificar de manera más eficaz la información de vídeo digital. Las técnicas de compresión de vídeo pueden definirse según una norma de codificación de vídeo, tal como AVC o HEVC. La norma ITU-T H.264 / MPEG-4 (AVC) fue formulada por el Grupo de Expertos en Codificación de Vídeo (VCEG) de la ITU-T junto con el Grupo de Expertos en Codificación de Imágenes en Movimiento (MPEG) de ISO / IEC, como el producto de una asociación colectiva conocida como el Equipo de Vídeo Conjunto (JVT). La norma H.264 se describe en la Recomendación H.264 de la UIT-T, Codificación de Vídeo Avanzada para servicios audiovisuales genéricos, por el Grupo de Estudio de la ITU-T, y con fecha de marzo de 2005, que puede denominarse en este documento la norma H.264 o la memoria descriptiva H.264, o la norma o memoria descriptiva H.264 / AVC. El Equipo de Vídeo Conjunto (JVT) continúa trabajando en extensiones para la norma H.264 / MPEG-4 AVC.

Un borrador de trabajo (WD) reciente de la norma HEVC, denominado "Borrador de trabajo HEVC 7" o "WD7", se describe en el documento JCTVC-I1003,d5 de Bross et al., "WD7: Borrador de trabajo 7 de codificación de vídeo de alta eficacia (HEVC)", Equipo de Colaboración Conjunta sobre Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 9^a conferencia: Ginebra, Suiza, 27 de abril de 2012 hasta 7 de mayo de 2012. Además, otro reciente borrador de trabajo de la HEVC, borrador de trabajo 9, se describe en el documento HCTVC-K1003_d7, Bross et al. "Borrador 9 de memoria descriptiva textual de codificación de vídeo de alta eficiencia (HEVC), Equipo de Colaboración Conjunta sobre Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 11^a conferencia, Shanghai, China, octubre de 2012. La inminente norma HEVC también puede denominarse ISO/IEC 23008-HEVC, que se pretende que sea el número de la norma para la versión publicada de la HEVC. En algunos aspectos, las técnicas descritas en esta divulgación pueden aplicarse a dispositivos que se ajustan en general a la norma H.264 y/o a la inminente norma HEVC. Aunque las técnicas de esta divulgación se describen con respecto a la norma H.264 y la inminente norma HEVC, las técnicas de esta divulgación pueden aplicarse, por lo general, a cualquier norma de codificación de vídeo.

Una secuencia de vídeo incluye normalmente una serie de tramas de vídeo, también denominadas imágenes. Los ejemplos de aplicaciones de vídeo, donde se codifica y/o decodifica una secuencia de vídeo, incluyen la reproducción local, el flujo de transmisión, la difusión, la multidifusión y las aplicaciones conversacionales. Las aplicaciones conversacionales incluyen la videotelefonía y la videoconferencia, y también se mencionan como aplicaciones de bajo retardo. Las aplicaciones conversacionales requieren un retardo relativamente bajo de extremo a extremo de la totalidad de los sistemas, es decir, el retardo entre el momento en que se captura una trama de vídeo en un primer dispositivo de vídeo digital y el momento en que se muestra la trama de vídeo en un segundo dispositivo de vídeo digital. Normalmente, el retardo aceptable de extremo a extremo para aplicaciones conversacionales debería ser de menos de 400 ms, y se considerara muy bueno un retardo de extremo a extremo de alrededor de 150 ms.

Cada etapa asociada al procesamiento de una secuencia de vídeo puede contribuir al retardo global de extremo a extremo. Los ejemplos de retardos asociados al procesamiento de una secuencia de vídeo incluyen el retardo de captura, el retardo de pre-procesamiento, el retardo de codificación, el retardo de transmisión, el retardo de almacenamiento en memoria intermedia de recepción (para eliminar la arritmia), el retardo de decodificación, el retardo de la salida de imagen decodificada, el retardo de pos-procesamiento y el retardo de visualización. El retardo asociado a la codificación de una secuencia de vídeo, según un estándar de codificación de vídeo en particular, puede denominarse el retardo de códec y puede incluir un retardo de codificación, un retardo de decodificación y un retardo de salida de imagen decodificada. El retardo de códec debería reducirse al mínimo en aplicaciones conversacionales. En particular, la estructura de codificación de una secuencia de vídeo debería garantizar que el orden de salida de las imágenes en la secuencia de vídeo es idéntico al orden de decodificación de imágenes en la secuencia de vídeo, de manera que el retardo de la salida de la imagen decodificada sea igual a cero. La estructura de codificación de una secuencia de vídeo se refiere, en parte, a la asignación de tipos de imagen utilizados para codificar una secuencia de vídeo.

Un grupo de imágenes (GOP) comprende generalmente una secuencia de una o más imágenes dispuestas según el orden de visualización. Según la HEVC, un codificador de vídeo puede dividir una trama o imagen de vídeo en una

serie de bloques de vídeo de igual tamaño. Un bloque de vídeo puede tener un componente de luminancia (indicado como Y) y dos componentes de crominancia (indicados como U y V o Cb y Cr). Estos bloques de vídeo también pueden denominarse unidades máximas de codificación (LCU), bloques arbolados o unidades de bloques arbolados de codificación (CTU). Las LCU de la HEVC puede ser mayormente análogas a los macro-bloques de las normas anteriores, tales como H.264 / AVC. Sin embargo, una LCU no se limita necesariamente a un tamaño particular. Según la HEVC, los datos sintácticos dentro de un flujo de bits pueden definir una LCU según el número de muestras de luminancia horizontales y/o verticales. Por ejemplo, una LCU se puede definir como incluyente de muestras de luminancia de dimensión 64x64 o 32x32. Además, una LCU puede dividirse en múltiples unidades de codificación (CU) según un esquema de partición de árbol de cuatro ramas. En general, la partición del árbol de cuatro ramas se refiere a la división recursiva de las CU en cuatro sub-CU. Los datos sintácticos asociados a un flujo de bits codificado pueden definir un número máximo de veces que puede dividirse una LCU, mencionado como una profundidad de CU máxima, y también pueden definir un tamaño mínimo de una CU. Por consiguiente, un flujo de bits también puede definir la unidad de codificación más pequeña (SCU). Por ejemplo, una SCU puede definirse como incluyente de muestras de luminancia de dimensión 8x8.

Además, según la HEVC, un codificador de vídeo puede dividir una imagen en una pluralidad de fragmentos, en el que cada uno de los fragmentos incluye un número entero de las LCU. Los fragmentos pueden ser fragmentos I, fragmentos P o fragmentos B, donde I, P y B definen cómo se utilizan otros bloques de vídeo para predecir las CU. Se predice un fragmento I utilizando una modalidad de intra-predicción (por ejemplo, a partir de bloques de vídeo dentro de la misma trama). La intra-codificación se apoya en la predicción espacial para reducir o eliminar la redundancia espacial en el vídeo dentro de una trama o imagen de vídeo dada. Un fragmento P se predice usando una modalidad de inter-predicción unidireccional (por ejemplo, desde el bloque de vídeo en una trama anterior). Un fragmento B se predice usando una modalidad de inter-predicción bidireccional (por ejemplo, a partir de bloques de vídeo dentro de una trama anterior y una trama posterior). La inter-codificación se apoya en la predicción temporal para reducir o eliminar la redundancia temporal en el vídeo dentro de tramas o imágenes adyacentes de una secuencia de vídeo.

La FIG. 1 es un diagrama conceptual que ilustra la secuencia de vídeo, codificada según las técnicas predictivas de codificación de vídeo. Como se ilustra en la FIG. 1, la secuencia de vídeo 100 incluye las imágenes Pic₁ a Pic₁₀. En el diagrama conceptual de la FIG. 1, las imágenes Pic₁ a Pic₁₀ están dispuestas y secuencialmente numeradas según el orden en que se van a mostrar. Como se describe en más detalle a continuación, el orden de visualización no se corresponde necesariamente con el orden de decodificación. Como se ilustra en la FIG. 1, la secuencia de vídeo 100 incluye el GOP₁ y el GOP₂, donde las imágenes Pic₁ a Pic₅ se incluyen en el GOP₁ y las imágenes Pic₆ a Pic₁₀ están incluidas en el GOP₂. La FIG. 1 ilustra donde Pic₅ se divide en el fragmento₁ y el fragmento₂, donde tanto el fragmento₁ como el fragmento₂ incluyen LCU consecutivas, según un barrido de cuadrícula de izquierda a derecha y de arriba a abajo. Aunque no se muestra, las otras imágenes ilustradas en la FIG. 1 pueden dividirse en uno o más fragmentos de una manera similar. La FIG. 1 también ilustra el concepto de fragmentos I, fragmentos P o fragmentos B con respecto al GOP₂. Las flechas asociadas a cada una entre Pic₆ y Pic₁₀ en el GOP₂ indican si una imagen incluye fragmentos I, fragmentos P o fragmentos B, en base a una imagen referida, indicada por las flechas. En la FIG. 1, las imágenes Pic₆ y Pic₉ representan unas imágenes que incluyen fragmentos I (es decir, las referencias son a la propia imagen), las imágenes Pic₇ y Pic₁₀ representan imágenes que incluyen fragmentos P (es decir, cada una hace referencia a una imagen anterior) y Pic₈ representa una imagen que incluye fragmentos B (es decir, hace referencia a una imagen anterior y a una posterior).

En la HEVC, cada uno entre una secuencia de vídeo, un GOP, una imagen, un fragmento y una CU puede estar asociado a datos sintácticos que describen las propiedades de codificación de vídeo. Por ejemplo, un fragmento incluye una cabecera que incluye un elemento sintáctico que indica si el fragmento es un fragmento I, un fragmento P o un fragmento B. Además, la HEVC incluye el concepto de conjuntos de parámetros. Un conjunto de parámetros es una estructura sintáctica que incluye elementos sintácticos que permiten a un decodificador de vídeo reconstruir una secuencia de vídeo. La HEVC emplea un mecanismo jerárquico de conjuntos de parámetros, donde los elementos sintácticos se incluyen en un tipo de conjunto de parámetros, en base a la frecuencia en la que se espera que los elementos sintácticos cambien. El mecanismo de conjuntos de parámetros en la HEVC desacopla la transmisión de información que cambia infrecuentemente de la transmisión de datos de bloques codificados. Además, en algunas aplicaciones, los conjuntos de parámetros pueden transmitirse "fuera de banda", es decir, no transportarse junto con las unidades que contienen datos de vídeo codificados. La transmisión fuera de la banda es normalmente fiable.

En el HEVC WD7, un conjunto determinado de parámetros se identifica mediante un identificador de conjunto de parámetros. En el HEVC WD7, un identificador de conjunto de parámetros es un elemento sintáctico entero sin signo con código Exp-Golomb, con el bit izquierdo en primer lugar. El HEVC WD7 define los siguientes conjuntos de parámetros:

Conjunto de parámetros de vídeo (VPS) Un VPS es una estructura sintáctica que contiene elementos sintácticos que se aplican a cero o más secuencias enteras de vídeo codificado. Es decir, un VPS incluye elementos sintácticos de los que se espera que permanezcan sin cambios durante una secuencia de tramas (por ejemplo, orden de imágenes, número de tramas de referencia y tamaño de imágenes). Un VPS se identifica

mediante un identificador de VPS. Un conjunto de parámetros de secuencia incluye un identificador de VPS.

Conjunto de parámetros de secuencia (SPS) - Un SPS es una estructura sintáctica que incluye elementos sintácticos que se aplican a cero o más secuencias enteras de vídeo codificado. Es decir, un SPS incluye elementos sintácticos de los que se espera que permanezcan sin cambios durante una secuencia de tramas (por ejemplo, orden de imágenes, número de tramas de referencia y tamaño de imágenes). Un SPS se identifica usando un identificador de SPS. Un conjunto de parámetros de imagen incluye un Identificador de SPS.

Conjunto de parámetros de imagen (PPS) - Un PPS es una estructura sintáctica que incluye elementos sintácticos que se aplican a una o más imágenes. Es decir, un PPS incluye elementos sintácticos que pueden cambiar de imagen a imagen dentro de una secuencia (por ejemplo, modalidad de codificación por entropía, parámetros de cuantización y profundidad de bits). Un conjunto de parámetros PPS se identifica utilizando un Identificador de PPS. Una cabecera de fragmento incluye un identificador de PPS.

Conjunto de parámetros adaptativos (APS) - Un APS es una estructura sintáctica que incluye elementos sintácticos que se aplican a una o más imágenes. Un APS incluye elementos sintácticos de los que se espera que cambien dentro de las imágenes de una secuencia (por ejemplo, el tamaño del bloque y el filtrado de desbloqueo). Un conjunto de APS se identifica con un Identificador de APS. Una cabecera de fragmento puede incluir un Identificador de APS.

Según los tipos de conjuntos de parámetros definidos en el HEVC WD7, cada SPS se refiere a un Identificador de VPS, cada PPS se refiere a un Identificador de SPS y cada cabecera de fragmento se refiere a un Identificador de PPS y, posiblemente, a un identificador de APS. Cabe señalar que, en algunos casos, la relación de referencia lineal de incluir un Identificador de VPS en un SPS y un Identificador de SPS en un PPS puede ser ineficaz. Por ejemplo, aunque el VPS tiene soporte en el HEVC WD7, la mayor parte de los parámetros de información de nivel de secuencia, sin embargo, solo están presentes en el SPS. Además del concepto de conjuntos de parámetros, la HEVC incluye el concepto de secuencias de vídeo codificado y unidades de acceso. Según el HEVC WD7, una secuencia de vídeo codificado y una unidad de acceso se definen de la forma siguiente:

Secuencia de vídeo codificado: Una secuencia de unidades de acceso que consta, en orden de decodificación, de una unidad de acceso CRA, que es la primera unidad de acceso en el flujo de bits, una unidad de acceso IDR o una unidad de acceso BLA, seguida de cero o más unidades de acceso no IDR y no BLA, incluyendo todas las unidades de acceso posteriores hasta cualquier, pero sin incluir ninguna, unidad de acceso IDR o BLA [las unidades de acceso CRA, IDR y BLA se describen en detalle más adelante].

Unidad de acceso: Un conjunto de unidades de NAL que son consecutivas en el orden de decodificación y que contienen una imagen codificada. Además de las unidades de NAL de fragmentos codificados de la imagen codificada, la unidad de acceso puede contener también otras unidades de NAL que no contienen fragmentos de la imagen codificada. La decodificación de una unidad de acceso siempre da como resultado una imagen decodificada.

Una unidad de NAL se refiere a una unidad de la capa de abstracción de red. Por lo tanto, según la HEVC, un flujo de bits de datos de vídeo codificados incluye una secuencia de unidades de NAL. Una unidad de acceso es un conjunto de unidades de NAL que están dispuestas consecutivamente en el orden de decodificación y que contienen exactamente una imagen codificada, y una secuencia de vídeo codificado incluye una secuencia de unidades de acceso dispuestas en el orden de decodificación. La FIG. 2 es un diagrama conceptual que ilustra un ejemplo de una secuencia de vídeo codificado. La FIG. 2 representa un ejemplo de una secuencia de vídeo codificado 200 que puede corresponder al GOP_2 ilustrado en la FIG. 1. Como se ilustra en la FIG. 2, la secuencia de vídeo codificado 200 incluye una unidad de acceso correspondiente a las Pic_6 a Pic_{10} . Las unidades de acceso de la secuencia de vídeo codificado 200 se disponen secuencialmente según el orden de decodificación. Debería tenerse en cuenta que la unidad de acceso correspondiente a la Pic_9 se encuentra antes de la unidad de acceso correspondiente a la Pic_8 . Por lo tanto, el orden de decodificación no se corresponde con el orden de visualización que se ilustra en la FIG. 1. En este ejemplo, esto se debe al hecho de que la Pic_8 hace referencia a la Pic_9 . Por lo tanto, la Pic_9 debe decodificarse antes de que la Pic_8 pueda decodificarse. La FIG. 2 ilustra donde la unidad de acceso correspondiente a la Pic_9 incluye las unidades de NAL: unidad de NAL de delimitador AU 202, unidad de NAL de PPS 204, unidad de NAL del fragmento₁ 206 y unidad de NAL del fragmento₂ 208. Cada unidad de NAL puede incluir una cabecera que identifica el tipo de unidad de NAL.

La HEVC define dos clases de tipos de unidades de NAL: unidades de NAL de fragmento codificado (VCL [Capa de Codificación de Vídeo]) y unidades de NAL no de VCL. Una unidad de NAL de fragmento codificado contiene un fragmento de datos de vídeo. En el ejemplo ilustrado en la FIG. 2, la unidad de NAL del fragmento₁ 206 y la unidad de NAL del fragmento₂ 208 contienen, cada una, un fragmento de datos de vídeo y son ejemplos de unidades de NAL de VCL. En el ejemplo de la FIG. 2, la unidad de NAL del fragmento₁ 206 y la unidad de NAL del fragmento₂ 208 pueden ser fragmentos I. Una no-VCL incluye que contiene información que no sea un fragmento de datos de vídeo. Por ejemplo, una no-VCL puede contener datos de delimitador o un conjunto de parámetros. En el ejemplo ilustrado en la FIG. 2, la unidad de NAL de delimitador de AU 202 incluye información para delimitar la unidad de acceso correspondiente a la Pic_9 a partir de la unidad de acceso correspondiente a la Pic_7 . Además, la unidad de

NAL de PPS 204 incluye un conjunto de parámetros de imagen. Por lo tanto, la unidad de NAL de delimitador de AU 202 y la unidad de NAL de PPS 204 son ejemplos de unidades de NAL no de VCL.

Otro ejemplo de una unidad de NAL no de VCL en la HEVC es la unidad de NAL de información de mejora suplementaria (SEI). El mecanismo de SEI, con soporte tanto en la AVC como en la HEVC, permite a los codificadores incluir metadatos en el flujo de bits que no se requieren para la correcta decodificación de los valores de muestra de las imágenes de salida, pero se pueden utilizar para otros diversos fines, tales como la temporización y visualización de salida de imágenes, así como la detección y la ocultación de pérdidas. Por ejemplo, las unidades de NAL de SEI pueden incluir mensajes de temporización de imágenes, que son utilizados por un decodificador de vídeo cuando se decodifica un flujo de bits. Los mensajes de temporización de imágenes pueden incluir información que indica cuándo un decodificador de vídeo debería empezar a decodificar una unidad de NAL de VCL. Los codificadores pueden incluir cualquier número de unidades de NAL de SEI en una unidad de acceso, y cada unidad de NAL de SEI puede contener uno o más mensajes de SEI. El borrador de la norma HEVC incluye la sintaxis y la semántica de varios mensajes de SEI, pero la gestión de los mensajes de SEI no se especifica, ya que no afectan al proceso de decodificación normativo. Una razón para tener mensajes de SEI en el borrador de la norma HEVC es permitir que los datos suplementarios sean interpretados de forma idéntica en los diferentes sistemas que utilizan la HEVC. Las especificaciones y los sistemas que utilizan la HEVC pueden requerir codificadores para generar ciertos mensajes de SEI o pueden definir una gestión específica de determinados tipos de mensajes de SEI recibidos. La Tabla 1 enumera los mensajes de SEI especificados en la HEVC y describe brevemente sus propósitos.

TABLA 1: Visión general de mensajes de SEI

Mensaje de SEI	Propósito
Período de almacenamiento temporal	Retardos iniciales para el funcionamiento del decodificador de referencia hipotética (HRD)
Temporización de imágenes	Tiempo de salida de la imagen y tiempo de eliminación de imagen / sub-imagen para el funcionamiento del HRD, así como información relacionada con la estructura de la imagen
Rectángulo escaneado panorámico	Exhibición con una razón de aspecto de imagen (PAR) diferente a la PAR de las imágenes de salida
Carga útil de relleno	Ajuste de la velocidad de bits para satisfacer limitaciones específicas
Datos de usuario registrados - Datos de usuario no registrados	Mensajes de SEI a especificar por parte de entidades externas
Punto de recuperación	Información adicional para acceso aleatorio puro. Actualización gradual de decodificación.
Información de escena	Información sobre cambios de escena y transiciones
Instantánea a pantalla completa	Indicación para etiquetar la imagen decodificada asociada como una instantánea de imagen fija del contenido de vídeo
Segmento de refinamiento progresivo	Indica que ciertas imágenes consecutivas representan un refinamiento progresivo de la calidad de una imagen, en lugar de una escena en movimiento
Características de textura de película	Permite a los decodificadores sintetizar la textura de la película
Preferencia de exhibición de filtro desbloqueador	Recomienda si las imágenes exhibidas deberían someterse o no al proceso del filtro desbloqueador de bucle interno
Sugerencia de pos-filtro	Proporciona coeficientes de pos-filtro sugeridos o información de correlación para el diseño de pos-filtro
Información de correlación tonos	Re-correlación con otro espacio de colores distinto al usado o adoptado en la codificación
Disposición de empaquetamiento de tramas	Empaquetamiento de vídeo estereoscópico en un flujo de bits de HEVC
Orientación de la pantalla	Especifica el vuelco y/o la rotación que deberían aplicarse a las imágenes de salida cuando se exhiben
Estructura de descripción	Describe la estructura temporal y de inter-predicción del flujo de bits

Mensaje de SEI	Propósito
imágenes	
Troceo de imagen decodificada	Suma de comprobación de la imagen decodificada, que puede ser utilizada para la detección de errores
Conjuntos de parámetros activos	Proporciona información sobre VPS, SPS, etc., activos
Información de unidad de decodificación	Tiempo de eliminación de sub-imagen para el funcionamiento del HRD, así como el índice de unidad de decodificación
Índice cero del nivel temporal	Proporciona valores de índice cero de nivel temporal
Anidamiento ajustable a escala	Proporciona un mecanismo para anidar mensajes de SEI para la asociación a diferentes puntos de operación y capas
Información de actualización de regiones	Proporciona información sobre regiones actualizadas y no actualizadas para la actualización de decodificación gradual

El acceso aleatorio se refiere a la decodificación de un flujo de bits de vídeo a partir de una imagen codificada que no es la primera imagen codificada en el flujo de bits. Se necesita acceso aleatorio a un flujo de bits en muchas aplicaciones de vídeo, tales como la radiodifusión y la transmisión por flujo, por ejemplo, para que los usuarios conmuten entre diferentes canales, para saltar a partes específicas del vídeo, o para conmutar a un flujo de bits diferente para adaptación del flujo (por ejemplo, para la velocidad de bits, la velocidad de tramas o la ajustabilidad a escala de resolución espacial). El acceso aleatorio se activa teniendo una estructura de codificación que incluye imágenes de punto de acceso aleatorio (RAP) o unidades de acceso, muchas veces a intervalos regulares, para una secuencia de vídeo. Las imágenes de actualización de decodificador instantánea (IDR), las imágenes de acceso aleatorio puro (CRA) y las imágenes de acceso a enlace roto (BLA) son tipos de imágenes de RAP definidas en el HEVC WD7. Cada una de las imágenes de IDR, imágenes de CRA e imágenes de BLA incluye solo fragmentos I. Sin embargo, cada una de las imágenes de IDR, imágenes de CRA e imágenes de BLA difiere, en base a las limitaciones de referencias definidas.

Las imágenes de IDR se especifican en la AVC y se definen según el HEVC WD7. Si bien las imágenes de IDR pueden utilizarse para el acceso aleatorio, las imágenes de IDR están limitadas en cuanto a que las imágenes siguientes a una imagen de IDR en el orden de decodificación no pueden utilizar imágenes decodificadas antes de la imagen de IDR como referencia. En el ejemplo ilustrado en las FIGs. 1 y 2, como se ha descrito anteriormente, la Pic_8 en la secuencia de vídeo 100 puede ser una imagen de IDR. Debido a las limitaciones asociadas a las imágenes de IDR, los flujos de bits que se apoyan en las imágenes de IDR para el acceso aleatorio pueden tener una eficacia de codificación significativamente menor.

Para mejorar la eficacia de la codificación, se introdujo el concepto de imágenes de CRA en la HEVC. Según el HEVC WD7, una imagen de CRA, como una imagen de IDR, incluye solo fragmentos I. Sin embargo, a las imágenes que siguen a una imagen de CRA en orden de decodificación, pero que preceden a las imágenes de CRA en orden de salida, se les permite usar las imágenes decodificadas antes de la imagen de CRA como referencia. Las imágenes que siguen a una imagen de CRA en orden de decodificación, pero que preceden a la imagen de CRA en el orden de salida, se denominan imágenes principales asociadas a la imagen de CRA (o imágenes principales de la imagen de CRA). Las imágenes principales de una imagen de CRA son decodificables correctamente si la decodificación se inicia a partir de una imagen de IDR o de CRA antes de la imagen de CRA actual. Sin embargo, las imágenes principales de una imagen de CRA tal vez no sean decodificables correctamente cuando se produce el acceso aleatorio desde la imagen de CRA. En el ejemplo ilustrado en las FIGs. 1 y 2, la Pic_9 puede ser una imagen de CRA y la Pic_8 puede ser una imagen principal de la Pic_9 . La Pic_8 es decodificable correctamente si se accede al GOP_2 en la Pic_6 , pero tal vez no sea decodificable correctamente si se accede al GOP_2 en la Pic_9 . Esto se debe al hecho de que la Pic_7 puede no estar disponible si se accede al GOP_2 en la Pic_9 . Para evitar la propagación de errores de imágenes de referencia que pueden no estar disponibles, en función del lugar donde se inicia la decodificación, según el HEVC WD7, todas las imágenes que siguen a una imagen de CRA, tanto en el orden de decodificación como en el orden de salida, se ven limitadas a no utilizar ninguna imagen que preceda a la imagen de CRA, ya sea en el orden de decodificación o en el orden de salida (que incluye las imágenes principales), como referencia. Además, las imágenes principales se descartan normalmente durante la decodificación del acceso aleatorio.

El empalme de flujos de bits se refiere a la concatenación de dos o más flujos de bits o partes de los mismos. Por ejemplo, un primer flujo de bits puede tener adosado un segundo flujo de bits, posiblemente con algunas modificaciones en uno cualquiera de, o ambos, flujos de bits para generar un flujo de bits empalmado. La primera imagen codificada en el segundo flujo de bits también se conoce como el punto de empalme. Por lo tanto, las imágenes después del punto de empalme en el flujo de bits empalmado se originaron a partir del segundo flujo de

bits, mientras que las imágenes anteriores al punto de empalme en el flujo de bits empalmado se originaron a partir del primer flujo de bits. El empalme de flujos de bits se realiza normalmente mediante empalmadores de flujos de bits. Los empalmadores de flujos de bits son a menudo ligeros y mucho menos inteligentes que los codificadores de vídeo. Por ejemplo, un empalmador de flujos de bits puede no estar equipado con capacidades de decodificación y codificación por entropía. La ajustabilidad temporal a escala es una aplicación que puede utilizar el empalme de flujo de bits. La ajustabilidad temporal a escala puede referirse a la decodificación de una secuencia de vídeo en una o más velocidades de tramas. Por ejemplo, una secuencia de vídeo puede poder decodificarse a 30 tramas por segundo (fps) o 60 fps, en base a las capacidades del sistema. Para lograr la ajustabilidad temporal a escala, una secuencia de vídeo puede incluir una pluralidad de capas temporales. Donde cada capa temporal es una secuencia de vídeo codificada asociada a una velocidad de tramas. La capa temporal con la velocidad de tramas más alta puede denominarse la capa temporal más alta. Una pluralidad de capas temporales se pueden empalmar entre sí para generar la secuencia de vídeo a la velocidad de tramas más alta, por ejemplo, la secuencia de vídeo codificado con 30 fps se empalma con la secuencia de vídeo codificado que permite 60 fps.

La conmutación de flujos de bits puede utilizarse en entornos de transmisión por flujo adaptativa. Una operación de conmutación de flujo de bits en cierta imagen en el flujo de bits al que se conmuta es efectivamente una operación de empalme de flujo de bits en la que el punto de empalme es el punto de conmutación del flujo de bits, es decir, la primera imagen procedente del flujo de bits al que se conmuta. Cabe señalar que la conmutación de flujos de bits se realiza generalmente sobre dos flujos con la misma estructura de codificación. Es decir, los dos flujos tienen la misma estructura de predicción y la misma asignación de imágenes de IDR, imágenes de CRA, imágenes P e imágenes B, etc.

El concepto de una imagen de acceso a enlace roto (BLA) se introdujo adicionalmente en el HEVC WD7 después de la introducción de imágenes de CRA, y se basa en el concepto de imágenes de CRA. Una imagen de BLA se origina generalmente a partir del empalme de flujos de bits en la posición de una imagen de CRA y, en el flujo de bits empalmado, la imagen de CRA del punto de empalme cambia a una imagen de BLA. La diferencia más esencial entre imágenes de BLA e imágenes de CRA es la siguiente: para una imagen de CRA, las imágenes principales asociadas son correctamente decodificables si la decodificación se inicia desde una imagen de RAP antes de la imagen de CRA en el orden de decodificación, y pueden no ser correctamente decodificables cuando el acceso aleatorio se inicia desde la imagen de CRA; para una imagen de BLA, las imágenes principales asociadas tal vez no sean correctamente decodificables en todos los casos, incluso cuando la decodificación comienza desde una imagen de RAP antes de la imagen de BLA en el orden de decodificación. Cabe señalar que, para una imagen de CRA o de BLA en particular, algunas de las imágenes principales asociadas son decodificables correctamente incluso cuando la imagen de CRA o de BLA es la primera imagen en el flujo de bits. Estas imágenes principales se denominan imágenes principales decodificables (DLP), y otras imágenes principales se denominan imágenes principales no decodificables (NLP). Las NLP también se mencionan como imágenes etiquetadas para el descarte (TFD) en el HEVC WD9. Cabe señalar que todas las imágenes principales asociadas a una imagen de IDR son imágenes DLP. La tabla 2 es una tabla incluida en el HEVC WD7 que especifica las unidades de NAL definidas según el HEVC WD7. Como se ilustra en la tabla 2, los tipos de unidades de NAL en el HEVC WD7 incluyen la imagen de CRA, la imagen de BLA, la imagen de IDR, los tipos de unidades de NAL de VPS, SPS, PPS y APS, que corresponden a las imágenes y a los conjuntos de parámetros descritos anteriormente.

TABLA 2 Códigos de tipo de unidad de NAL del HEVC WD7 y clases de tipos de unidad de NAL

tipo_unidad_nal	Contenido de la unidad de NAL y la estructura sintáctica RBSP	Clase de tipo de unidad de NAL
0	Sin especificar	no VCL
1	Fragmento codificado de una imagen no de RAP, no TFD y no de TLA rbsp_capa_fragmento ()	VCL
2	Fragmento codificado de una imagen TFD rbsp_capa_fragmento ()	VCL
3	Fragmento codificado de una imagen no TFD de TLA rbsp_capa_fragmento ()	VCL
4,5	Fragmento codificado de una imagen de CRA RBSP de capa de fragmento ()	VCL
6,7	Fragmento codificado de una imagen de BLA rbsp_capa_fragmento ()	VCL
8	Fragmento codificado de una imagen de IDR rbsp_capa_fragmento ()	VCL

tipo_unidad_nal	Contenido de la unidad de NAL y la estructura sintáctica RBSP	Clase de tipo de unidad de NAL
9..24	Reservados	no disponible
25	Conjunto de parámetros de vídeo rbsp_conjunto_parámetros_vídeo ()	no VCL
26	Conjunto de parámetros de secuencia rbsp_conjunto_parámetros_sec ()	no VCL
27	Conjunto de parámetros de imagen rbsp_conjunto_parámetros_imagen ()	no VCL
28	Conjunto de parámetros de adaptación rbsp_aps ()	no VCL
29	Delimitador de unidad de acceso rbsp_delimitador_unidad_acceso ()	no VCL
30	Datos de relleno rbsp_datos_relleno ()	no VCL
31	Información de mejora suplementaria (SEI) rbsp_sei ()	no VCL
32..47	Reservados	no disponible
48..63	Sin especificar	no VCL

Con el fin de simplificar las asignaciones de unidades de NAL, el artículo de S. Kanumuri y G. Sullivan, "Perfeccionamiento del soporte de puntos de acceso aleatorio", 10ª Conferencia, Estocolmo, SE, julio de 2012, Doc. JCTVC-J0344 (en adelante, "Kanumuri"), que se incorpora por referencia en su totalidad, propone (1) una restricción sobre imágenes de IDR, de tal manera que no haya imágenes principales asociadas a cualquier imagen de IDR (es decir, ninguna imagen puede seguir a una imagen de IDR en el orden de decodificación y preceder a una imagen de IDR en el orden de salida), y (2) los tipos 4 a 7 de unidad de NAL de asignación modificada, definidos según la Tabla 2 anterior para imágenes de RAP de la forma siguiente:

TABLA 3: Tipos de unidades de JNAL propuestas según Kanumuri

Tipo de unidad de NAL	Descripción	Tipos posibles de SAP
4	Imagen de CRA	1,2,3
5	Imagen de BLA	1,2,3
6	Imagen de BLA sin imágenes asociadas TFD	1,2
7	Imagen de BLA sin imágenes principales	1

En la Tabla 3, los tipos de SAP se refieren a tipos de punto de acceso al flujo, definidos en la norma ISO / IEC 14496-12, 4ª edición, "Tecnología de la información - Codificación de objetos audiovisuales - Parte 12: Formato de fichero de medios de base ISO", wI2640, 100ª conferencia de MPEG, Ginebra, abril de 2012, que se incorpora por referencia en su totalidad. Como se ha descrito anteriormente, las imágenes de IDR y las imágenes de BLA / CRA son funcionalmente diferentes para la conmutación de flujos de bits, aunque son funcionalmente las mismas para el acceso aleatorio (por ejemplo, aplicaciones de búsqueda). Para la conmutación de flujos de bits en imágenes de IDR, un sistema de codificación de vídeo puede conocer o suponer que la presentación puede ser continua sin fallos (por ejemplo, falta de imágenes no presentadas). Esto es porque las imágenes siguientes a una imagen de IDR en el orden de decodificación no pueden utilizar imágenes decodificadas antes de la imagen de IDR como referencia (es decir, las imágenes principales asociadas a una imagen de IDR son DLP). Sin embargo, para la conmutación de flujos de bits en las imágenes de BLA, puede ser necesaria cierta decodificación de solapamiento de una o más imágenes procedentes de ambos flujos para asegurar que la presentación sea continua. Esta decodificación de solapamiento puede no ser posible actualmente para decodificadores compatibles con el HEVC WD7 sin capacidad adicional. Sin capacidad adicional, puede no haber ninguna imagen en las posiciones de imágenes TFD asociadas

para presentar, ya que pueden haber sido descartadas. Esto puede dar como resultado que la presentación no sea necesariamente continua. Además, incluso si la imagen de BLA es una imagen de BLA sin imágenes TFD asociadas, el problema es el mismo, porque las imágenes TFD que estaban presentes en el flujo de bits original pueden estar descartadas. Además, si no había imágenes TFD en el flujo de bits original, entonces la imagen de CRA (más tarde cambiada para que sea una imagen de BLA, debido al empalme / conmutación de flujos de bits, etc.) podría haberse codificado como una imagen de IDR. Por lo tanto, no marcar las imágenes de IDR con imágenes principales como imágenes de IDR (es decir, no permitir que las imágenes de IDR tengan imágenes principales), según lo propuesto por Kanumuri, hace que las imágenes de IDR sean menos fáciles de utilizar para los sistemas, para la conmutación de flujos de bits.

Desde el punto de vista de los sistemas de transmisión por flujo, por ejemplo, la transmisión por flujo dinámica por el HTTP (DASH), es beneficioso poder identificar fácilmente qué imagen es una imagen de RAP y, si la decodificación se inicia desde una imagen de RAP, cuál es el primer momento de la presentación (por ejemplo, un primer valor de recuento de orden de imágenes (POC)). Por lo tanto, los diseños existentes de asignación de tipos de unidad de NAL a diferentes imágenes de RAP, así como imágenes DLP e imágenes TFD, pueden mejorarse adicionalmente para que sean más fáciles de utilizar para los sistemas de transmisión por flujo. Según los diseños existentes, para cada imagen de RAP, los sistemas tienen que comprobar si hay o no imágenes DLP asociadas, para saber si el tiempo de presentación de la propia imagen de RAP es el tiempo de presentación más temprano cuando se inicia la decodificación a partir de la imagen de RAP. Además, el sistema tiene que comprobar y comparar los momentos de presentación de todas las imágenes DLP para averiguar el valor del tiempo de presentación más temprano.

Las normas de codificación de vídeo incluyen una memoria descriptiva de un modelo de almacenamiento temporal de vídeo. En la AVC y la HEVC, el modelo de almacenamiento temporal se menciona como un decodificador de referencia hipotético (HRD), que incluye un modelo de almacenamiento temporal, tanto de una memoria temporal de imágenes codificadas (CPB) como de una memoria temporal de imágenes decodificadas (DPB). Según el HEVC WD7, el HRD se define como un modelo de decodificador hipotético que especifica las restricciones en la variabilidad de flujos de unidades de NAL compatibles o de flujos de bits compatibles que un proceso de codificación puede producir. En la AVC y la HEVC, la conformidad del flujo de bits y la conformidad del decodificador se especifican como partes de la memoria descriptiva del HRD. Según el HEVC WD7, una CPB es una memoria intermedia del tipo 'primero en entrar, primero en salir', que contiene unidades de acceso en orden de decodificación, y una DPB es una memoria intermedia que contiene imágenes decodificadas para referencia. Los comportamientos de las CPB y DPB se especifican matemáticamente según el HRD. El HRD impone directamente restricciones sobre temporización, tamaños de memorias intermedias y velocidades de bits, e impone indirectamente restricciones sobre características y estadísticas de flujos de bits. Un conjunto completo de parámetros de HRD incluye cinco parámetros básicos: el retardo inicial de eliminación de la CPB, el tamaño de la CPB, la velocidad de bits, el retardo inicial de salida de la DPB y el tamaño de la DPB. Según el HEVC WD7, los parámetros del HRD se pueden incluir en los parámetros de información de usabilidad de vídeo (VUI) y los parámetros de VUI se pueden incluir en un SPS. Aunque el HRD se menciona como un decodificador, el HRD se necesita habitualmente en el sector del codificador para garantizar la conformidad del flujo de bits, y habitualmente no se necesita en el sector del decodificador. El HEVC WD7 especifica dos tipos de flujos de bits para la conformidad del HRD: esto es, Tipo I y Tipo II. Además, el HEVC WD7 especifica dos tipos de conformidad de decodificador, es decir, la conformidad de decodificador de la temporización de salida y la conformidad de decodificador del orden de salida.

En los modelos del HRD de la AVC y la HEVC, la decodificación o la eliminación de la CPB está basada en unidades de acceso, y se supone que la decodificación de imágenes es instantánea. El tiempo necesario para decodificar una imagen en aplicaciones prácticas no puede ser igual a cero. De este modo, en aplicaciones prácticas, si un decodificador compatible sigue estrictamente los tiempos de decodificación señalizados, por ejemplo, en los mensajes de SEI de temporización de imágenes, para iniciar la decodificación de unidades de acceso, entonces el momento más temprano posible para emitir una imagen decodificada específica es igual al momento de decodificación de esa imagen específica, más el tiempo necesario para decodificar esa imagen específica.

Un comportamiento de CPB basado en sub-imágenes, similar al comportamiento de CPB descrito en el artículo de Ye-Kui Wang, et al, "Funcionamiento de CPB basado en sub-imágenes," 9ª Conferencia: Ginebra, CH, mayo de 2012, JCTVC-I0588 (en adelante, "Wang"), se ha incluido en el HEVC WD7. La CPB basada en sub-imágenes de Wang permite llevar a cabo la eliminación de CPB tanto a nivel de unidad de acceso (AU) como a nivel de sub-imagen. Permitir la retirada de CPB a nivel de AU o de sub-imagen ayuda a lograr un retardo de códec reducido de manera interoperable. Cuando se produce la eliminación de CPB a nivel de unidad de acceso, se retira una unidad de acceso de la CPB cada vez que se produce una operación de eliminación. Cuando se produce la eliminación de CPB a nivel de sub-imagen, se elimina de la CPB una unidad de decodificación (DU) que contiene uno o más fragmentos cada vez que se produce una operación de eliminación.

Se puede señalar la información de temporización de retirada de CPB a nivel de sub-imagen, además de la información de temporización de retirada de CPB a nivel de AU. Cuando la información de temporización de retirada de CPB está presente tanto para la retirada a nivel de AU como para la retirada a nivel de sub-imagen, un decodificador puede escoger operar la CPB a nivel de AU o a nivel de sub-imagen. Cabe señalar que, para que el mensaje actual de SEI de temporización de imagen actual, y el mecanismo para permitir la retirada de la CPB del

HRD a nivel de AU y a nivel de DU al mismo tiempo, logren el retardo de sub-imágenes, deben enviarse las DU antes de que se codifique toda la AU, y los mensajes de SEI a nivel de AU todavía no pueden enviarse antes de que se codifique toda la AU.

- 5 Según el HEVC WD7, la información de temporización puede incluir información que define la distancia temporal entre los tiempos de salida del HRD de dos imágenes consecutivas. El HEVC WD7 define los siguientes elementos sintácticos de información de temporización:

10 La **escala_temporal** es el número de unidades de tiempo que pasan en un segundo. Por ejemplo, un sistema de coordenadas temporales que mide el tiempo usando un reloj de 27 MHz tiene una **escala_temporal** de 27.000.000. La **escala_temporal** será mayor que 0.

15 El **núm_unidades_en_pulso** es el número de unidades de tiempo de un reloj que funciona en la frecuencia de **escala_temporal** Hz, que corresponde a un incremento (llamado un pulso de reloj) de un contador de pulsos de reloj; el **núm_unidades_en_pulso** será mayor que 0.

Por lo tanto, en base a los valores de **escala_temporal** y **núm_unidades_en_pulso**, la llamada variable de pulso de reloj, t_c , se puede obtener de la forma siguiente:

$$20 \quad t_c = \text{núm_unidades_en_pulso} \div \text{escala_temporal} \quad (1)$$

Según el HEVC WD7, la variable de pulso de reloj se puede utilizar para limitar los tiempos de salida del HRD. Es decir, en algunos casos, puede ser necesario que la diferencia entre los tiempos de presentación de dos imágenes continuas en orden de salida (es decir, primera y segunda imagen) sea igual al pulso de reloj. El HEVC WD7 incluye el elemento sintáctico **indicador_velocidad_imágenes_fija** que indica si la diferencia entre los tiempos de presentación de dos imágenes continuas en orden de salida es o no igual al pulso de reloj. El elemento sintáctico **indicador_velocidad_imágenes_fija** puede incluirse en un conjunto de parámetros de VUI, que pueden incluirse en un SPS. En el HEVC WD7, cuando el elemento sintáctico **indicador_velocidad_imágenes_fija** es igual a uno, la distancia temporal entre los tiempos de salida del HRD de dos imágenes consecutivas cualesquiera en orden de salida se ve limitada a coincidir con el pulso de reloj determinado, siempre que cualquiera de las siguientes condiciones sea verdadera: (1) la segunda imagen está en la misma secuencia de vídeo codificado que la primera imagen; o (2) la segunda imagen está en una secuencia de vídeo codificado diferente a la primera imagen y el **indicador_velocidad_imágenes_fija** es igual a 1 en la secuencia de vídeo codificado que contiene la segunda imagen y el valor de **núm_unidades_en_pulso** ÷ **escala_temporal** es el mismo para ambas secuencias de vídeo codificado. Cuando el elemento sintáctico **indicador_velocidad_imágenes_fija** es igual a cero, no se aplica ninguna limitación de ese tipo a la distancia temporal entre los tiempos de salida del HRD de dos imágenes consecutivas cualesquiera (es decir, primera y segunda imagen) en orden de salida. Cabe señalar que cuando el **indicador_velocidad_imágenes_fija** no está presente, se deduce que es igual a 0. Cabe señalar que, según el HEVC WD7, cuando el **indicador_velocidad_imágenes_fija** es igual a 1, la adaptación del flujo en base a la ajustabilidad temporal a escala requeriría cambiar el valor de **escala_temporal** o de **núm_unidades_en_pulso** en el caso en que se descartan algunas capas temporales más altas. Cabe señalar que el HEVC WD7 proporciona la semántica siguiente para el **indicador_velocidad_imágenes_fija**:

45 Cuando el **indicador_velocidad_imágenes_fija** es igual a 1 para una secuencia de vídeo codificado que contiene la imagen n , el valor calculado para $\Delta t_{o,dpb}(n)$, tal como se especifica en la Ecuación C-13, será igual a t_c , como se especifica en la Ecuación C-1 (utilizando el valor de t_c para la secuencia de vídeo codificado que contiene la imagen n) cuando una o más de las siguientes condiciones son verdaderas para la siguiente imagen n_n que se especifica para su uso en la Ecuación C-13:

50 - la imagen n_n está en la misma secuencia de vídeo codificado que la imagen n .

- la imagen n_n está en una secuencia de vídeo codificado diferente y el **indicador_velocidad_imágenes_fija** es igual a 1 en la secuencia de vídeo codificado que contiene la imagen n_n y el valor de **núm_unidades_en_pulso** ÷ **escala_temporal** es el mismo para ambas secuencias de vídeo codificado.

55 En donde la Ecuación C-1 corresponde a la ecuación (1) y la Ecuación C-13 se define en el HEVC WD7 de la forma siguiente:

$$60 \quad \Delta t_{o,dpb}(n) = t_{o,dpb}(n_n) - t_{o,dpb}(n) \quad (2)$$

En vista de la sincronización anteriormente mencionada y las características de acceso aleatorio asociadas al HEVC WD7, esta divulgación describe las técnicas que pueden utilizarse para reducir el retardo en aplicaciones de vídeo, tales como aplicaciones de conversación, y proporcionar mejoras en el acceso aleatorio a una secuencia de vídeo codificado. En un ejemplo, esta divulgación describe las técnicas para la asignación de los tipos de unidades de NAL. En otro ejemplo, esta divulgación describe el comportamiento del HRD, a nivel de sub-imagen o a nivel de

unidad de decodificación. En otro ejemplo, esta divulgación describe técnicas para hacer referencia a los Identificadores de conjuntos de parámetros. En otro ejemplo más, esta divulgación describe las técnicas para proporcionar una semántica mejorada para el elemento sintáctico **indicador_velocidad_imágenes_fija**. Debería tenerse en cuenta que cualquiera de, y todas, las combinaciones de estas técnicas y otras técnicas descritas en este documento pueden incorporarse en un sistema de codificación y decodificación de vídeo.

La FIG. 3 es un diagrama de bloques que ilustra un sistema ejemplar de codificación y decodificación de vídeo 10 que puede utilizar las técnicas descritas en este documento. En particular, el sistema de codificación y decodificación de vídeo puede utilizar las técnicas descritas en el presente documento en relación con (1) la asignación de tipos de unidades de NAL, (2) el comportamiento del HRD a nivel de sub-imagen o a nivel de unidad de decodificación, (3) la referencia a Identificadores de conjuntos de parámetros, (4) la semántica mejorada para el **indicador_velocidad_imágenes_fija**, o cualquiera de, y todas, las combinaciones de estas técnicas. El sistema de codificación y decodificación de vídeo 10 es un ejemplo de un sistema de vídeo que puede ser utilizado para cualquiera de las siguientes aplicaciones de vídeo: reproducción local, transmisión por flujo, difusión, multidifusión y/o aplicaciones conversacionales. El dispositivo de origen 12 y el dispositivo de destino 14 son ejemplos de tales dispositivos de codificación, en los que el dispositivo de origen 12 genera datos de vídeo codificado para su transmisión al dispositivo de destino 14. En algunos ejemplos, los dispositivos de origen 12 y el dispositivo de destino 14 pueden funcionar de una manera esencialmente simétrica, de modo que cada uno entre el dispositivo de origen 12 y el dispositivo de destino 14 incluya componentes de codificación y decodificación de vídeo. Por lo tanto, el sistema 10 puede configurarse para dar soporte a la transmisión de vídeo unidireccional o bidireccional entre el dispositivo de origen 12 y el dispositivo de destino 14.

Aunque las técnicas descritas en el presente documento se describen conjuntamente con el dispositivo de origen 12 y el dispositivo de destino 14, las técnicas pueden ser realizadas por cualquier dispositivo de codificación y/o decodificación de vídeo digital. Las técnicas de esta divulgación también pueden llevarse a cabo por un preprocesador de vídeo. Además, aunque, por lo general, las técnicas de esta divulgación se describen como llevadas a cabo mediante un dispositivo de codificación de vídeo y un dispositivo de decodificación de vídeo, las técnicas también pueden llevarse a cabo mediante un codificador/decodificador de vídeo, denominado habitualmente un "CÓDEC". Por lo tanto, el codificador de vídeo 20 y el decodificador de vídeo 30 en la FIG. 3 pueden incluirse en uno o más codificadores o decodificadores, donde cualquiera de los mismos puede estar integrado como parte de un codificador/decodificador combinado (CÓDEC) en un dispositivo respectivo. Además, un dispositivo que incluye el codificador de vídeo 20 y/o el decodificador de vídeo 30 puede comprender un circuito integrado, un microprocesador y/o un dispositivo de comunicación inalámbrica, tal como un teléfono celular. Aunque no se muestra en la FIG. 3, en algunos aspectos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden estar integrados en un codificador y decodificador de audio, y pueden incluir unidades adecuadas de multiplexado y demultiplexado, u otro hardware y software, para llevar a cabo la codificación, tanto de audio como de vídeo, en un flujo de datos común o en flujos de datos diferentes. Si procede, las unidades de multiplexado y demultiplexado pueden ajustarse al protocolo de multiplexado ITU H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

Como se ilustra en la FIG. 3, el sistema 10 incluye un dispositivo de origen 12 que proporciona datos de vídeo codificados, a decodificar en un momento posterior por parte de un dispositivo de destino 14. En particular, el dispositivo de origen 12 proporciona datos de vídeo codificados al dispositivo destino 14 a través de un medio legible por ordenador 16. El dispositivo de destino 14 puede recibir los datos de vídeo codificados que se van a decodificar, mediante un medio legible por ordenador 16. El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender cualquiera entre una amplia gama de dispositivos, incluyendo ordenadores de sobremesa, ordenadores plegables (es decir, portátiles), ordenadores de tableta, decodificadores, equipos telefónicos portátiles tales como los denominados teléfonos "inteligentes", los denominados paneles "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, un dispositivo de flujos de transmisión de vídeo o similares. En algunos casos, el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar equipados para la comunicación inalámbrica.

El medio legible por ordenador 16 puede comprender cualquier tipo de medio o dispositivo capaz de desplazar los datos de vídeo codificado desde el dispositivo de origen 12 al dispositivo de destino 14. El medio legible por ordenador 16 puede incluir medios transitorios, tales como una emisión inalámbrica o transmisión de la red cableada, o medios de almacenamiento (es decir, medios de almacenamiento no transitorio), tales como un disco duro, una unidad flash, un disco compacto, un disco de vídeo digital, un disco Blu-ray u otros medios legibles por ordenador. En algunos ejemplos, un servidor de red (no mostrado) puede recibir datos de vídeo codificado desde el dispositivo de origen 12 y proporcionar los datos de vídeo codificado al dispositivo de destino 14, por ejemplo, mediante transmisión por red. De manera similar, un dispositivo informático de una instalación de producción de un medio, tal como una instalación de estampación de discos, puede recibir datos de vídeo codificado desde el dispositivo de origen 12 y producir un disco que contenga los datos de vídeo codificado.

En un ejemplo, el medio legible por ordenador 16 puede comprender un medio de comunicación para permitir al dispositivo de origen 12 transmitir datos de vídeo codificado directamente al dispositivo de destino 14 en tiempo real. Los datos de vídeo codificado pueden ser modulados según una norma de comunicación, tal como un protocolo de

comunicación inalámbrica, y transmitidos al dispositivo de destino 14. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrico o cableado, tal como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física. El medio de comunicación puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área extensa o una red global tal como Internet. El medio de comunicación puede incluir encaminadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 al dispositivo de destino 14.

El dispositivo de almacenamiento puede incluir cualquiera entre una diversidad de medios de almacenamiento de datos, de acceso distribuido o local, tales como una unidad de disco, discos Blu-ray, discos DVD, discos CD-ROM, memoria flash, memoria volátil o no volátil, u otros medios adecuados cualesquiera de almacenamiento digital, para almacenar datos de vídeo codificados. En un ejemplo adicional, el dispositivo de almacenamiento puede corresponder a un servidor de ficheros o a otro dispositivo de almacenamiento intermedio que pueda almacenar los datos de vídeo codificado, generados por el dispositivo de origen 12. El dispositivo de destino 14 puede acceder a datos de vídeo almacenados procedentes del dispositivo de almacenamiento, mediante transmisión por flujo o descarga. El servidor de ficheros puede ser cualquier tipo de servidor capaz de almacenar datos de vídeo codificado y transmitir esos datos de vídeo codificado al dispositivo de destino 14. Entre los ejemplos de servidores de ficheros se incluyen un servidor de Internet (por ejemplo, para una sede de la Red), un servidor del FTP, dispositivos de almacenamiento conectados a la red (NAS) o una unidad de disco local. El dispositivo de destino 14 puede acceder a los datos de vídeo codificados a través de cualquier conexión de datos estándar, incluyendo una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión de Wi-Fi), una conexión cableada (por ejemplo, DSL, módem de cable, etc.), o una combinación de ambos que sea adecuada para acceder a datos de vídeo codificado, almacenados en un servidor de ficheros. La transmisión de datos de vídeo codificado desde el dispositivo de almacenamiento puede ser una transmisión por flujo, una transmisión de descarga o una combinación de las mismas.

Las técnicas de esta divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas pueden aplicarse a la codificación de vídeo, como soporte de cualquiera entre una diversidad de aplicaciones de multimedia, tales como difusiones de televisión por el aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones de vídeo por flujo por Internet, tales como la transmisión por flujo adaptativa dinámica sobre HTTP (DASH), vídeo digital que se codifica en un medio de almacenamiento de datos, decodificación de vídeo digital almacenado en un medio de almacenamiento de datos, u otras aplicaciones.

En el ejemplo de la FIG. 3, el dispositivo de origen 12 incluye el origen de vídeo 18, la unidad de estructura de codificación 19, el codificador de vídeo 20, la unidad de encapsulación 21 y la interfaz de salida 22. El dispositivo de destino 14 incluye la interfaz de entrada 28, la unidad de desencapsulación 29, el decodificador de vídeo 30 y el dispositivo de visualización 32. En otros ejemplos, un dispositivo de origen 12 y un dispositivo de destino 14 pueden incluir otros componentes o disposiciones. Por ejemplo, el dispositivo de origen 12 puede recibir datos de vídeo desde un origen de vídeo externo 18, tal como una cámara externa. Asimismo, el dispositivo de destino 14 puede interactuar con un dispositivo de visualización externo, en lugar de incluir un dispositivo de visualización integrado. Cada uno de los componentes del dispositivo de origen 12 y del dispositivo de destino 14 puede implementarse como cualquiera entre una diversidad de circuitos adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), formaciones de compuertas programables en el terreno (FPGA), lógica discreta, software, hardware, firmware, o combinaciones cualesquiera de los mismos. Cuando las técnicas descritas en este documento son implementadas parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio no transitorio adecuado, legible por ordenador, y ejecutar las instrucciones en hardware usando uno o más procesadores para realizar las técnicas.

El origen de vídeo 18 del dispositivo de origen 12 puede incluir un dispositivo de captura de vídeo, tal como una cámara de vídeo, un fichero de vídeo que contiene vídeo grabado previamente y/o una interfaz de alimentación de vídeo para recibir vídeo procedente de un proveedor de contenido de vídeo. Como una alternativa adicional, el origen de vídeo 18 puede generar datos basados en gráficos de ordenador como el vídeo de origen, o una combinación de vídeo en directo, vídeo archivado y vídeo generado por ordenador. En algunos casos, si el origen de vídeo 18 es una videocámara, el dispositivo de origen 12 y el dispositivo de destino 14 pueden formar los denominados teléfonos con cámara o videoteléfonos. Sin embargo, como se ha mencionado anteriormente, las técnicas descritas en esta divulgación pueden aplicarse a la codificación de vídeo en general, y pueden aplicarse a aplicaciones inalámbricas y/o cableadas. En cada caso, el vídeo capturado, pre-capturado o generado por ordenador puede ser recibido por el codificador de vídeo 20. La interfaz de salida 22 se puede configurar para emitir datos de vídeo codificado, tales como una secuencia de vídeo codificado, en un medio legible por ordenador 16. En algunos ejemplos, una secuencia de vídeo codificado puede emitirse desde la interfaz de salida 22 a un dispositivo de almacenamiento. La interfaz de entrada 28 del dispositivo de destino 14 recibe los datos de vídeo codificado desde el medio legible por ordenador 16. El dispositivo de visualización 32 muestra los datos de vídeo decodificado a un usuario y puede comprender cualquiera entre una diversidad de dispositivos de visualización, tales como un tubo de rayos catódicos (CRT), una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos de emisión de luz (OLED) u otro tipo de dispositivo de visualización.

La unidad de la estructura de codificación 19, el codificador de vídeo 20, la unidad de encapsulación 21, la unidad de

desencapsulación 29 y el decodificador de vídeo 30 pueden funcionar según un estándar de codificación de vídeo, como la inminente HEVC, descrita anteriormente y, en general, puede ajustarse al modelo de prueba de HEVC (HM). Como alternativa, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar según otras normas patentadas o industriales, tales como la norma ITU-T H.264, alternativamente denominada MPEG-4, Parte 10, Codificación Avanzada de Vídeo (AVC), o extensiones de tales normas. La unidad de estructura de codificación 19, el codificador de vídeo 20, la unidad de encapsulación 21, la unidad de desencapsulación 29 y el decodificador de vídeo 30 también pueden funcionar según una versión modificada de una norma de codificación de vídeo, en la que la versión modificada de la norma de codificación de vídeo se modifica para incluir todas, y cualquiera de, las combinaciones de las técnicas descritas en el presente documento.

El codificador de vídeo 20 puede dividir una trama o imagen de vídeo en una serie de bloques de vídeo de igual tamaño, tales como una CU, como se describe en el HEVC WD7. Una CU incluye un nodo de codificación y unidades de predicción (PU) y unidades de transformación (TU) asociadas al nodo de codificación. Un tamaño de la CU corresponde a un tamaño del nodo de codificación, y debe ser de forma cuadrada. El tamaño de la CU puede variar desde 8 x 8 píxeles hasta el tamaño del bloque arbolado, con un máximo de 64 x 64 píxeles, o más. Cada CU puede contener una o más PU y una o más TU. Los datos sintácticos asociados a una CU pueden describir, por ejemplo, la división de la CU en una o más PU. Las modalidades de división pueden diferir entre si la CU está codificada en modalidad de omisión o directa, codificada en modalidad de intra-predicción y codificada en modalidad de inter-predicción. Las PU pueden dividirse para que no tengan forma cuadrada. Los datos sintácticos asociados a una CU también pueden describir, por ejemplo, la división de la CU en una o más TU según un árbol cuádruple. Una TU puede tener forma cuadrada o no cuadrada (por ejemplo, rectangular).

La norma HEVC admite transformaciones según las TU, que pueden ser distintas para distintas CU. Las TU son habitualmente dimensionadas en base al tamaño de las PU dentro de una CU dada, definida para una LCU dividida, aunque esto puede no ser siempre el caso. Las TU tienen habitualmente el mismo tamaño, o un tamaño más pequeño, que las PU. En algunos ejemplos, las muestras residuales correspondientes a una CU pueden subdividirse en unidades más pequeñas usando una estructura de árbol cuádruple conocida como "árbol cuádruple residual" (RQT). Los nodos de hoja del RQT pueden denominarse unidades de transformación (TU). Los valores de diferencias de píxeles asociados a las TU pueden ser transformados para producir coeficientes de transformación, que pueden cuantizarse.

Una CU hoja puede incluir una o más unidades de predicción (PU). En general, una PU representa una zona espacial correspondiente a la totalidad o a una parte de la CU correspondiente, y puede incluir datos para recuperar una muestra de referencia para la PU. Además, una PU incluye datos relacionados con la predicción. Por ejemplo, cuando la PU está codificada en la intra-modalidad, los datos para la PU pueden incluirse en un árbol cuádruple residual (RQT) que pueden incluir datos que describen una modalidad de intra-predicción para una TU correspondiente a la PU. Como otro ejemplo, cuando la PU está codificada en la inter-modalidad, la PU puede incluir datos que definen uno o más vectores de movimiento para la PU. Los datos que definen el vector de movimiento para una PU pueden describir, por ejemplo, un componente horizontal del vector de movimiento, un componente vertical del vector de movimiento, una resolución para el vector de movimiento (por ejemplo, precisión de un cuarto de píxel o precisión de un octavo de píxel), una imagen de referencia a la que apunta el vector de movimiento y/o una lista de imágenes de referencia (por ejemplo, la Lista 0, la Lista 1 o la Lista C) para el vector de movimiento.

Una CU hoja con una o más PU también puede incluir una o más unidades de transformación (TU). Las unidades de transformación pueden especificarse usando un RQT (también denominada una estructura de árbol cuádruple de TU), como se ha expuesto anteriormente. Por ejemplo, un indicador de división puede indicar si una CU hoja está dividida o no en cuatro unidades de transformación. Después, cada unidad de transformación puede dividirse adicionalmente en más sub-TU. Cuando una TU no está dividida adicionalmente, puede denominarse una TU hoja. Generalmente, para la intra-codificación, todas las TU hoja que pertenecen a una CU hoja comparten la misma modalidad de intra-predicción. Es decir, la misma modalidad de intra-predicción se aplica generalmente para calcular valores predichos para todas las TU de una CU hoja. Para la intra-codificación, un codificador de vídeo puede calcular un valor residual para cada TU hoja usando la modalidad de intra-predicción, como una diferencia entre la parte de la CU correspondiente a la TU y el bloque original. Una TU no está necesariamente limitada al tamaño de una PU. De este modo, las TU pueden ser más grandes o más pequeñas que una PU. Para la intra-codificación, una PU puede estar co-situada con una correspondiente TU hoja para la misma CU. En algunos ejemplos, el tamaño máximo de una TU hoja puede corresponderse con el tamaño de la CU hoja correspondiente.

Además, las TU de las CU hojas también pueden asociarse a estructuras de datos respectivas de árbol cuádruple, denominadas árboles cuádruples residuales (RQT). Es decir, una CU hoja puede incluir un árbol cuádruple que indica cómo la CU hoja está dividida en varias TU. El nodo raíz de un árbol cuádruple de TU corresponde generalmente a una CU hoja, mientras que el nodo raíz de un árbol cuádruple de CU corresponde generalmente a un bloque de árbol (o LCU). Las TU del RQT que no están divididas se denominan TU hojas. En general, esta divulgación usa los términos CU y TU para hacer referencia a una CU hoja y a una TU hoja, respectivamente, a no ser que se indique lo contrario. Esta divulgación utiliza el término "bloque" para referirse a cualquiera entre una CU, PU o TU, en el contexto de la HEVC, o a estructuras de datos similares en el contexto de otras normas (por ejemplo, macrobloques y sub-bloques de los mismos en la norma H.264 / AVC).

Como ejemplo, el HM da soporte a la predicción en diversos tamaños de PU. Suponiendo que el tamaño de una CU específica sea $2N \times 2N$, el HM presta soporte a la intra-predicción en tamaños de PU de $2N \times 2N$ o $N \times N$, y a la intra-predicción en tamaños de PU simétricos de $2N \times 2N$, $2N \times N$, $N \times 2N$ o $N \times N$. El HM también presta soporte a la división asimétrica para la inter-predicción en tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$. En la división asimétrica, una dirección de una CU no está dividida, mientras que la otra dirección está dividida entre el 25 % y el 75 %. La parte de la CU correspondiente a la división del 25 % está indicada por una "n" seguida por una indicación de "Arriba", "Abajo", "Izquierda" o "Derecha". Así, por ejemplo, "2NxnU" se refiere a una CU de tamaño $2N \times 2N$ que está dividida horizontalmente, con una PU de tamaño $2N \times 0,5N$ encima y una PU de tamaño $2N \times 1,5N$ debajo.

En esta divulgación, "NxN" y "N por N" pueden usarse de manera intercambiable para hacer referencia a las dimensiones de píxeles de un bloque de vídeo, en lo que respecta a dimensiones verticales y horizontales, por ejemplo, 16×16 píxeles o 16 por 16 píxeles. En general, un bloque de tamaño 16×16 tendrá 16 píxeles en la dirección vertical ($y = 16$) y 16 píxeles en la dirección horizontal ($x = 16$). Asimismo, un bloque de tamaño $N \times N$ tiene generalmente N píxeles en la dirección vertical y N píxeles en la dirección horizontal, donde N representa un valor entero no negativo. Los píxeles en un bloque pueden estar dispuestos en filas y columnas. Además, los bloques no necesitan tener necesariamente el mismo número de píxeles en la dirección horizontal y en la dirección vertical. Por ejemplo, los bloques pueden comprender $N \times M$ píxeles, donde M no es necesariamente igual a N .

Tras la codificación intra-predictiva o inter-predictiva, usando las PU de una CU, el codificador de vídeo 20 puede calcular datos residuales para las TU de la CU. Las PU pueden comprender datos sintácticos que describen un procedimiento o modalidad de generación de datos de píxeles predictivos en el dominio espacial (también denominado el dominio de píxeles) y las TU pueden comprender coeficientes en el dominio de transformación, tras la aplicación de una transformación, por ejemplo, una transformación de coseno discreta (DCT), una transformación entera, una transformación de ondículas o una transformación conceptualmente similar, a los datos de vídeo residuales. Los datos residuales pueden corresponder a diferencias de píxeles entre píxeles de la imagen no codificada y valores de predicción correspondientes a las PU. El codificador de vídeo 20 puede formar las TU incluyendo los datos residuales para la CU, y luego transformar las TU para producir coeficientes de transformación para la CU.

Tras cualquier transformación para producir coeficientes de transformación, el codificador de vídeo 20 puede realizar la cuantización de los coeficientes de transformación. La cuantización se refiere generalmente a un proceso en el que los coeficientes de transformación se cuantizan para reducir posiblemente la cantidad de datos usados para representar los coeficientes, proporcionando una compresión adicional. El proceso de cuantización puede reducir la profundidad de bits asociada a algunos de, o todos, los coeficientes. Por ejemplo, un valor de n bits puede redondearse por lo bajo a un valor de m bits durante la cuantización, donde n es mayor que m .

Después de la cuantización, el codificador de vídeo puede escanear los coeficientes de transformación, produciendo un vector unidimensional a partir de la matriz bidimensional, incluyendo los coeficientes de transformación cuantizados. La exploración puede diseñarse para poner coeficientes de energía más alta (y, por lo tanto, frecuencia más baja) en la parte delantera de la matriz y para poner coeficientes de energía más baja (y, por lo tanto, frecuencia más alta) en la parte posterior de la matriz. En algunos ejemplos, el codificador de vídeo 20 puede utilizar un orden de escaneado predefinido para escanear los coeficientes de transformación cuantizados, para producir un vector serializado que pueda ser codificado por entropía. En otros ejemplos, el codificador de vídeo 20 puede realizar un escaneado adaptativo. Después de escanear los coeficientes de transformación cuantizados, para formar un vector unidimensional, el codificador de vídeo 20 puede codificar por entropía el vector unidimensional, por ejemplo, según la codificación de longitud variable adaptativa al contexto (CAVLC), la codificación aritmética binaria adaptativa al contexto (CABAC), la codificación aritmética binaria adaptativa al contexto y de base sintáctica (SBAC), la codificación por Entropía por División de Intervalos de Probabilidad (PIPE) u otra metodología de codificación por entropía. El codificador de vídeo 20 también puede codificar por entropía elementos sintácticos asociados a los datos de vídeo codificados, para su uso por el decodificador de vídeo 30, en la decodificación de los datos de vídeo.

Para realizar la CABAC, el codificador de vídeo 20 puede asignar un contexto dentro de un modelo contextual a un símbolo a transmitir. El contexto puede referirse, por ejemplo, a si los valores adyacentes del símbolo son distintos de cero o no. Para realizar la CAVLC, el codificador de vídeo 20 puede seleccionar un código de longitud variable para un símbolo a transmitir. Las palabras de código en la VLC pueden ser construidas de modo que los códigos relativamente más cortos correspondan a símbolos más probables, mientras que los códigos más largos correspondan a símbolos menos probables. De esta manera, el uso de la VLC puede lograr un poco de ahorro con respecto, por ejemplo, a usar palabras de código de igual longitud para cada símbolo a transmitir. La determinación de la probabilidad puede basarse en un contexto asignado al símbolo.

Como se ha descrito anteriormente, una secuencia de vídeo puede codificarse según una estructura de codificación de vídeo determinada, donde la estructura de codificación define la asignación de tipos de imágenes (por ejemplo, imágenes de RAP e imágenes no de RAP) utilizados para codificar una secuencia de vídeo. Por ejemplo, una secuencia de vídeo puede codificarse con imágenes de RAP incluidas en intervalos de predeterminación con el fin de facilitar el acceso aleatorio de una secuencia de vídeo. Tal estructura de codificación puede ser útil para

aplicaciones de difusión. Además, una secuencia de vídeo puede codificarse según una estructura de codificación que minimiza el retardo para aplicaciones de bajo retardo. La unidad de estructura de codificación 19 puede configurarse para determinar una estructura de codificación, a ser utilizada por el codificador de vídeo 20 para la codificación de una secuencia de vídeo recibida desde el origen de vídeo 18. En un ejemplo, la unidad de estructura de codificación 19 puede almacenar estructuras de codificación predefinidas que corresponden a aplicaciones de vídeo respectivas. La unidad de estructura de codificación 19 puede configurarse para emitir información que indica una estructura de codificación particular para cada codificador de vídeo 20 y unidad de encapsulación 21. El codificador de vídeo 20 recibe una secuencia de vídeo desde el origen de vídeo 18 e información de estructura de codificación desde la unidad de estructura de codificación 19, y genera datos de vídeo codificado. La unidad de encapsulación 21 recibe datos de vídeo codificado desde el codificador de vídeo 20 e información que indica una estructura de codificación particular, y genera una secuencia de vídeo codificado que incluye unidades de acceso. La unidad de desencapsulación 29 puede configurarse para recibir la secuencia de vídeo codificado y analizar sintácticamente las unidades de acceso y unidades de NAL. El decodificador de vídeo 30 puede configurarse para recibir unidades de NAL y reconstruir los datos de vídeo en base a la información incluida en las unidades de NAL recibidas.

Debería tenerse en cuenta que la unidad de estructura de codificación 19 y/o el codificador de vídeo 20 puede configurarse para generar elementos sintácticos incluidos en un conjunto de parámetros. En algunos ejemplos, la unidad de estructura de codificación 19 puede configurarse para generar elementos sintácticos incluidos en conjuntos de parámetros de alto nivel, tales como un SPS, y el codificador de vídeo 20 puede configurarse para realizar la codificación de vídeo en base a elementos sintácticos recibidos desde la estructura de unidad de codificación, así como elementos sintácticos de salida codificados por entropía, como parte de datos de vídeo codificado.

Según las técnicas de esta divulgación, la asignación de los tipos de unidad de NAL se puede realizar de tal manera que un dispositivo, tal como el dispositivo de destino 14, pueda identificar fácilmente una imagen de RAP e información de temporización asociada. En un ejemplo, imágenes de IDR sin imágenes principales asociadas tienen un tipo distinto de unidad de NAL que imágenes de IDR que pueden tener asociadas imágenes principales. Por ejemplo, las imágenes de IDR sin imágenes principales asociadas tienen un tipo M de unidad de NAL, mientras que las imágenes de IDR que pueden tener asociadas imágenes principales tienen el tipo N de unidad de NAL, en donde M no es igual a N, como se ilustra en la Tabla 4. Cabe señalar que, en el ejemplo ilustrado en la Tabla 4, las imágenes principales asociadas a una imagen de IDR pueden ser imágenes DLP. En un ejemplo, los tipos de unidad de NAL ilustrados en la Tabla 4 se pueden incorporar en los códigos de tipo de unidad de NAL y las clases de tipos de unidad de NAL del HEVC WD7, que se ilustran en la Tabla 2. Por ejemplo, los valores del tipo de unidad de NAL invertidos en la Tabla 2 se pueden utilizar para los tipos M y N de unidad de NAL en la Tabla 4.

TABLA 4 Tipos distintos de unidades de NAL de IDR

M	Fragmento codificado de una imagen de IDR sin imágenes principales asociadas rbsp_capa_fragmento ()	VCL
N	Fragmento codificado de una imagen de IDR que puede tener asociadas imágenes principales rbsp_capa_fragmento ()	VCL

En otro ejemplo, las imágenes de CRA sin imágenes principales asociadas tienen un tipo de unidad de NAL distinto, diferente a imágenes de CRA que pueden tener asociadas imágenes principales. Además, las imágenes de CRA sin imágenes TFD asociadas tienen una unidad de NAL distinta, diferente a imágenes de CRA que pueden tener asociadas imágenes TFD. En consecuencia, tres tipos de unidades de NAL diferentes se pueden utilizar para diferentes tipos de imágenes de CRA, como se ilustra en la Tabla 5. En un ejemplo, los tipos de unidad de NAL ilustrados en la Tabla 5 se pueden incorporar en los códigos de tipos de unidad de NAL y clases de tipos de unidad de NAL del HEVC WD7 que se ilustran en la Tabla 2. Por ejemplo, los valores de tipos de unidad de NAL invertidos en la Tabla 1 se pueden utilizar para los tipos de unidad de NAL X, Y y Z en la Tabla 5.

TABLA 5 Tipos distintos de unidad de NAL de CRA

X	Fragmento codificado de una imagen de CRA sin imágenes principales asociadas rbsp_capa_fragmento ()	VCL
Y	Fragmento codificado de una imagen de CRA que no tiene TFD asociado (pero puede tener imágenes DLP asociadas) rbsp_capa_fragmento ()	VCL
Z	Fragmento codificado de imagen de CRA que puede tener imágenes TFD asociadas rbsp_capa_fragmento ()	VCL

En otro ejemplo, las imágenes de BLA sin imágenes principales asociadas pueden tener un tipo distinto de unidad de NAL, diferente a las imágenes de BLA que pueden tener asociadas imágenes principales. Por otra parte, las imágenes de BLA sin imágenes TFD asociadas pueden tener una unidad de NAL distinta, diferente a las imágenes de BLA que pueden tener asociadas imágenes TFD. En consecuencia, se pueden utilizar tres tipos de unidades de NAL diferentes para diferentes tipos de BLA, como se ilustra en la Tabla 6. En un ejemplo, los tipos de unidad de NAL ilustrados en la Tabla 6 se pueden incorporar en los códigos de tipo de unidad de NAL y las clases de tipos de unidad de NAL del HEVC WD7, que se ilustran en la Tabla 2. Por ejemplo, los valores de tipos de unidad de NAL invertidos en la Tabla 2 se pueden utilizar para los tipos de unidad de NAL A, B y C en la Tabla 6.

TABLA 6 Tipos distintos de unidad de NAL de BLA

A	Fragmento codificado de una imagen de BLA sin imágenes principales asociadas rbsp_capa_fragmento ()	VCL
B	Fragmento codificado de una imagen de BLA que no tiene TFD asociado (pero puede tener imágenes DLP asociadas) rbsp_capa_fragmento ()	VCL
C	Fragmento codificado de imagen de BLA que puede tener asociadas imágenes TFD rbsp_capa_fragmento ()	VCL

Cualquiera de, y todas, las combinaciones de los tipos de unidad de NAL, descritas con respecto a las Tablas 4 a 6, se pueden usar para la asignación de los tipos de unidad de NAL. En un ejemplo, todos los tipos de unidad de NAL descritos con respecto a las Tablas 4 a 6 se pueden usar para la asignación de los tipos de unidad de NAL. La Tabla 7 ilustra un ejemplo en el que todos los tipos de NAL ilustrados en las Tablas 4 a 6 se usan para la asignación de los tipos de unidad de NAL. Como se ilustra en la Tabla 7, los tipos de unidad de NAL incluyen los tipos de unidad de NAL de imagen de CRA, de imagen de BLA y de imagen de IDR, descritos con respecto a las Tablas 4 a 6, así como los tipos de unidad de NAL de VPS, SPS, PPS y APS, descritos anteriormente. La Tabla 7 se puede contrastar con la Tabla 2 anterior en cuanto a que la asignación de tipos de unidad de NAL, proporcionados en la Tabla 7, incluye múltiples tipos de unidad de NAL para imágenes de IDR, de CRA y de BLA, mientras que la asignación de tipos de unidad de NAL, proporcionados en la Tabla 1, incluye un único tipo de unidad de NAL para cada una de las imágenes de IDR, de CRA y de BLA.

TABLA 7 Códigos de tipo de unidad de NAL y clases de tipos de unidad de NAL

tipo_unidad_nal	Contenido de la unidad de NAL y estructura sintáctica de RBSP	Clase de tipo de unidad de NAL
0	Sin especificar	no VCL
1	Fragmento codificado de una imagen no de RAP, no TFD, no DLP y no de TLA rbsp_capa_fragmento ()	VCL
2	Fragmento codificado de una imagen de TLA rbsp_capa_fragmento ()	VCL
3	Fragmento codificado de una imagen TFD rbsp_capa_fragmento ()	VCL
4	Fragmento codificado de una imagen DLP rbsp_capa_fragmento ()	VCL
5	Fragmento codificado de una imagen de CRA sin imágenes principales asociadas rbsp_capa_fragmento ()	VCL
6	Fragmento codificado de una imagen de CRA que no tiene TFD asociado (pero puede tener imágenes DLP asociadas) rbsp_capa_fragmento ()	VCL
7	Fragmento codificado de imagen de CRA que puede tener imágenes TFD asociadas rbsp_capa_fragmento ()	VCL
8	Fragmento codificado de una imagen de BLA sin imágenes principales asociadas rbsp_capa_fragmento ()	VCL
9	Fragmento codificado de una imagen de BLA que no tiene TFD asociado (pero puede tener imágenes DLP asociadas) rbsp_capa_fragmento ()	VCL
10	Fragmento codificado de imagen de BLA que puede tener asociadas	VCL

tipo_unidad_nal	Contenido de la unidad de NAL y estructura sintáctica de RBSP	Clase de tipo de unidad de NAL
	imágenes TFD rbsp_capa_fragmento ()	
11	Fragmento codificado de una imagen de IDR sin imágenes principales asociadas rbsp_capa_fragmento ()	VCL
12	Fragmento codificado de una imagen de IDR que puede tener asociadas imágenes principales rbsp_capa_fragmento ()	VCL
13..24	Reservados	no disponible
25	Conjunto de parámetros de vídeo rbsp_conjunto_parámetros_vídeo ()	no VCL
26	Conjunto de parámetros de secuencia rbsp_conjunto_parámetros_secuencia ()	no VCL
27	Conjuntos de parámetros de imagen rbsp_conjunto_parámetros_imagen ()	no VCL
28	Conjunto de parámetros de adaptación rbsp_aps ()	no VCL
29	Delimitador de unidad de acceso rbsp_delimitador_unidad_acceso ()	no VCL
30	Datos de relleno rbsp_datos_relleno ()	no VCL
31	Información de mejora suplementaria (SEI) rbsp_sei ()	no VCL
32..47	Reservados	no disponible
48..63	Sin especificar	no VCL

La unidad de encapsulación 21 puede configurarse para recibir datos de vídeo codificado desde el codificador de vídeo 20 e información que indica una estructura de codificación particular, y generar una secuencia de vídeo codificado que incluye unidades de acceso en base a la asignación de los tipos de unidad de NAL ilustrados en cualquiera de, y todas, las combinaciones de asignaciones de unidades de NAL, ilustradas en las Tablas 2 a 7. Además, la unidad de desencapsulación 29 puede configurarse para recibir la secuencia de vídeo codificado y analizar sintácticamente las unidades de acceso y las unidades de NAL, donde las unidades de NAL se asignan basándose en todas y cada una de las combinaciones de las asignaciones de unidades de NAL, ilustradas en las Tablas 2 a 7.

Como se ha descrito anteriormente, según el HEVC WD7, para que el mensaje de SEI de temporización de imagen actual, y el mecanismo, permitan la retirada de CPB del HRD, tanto a nivel de AU como a nivel de DU al mismo tiempo, para lograr el retardo de sub-imágenes, las DU deben enviarse antes de que toda la AU se codifique, y los mensajes de SEI a nivel de AU aún no se pueden enviar antes de que toda la AU se codifique. Según las técnicas de esta divulgación, la unidad de encapsulación 21 y la unidad de desencapsulación 29 pueden configurarse de manera que el comportamiento del HRD, a nivel de sub-imagen o a nivel de unidad de decodificación, pueda modificarse en comparación con el HEVC WD7.

Por ejemplo, la unidad de encapsulación 21 puede estar configurada de tal manera que los mensajes de SEI a nivel de AU se envíen después de que toda la AU sea codificada. Tal mensaje de SEI a nivel de AU se puede incluir en una unidad de NAL de SEI con un tipo distinto de unidad de NAL. Una diferencia entre una unidad de NAL de SEI de este tipo y las definiciones existentes de unidades de NAL de SEI, por ejemplo, tal como se define en el HEVC WD7, es que se puede permitir que este tipo distinto de unidad de NAL de SEI siga a la última unidad de NAL de VCL en la misma AU, en orden de decodificación, y se puede limitar de manera que no preceda a la primera unidad de NAL de VCL en la misma AU en el orden de decodificación. Las unidades de NAL de SEI y los mensajes de SEI convencionales pueden denominarse, respectivamente, unidades de NAL de SEI de prefijo y mensajes de SEI de prefijo, mientras que la unidad distinta de NAL de SEI y el mensaje de SEI que se describen en el presente documento pueden denominarse, respectivamente, unidades de NAL de SEI de sufijo y mensajes de SEI de sufijo.

Además de configurarse para generar una secuencia de vídeo codificado en base a cualquiera de, y todas, las combinaciones de asignaciones de unidades de NAL, ilustradas en las Tablas 2 a 7, la unidad de encapsulación 21 puede configurarse para generar una secuencia de vídeo codificado que incluye unidades de NAL de SEI de prefijo y de sufijo. Del mismo modo, la unidad de desencapsulación 29 puede configurarse para recibir una secuencia de vídeo codificado y analizar sintácticamente las unidades de acceso y las unidades de NAL, donde las unidades de NAL incluyen tipos de unidad de NAL de SEI de prefijo y de sufijo. Es decir, la unidad de desencapsulación 29 puede configurarse para extraer las unidades de NAL de SEI de sufijo desde las unidades de acceso. La Tabla 8 ilustra un ejemplo en el que todos los tipos de NAL ilustrados en las Tablas 4 a 6 se usan para la asignación de los tipos de

unidad de NAL, así como las unidades de NAL de SEI de prefijo y de sufijo.

tipo_unidad_nal	Contenido de la unidad de NAL y estructura sintáctica de RBSP	Clase de tipo de unidad de NAL
0	Sin especificar	no VCL
1	Fragmento codificado de imagen no-RAP, no TFD, no DLP y no TLA rbsp_capa_fragmento ()	VCL
2	Fragmento codificado de una imagen de TLA rbsp_capa_fragmento ()	VCL
3	Fragmento codificado de una imagen TFD rbsp_capa_fragmento ()	VCL
4	Fragmento codificado de una imagen DLP rbsp_capa_fragmento ()	VCL
5	Fragmento codificado de una imagen de CRA sin imágenes principales asociadas rbsp_capa_fragmento ()	VCL
6	Fragmento codificado de una imagen de CRA que no tiene TFD asociado (pero puede tener imágenes DLP asociadas) rbsp_capa_fragmento ()	VCL
7	Fragmento codificado de imagen de CRA que puede tener imágenes TFD asociadas rbsp_capa_fragmento ()	VCL
8	Fragmento codificado de una imagen de BLA sin imágenes principales asociadas rbsp_capa_fragmento ()	VCL
9	Fragmento codificado de una imagen de BLA que no tiene TFD asociada (pero puede tener imágenes DLP asociadas) rbsp_capa_fragmento ()	VCL
10	Fragmento codificado de imagen de BLA que puede tener asociadas imágenes TFD rbsp_capa_fragmento ()	VCL
11	Fragmento codificado de una imagen de IDR sin imágenes principales asociadas rbsp_capa_fragmento ()	VCL
12	Fragmento codificado de una imagen de IDR que puede tener asociadas imágenes principales rbsp_capa_fragmento ()	VCL
13..24	Reservados	no disponible
25	Conjunto de parámetros de vídeo rbsp_conjunto_parámetros_video ()	no VCL
26	Conjunto de parámetros de secuencia rbsp_conjunto_parámetros_sec ()	no VCL
27	Conjuntos de parámetros de imagen rbsp_conjunto_parámetros_imagen ()	no VCL
28	Conjunto de parámetros de adaptación rbsp_aps ()	no VCL
29	Delimitador de unidad de acceso rbsp_delimitador_unidad_acceso ()	no VCL
30	Datos de relleno rbsp_datos_relleno ()	no VCL
31	Información de mejora suplementaria (SEI) de prefijo rbsp_sei ()	no VCL
32	Información de mejora suplementaria (SEI) de sufijo rbsp_sei ()	no VCL
33..47	Reservados	no disponible
48..63	Sin especificar	no VCL

TABLA 8 Códigos de tipo de unidad de NAL y clases de tipo de unidad de NAL

5 Como se ha descrito anteriormente, además de las unidades de NAL de SEI, los tipos de unidad de NAL, no de VCL, incluyen unidades de NAL de VPS, SPS, PPS y APS. Según los tipos definidos de conjuntos de parámetros en el HEVC WD7, cada SPS se refiere a un Identificador de VPS, cada PPS se refiere a un Identificador de SPS y cada cabecera de fragmento se refiere a un Identificador de PPS y, posiblemente, a un identificador de APS. El
10 codificador de vídeo 20 y/o la unidad de estructura de codificación 19 pueden configurarse para generar conjuntos

de parámetros según los conjuntos de parámetros definidos en el HEVC WD7. Además, el codificador de vídeo 20 y/o la unidad de estructura de codificación 19 pueden configurarse para generar conjuntos de parámetros donde el Identificador de VPS y el Identificador de SPS (por ejemplo, con el Identificador de VPS anterior al Identificador de SPS) pueden señalizarse optativamente en las cabeceras de fragmento. En un ejemplo, donde el Identificador de VPS y el Identificador de SPS se señalizan en una cabecera de fragmento, ningún Identificador de VPS se encontrará en un SPS y ningún Identificador de SPS se encontrará en un PPS. Además, en un ejemplo, el Identificador de VPS y el Identificador de SPS pueden estar presentes en las cabeceras de fragmento de cada imagen de RAP y cada imagen puede estar asociada a un mensaje de SEI de punto de recuperación. Además, en otros ejemplos, el Identificador de VPS y el Identificador de SPS pueden estar presentes en la cabecera del fragmento para otras imágenes.

La FIG. 4 es un diagrama de bloques que ilustra una unidad ejemplar de encapsulación que puede implementar las técnicas descritas en esta divulgación. En el ejemplo ilustrado en la FIG. 4, la unidad de encapsulación 21 incluye un constructor de unidades de NAL de VCL 402, un constructor de unidades de NAL no de VCL 404, un constructor de unidades de acceso 406 y la interfaz de salida de flujo de bits 408. La unidad de encapsulación 21 recibe datos de vídeo codificado y sintaxis de alto nivel, y emite un flujo de bits de vídeo codificado. Los datos de vídeo codificado pueden incluir datos de vídeo residuales y datos sintácticos asociados a un fragmento. Los datos sintácticos de alto nivel pueden incluir, por ejemplo, elementos sintácticos incluidos en un conjunto de parámetros, mensajes de SEI u otros elementos sintácticos definidos por una norma de codificación de vídeo, tal como la inminente norma HEVC. Un flujo de bits de vídeo codificado puede incluir una o más secuencias de vídeo codificado y, en general, puede ajustarse a una norma de codificación de vídeo, tal como la inminente norma HEVC. Como se ha descrito anteriormente, las unidades de NAL de VCL incluyen un fragmento de datos de vídeo. El constructor de unidades de NAL de VCL 402 puede configurarse para recibir fragmentos de datos de vídeo codificado y generar unidades de NAL de VCL en base al tipo de imagen que incluye un fragmento. El constructor de unidades de NAL de VCL 402 puede configurarse para generar unidades de NAL de VCL según cualquiera de, y todas, las combinaciones de las asignaciones de NAL descritas anteriormente con respecto a las Tablas 2 a 8. El constructor de unidades de NAL de VCL 402 puede configurarse para incluir una cabecera en la unidad de NAL de VCL, donde la cabecera identifica el tipo de unidad de NAL de VCL.

Por ejemplo, el constructor de unidades de NAL de VCL 402 puede configurarse para recibir un fragmento de datos de vídeo, incluidos en una imagen de IDR y, (1) si la imagen de IDR no tiene asociadas imágenes principales, encapsular el fragmento de datos de vídeo en una unidad de NAL con un tipo que indica que la imagen de IDR no tiene imágenes principales o, (2) si la imagen de IDR tiene asociadas imágenes principales, encapsular el fragmento de datos de vídeo en una unidad de NAL con un tipo que indica que la imagen de IDR tiene imágenes principales. El constructor de unidades de NAL de VCL 402 puede configurarse para recibir un fragmento de datos de vídeo, incluidos en una imagen de CRA y, (1) si la imagen de CRA no tiene asociadas imágenes principales, encapsular el fragmento de datos de vídeo en una unidad de NAL con un tipo que indica que la imagen de CRA no tiene imágenes principales o, (2) si la imagen de CRA tiene asociadas imágenes principales, encapsular el fragmento de datos de vídeo en una unidad de NAL con un tipo que indica que la imagen de CRA tiene imágenes principales. Además, si las imágenes principales asociadas a la imagen de CRA son imágenes TFD, el constructor de unidades de NAL de VCL 402 puede configurarse para encapsular un fragmento de datos de vídeo en una unidad de NAL con un tipo que indica que la imagen principal asociada a las imágenes de CRA es TFD.

Además, si las imágenes principales asociadas a la imagen de CRA no son imágenes TFD, el constructor de unidades de NAL de VCL 402 puede configurarse para encapsular el fragmento de datos de vídeo en una unidad de NAL con un tipo que indica que las imágenes principales asociadas a las imágenes de CRA no son TFD. Además, el constructor de unidades de NAL de VCL 402 puede configurarse para recibir un fragmento de datos de vídeo, incluidos en una imagen de BLA y, (1) si la imagen de BLA no tiene asociadas imágenes principales, encapsular el fragmento de datos de vídeo en una unidad de NAL con un tipo que indica que la imagen de BLA no tiene imágenes principales o, (2) si la imagen de BLA tiene asociadas imágenes principales, encapsular el fragmento de datos de vídeo en una unidad de NAL con un tipo que indica que la imagen de BLA tiene imágenes principales. Además, si las imágenes principales asociadas a la imagen de BLA son imágenes TFD, el constructor de unidades de NAL de VCL 402 puede configurarse para encapsular un fragmento de datos de vídeo en una unidad de NAL con un tipo que indica que las imágenes principales asociadas a las imágenes de BLA son TFD. Además, si las imágenes principales asociadas a la imagen de BLA no son imágenes TFD, el constructor de unidades de NAL de VCL 402 puede configurarse para encapsular el fragmento de datos de vídeo en una unidad de NAL con un tipo que indica que la imagen principal asociada a las imágenes de BLA no es TFD.

La FIG. 5 es un diagrama de flujo que ilustra un ejemplo de generación de unidades de NAL de VCL según las técnicas de esta divulgación. Aunque el ejemplo de la generación de unidades de NAL de VCL, ilustrado en la FIG. 5, se describe como realizado por el constructor de unidades de NAL de VCL 402, cualquier combinación del dispositivo de origen 12, el codificador de vídeo 20, la unidad de encapsulación 21 y las combinaciones de componentes de los mismos puede llevar a cabo el ejemplo de generación de unidades de NAL de VCL, ilustrado en la FIG. 5. Como se ilustra en la FIG. 5, el constructor de unidades de NAL de VCL 402 recibe un fragmento de datos de vídeo (502). El fragmento de datos de vídeo puede ser de datos de vídeo codificado según cualquiera de las técnicas de codificación descritas en este documento. El fragmento de datos de vídeo puede incluirse en uno de los

tipos de imágenes descritos en este documento. El constructor de unidades de NAL de VCL 402 determina si el fragmento de datos de vídeo se incluye en una imagen de IDR o de CRA (504).

5 Si el fragmento de datos de vídeo está incluido en una imagen de IDR (rama "IDR" de 504), el constructor de unidades de NAL de VCL 402 determina si la imagen de IDR tiene o no asociadas imágenes principales (506). Si la imagen de IDR no tiene asociadas imágenes principales (rama "NO" de 506), el constructor de unidades de NAL de VCL 402 genera una unidad de NAL de VCL que indica que la imagen de IDR no tiene asociadas imágenes principales (508). Si la imagen de IDR tiene asociadas imágenes principales (rama "SÍ" de 506), el constructor de unidades de NAL de VCL 402 genera una unidad de NAL de VCL que indica que la imagen de IDR tiene asociadas imágenes principales (510).

15 Si el fragmento de datos de vídeo está incluido en una imagen de CRA, el constructor de unidades de NAL de VCL 402 determina si la imagen de CRA tiene o no asociadas imágenes principales (512). Si la imagen de CRA no tiene asociadas imágenes principales (rama "NO" de 512), el constructor de unidades de NAL de VCL 402 genera una unidad de NAL de VCL que indica que la imagen de CRA no tiene asociadas imágenes principales (514). Si la imagen de CRA tiene asociadas imágenes principales (rama "SÍ" de 512), el constructor de unidades de NAL de VCL 402 determina si las imágenes principales asociadas son o no imágenes TFD (516).

20 Si las imágenes principales asociadas de la imagen de CRA son imágenes TFD (rama "SÍ" de 516), el constructor de unidades de NAL de VCL 402 genera una unidad de NAL de VCL que indica que las imágenes principales asociadas de la imagen de CRA son imágenes TFD (518). Si las imágenes principales asociadas de la imagen de CRA no son imágenes TFD (rama "NO" de 516), el constructor de unidades de NAL de VCL 402 genera una unidad de NAL de VCL que indica que las imágenes principales asociadas no son imágenes TFD (520).

25 El constructor de unidades de NAL de VCL 402 puede generar unidades de NAL mediante la encapsulación de datos de fragmento en una unidad de NAL, e incluyendo un valor de tipo de unidad de NAL en una cabecera de unidad de NAL. Cada valor de tipo de unidad de NAL puede corresponder a un respectivo tipo de unidad de NAL. En un ejemplo, los valores de tipos de unidad de NAL se pueden definir según la Tabla 7. Las unidades de NAL generadas pueden ser emitidas por el constructor de unidades de NAL 402 para acceder al constructor de unidades 30 406, para su inclusión en una unidad de acceso (522).

35 De esta manera, la unidad de encapsulación 21 representa un ejemplo de un dispositivo para generar un flujo de bits que incluye datos de vídeo, incluyendo el dispositivo un procesador configurado para determinar si una imagen de punto de acceso aleatorio (RAP) es o no de un tipo que pueda tener asociadas imágenes principales y, si la imagen de RAP comprende una imagen de actualización de decodificador instantánea (IDR) o una imagen de acceso aleatorio puro (CRA), encapsular un fragmento de la imagen de RAP en una unidad de capa de abstracción de red (NAL), en donde la unidad de NAL incluye un valor de tipo de unidad de NAL que indica si la imagen de RAP es o no de un tipo que pueda tener asociadas imágenes principales, y generar un flujo de bits que incluya la unidad de NAL.

40 Del mismo modo, el procedimiento de la FIG. 5 representa un ejemplo de un procedimiento de generación de un flujo de bits que incluye datos de vídeo, incluyendo el procedimiento la determinación de si la imagen de un punto de acceso aleatorio (RAP) es o no de un tipo que pueda tener asociadas imágenes principales y si la imagen de RAP comprende una imagen de actualización de decodificador instantánea (IDR) o una imagen de acceso aleatorio puro (CRA), la encapsulación de un fragmento de la imagen de RAP en una unidad de capa de abstracción de red (NAL), 45 en donde la unidad de NAL incluye un valor de tipo de unidad de NAL que indica si la imagen de RAP es o no de un tipo que pueda tener asociadas imágenes principales, y la generación de un flujo de bits que incluye la unidad de NAL.

50 Haciendo de nuevo referencia a la FIG. 4, el constructor de unidades de NAL no de VCL 404 puede configurarse para recibir los elementos sintácticos de alto nivel, tales como elementos sintácticos incluidos en los conjuntos de parámetros y en mensajes de SEI, como se ha descrito anteriormente, y generar unidades de NAL, no de VCL, en base a cualquiera de, y todas, las combinaciones de las asignaciones de unidades de NAL descritas anteriormente con respecto a las Tablas 2 a 8. El constructor de unidades de NAL no de VCL 404 puede configurarse para generar unidades de NAL, no de VCL, encapsulando datos sintácticos en una unidad de NAL e incluyendo un valor de tipo 55 de unidad de NAL en una cabecera de unidad de NAL. Por ejemplo, el constructor de unidades de NAL, no de VCL, puede configurarse para recibir elementos sintácticos incluidos en un conjunto de parámetros, e incluye un valor de tipo de unidad de NAL que indica el tipo de conjunto de parámetros en la cabecera de unidad de NAL.

60 Además, el constructor de unidades de NAL, no de VCL, 404 puede configurarse para recibir mensajes de SEI de nivel AU y generar unidades de NAL de mensajes de SEI. En un ejemplo, el constructor de unidades de NAL, no de VCL, 404 puede configurarse para generar dos tipos de unidades de NAL de mensajes de SEI, donde un primer tipo de unidad de NAL de SEI indica que dicha unidad de NAL de SEI puede seguir a la última unidad de NAL de VCL en una unidad de acceso, en orden de decodificación, y el segundo tipo de unidad de NAL de SEI indica que dicha unidad de NAL de SEI tal vez no siga a la última unidad de NAL de VCL en una unidad de acceso en orden 65 de decodificación. Además, el primer tipo de NAL de SEI puede estar limitado de manera que no se pueda permitir que preceda a la primera unidad de NAL VCL en la misma unidad de acceso, en orden de decodificación. El primer tipo

de unidad de NAL puede denominarse unidad de NAL de SEI de sufijo y el segundo tipo de unidades de NAL puede mencionarse como unidades de NAL de SEI de prefijo. El constructor de unidades de NAL, no de VCL, 404 emite unidades de NAL, no de VCL, al constructor de unidades de acceso 406.

5 El constructor de unidades de acceso 406 puede configurarse para recibir unidades de NAL de VCL y unidades de NAL, no de VCL, y generar unidades de acceso. El constructor de unidades de acceso 406 puede recibir cualquier tipo de unidad de NAL definida en las Tablas 2 a 8. El constructor de unidades de acceso de VCL 406 puede configurarse para generar unidades de acceso en base a cualquiera, y todas y cada una, las combinaciones de tipos de unidad de NAL descritos en este documento. Como se ha descrito anteriormente, según el HEVC WD7, una
10 unidad de acceso es un conjunto de unidades de NAL que son consecutivas en el orden de decodificación y contienen una imagen codificada. Por lo tanto, el constructor de unidades de acceso 406 puede configurarse para recibir una pluralidad de unidades de NAL y disponer la pluralidad de unidades de NAL según un orden de decodificación. Además, el constructor de unidades de acceso de 406 puede configurarse para disponer una unidad de NAL de SEI de sufijo, como se ha descrito anteriormente, de manera que siga a la última unidad de NAL de VCL
15 en una unidad de acceso y/o no preceda a la primera unidad de NAL de VCL en la misma unidad de acceso.

La FIG. 6 es un diagrama de flujo que ilustra un ejemplo de generación de unidades de NAL, no de VCL, según las técnicas de esta divulgación. Aunque el ejemplo de la generación de unidades de NAL, no de VCL, ilustrado en la FIG. 6, se describe como realizado por un constructor de unidades de NAL, no de VCL, 404 y un constructor de
20 unidades de acceso 406, cualquier combinación del dispositivo de origen 12, el codificador de vídeo 20, la unidad de encapsulación 21 y las combinaciones de componentes de los mismos pueden llevar a cabo el ejemplo de generación de unidades de NAL no de VCL, ilustrado en la FIG. 6.

Como se muestra en la FIG. 6, el constructor de unidades de NAL, no de VCL, 404 recibe un mensaje de SEI (602). El mensaje de SEI puede ser cualquier tipo de mensaje de SEI descrito anteriormente con respecto a la Tabla 1. El constructor de unidades de NAL, no de VCL, 404 determina si el mensaje de SEI es un mensaje de SEI de prefijo o un mensaje de SEI de sufijo (604).
25

Si el mensaje de SEI es un mensaje de SEI de sufijo (rama "SUFIJO" de 604), el constructor de unidades de NAL, no de VCL, 404 genera un valor de tipo para una unidad de NAL de SEI, que indica que la unidad de NAL de SEI es un mensaje de SEI de sufijo (606). Si el mensaje de SEI es un mensaje de SEI de prefijo (rama "PREFIJO" de 604), el constructor de unidades de NAL, no de VCL, 404 genera un valor de tipo para una unidad de NAL de SEI, que indica que la unidad de NAL de SEI es un mensaje de SEI convencional (608).
30

El constructor de unidades de acceso 406 recibe las unidades de NAL generadas, que pueden incluir cualquier combinación de los tipos de unidades de NAL descritas anteriormente con respecto a las Tablas 2 a 8 (610). El constructor de unidades de acceso 406 genera las unidades de acceso incluidas en las unidades de NAL recibidas (612). Si la unidad de acceso generada incluye una unidad de NAL de SEI de sufijo, las unidades de NAL de la unidad de acceso pueden disponerse de tal manera que la unidad de NAL de SEI de sufijo no preceda a la primera
35 unidad de NAL de VCL en la misma unidad de acceso, pero pueda seguir a la última unidad de NAL de VCL en una unidad de acceso en orden de decodificación.
40

De esta manera, la unidad de encapsulación 21 representa un ejemplo de un procesador configurado para determinar si un mensaje de información de mejora suplementaria (SEI) es un mensaje de SEI de prefijo o un mensaje de SEI de sufijo, en donde el mensaje de SEI incluye datos relacionados con los datos de vídeo codificado, encapsular el mensaje de SEI en una unidad de NAL de SEI, en donde la unidad de NAL de SEI incluye un valor de tipo de unidad de NAL que indica si la unidad de NAL de SEI es una unidad de NAL de SEI de prefijo o una unidad de NAL de SEI de sufijo y si el mensaje de SEI es un mensaje de SEI de prefijo o un mensaje de SEI de sufijo, y generar un flujo de bits que incluye al menos la unidad de NAL de SEI.
45
50

Del mismo modo, el procedimiento de la FIG. 6 representa un ejemplo de un procedimiento de generación de un flujo de bits que incluye datos de vídeo, incluyendo el procedimiento la determinación de si un mensaje de información de mejora suplementaria (SEI) es un mensaje de SEI de prefijo o un mensaje de SEI de sufijo, en donde el mensaje de SEI incluye datos relacionados con los datos de vídeo codificado, la encapsulación del mensaje de SEI en una unidad de NAL de SEI, en donde la unidad de NAL de SEI incluye un valor de tipo de unidad de NAL, que indica si la unidad de NAL de SEI es una unidad de NAL de SEI de prefijo o una unidad de NAL de SEI de sufijo y si el mensaje de SEI es un mensaje de SEI de prefijo o un mensaje de SEI de sufijo, y la generación de un flujo de bits que incluye al menos la unidad de NAL de SEI.
55

Haciendo de nuevo referencia a la FIG. 4, la interfaz de salida de flujo de bits 408 puede configurarse para recibir unidades de acceso y generar una secuencia de vídeo codificado. La interfaz de salida de flujo de bits 408 se puede configurar además para la emisión de una secuencia de vídeo codificado como parte de un flujo de bits de vídeo codificado, donde un flujo de bits de vídeo codificado incluye una o más secuencias de vídeo codificado, en base a cualquiera de, y todas, las combinaciones de los tipos de unidad de NAL descritos en este documento. Como se ha descrito anteriormente, según el HEVC WD7, una secuencia de vídeo codificado es un conjunto de unidades de acceso que son consecutivas en el orden de decodificación. Por lo tanto, la interfaz de salida de flujo de bits 408
60
65

puede configurarse para recibir una pluralidad de unidades de acceso y disponer la pluralidad de unidades de acceso según un orden de decodificación.

5 Como se ha descrito anteriormente, la unidad de estructura de codificación 19 y/o el codificador de vídeo 20 pueden configurarse para generar elementos sintácticos incluidos en un conjunto de parámetros que incluye el elemento sintáctico **indicador_velocidad_imágenes_fija**, que puede estar incluido en un conjunto de parámetros de VUI, que puede estar incluido en un SPS, según lo dispuesto en el HEVC WD7. Además, la unidad de estructura de codificación 19 y/o el codificador de vídeo 20 pueden configurarse para generar el elemento sintáctico **indicador_velocidad_imágenes_fija**, donde el elemento sintáctico **indicador_velocidad_imágenes_fija** incluye semántica que se modifica a partir de las proporcionadas en el HEVC WD7. Por ejemplo, según la semántica actual del **indicador_velocidad_imágenes_fija** en el HEVC WD7, cuando el **indicador_velocidad_imágenes_fija** es igual a 1, se requiere que la diferencia entre los tiempos de presentación de dos imágenes continuas en orden de salida sea igual al pulso del reloj. Sin embargo, esto requeriría el cambio del valor de la **escala_temporal** o del **núm_unidades_en_pulso** cuando algunas capas temporales máximas se descarten para la adaptación del flujo, en base a la ajustabilidad temporal a escala.

20 En un ejemplo, en lugar de requerir que el delta (es decir, la diferencia entre los tiempos de presentación de dos imágenes seguidas en orden de salida) sea exactamente igual al pulso del reloj, puede requerirse que el delta sea un número entero de pulso(s) de reloj. De esta manera, la unidad de estructura de codificación 19 y/o el codificador de vídeo 20 pueden estar configurados para generar el elemento sintáctico **indicador_velocidad_imágenes_fija**, de modo tal que, cuando el **indicador_velocidad_imágenes_fija** es igual a 1, se requiera que la diferencia entre los tiempos de presentación de dos imágenes continuas en orden de salida sea igual a un número entero del pulso de reloj.

25 En otro ejemplo, puede requerirse que la unidad de estructura de codificación 19 y/o el codificador de vídeo 20 señalicen un **indicador_velocidad_imágenes_fija** para cada capa temporal. Además, en este ejemplo, si el **indicador_velocidad_imágenes_fija** para una capa temporal particular es igual a 1, es decir, la representación de la capa temporal tiene una velocidad de imágenes constante, puede señalizarse un valor N, y el delta (entre los tiempos de presentación de dos imágenes continuas en orden de salida) para la representación de capa temporal puede ser igual a N pulsos de reloj.

35 En otro ejemplo, la unidad de estructura de codificación 19 y/o el codificador de vídeo 20 pueden configurarse para señalar optativamente un **indicador_velocidad_imágenes_fija** para cada capa temporal. En este ejemplo, si el **indicador_velocidad_imágenes_fija** para una capa particular está presente y es igual a 1, es decir, la representación de capa temporal tiene una velocidad de imágenes constante, puede señalizarse un valor N y el delta (entre los tiempos de presentación de dos imágenes continuas en orden de salida) para la representación de capa temporal es igual a N pulsos de reloj. En el caso en que el **indicador_velocidad_imágenes_fija** se señalice optativamente para cada capa temporal, suponiendo que el **indicador_velocidad_imágenes_fija** sea señalado para la capa temporal más alta y que el valor sea igual a 1, a continuación, para cada capa temporal particular que no tenga ningún **indicador_velocidad_imagen_fija** señalado, el valor del **indicador_velocidad_imágenes_fija** puede obtenerse igual al **indicador_velocidad_imágenes_fija** señalado para la capa temporal más alta), y el valor de N se obtiene igual a $2^{\max_Tid - currTid}$, en donde \max_Tid es igual al valor más alto de $temporal_id$ y $currTid$ es igual al $temporal_id$ de la capa temporal particular.

45 La FIG. 7 es un diagrama de flujo que ilustra un ejemplo de señalización de un valor de delta de tiempo de presentación. Aunque el ejemplo de señalización de un valor de delta de tiempo de presentación que se ilustra en la FIG. 7 se describe como realizado por la unidad de encapsulación 21, cualquier combinación del dispositivo de origen 12, el codificador de vídeo 20, la unidad de encapsulación 21 y las combinaciones de componentes de los mismos pueden llevar a cabo el ejemplo de señalización del valor de delta de tiempo de presentación ilustrado en la FIG. 7.

55 Como se ilustra en el ejemplo de la FIG. 7, la unidad de encapsulación 21 genera un indicador que indica si un delta entre un tiempo de presentación (por ejemplo, un valor de POC) de una primera imagen y un tiempo de presentación de una segunda imagen es o no un número entero del valor de un pulso de reloj (702). En otras palabras, la unidad de encapsulación 21 puede generar datos que indican si una diferencia (por ejemplo, el delta) entre los tiempos de presentación de una primera imagen y una segunda imagen es o no un múltiplo entero del valor del pulso de reloj. El indicador descrito en la FIG. 7 representa un ejemplo de tales datos generados. En algunos casos, la unidad de encapsulación 21 puede recibir un valor para el indicador desde la unidad de estructura de codificación 19 o el codificador de vídeo 20. El indicador puede ser cualquiera de los elementos sintácticos **indicador_velocidad_imágenes_fija** descritos anteriormente.

65 En un ejemplo, la unidad de encapsulación 21 determina si un valor para el indicador puede o no indicar que el delta es un número entero del valor del pulso de reloj (704). Cuando el indicador indica que el delta es un valor entero del pulso de reloj (rama "Sí" de 704), la unidad de encapsulación 21 puede generar un valor entero N (706) representativo del múltiplo entero del valor del pulso de reloj. El valor entero N puede ser utilizado por un dispositivo de decodificación, tal como el dispositivo de destino 14, para determinar el valor delta en el que el delta es un

múltiplo entero del valor del pulso de reloj. En un ejemplo, el valor entero N puede ser un valor entre 0 y 2.047 y puede indicar un valor de uno menos que el número entero de pulsos de reloj al que es igual el delta. A continuación, la unidad de encapsulación 21 puede emitir el indicador y el valor entero N como parte de un flujo de bits (708).

5 Por otro lado, cuando la unidad de encapsulación 21 determina que el indicador indica que el valor de delta no es un múltiplo entero del pulso de reloj (rama "NO" de 704), la unidad de encapsulación 21 puede simplemente emitir el indicador (710).

10 De esta manera, el dispositivo de origen 12 representa un ejemplo de un procesador configurado para generar datos que indican si una diferencia entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen es o no un múltiplo entero de un valor de pulso de reloj y, cuando los datos indican que la diferencia es el múltiplo entero del valor del pulso de reloj, generar datos representativos del múltiplo entero.

15 Del mismo modo, el procedimiento de la FIG. 7 representa un ejemplo de un procedimiento para generar un flujo de bits que incluye datos de vídeo, incluyendo el procedimiento la generación de datos que indican si una diferencia entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen es o no un múltiplo entero de un valor de pulso de reloj y, cuando los datos indican que la diferencia es el múltiplo entero del valor del pulso de reloj, la generación de datos representativos del múltiplo entero.

20 Como se ha descrito anteriormente, la unidad de encapsulación 21 recibe datos de vídeo codificado. La FIG. 8 es un diagrama de bloques que ilustra un ejemplo del codificador de vídeo 20, que puede generar datos de vídeo codificado. Como se muestra en la FIG. 8, el codificador de vídeo 20 recibe datos de vídeo y datos sintácticos de alto nivel. Un codificador de vídeo 20 actúa habitualmente sobre bloques de vídeo dentro de los fragmentos de vídeo individuales, con el fin de codificar los datos de vídeo. Un bloque de vídeo puede corresponder a un nodo de codificación dentro de una CU. Los bloques de vídeo pueden tener tamaños fijos o variables y pueden diferir en tamaño, según una norma de codificación especificada. Además, el codificador de vídeo 20 puede generar datos sintácticos, tales como datos sintácticos basados en bloques, datos sintácticos basados en tramas y datos sintácticos basados en GOP, por ejemplo, en una cabecera de trama, una cabecera de bloque, una cabecera de fragmento o una cabecera de GOP. Los datos sintácticos del GOP pueden describir un cierto número de tramas en el GOP respectivo, y los datos sintácticos de trama pueden indicar una modalidad de codificación/predicción usada para codificar la trama correspondiente.

35 En el ejemplo de la FIG. 8, el codificador de vídeo 20 incluye una unidad de selección de modalidad 40, una memoria de imágenes de referencia 64, un sumador 50, una unidad de procesamiento de transformación 52, una unidad de cuantización 54 y una unidad de codificación por entropía 56. A su vez, la unidad de selección de modalidad 40 incluye una unidad de compensación de movimiento 44, una unidad de estimación de movimiento 42, una unidad de intra-predicción 46 y una unidad de división 48. Para la reconstrucción de bloques de vídeo, el codificador de vídeo 20 incluye además una unidad de cuantización inversa 58, una unidad de transformación inversa 60 y un sumador 62. También puede incluirse un filtro de desbloqueo (no mostrado en la FIG. 8) para filtrar límites de bloque, para eliminar distorsiones de efecto pixelado del vídeo reconstruido. Si se desea, el filtro de desbloqueo filtrará habitualmente la salida del sumador 62. También pueden usarse filtros adicionales (en bucle o pos-bucle), además del filtro de desbloqueo. Estos filtros no se muestran por razones de brevedad pero, si se desea, pueden filtrar la salida del sumador 50 (tal como un filtro en bucle).

45 Durante el proceso de codificación, el codificador de vídeo 20 recibe una trama o un fragmento de vídeo que va a codificarse. La trama o el fragmento pueden ser divididos en múltiples bloques de vídeo. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 llevan a cabo la codificación inter-predictiva del bloque de vídeo recibido con respecto a uno o más bloques en una o más tramas de referencia, para proporcionar la predicción temporal. La unidad de intra-predicción 46, como alternativa, puede llevar a cabo la codificación intra-predictiva del bloque de vídeo recibido, con respecto a uno o más bloques vecinos en la misma trama o fragmento que el bloque que va a codificarse, para proporcionar la predicción espacial. El codificador de vídeo 20 puede llevar a cabo múltiples pasadas de codificación, por ejemplo, para seleccionar una modalidad de codificación adecuada para cada bloque de datos de vídeo.

55 Además, la unidad de división 48 puede dividir bloques de datos de vídeo en sub-bloques, en base a la evaluación de los anteriores esquemas de división en las pasadas de codificación anteriores. Por ejemplo, la unidad de división 48 puede dividir inicialmente una trama o un fragmento en varias LCU, y dividir cada una de las LCU en varias sub-CU, basándose en un análisis de distorsión de velocidad (por ejemplo, optimización de distorsión de velocidad). La unidad de selección de modalidad 40 puede producir además una estructura de datos de árbol cuádruple que indica la división de una LCU en varias sub-CU. Las CU de nodos de hojas del árbol cuádruple pueden incluir una o más PU y una o más TU.

60 La unidad de selección de modalidad 40 puede seleccionar una de las modalidades de codificación (intra o inter), por ejemplo, en función de los resultados de errores, y proporciona el bloque intra-codificado o inter-codificado resultante al sumador 50 para generar datos de bloque residuales, y al sumador 62 para reconstruir el bloque

codificado, para su uso como una trama de referencia. La unidad de selección de modalidad 40 proporciona además elementos sintácticos, tales como vectores de movimiento, indicadores de intra-modalidad, información de división y otra información sintáctica de este tipo, a la unidad de codificación por entropía 56.

5 La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar sumamente integradas, pero se ilustran por separado con fines conceptuales. La estimación de movimiento, realizada por la unidad de estimación de movimiento 42, es el proceso de generación de vectores de movimiento, que estiman el movimiento para los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una PU de un bloque de vídeo dentro de una trama o imagen actual de vídeo, con respecto a un
10 bloque predictivo dentro de una trama de referencia (u otra unidad codificada), con respecto al bloque actual que está siendo codificado dentro de la trama actual (u otra unidad codificada). Un bloque predictivo es un bloque que se revela como estrechamente coincidente con el bloque a codificar, en lo que respecta a la diferencia de píxeles, que puede determinarse mediante la suma de una diferencia absoluta (SAD), una suma de diferencia de cuadrados (SSD) u otras métricas de diferencia. En algunos ejemplos, el codificador de vídeo 20 puede calcular valores para
15 posiciones de fracciones de píxel de imágenes de referencia almacenadas en la memoria de imágenes de referencia 64. Por ejemplo, el codificador de vídeo 20 puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones fraccionarias de píxel de la imagen de referencia. Por lo tanto, la unidad de estimación de movimiento 42 puede realizar una búsqueda de movimiento con respecto a las posiciones de píxeles completos y a las posiciones de fracciones de píxel, y emitir un vector de movimiento con una precisión de
20 fracciones de píxel.

La unidad de estimación de movimiento 42 calcula un vector de movimiento para una PU de un bloque de vídeo en un fragmento inter-codificado, comparando la posición de la PU con la posición de un bloque predictivo de una imagen de referencia. La imagen de referencia puede seleccionarse a partir de una primera lista de imágenes de
25 referencia (Lista 0) o una segunda lista de imágenes de referencia (Lista 1), cada una de las cuales identifica una o más imágenes de referencia almacenadas en la memoria de imágenes de referencia 64. La unidad de estimación de movimiento 42 envía el vector de movimiento calculado a la unidad de codificación por entropía 56 y a la unidad de compensación de movimiento 44.

30 La compensación de movimiento, llevada a cabo por la unidad de compensación de movimiento 44, puede implicar capturar o generar el bloque predictivo en base al vector de movimiento determinado por la unidad de estimación de movimiento 42. De nuevo, la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden integrarse funcionalmente, en algunos ejemplos. Tras recibir el vector de movimiento para la PU del bloque de vídeo actual, la unidad de compensación de movimiento 44 puede localizar el bloque predictivo al que
35 apunta el vector de movimiento en una de las listas de imágenes de referencia. El sumador 50 forma un bloque de vídeo residual restando los valores de píxel del bloque predictivo a los valores de píxel del bloque de vídeo actual que está siendo codificado, generando valores de diferencia de píxel, como se expone posteriormente. En general, la unidad de estimación de movimiento 42 lleva a cabo una estimación de movimiento con respecto a los componentes de luminancia, y la unidad de compensación de movimiento 44 usa vectores de movimiento calculados
40 en función de los componentes de luminancia, tanto para los componentes de crominancia como para los componentes de luminancia. La unidad de selección de modalidad 40 también puede generar elementos sintácticos asociados a los bloques de vídeo y al fragmento de vídeo, para su uso por parte del decodificador de vídeo 30 a la hora de decodificar los bloques de vídeo del fragmento de vídeo.

45 La unidad de intra-predicción 46 puede intra-predicir un bloque actual, como alternativa a la inter-predicción llevada a cabo por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como se ha descrito anteriormente. En particular, la unidad de intra-predicción 46 puede determinar una modalidad de intra-predicción a usar para codificar un bloque actual. En algunos ejemplos, la unidad de intra-predicción 46 puede codificar un bloque actual usando varias modalidades de intra-predicción, por ejemplo, durante diferentes pasadas
50 de codificación, y la unidad de intra-predicción 46 (o la unidad de selección de modalidad 40, en algunos ejemplos) puede seleccionar una modalidad adecuada de intra-predicción a usar, entre las modalidades probadas.

Por ejemplo, la unidad de intra-predicción 46 puede calcular valores de distorsión de velocidad usando un análisis de distorsión de velocidad para las diversas modalidades de intra-predicción probadas, y seleccionar la modalidad de
55 intra-predicción que tenga las mejores características de distorsión de velocidad entre las modalidades probadas. El análisis de distorsión de velocidad determina generalmente una magnitud de distorsión (o error) entre un bloque codificado y un bloque original, no codificado, que se codificó para generar el bloque codificado, así como una velocidad binaria (es decir, un número de bits) usada para generar el bloque codificado. La unidad de intra-predicción 46 puede calcular razones a partir de las distorsiones y velocidades para los diversos bloques
60 codificados, para determinar qué modalidad de intra-predicción presenta el mejor valor de distorsión de velocidad para el bloque.

Después de seleccionar una modalidad de intra-predicción para un bloque, la unidad de intra-predicción 46 puede proporcionar información, que indica la modalidad de intra-predicción seleccionada para el bloque, a la unidad de
65 codificación por entropía 56. La unidad de codificación por entropía 56 puede codificar la información que indica la modalidad de intra-predicción seleccionada. El codificador de vídeo 20 puede incluir datos de configuración en el

flujo de bits transmitido, que pueden incluir una pluralidad de tablas de índices de modalidades de intra-predicción y una pluralidad de tablas de índices de modalidades de intra-predicción modificadas (también denominadas tablas de correlación de palabras de código), definiciones de contextos de codificación para varios bloques e indicaciones de la modalidad de intra-predicción más probable, una tabla de índices de modalidades de intra-predicción y una tabla modificada de índices de modalidades de intra-predicción, a usar en cada uno de los contextos.

El codificador de vídeo 20 forma un bloque de vídeo residual, restando los datos de predicción de la unidad de selección de modalidad 40 del bloque de vídeo original que está siendo codificado. El sumador 50 representa el componente o los componentes que realizan esta operación de resta. La unidad de procesamiento de transformaciones 52 aplica una transformación, tal como una transformación discreta de coseno (DCT) o una transformación conceptualmente similar, al bloque residual, generando un bloque de vídeo que comprende valores residuales de coeficientes de transformación. La unidad de procesamiento de transformaciones 52 puede llevar a cabo otras transformaciones que son conceptualmente similares a la DCT. También podrían usarse transformaciones de ondículas, transformaciones de enteros, transformaciones de sub-bandas u otros tipos de transformaciones. En cualquier caso, la unidad de procesamiento de transformaciones 52 aplica la transformación al bloque residual, generando un bloque de coeficientes de transformación residuales. La transformación puede convertir la información residual, desde un dominio de valores de píxel a un dominio de transformación, tal como un dominio de frecuencia. La unidad de procesamiento de transformaciones 52 puede enviar los coeficientes de transformación resultantes a la unidad de cuantización 54. La unidad de cuantización 54 cuantiza los coeficientes de transformación para reducir adicionalmente la velocidad de bits. El proceso de cuantización puede reducir la profundidad de bits asociada a algunos de, o a todos, los coeficientes. El grado de cuantización puede modificarse ajustando un parámetro de cuantización. En algunos ejemplos, la unidad de cuantización 54 puede realizar a continuación un escaneado de la matriz, incluyendo los coeficientes de transformación cuantizados. Como alternativa, la unidad de codificación por entropía 56 puede realizar el escaneado.

Tras la cuantización, la unidad de codificación por entropía 56 codifica por entropía los coeficientes de transformación cuantizados. Por ejemplo, la unidad de codificación por entropía 56 puede llevar a cabo la codificación de longitud variable adaptable al contexto (CAVLC), la codificación aritmética binaria adaptable al contexto (CABAC), la codificación aritmética binaria adaptable al contexto y basada en la sintaxis (SBAC), la codificación por entropía mediante la división en intervalos de probabilidades (PIPE) u otra técnica de codificación por entropía. En el caso de la codificación por entropía basada en el contexto, el contexto puede basarse en bloques vecinos. Tras la codificación por entropía realizada por la unidad de codificación por entropía 56, el flujo de bits codificado puede transmitirse a otro dispositivo (por ejemplo, el decodificador de vídeo 30) o guardarse para su posterior transmisión o recuperación.

La unidad de cuantización inversa 58 y la unidad de transformación inversa 60 aplican la cuantización inversa y la transformación inversa, respectivamente, para reconstruir el bloque residual en el dominio de píxeles, por ejemplo, para su uso posterior como un bloque de referencia. La unidad de compensación de movimiento 44 puede calcular un bloque de referencia añadiendo el bloque residual a un bloque predictivo de una de las tramas de la memoria de imágenes de referencia 64. La unidad de compensación de movimiento 44 también puede aplicar uno o más filtros de interpolación al bloque residual reconstruido para calcular valores de fracciones de píxel para su uso en la estimación de movimiento. El sumador 62 añade el bloque residual reconstruido al bloque de predicción compensado por movimiento, generado por la unidad de compensación de movimiento 44, para generar un bloque de vídeo reconstruido para su almacenamiento en la memoria de imágenes de referencia 64. El bloque de vídeo reconstruido puede ser usado por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 como un bloque de referencia para inter-codificar un bloque en una trama de vídeo posterior.

Como se ha descrito anteriormente, la unidad de desencapsulación 29 puede configurarse para recibir la secuencia de vídeo codificado y analizar sintácticamente las unidades de acceso y unidades de NAL, donde las unidades de NAL se asignan basándose en todas y cada una de las combinaciones de las asignaciones de unidad de NAL ilustradas en las Tablas 2 a 7. Además, la unidad de desencapsulación 29 y el decodificador de vídeo 30 pueden reconstruir los datos de vídeo en base a las asignaciones de tipo de unidad de NAL. En un ejemplo, la unidad de desencapsulación 29 puede configurarse para recibir una unidad de NAL, donde la unidad de NAL incluye un valor de tipo de NAL, y determinar si la unidad de NAL encapsula o no un fragmento codificado de datos de vídeo incluidos en una imagen de RAP asociada a una imagen principal, en base al valor del tipo de NAL, y el decodificador de vídeo 30 puede configurarse para reconstruir datos de vídeo en base a si la unidad de NAL encapsula o no un fragmento codificado de datos de vídeo incluidos en una imagen de RAP asociada a una imagen principal. En otro ejemplo, la unidad de desencapsulación 29 puede configurarse para recibir una unidad de NAL, donde la unidad de NAL incluye un valor de tipo de NAL, y determinar si la unidad de NAL encapsula o no un mensaje de SEI de nivel de AU, en base al valor de tipo de NAL, y el decodificador de vídeo 30 puede configurarse para reconstruir los datos de vídeo en base a si la unidad de NAL encapsula o no un mensaje de SEI de nivel AU. En algunos casos, la reconstrucción de datos de vídeo puede incluir la generación de un flujo de bits empalmado, como se ha descrito anteriormente, y el decodificador de vídeo 30 puede determinar los tiempos de presentación de imágenes en el flujo de vídeo empalmado en base a las determinaciones del tipo de unidad de NAL.

Además, como se ha descrito anteriormente, un dispositivo de origen, como el dispositivo de origen 12, puede

configurarse para señalar el delta entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen, donde la señalización utiliza cualquiera de los elementos sintácticos **indicador_velocidad_imágenes_fija** descritos anteriormente. Por lo tanto, el dispositivo de destino 14, la unidad de desencapsulación 29 y el decodificador de vídeo 30 pueden configurarse para determinar los tiempos de presentación de una primera imagen y una segunda imagen, y presentar las imágenes en consecuencia.

La FIG. 9 es un diagrama de flujo que ilustra un procedimiento ejemplar de determinación de un valor de delta de tiempo de presentación. Aunque el ejemplo de señalización de un valor de delta de tiempo de presentación que se ilustra en la FIG. 9 se describe como realizado por la unidad de desencapsulación 29, cualquier combinación del dispositivo de destino 14, el decodificador de vídeo 30, la unidad de desencapsulación 29 y las combinaciones de componentes de los mismos pueden llevar a cabo el ejemplo de la determinación de un valor de delta de tiempo de presentación ilustrado en la FIG. 9. Como se ilustra en la FIG. 9, la unidad de desencapsulación 29 obtiene una primera imagen (902). La primera imagen puede ser una imagen codificada correspondiente a una unidad de acceso. La unidad de desencapsulación 29 obtiene una segunda imagen (904). La segunda imagen puede ser una imagen codificada correspondiente a una unidad de acceso. La segunda imagen puede estar incluida en la misma capa temporal que la primera imagen. Además, la primera y la segunda imagen pueden estar incluidas en una capa temporal máxima de datos de vídeo.

La unidad de desencapsulación 29 puede entonces obtener un valor entero N (906). Esto es suponiendo que la unidad de desencapsulación 29 hubiera obtenido previamente datos, tales como el valor de un indicador, que indican que el valor entero N puede estar incluido en un conjunto de parámetros de VUI, que se pueden incluir en un SPS. La unidad de desencapsulación 29 determina un valor de pulso de reloj (908). La unidad de desencapsulación 29 puede determinar el valor de pulso de reloj en base a los elementos sintácticos **escala_temporal** y **núm_unidades_en_pulso** según la ecuación (1) descrita anteriormente.

La unidad de desencapsulación 29 puede determinar entonces un delta entre un tiempo de presentación de la primera imagen y un tiempo de presentación de la segunda imagen (910). El delta puede ser igual a un número entero del valor del pulso de reloj, en base al valor entero N. Por ejemplo, el delta puede ser igual al $(N + 1) * \text{pulso de reloj}$.

La unidad de desencapsulación 29 y el decodificador de vídeo 30 pueden a continuación presentar la primera imagen y la segunda imagen según el delta determinado (912). En un ejemplo, la unidad de desencapsulación 29 puede indicar el valor de delta al decodificador de vídeo 30 y el decodificador de vídeo 30 puede llevar a cabo un proceso de decodificación basándose en el valor de delta. De esta manera, el dispositivo de destino 14 representa un ejemplo de un dispositivo que incluye un procesador configurado para determinar un valor de diferencia entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen, en donde el valor de la diferencia es igual a un valor entero multiplicado por un valor de pulso de reloj, y presentar la primera imagen y la segunda imagen según el valor de la diferencia determinada.

Del mismo modo, el procedimiento de la FIG. 9 representa un ejemplo de un procedimiento que incluye la determinación de un valor de diferencia entre un tiempo de presentación de una primera imagen y un tiempo de presentación de una segunda imagen, en donde el valor de la diferencia es igual a un valor entero multiplicado por un valor de pulso de reloj, y la presentación de la primera imagen y la segunda imagen según el valor de la diferencia determinada.

La FIG. 10 es un diagrama de bloques que ilustra un ejemplo de decodificador de vídeo 30 que puede poner en práctica técnicas para (1) la recepción de datos que incluyen los tipos de unidad de NAL, (2) el procesamiento del comportamiento al nivel de sub-imagen recibida o del comportamiento del HRD al nivel de unidad de decodificación, (3) procesar datos que incluyen la referencia a los Identificadores de conjuntos de parámetros, (4) procesar los datos recibidos, que incluyen semántica mejorada para el **indicador_velocidad_imágenes_fija** o todas y cada una de las combinaciones de éstos. En el ejemplo de la FIG. 10, el decodificador de vídeo 30 incluye una unidad de decodificación por entropía 70, una unidad de compensación de movimiento 72, una unidad de intra-predicción 74, una unidad de cuantización inversa 76, una unidad de transformación inversa 78, una memoria de imágenes de referencia 82 y un sumador 80. En algunos ejemplos, el decodificador de vídeo 30 puede llevar a cabo una pasada de decodificación generalmente recíproca a la pasada de codificación descrita con respecto al codificador de vídeo 20 (FIG. 2). La unidad de compensación de movimiento 72 puede generar datos de predicción basados en vectores de movimiento recibidos desde la unidad de decodificación por entropía 70, mientras que la unidad de intra-predicción 74 puede generar datos de predicción basados en indicadores de modalidad de intra-predicción recibidos desde la unidad de decodificación por entropía 70.

Durante el proceso de decodificación, el decodificador de vídeo 30 recibe un flujo de bits de vídeo codificado que representa bloques de vídeo de un fragmento de vídeo codificado y elementos sintácticos asociados, desde el codificador de vídeo 20. La unidad de decodificación por entropía 70 del decodificador de vídeo 30 decodifica por entropía el flujo de bits para generar coeficientes cuantizados, vectores de movimiento o indicadores de modalidad de intra-predicción, y otros elementos sintácticos. La unidad de decodificación por entropía 70 remite los vectores de movimiento y otros elementos sintácticos a la unidad de compensación de movimiento 72. El decodificador de vídeo

30 puede recibir los elementos sintácticos al nivel del fragmento de vídeo y/o al nivel del bloque de vídeo.

5 Cuando el fragmento de vídeo se codifica como un fragmento intra-codificado (I), la unidad de intra-predicción 74 puede generar datos de predicción para un bloque de vídeo del fragmento de vídeo actual, en base a una modalidad de intra-predicción señalizada, y datos de bloques previamente decodificados de la trama o imagen actual. Cuando la trama de vídeo se codifica como un fragmento inter-codificado (es decir, B, P o GPB), la unidad de compensación de movimiento 72 genera bloques predictivos para un bloque de vídeo del fragmento de vídeo actual, en base a los vectores de movimiento y a otros elementos sintácticos recibidos desde la unidad de decodificación por entropía 70. Los bloques predictivos pueden ser generados a partir de una de las imágenes de referencia dentro de una de las listas de imágenes de referencia. El decodificador de vídeo 30 puede construir las listas de tramas de referencia, la Lista 0 y la Lista 1, usando técnicas de construcción por omisión, en base a las imágenes de referencia almacenadas en la memoria de imágenes de referencia 82. La unidad de compensación de movimiento 72 determina la información de predicción para un bloque de vídeo del fragmento de vídeo actual, analizando sintácticamente los vectores de movimiento y otros elementos sintácticos, y usa la información de predicción para generar los bloques predictivos para el bloque de vídeo actual que está siendo decodificado. Por ejemplo, la unidad de compensación de movimiento 72 usa algunos de los elementos sintácticos recibidos para determinar una modalidad de predicción (por ejemplo, intra-predicción o inter-predicción), usada para codificar los bloques de vídeo del fragmento de vídeo, un tipo de fragmento de inter-predicción (por ejemplo, fragmento B, fragmento P o fragmento GPB), información de construcción para una o más de las listas de imágenes de referencia del fragmento, vectores de movimiento para cada bloque de vídeo inter-codificado del fragmento, el estado de inter-predicción para cada bloque de vídeo inter-codificado del fragmento y otra información, para decodificar los bloques de vídeo en el fragmento de vídeo actual.

25 La unidad de compensación de movimiento 72 también puede realizar la interpolación en base a filtros de interpolación. La unidad de compensación de movimiento 72 puede usar filtros de interpolación como los usados por el codificador de vídeo 20 durante la codificación de los bloques de vídeo, para calcular valores interpolados para fracciones de píxeles de bloques de referencia. En este caso, la unidad de compensación de movimiento 72 puede determinar los filtros de interpolación usados por el codificador de vídeo 20 a partir de los elementos sintácticos recibidos y usar los filtros de interpolación para generar bloques predictivos.

30 La unidad de cuantización inversa 76 cuantiza de manera inversa, es decir, descuantiza, los coeficientes de transformación cuantizados, proporcionados en el flujo de bits y decodificados por la unidad de decodificación por entropía 70. El proceso de cuantización inversa puede incluir el uso de un parámetro de cuantización QP_Y calculado por el codificador de vídeo 30 para cada bloque de vídeo en el fragmento de vídeo, para determinar un grado de cuantización y, asimismo, un grado de cuantización inversa que debería aplicarse.

La unidad de transformación inversa 78 aplica una transformación inversa, por ejemplo, una DCT inversa, una transformación inversa entera o un proceso de transformación inversa conceptualmente similar, a los coeficientes de transformación con el fin de generar bloques residuales en el dominio de píxeles.

40 Después de que la unidad de compensación de movimiento 72 genera el bloque predictivo para el bloque de vídeo actual, en base a los vectores de movimiento y a otros elementos sintácticos, el decodificador de vídeo 30 forma un bloque de vídeo decodificado sumando los bloques residuales procedentes de la unidad de transformación inversa 78 a los correspondientes bloques predictivos generados por la unidad de compensación de movimiento 72. El sumador 80 representa el componente o los componentes que llevan a cabo esta operación de suma. Si se desea, también puede aplicarse un filtro de desbloqueo para filtrar los bloques decodificados, con el fin de eliminar distorsiones de efecto pixelado. Otros filtros de bucle (ya sea en el bucle de codificación o después del bucle de codificación) también pueden ser usados para allanar las transiciones de píxeles, o mejorar de otro modo la calidad del vídeo. Los bloques de vídeo decodificados en una trama o imagen dada son a continuación almacenados en la memoria de imágenes de referencia 82, que almacena imágenes de referencia usadas para la posterior compensación de movimiento. La memoria de imágenes de referencia 82 almacena también vídeo decodificado para su presentación posterior en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la FIG. 3.

55 Ha de reconocerse que, según el ejemplo, ciertos actos o sucesos de cualquiera de las técnicas descritas en el presente documento pueden realizarse en una secuencia distinta, pueden añadirse, fundirse u omitirse por completo (por ejemplo, no todos los actos o sucesos descritos son necesarios para la puesta en práctica de las técnicas). Además, en ciertos ejemplos, los actos o sucesos pueden realizarse simultáneamente, por ejemplo, mediante el procesamiento de múltiples hebras, el procesamiento de interrupciones o múltiples procesadores, en lugar de secuencialmente.

60 En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o cualquier combinación de los mismos. Si se implementan en software, las funciones, como una o más instrucciones o código, pueden almacenarse en, o transmitirse por, un medio legible por ordenador, y ejecutarse mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, lo que corresponde a un medio tangible tal como medios de almacenamiento de datos o medios de comunicación que incluyen cualquier medio que facilite la transferencia de un programa informático desde

un lugar a otro, por ejemplo, según un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder, generalmente, a (1) medios de almacenamiento tangibles y legibles por ordenador, que sean no transitorios, o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser medios disponibles cualesquiera, a los que se pueda acceder desde uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

A modo de ejemplo, y no de manera limitativa, tales medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda usarse para almacenar código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. Además, cualquier conexión puede denominarse adecuadamente un medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde una sede de la Red, un servidor u otro origen remoto, usando un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, la DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas se incluyen en la definición de medio. Sin embargo, debería entenderse que los medios de almacenamiento legibles por ordenador y los medios de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales u otros medios transitorios, sino que, en cambio, se orientan a medios de almacenamiento tangibles no transitorios. Los discos, como se usan en el presente documento, incluyen el disco compacto (CD), el disco de láser, el disco óptico, el disco versátil digital (DVD), el disco flexible y el disco Blu-ray, donde algunos discos normalmente reproducen datos de manera magnética, mientras que otros discos reproducen los datos de manera óptica con láser. Las combinaciones de lo anterior también deberían incluirse dentro del alcance de los medios legibles por ordenador.

Las instrucciones pueden ser ejecutadas por uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados específicos de la aplicación (ASIC), formaciones lógicas programables en el terreno (FPGA) u otros circuitos lógicos integrados o discretos equivalentes. Por consiguiente, el término "procesador", como se usa en el presente documento, puede referirse a cualquier estructura anterior o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento puede proporcionarse dentro de hardware especializado y/o módulos de software configurados para la codificación y la decodificación, o incorporarse en un códec combinado. Además, las técnicas podrían implementarse completamente en uno o más circuitos o elementos lógicos.

Las técnicas de esta divulgación se pueden implementar en una gran variedad de dispositivos o aparatos, incluyendo un equipo de mano inalámbrico, un circuito integrado (IC) o un conjunto de IC (por ejemplo, un conjunto de chips). Varios componentes, módulos o unidades se describen en esta divulgación para enfatizar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no requieren necesariamente la realización mediante diferentes unidades de hardware. Más bien, como se ha descrito anteriormente, pueden combinarse diversas unidades en una unidad de hardware de códec, o ser proporcionadas por una colección de unidades de hardware inter-operativas, incluyendo uno o más procesadores, como se ha descrito anteriormente, junto con el software y/o firmware adecuado.

Se han descrito diversos ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

Reivindicaciones

1. Un procedimiento de presentación de datos de vídeo, comprendiendo el procedimiento:
 - 5 basándose en una determinación de que una capa temporal, incluyendo una primera imagen y una segunda imagen, tiene una velocidad de imágenes constante, decodificar la información de temporización para los datos de vídeo, incluyendo la información de temporización un valor entero para los datos de vídeo;
 - 10 determinar un valor de diferencia entre un tiempo de presentación para la primera imagen y un tiempo de presentación de la segunda imagen, de forma que el valor de diferencia sea igual al valor entero multiplicado por el valor de pulso en el que la primera imagen y la segunda imagen sean consecutivas en orden de salida; y
 - 15 presentar la primera imagen en un primer momento y presentar la segunda imagen en un segundo momento, de forma que una diferencia entre el segundo momento y el primer momento sea el valor de diferencia determinado.
2. El procedimiento de la reivindicación 1, que comprende además visualizar, mediante un dispositivo de visualización, los datos de vídeo presentados.
3. El procedimiento de la reivindicación 1, que comprende además determinar que la capa temporal que incluye la primera imagen y la segunda imagen tiene la velocidad de imágenes constante, en el que la información de temporización incluye la definición de datos del valor entero, en el que la decodificación de la información de temporización para los datos de vídeo incluye determinar el valor entero, y en el que la determinación del valor entero comprende, basándose en la determinación de que la capa temporal tiene la velocidad de imágenes constante, decodificar los datos que definen el valor entero.
4. El procedimiento de la reivindicación 3, en el que la determinación de que la capa temporal tiene la velocidad de imágenes constante comprende la determinación de que un `fixed_pic_rate_flag` tiene un valor que indica que la capa temporal tiene la velocidad de imágenes constante.
5. El procedimiento de la reivindicación 3, que comprende además:
 - 35 determinar, para cada capa temporal que tenga una velocidad de imágenes constante respectiva, un valor entero indicado respectivo; y
 - 40 presentar imágenes de cada capa temporal que tengan una velocidad de imágenes constante respectiva de acuerdo con los valores enteros respectivos multiplicados por el valor de pulso de reloj.
6. El procedimiento de la reivindicación 1, en el que la decodificación de la información de temporización para los datos de vídeo incluye la determinación del valor de pulso de reloj, en el que la determinación del valor de pulso de reloj incluye la determinación del valor de pulso de reloj de forma que el valor de pulso de reloj se base, al menos en parte, en un valor de escala temporal, en el que el valor de escala temporal se base en la frecuencia de reloj.
7. El procedimiento de la reivindicación 1, en el que la capa temporal incluye una capa temporal más alta, comprendiendo además el procedimiento:
 - 50 determinar un segundo valor de diferencia entre un tiempo de presentación de una tercera imagen y un tiempo de presentación de una cuarta imagen en una capa temporal inferior teniendo un valor de identificación de capa temporal que sea inferior al valor de identificación temporal que está asociado con la capa temporal más alta, un segundo valor entero asociado con la capa temporal inferior, y el valor de pulso de reloj; y
 - 55 presentar imágenes de la capa temporal inferior basándose en el segundo valor de diferencia.
8. El procedimiento de la reivindicación 1, en el que el tiempo de presentación para la primera imagen comprende un primer valor de recuento de orden de imágenes, POC, y en el que el tiempo de presentación para la segunda imagen comprende un segundo valor de POC.
9. Un dispositivo para presentar datos de vídeo, comprendiendo el dispositivo:
 - 65 medios para decodificar información de temporización para los datos de vídeo en los que la información de temporización se determina basándose en una determinación de que una capa temporal, incluyendo una primera imagen y una segunda imagen, tiene una velocidad de imágenes constante, incluyendo la

información de temporización un valor entero para los datos de vídeo;

5 medios para determinar un valor de diferencia entre un tiempo de presentación de la primera imagen y un tiempo de presentación de la segunda imagen de forma que el valor de diferencia sea igual a un valor entero multiplicado por un valor de pulso de reloj en el que la primera imagen y la segunda imagen sean consecutivas en orden de salida; y

10 medios para presentar la primera imagen en un primer momento y presentar la segunda imagen en un segundo momento, de forma que una diferencia entre el segundo momento y el primer momento sea el valor de diferencia determinado.

10. El dispositivo de la reivindicación 9, que comprende además medios para determinar que la capa temporal que incluye la primera imagen y la segunda imagen tiene la velocidad de imágenes constante, en el que la información de temporización incluye datos que definen el valor de entero, en el que los medios para decodificar la información de temporización para los datos de vídeo incluye medios para determinar el valor entero, y en el que los medios para determinar el valor entero comprenden medios para decodificar, basándose en la determinación de que la capa temporal tenga la velocidad de imágenes constante, los datos que definen el valor entero.

20 11. El dispositivo de la reivindicación 10, en el que los medios para determinar que la capa temporal tiene la velocidad de imágenes constante comprende medios para determinar que un `fixed_pic_rate_flag` tiene un valor que indica que la capa temporal tiene la velocidad de imágenes constante.

25 12. El dispositivo de la reivindicación 10, que comprende además:
medios para determinar, para cada capa temporal que tiene una velocidad de imágenes constante respectiva, un valor entero indicado respectivo; y

30 medios para presentar imágenes de cada capa temporal que tienen una velocidad de imágenes constante respectiva de acuerdo con los valores enteros respectivos multiplicados por el valor de pulso de reloj.

13. El dispositivo de la reivindicación 9, en el que la capa temporal incluye una capa temporal más alta, comprendiendo además el dispositivo:

35 medios para determinar un segundo valor de diferencia entre un tiempo de presentación de una tercera imagen y un tiempo de presentación de una cuarta imagen en una capa temporal inferior que tiene un valor de identificación de capa temporal que es inferior que un valor de identificación temporal que está asociado con la capa temporal más alta, un segundo valor entero asociado con la capa temporal inferior, y el valor de pulso de reloj; y

40 medios para presentar imágenes de la capa temporal inferior basándose en el segundo valor de diferencia.

45 14. El dispositivo de la reivindicación 9, en el que el tiempo de presentación para la primera imagen comprende un primer valor de recuento de orden de imágenes, POC, y en el que el tiempo de presentación para la segunda imagen comprende un segundo valor de POC.

50 15. Un medio de almacenamiento legible por ordenador que tiene instrucciones almacenadas en el mismo que, cuando se ejecutan, hacen que un procesador:

decodifique, basándose en una determinación de que una capa temporal, incluyendo una primera imagen y una segunda imagen, tiene una velocidad de imágenes constante, información de temporización para los datos de vídeo, incluyendo la información de temporización un valor entero para los datos de vídeo;

55 determine un valor de diferencia entre un tiempo de presentación de la primera imagen y un tiempo de presentación de la segunda imagen, de forma que el valor de diferencia sea igual al valor entero multiplicado por el valor de pulso de reloj en el que la primera imagen y la segunda imagen son consecutivas en orden de salida; y presente la primera imagen en un primer momento y presente la segunda imagen en un segundo momento, de forma que la diferencia entre el segundo momento y el primer momento sea el valor de diferencia determinado.

60

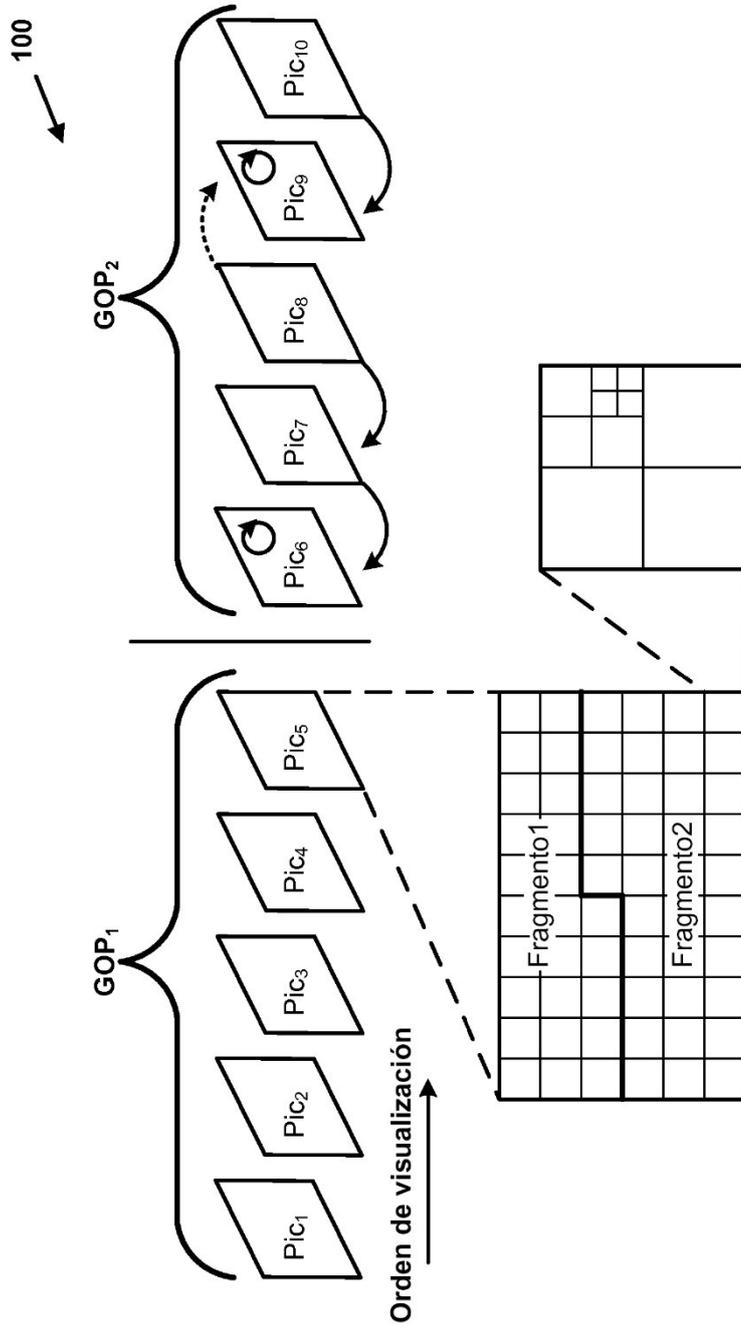


FIG. 1

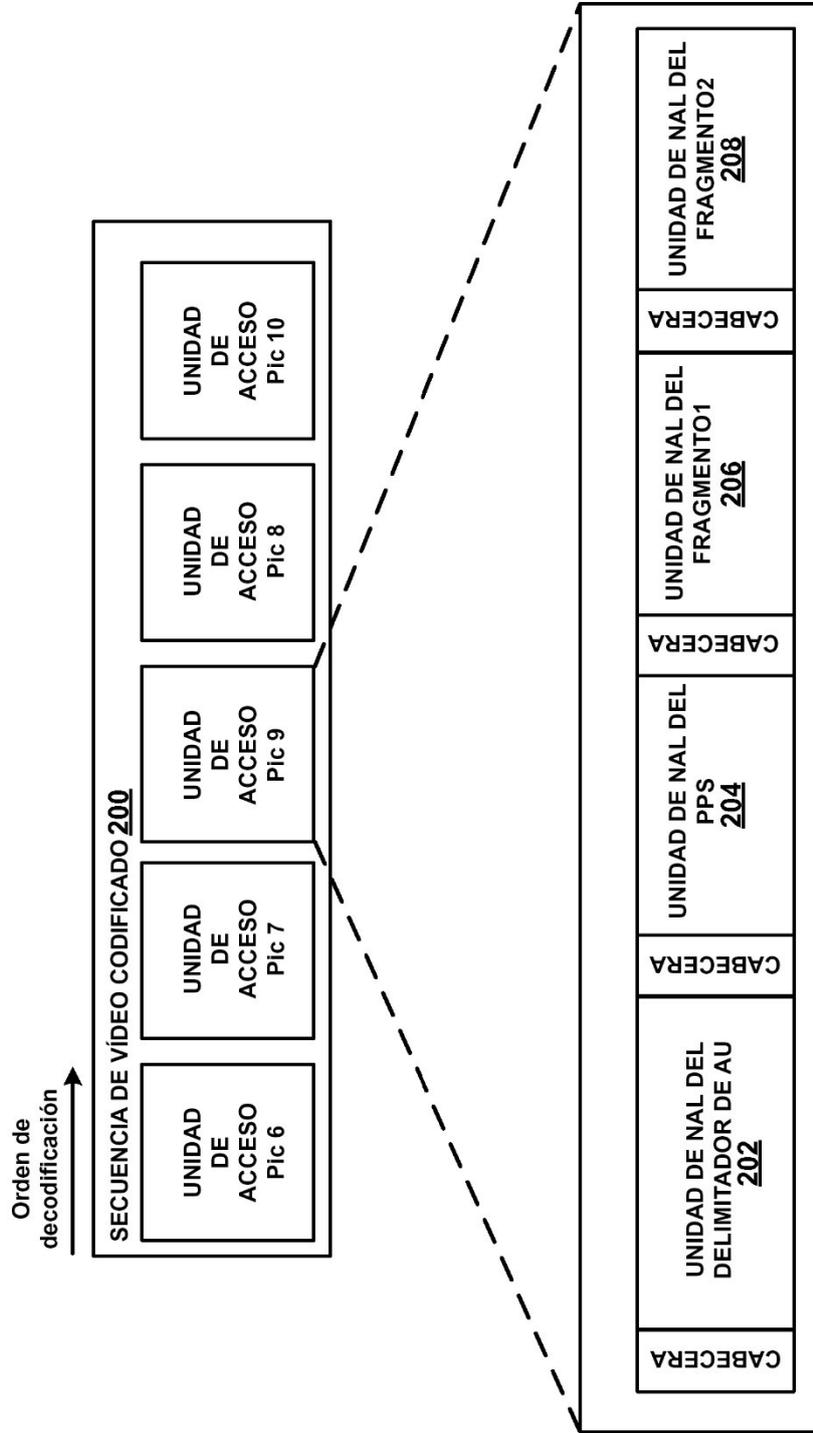


FIG. 2

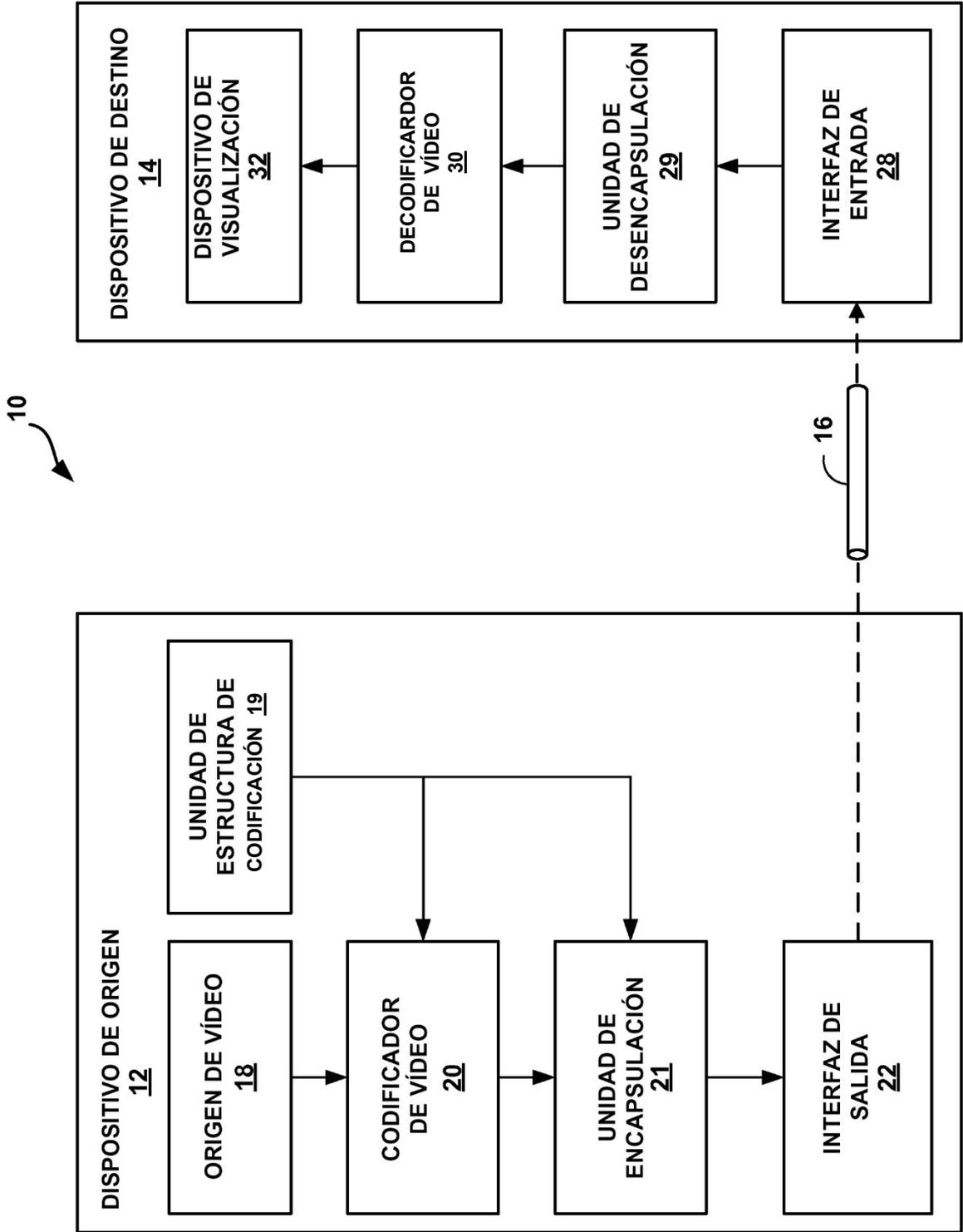


FIG. 3

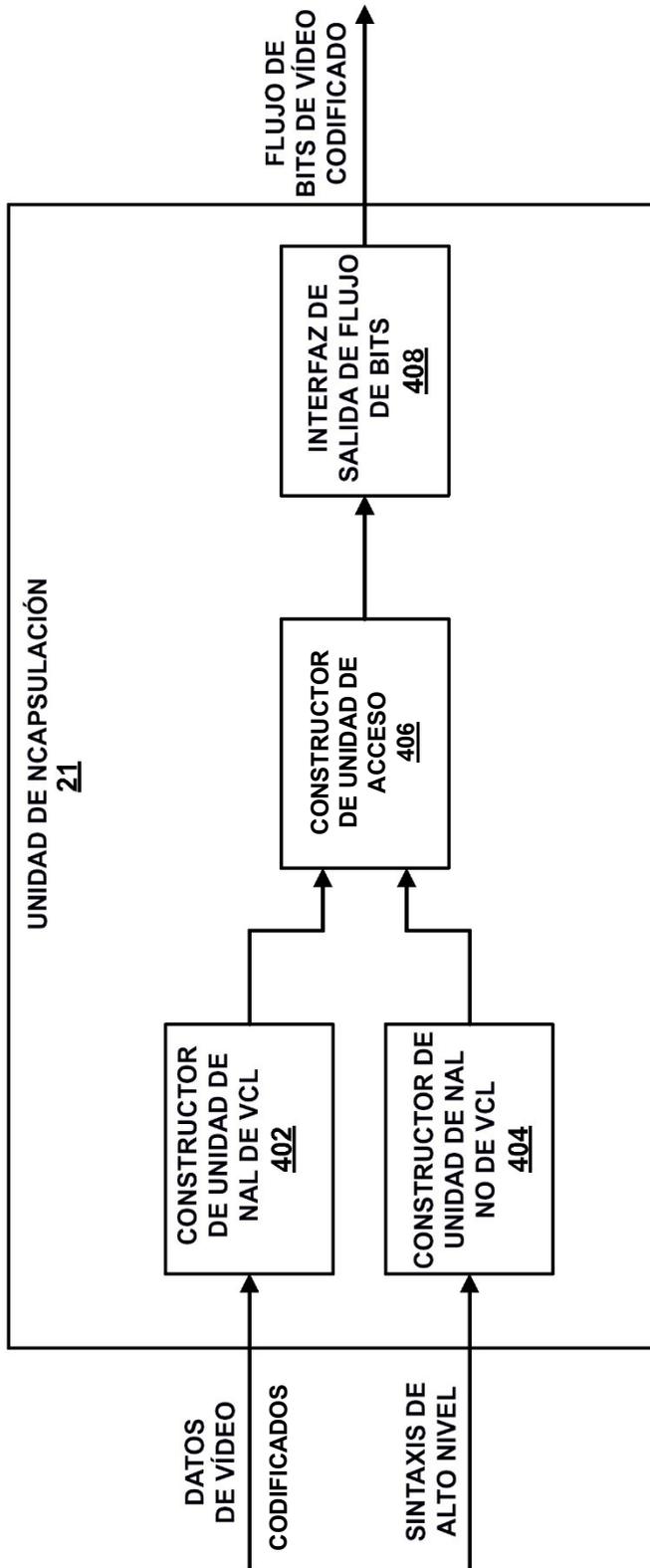


FIG. 4

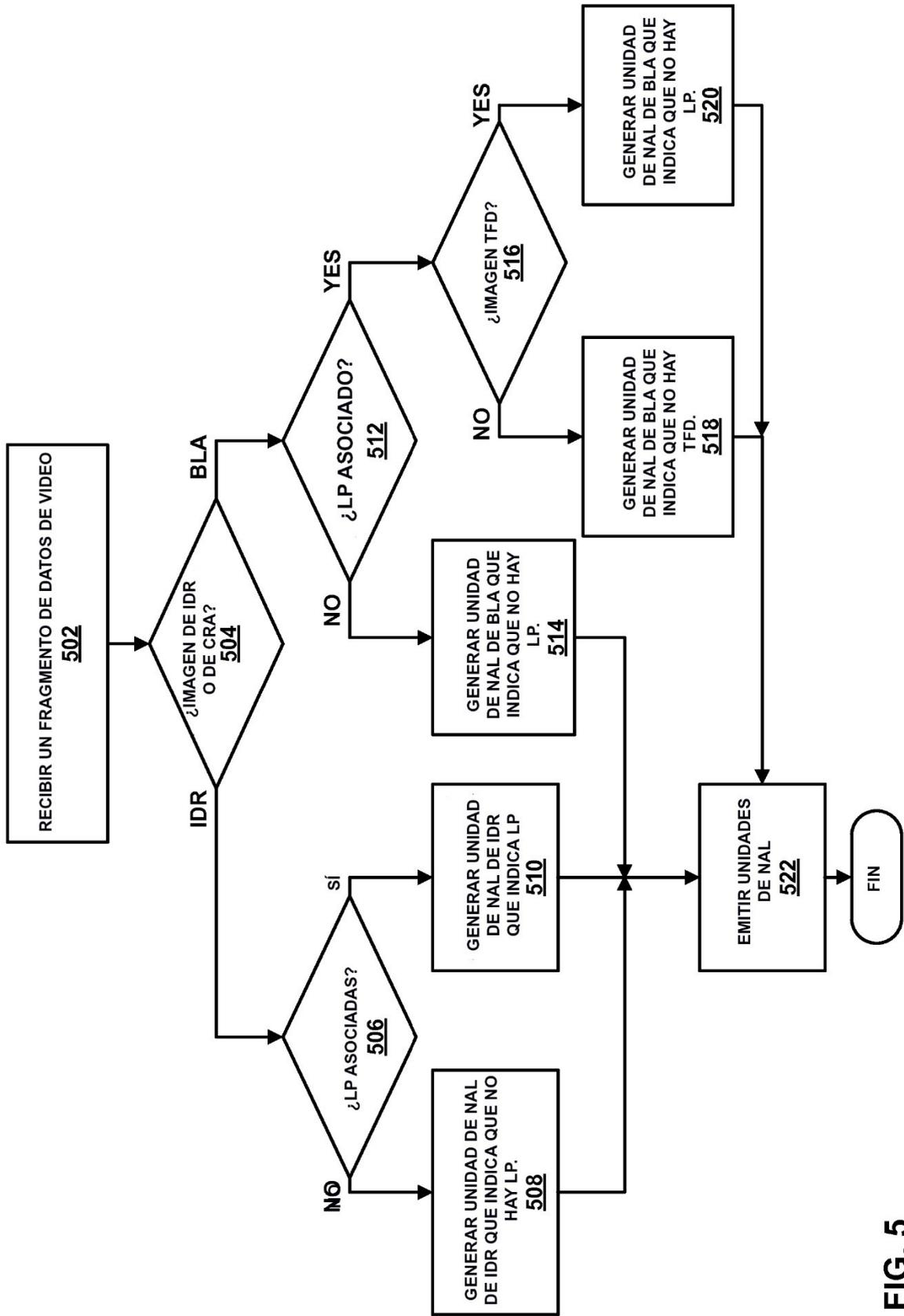


FIG. 5

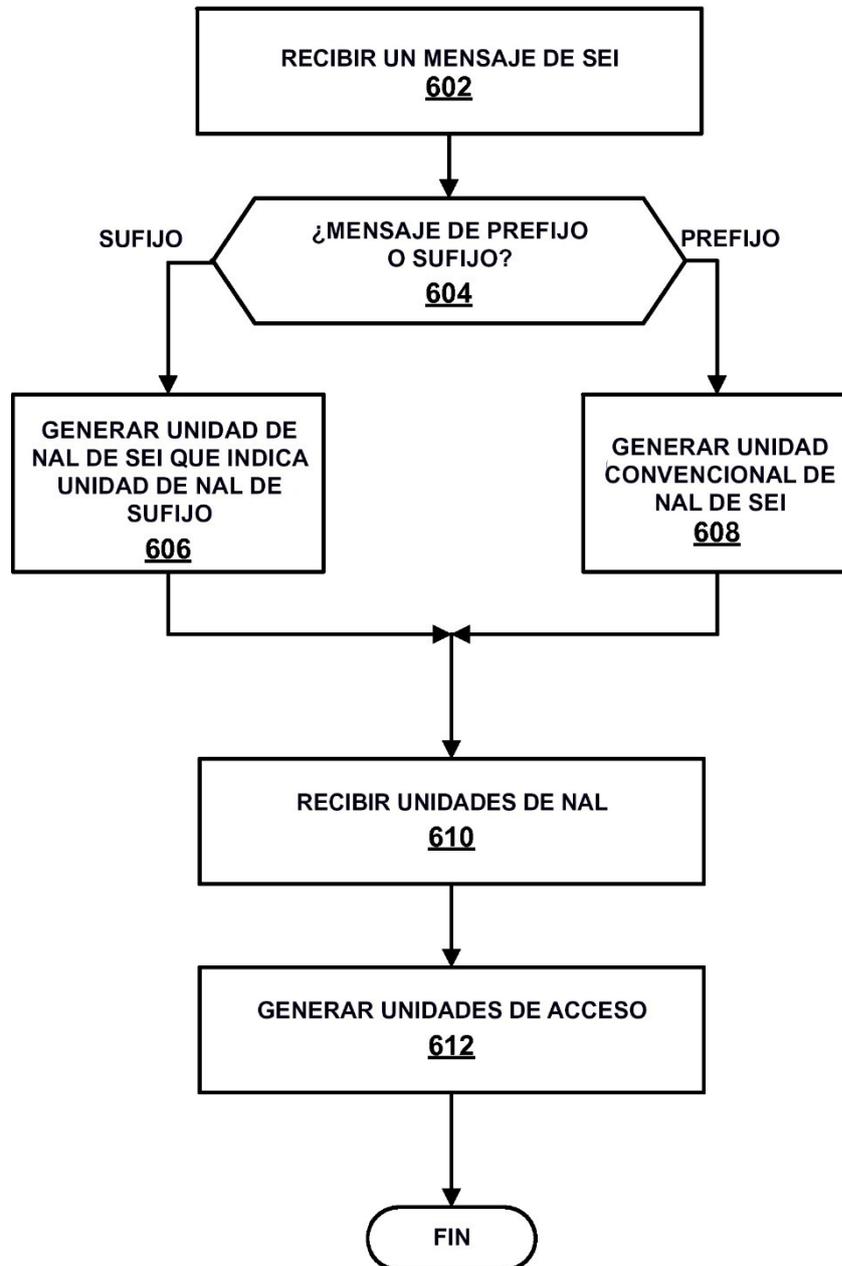


FIG. 6

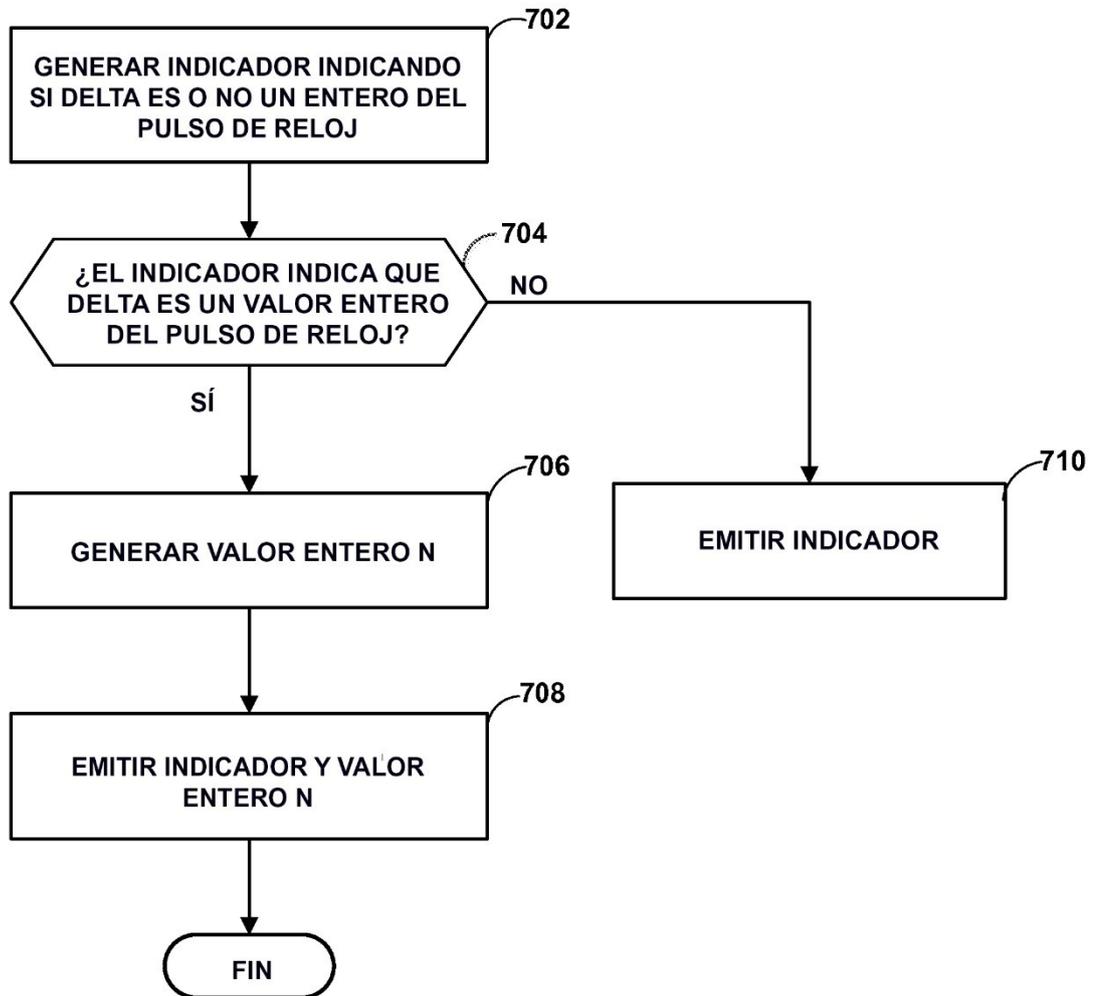


FIG. 7

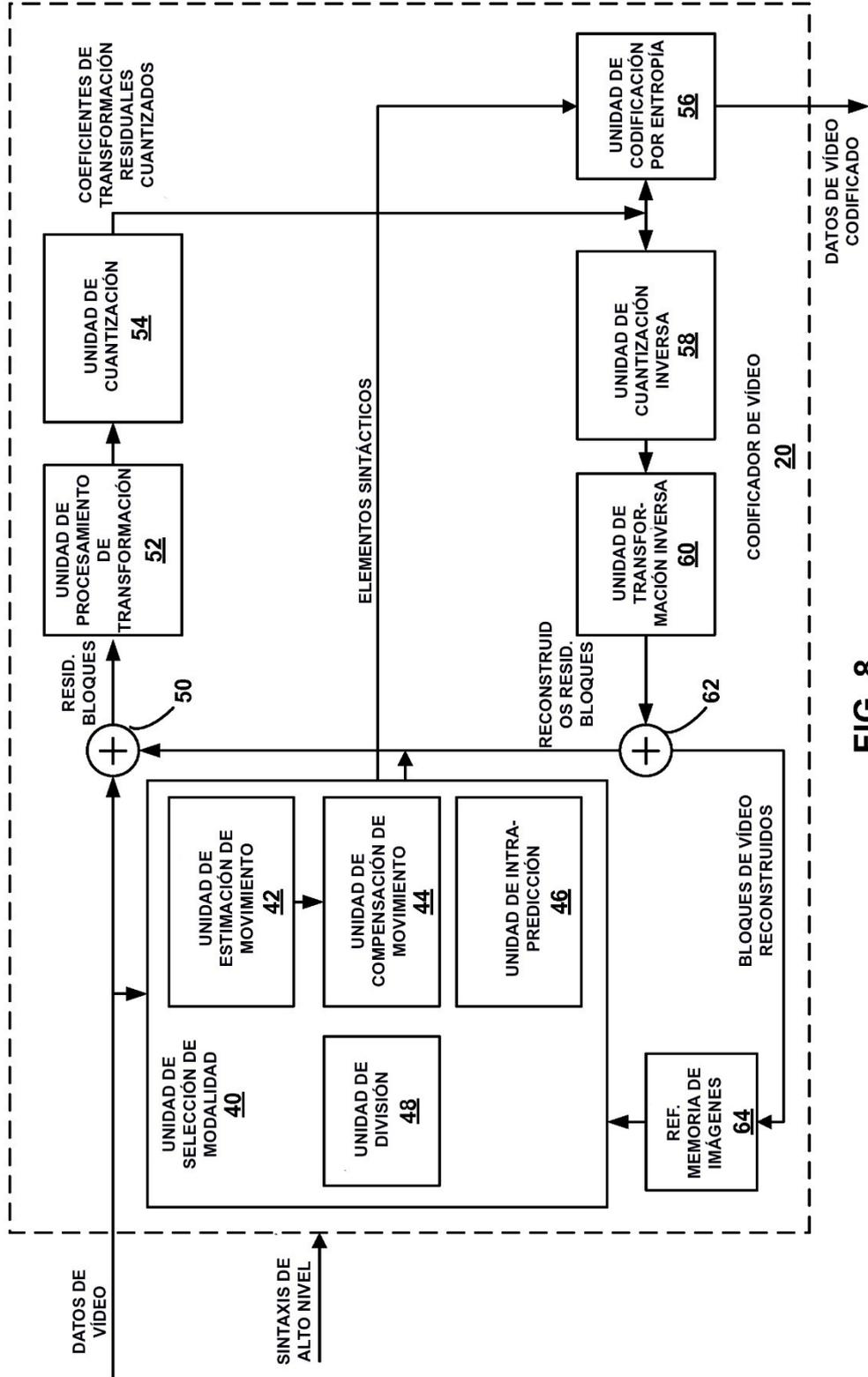


FIG. 8

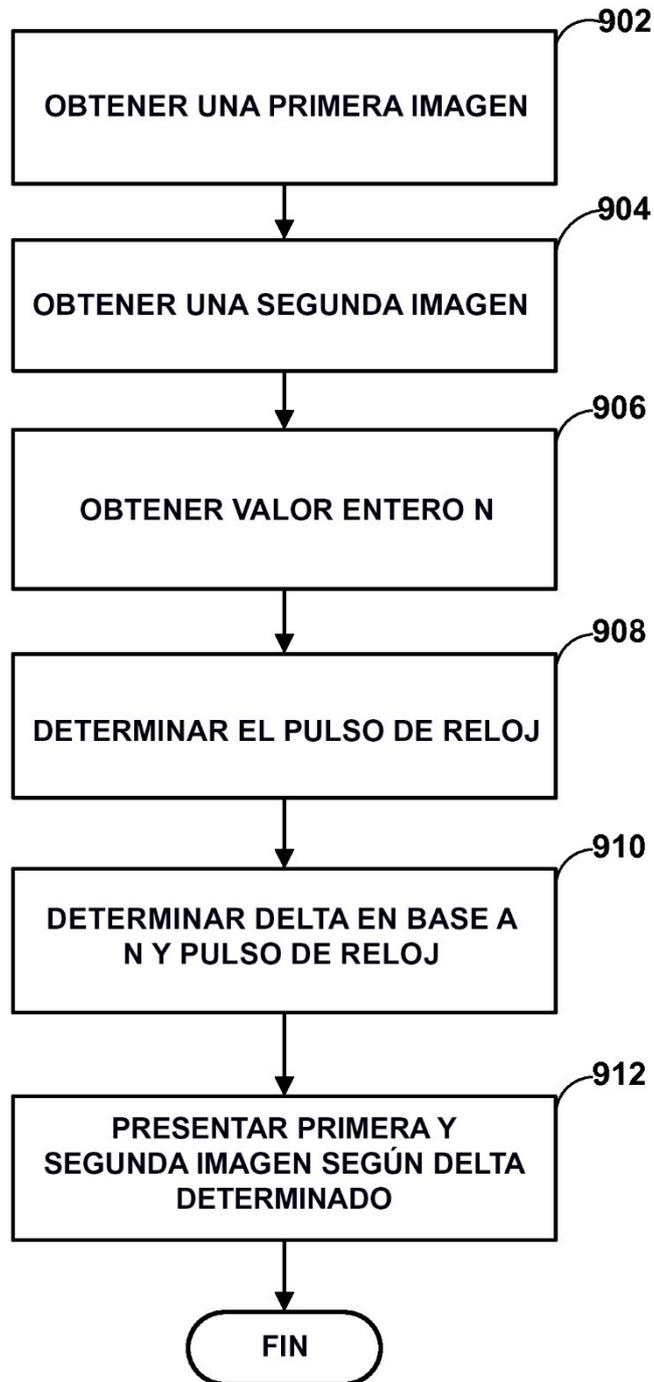


FIG. 9

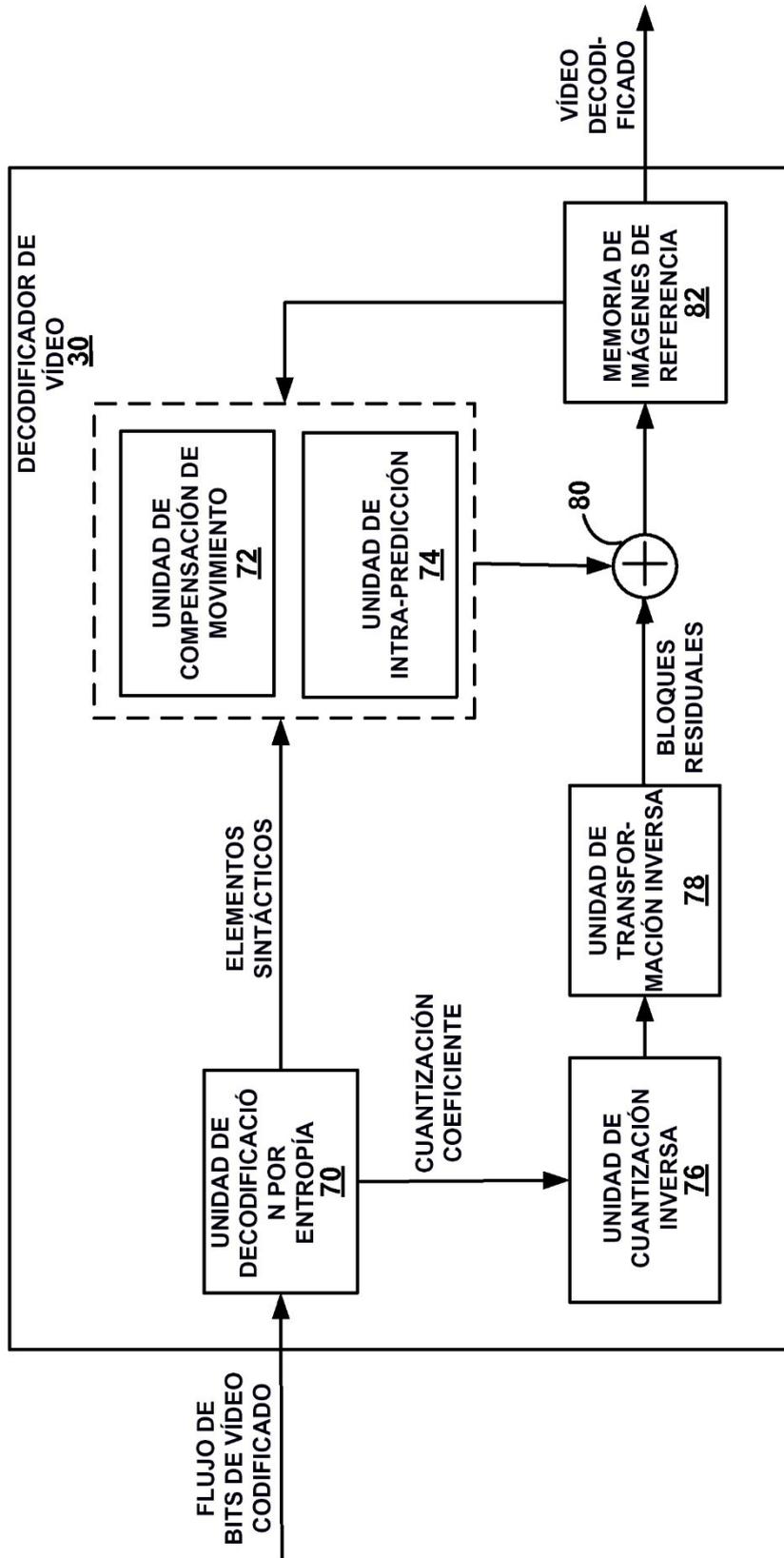


FIG. 10