

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 707 892**

51 Int. Cl.:

H04N 19/107 (2014.01)

H04N 19/70 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **26.11.2013 PCT/US2013/072086**

87 Fecha y número de publicación internacional: **10.07.2014 WO14107250**

96 Fecha de presentación y número de la solicitud europea: **26.11.2013 E 13803402 (0)**

97 Fecha y número de publicación de la concesión europea: **24.10.2018 EP 2941869**

54 Título: **Operaciones de almacenamiento en memoria intermedia de vídeo para acceso aleatorio en la codificación de vídeo**

30 Prioridad:

07.01.2013 US 201361749820 P
25.11.2013 US 201314089495

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
05.04.2019

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)
ATTN: International IP Administration, 5775
Morehouse Drive
San Diego, California 92121-1714, US

72 Inventor/es:

WANG, YE-KUI;
RAMASUBRAMONIAN, ADARSH KRISHNAN y
CHEN, YING

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 707 892 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

Operaciones de almacenamiento en memoria intermedia de vídeo para acceso aleatorio en la codificación de vídeo

5 **CAMPO TÉCNICO**

[0001] Esta divulgación se refiere a la codificación de vídeo, y más particularmente a técnicas para controlar las operaciones de almacenamiento en memoria intermedia de vídeo de un decodificador de vídeo.

10 **ANTECEDENTES**

[0002] Las capacidades de 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 tablet, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos móviles o de radio por satélite, los denominados "teléfonos inteligentes", dispositivos de videoconferencia, dispositivos de transmisión de vídeo y similares. Los dispositivos de vídeo digitales implementan técnicas de compresión de vídeo, tales como las descritas en las normas definidas por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, parte 10, codificación de vídeo avanzada (AVC), la norma de codificación de vídeo de alta eficiencia (HEVC) actualmente en desarrollo y las extensiones de dichas normas. Los dispositivos de vídeo pueden transmitir, recibir, codificar, decodificar y/o almacenar información de vídeo digital con más eficiencia implementando dichas técnicas de compresión de vídeo.

[0003] Las técnicas de compresión de vídeo realizan la predicción espacial (intraimagen) y/o la predicción temporal (interimagen) para reducir o eliminar la redundancia inherente en las secuencias de vídeo. Para la codificación de vídeo basada en bloques, un fragmento de vídeo (por ejemplo, una trama de vídeo o una porción de una trama de vídeo) puede particionarse en bloques de vídeo, que también pueden llamarse bloques de árbol, unidades de codificación (CU) y/o nodos de codificación. Los bloques de vídeo en un fragmento intracodificado (I) de una imagen se codifican usando predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen. Los bloques de vídeo en un fragmento intercodificado (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 llamarse tramas y las imágenes de referencia pueden llamarse tramas de referencia.

[0004] La predicción espacial o temporal utiliza un bloque predictivo. Los datos residuales representan diferencias de píxeles entre el bloque original que se va a codificar y el bloque predictivo. Un bloque intercodificado se codifica de acuerdo con 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 intracodificado se codifica de acuerdo con un modo de intracodificación y con los datos residuales. Para una mayor compresión, los datos residuales pueden transformarse desde el dominio del píxel en un dominio de transformada, dando como resultado unos coeficientes de transformada residuales, que a continuación se pueden cuantizar. Los coeficientes de transformada cuantizados, dispuestos inicialmente en una matriz bidimensional, pueden escanearse con el fin de producir un vector unidimensional de coeficientes de transformada, y puede aplicarse codificación por entropía para lograr aún más compresión.

[0005] El documento CHEN Y ET AL: "AHG15: Conforming bitstreams starting with CRA pictures (Conformación de flujos de bits a partir de imágenes CRA)", 7. REUNIÓN DE JCT-VC; 98. REUNIÓN JCT-VC; 21-11-2011 - 30-11-2011; GINEBRA; (EQUIPO COLABORATIVO CONJUNTO SOBRE LA CODIFICACIÓN DE VIDEO DE ISO/CEI JTC1/SC29/WG11 Y UIT-T SG.16); URL: [HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/](http://wftp3.itu.int/av-arch/jctvc-site/), no. JCTVC-G319, 9 de noviembre de 2011 (2011-11-09), divulga un procedimiento para procesar flujos de bits a partir de imágenes CRA.

SUMARIO

[0006] En general, esta divulgación describe técnicas para controlar las operaciones de almacenamiento en memoria intermedia en un decodificador de vídeo para imágenes de puntos de acceso aleatorio (RAP) que tengan imágenes principales descartadas. En algunos ejemplos, cuando un flujo de bits incluye una imagen RAP subsiguiente que sigue a una imagen RAP inicial, se puede configurar un decodificador para conmutar un tiempo de eliminación de la memoria intermedia de imágenes codificadas (CPB) para las unidades de acceso que siguen a la imagen RAP subsiguiente cuando se muestran imágenes, tal como las imágenes principales omitidas de acceso aleatorio (RASL), para la imagen RAP subsiguiente no están presentes.

[0007] De acuerdo con un aspecto de la invención, se proporciona un procedimiento para procesar datos de vídeo, como se define en la reivindicación 1. De acuerdo con otro aspecto de la invención, se proporciona un dispositivo para procesar datos de vídeo, como se define en la reivindicación 5. De acuerdo con otro aspecto de la presente invención, se proporciona un medio de almacenamiento no transitorio legible por ordenador que almacena instrucciones que, al

ejecutarse por uno o más procesadores, causan que uno o más procesadores realicen el procedimiento como se define en la reivindicación 1.

5 **[0008]** En algunos ejemplos, la descripción describe diversos procedimientos. Se puede configurar una amplia variedad de procesadores, unidades de procesamiento y aparatos para implementar los procedimientos de ejemplo. La divulgación también describe medios de almacenamiento legibles por ordenador que pueden configurarse para realizar las funciones de uno cualquiera o más de los procedimientos de ejemplo recibidos, y la reinicialización de un tiempo de eliminación de la memoria intermedia de imágenes y un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes en respuesta a la recepción de la unidad de acceso y a la no recepción de las unidades de acceso asociadas que contienen imágenes RASL.

15 **[0009]** En otro ejemplo, la divulgación describe un dispositivo para procesar datos de vídeo que incluye un procesador configurado para recibir en un flujo de bits de vídeo una unidad de acceso que tenga una imagen RAP, recibir en el flujo de bits de vídeo, después de la unidad de acceso en el flujo de bits, una unidad de acceso subsiguiente que tenga una segunda imagen RAP y, en el caso de que una o más imágenes principales omitidas de acceso aleatorio (RASL) para la unidad de acceso subsiguiente no estén presentes en el flujo de bits recibido, conmutar el tiempo de eliminación de la memoria intermedia de imágenes antes basándose en un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes.

20 **[0010]** En otro ejemplo, la divulgación describe un dispositivo para procesar datos de vídeo que incluye un procesador configurado para recibir una unidad de acceso después de una inicialización anterior de un decodificador hipotético de referencia (HRD), teniendo la unidad de acceso una imagen RAP, en donde las unidades de acceso asociadas que contienen imágenes RASL no se reciben, y se reinicializa el HRD, incluida la reinicialización de un tiempo de eliminación de la memoria intermedia de imágenes y un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes en respuesta a la recepción de la unidad de acceso y a la no recepción de las unidades de acceso asociadas que contienen imágenes RASL.

30 **[0011]** En otro ejemplo, la divulgación describe un dispositivo para procesar datos de vídeo que incluye medios para recibir en un flujo de bits de vídeo una unidad de acceso que tenga una imagen RAP, medios para recibir en el flujo de bits de vídeo, después de la unidad de acceso en el flujo de bits, una unidad de acceso subsiguiente que tenga una segunda imagen RAP, y medios para conmutar un tiempo de eliminación de la memoria intermedia de imágenes antes basándose en un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes en el caso de una o más imágenes principales omitidas de acceso aleatorio (RASL) para la subsiguiente unidad de acceso.

35 **[0012]** En otro ejemplo, la divulgación describe un dispositivo para procesar datos de vídeo que incluye medios para recibir una unidad de acceso después de una inicialización anterior del decodificador hipotético de referencia (HRD), teniendo la unidad de acceso una imagen RAP, en donde las unidades de acceso asociadas que contienen imágenes RASL no se reciben, y medios para reinicializar el HRD, incluida la reinicialización del tiempo de eliminación de la memoria intermedia de imágenes y un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes en respuesta a la recepción de la unidad de acceso y a la no recepción de las unidades de acceso asociadas que contienen imágenes RASL.

45 **[0013]** En otro ejemplo, la divulgación describe un medio de almacenamiento legible por ordenador. El medio de almacenamiento legible por ordenador que tiene almacenadas en el mismo las instrucciones que, al ejecutarse, causan que uno o más procesadores reciban en un flujo de bits de vídeo una unidad de acceso que tenga una imagen RAP, recibe en el flujo de bits de vídeo, después de la unidad de acceso en el flujo de bits, una unidad de acceso subsiguiente que tiene una segunda imagen RAP y, en el caso de que una o más imágenes principales omitidas de acceso aleatorio (RASL) para la unidad de acceso subsiguiente no estén presentes en el flujo de bits recibido, conmuta el tiempo de eliminación de la memoria intermedia de imágenes antes basándose en el desplazamiento de retardo de eliminación de la memoria intermedia de imágenes.

55 **[0014]** En otro ejemplo, la divulgación describe un medio de almacenamiento legible por ordenador. El medio de almacenamiento legible por ordenador que tiene almacenadas en el mismo las instrucciones que, al ejecutarse, causan que uno o más procesadores reciban una unidad de acceso después de una inicialización anterior del decodificador hipotético de referencia (HRD), teniendo la unidad de acceso una imagen de punto de acceso aleatorio (RAP), en donde las unidades de acceso asociadas que contienen imágenes iniciales omitidas de acceso aleatorio (RASL) no se reciben y reinicializan el HRD, incluida la reinicialización de un tiempo de eliminación de la memoria intermedia de imágenes y un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes en respuesta a la recepción de la unidad de acceso y a la no recepción de las unidades de acceso asociadas que contienen imágenes RASL.

60 **[0015]** En otro ejemplo, la divulgación describe un procedimiento de procesamiento de datos de vídeo que incluye señalar un desplazamiento de retardo de eliminación de la CPB para cada unidad de acceso CRA o BLA, conmutar el tiempo de eliminación de CBP de cada una de las unidades de acceso que siguen a la unidad de acceso CRA o BLA en orden de decodificación es antes por el desplazamiento de retardo de eliminación de la CPB para cada unidad

de acceso CRA o BLA para la que no están presentes las imágenes RASL asociadas, independientemente de si el HRD se inicializa en la unidad de acceso CRA o BLA.

5 [0016] En algunos ejemplos, la descripción describe diversos procedimientos. Se puede configurar una amplia variedad de procesadores, unidades de procesamiento y aparatos para implementar los procedimientos de ejemplo. La divulgación también describe medios de almacenamiento legibles por ordenador que pueden configurarse para realizar las funciones de uno o más de los procedimientos de ejemplo.

10 [0017] Los detalles de uno o más ejemplos se exponen en los dibujos adjuntos y en la siguiente descripción. Otras características, objetos y ventajas resultarán evidentes a partir de la descripción y de los dibujos, y a partir de las reivindicaciones.

15 [0018] 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 de los dibujos, y a partir de las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

20 [0019]

La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación y decodificación de vídeo de ejemplo que puede utilizar las técnicas descritas en esta divulgación.

25 La FIG. 2 es un diagrama de bloques que ilustra un codificador de vídeo de ejemplo que puede implementar las técnicas descritas en esta divulgación.

La FIG. 3 es un diagrama de bloques que ilustra un decodificador de vídeo de ejemplo que puede implementar las técnicas descritas en esta divulgación.

30 La FIG. 4 es un diagrama de flujo que ilustra un procedimiento de ejemplo de acuerdo con uno o más ejemplos descritos en esta divulgación.

35 La FIG. 5 es un diagrama de flujo que ilustra un procedimiento de ejemplo de acuerdo con uno o más ejemplos descritos en esta divulgación.

DESCRIPCIÓN DETALLADA

40 [0020] Esta divulgación describe diversas técnicas para controlar las operaciones de almacenamiento en memoria intermedia en un decodificador de vídeo para imágenes de puntos de acceso aleatorios (RAP) que han descartado imágenes principales. Las técnicas pueden soportar operaciones mejoradas del decodificador hipotético de referencia (HRD), que se pueden aplicar a través de unidades de acceso intermedio (AU) que contienen imágenes de acceso aleatorio limpio (CRA) o de acceso de enlace roto (BLA), para las cuales las unidades principales omitidas de acceso aleatorio (RASL) asociadas se descartan. Una imagen CRA contiene solo I fragmentos, y puede ser la primera imagen en el flujo de bits en orden de decodificación, o puede aparecer más adelante en el flujo de bits. Una imagen CRA puede tener imágenes RADL o RASL asociadas. Cuando una imagen CRA es la primera imagen en el flujo de bits en orden de decodificación, la imagen CRA es la primera imagen de una secuencia de vídeo codificada en orden de decodificación, y cualquier imagen RASL asociada no se emite por el decodificador y puede no ser decodificable, ya que puede contener referencias a imágenes que no estén presentes en el flujo de bits. Una imagen BLA solo contiene fragmentos I, y puede ser la primera imagen en el flujo de bits en orden de decodificación, o puede aparecer más adelante en el flujo de bits. Cada imagen BLA comienza una nueva secuencia de vídeo codificada, y tiene el mismo efecto en el proceso de decodificación que una imagen IDR. Sin embargo, una imagen BLA contiene elementos de sintaxis que especifican un conjunto de imágenes de referencia no vacío. Las imágenes RASL son imágenes principales de una imagen BLA o CRA asociada. Todas las imágenes RASL son imágenes principales de una imagen BLA o CRA asociada. Cuando la imagen RAP asociada es una imagen BLA o es la primera imagen codificada en el flujo de bits, la imagen RASL no se emite y es posible que no se pueda decodificar correctamente, ya que la imagen RASL puede contener referencias a imágenes que no estén presentes en el flujo de bits. Las imágenes RASL no se usan como imágenes de referencia para el proceso de decodificación de imágenes que no son RASL. Cuando están presentes, todas las imágenes RASL preceden, en orden de decodificación, todas las imágenes secundarias de la misma imagen RAP asociada.

60 [0021] Entre las normas de codificación de vídeo se incluyen ITU-T H.261, MPEG-1 Visual de la ISO/IEC, ITU-T H.262 o MPEG-2 Visual de la ISO/IEC, ITU-T H.263, MPEG-4 Visual de la ISO/IEC e ITU-T H.264 (también conocida como AVC del MPEG-4 de la ISO/IEC), incluyendo sus ampliaciones de codificación de vídeo ajustable a escala (SVC) y de codificación de vídeo multivistas (MVC).

65

- 5 **[0022]** Además, existe una nueva norma de codificación de vídeo, concretamente la codificación de vídeo de alta eficiencia (HEVC), desarrollada por el Equipo de Colaboración Conjunta sobre Codificación de Vídeo (JCT-VC) del Grupo de Expertos sobre Codificación de Vídeo (VCEG) de ITU-T y el Grupo de Expertos sobre Imágenes en Movimiento (MPEG) de ISO/IEC. Un borrador de trabajo (WD) de HEVC, Bross, et al., "High Efficiency Video Coding (HEVC) text specification draft 9 (Borrador 9 de memoria descriptiva de texto de la codificación de vídeo de alta eficiencia (HEVC))", y denominado WD9 de HEVC en adelante, está disponible en http://phenix.intevry.fr/jct/doc_end_user/documents/11_Shanghai/wg11/JCTVC-K1003-v13.zip, a partir del 7 de enero de 2013.
- 10 **[0023]** Un borrador reciente de la norma de HEVC, denominado "HEVC Working Draft 10 (Borrador 10 de trabajo de HEVC)" o "WD10", se describe en el documento JCTVC-H1003, de Bross et al., titulado "High Efficiency Video Coding (HEVC) Text Specification Draft 10 (for FDIS & Last Call)" ("Memoria descriptiva textual de la codificación de vídeo de alta eficiencia (HEVC), Borrador 10 (para FDIS y Última Llamada)"), Equipo de Colaboración Conjunta sobre Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 12ª conferencia: Ginebra, Suiza,
- 15 14-23 de enero de 2013, que se puede descargar en: http://phenix.intevry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v34.zip.
- [0024]** Otro borrador de la norma de HEVC, mencionado en el presente documento como "revisiones de WD10", se describe en el artículo de Bross et al., "Editors' proposed corrections to HEVC version 1 (Correcciones propuestas de los editores para la HEVC, versión 1)", Equipo de Colaboración Conjunta sobre Codificación de Vídeo (JCT-VC) de UIT-T SG16 WP3 e ISO/IEC JTC 1/SC29/WG11, 13ª Conferencia, Incheon, KR, abril de 2013, que, al 7 de junio de 2013, está disponible en: http://phenix.int-evry.fr/jct/doc_end_user /documents/13_Incheon/wg11/JCTVC-M0432-v3.zip
- 20 **[0025]** Los esfuerzos de normalización de la HEVC se basan en un modelo de un dispositivo de codificación de vídeo denominado Modelo de Prueba de HEVC (HM). El HM supone mejoras en las capacidades de los dispositivos de codificación de vídeo actuales con respecto a los dispositivos de codificación de vídeo disponibles durante el desarrollo de otras normas de codificación de vídeo anteriores, por ejemplo, la ITU-T H.264/AVC. Por ejemplo, mientras que la norma H.264 proporciona nueve modos de codificación de intrapredicción, el HM proporciona hasta
- 25 treinta y cinco modos de codificación de intrapredicción.
- [0026]** Las normas de codificación de vídeo incluyen típicamente una memoria descriptiva de un modelo de almacenamiento en memoria intermedia de vídeo. En AVC y HEVC, el modelo de almacenamiento en memoria intermedia se denomina decodificador hipotético de referencia (HRD), que incluye un modelo de almacenamiento en
- 30 memoria intermedia de tanto la memoria intermedia de imágenes codificadas (CPB) como la memoria intermedia de imágenes decodificadas (DPB). Los comportamientos de la CPB y de la DPB se especifican matemáticamente. El HRD impone directamente restricciones sobre la temporización, los tamaños de la memoria intermedia y las velocidades de bits diferentes, 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 de eliminación de la CPB inicial, el tamaño de la CPB, la velocidad de bits, el retardo de salida de DPB inicial y el tamaño de la DPB.
- 35 **[0027]** En la AVC y en la HEVC, la conformidad del flujo de bits y la conformidad del decodificador se especifican como partes de la memoria descriptiva del HRD. Aunque el HRD se denomina decodificador, la memoria descriptiva del HRD es típicamente necesaria en el lado del codificador para garantizar la conformidad del flujo de bits. Se especifican dos tipos de conformidad de flujo de bits o de HRD, concretamente, Tipo I y Tipo II. Además, se especifican dos tipos de conformidad del decodificador (conformidad del decodificador de temporización de salida y conformidad del decodificador de orden de salida).
- 40 **[0028]** El HRD se inicializa típicamente en una imagen de punto de acceso aleatorio (RAP). Tanto en HEVC como en AVC, una vez que se inicializa el HRD, continúa funcionando hasta el final de un flujo de bits sin reinicialización.
- 45 **[0029]** En el WD9 de HEVC, si el HRD se inicializa en una unidad de acceso CRA o BLA (esta unidad de acceso inicial se denomina unidad de acceso 0), se elige el retardo de eliminación de la CPB inicial predeterminado o el alternativo y el desplazamiento de retardo asociado con la unidad de acceso inicial 0 para su uso en las operaciones de HRD, dependiendo de si las unidades de acceso RASL asociadas con la unidad de acceso 0 están presentes. Si las unidades de acceso RASL asociadas con la unidad de acceso 0 no están presentes, el retardo de eliminación de la CPB inicial alternativo y el desplazamiento de retardo se eligen, por ejemplo, mediante un decodificador de vídeo; de lo contrario (es decir, las unidades de acceso RASL asociadas con la unidad de acceso 0 están presentes), el retardo de eliminación de la CPB inicial predeterminado y el desplazamiento de retardo se eligen, por ejemplo,
- 50 mediante un decodificador de vídeo. El conjunto elegido del desplazamiento de retardo de eliminación de la CPB inicial y del desplazamiento de retardo se usa, por ejemplo, por un decodificador de vídeo hasta el final del flujo de bits.
- 55 **[0030]** En la Publicación de Patente de los EE.UU 2013/0107953, fecha de publicación 2 de mayo de 2013, presentada el 30 de octubre de 2012, se describe el siguiente procedimiento. Se puede señalar un desplazamiento de eliminación de la CPB para cada unidad de acceso CRA. Si el HRD se inicializa en dicha unidad de acceso CRA
- 60
- 65

(también denominada unidad de acceso 0), y las imágenes RASL asociadas no están presentes, el tiempo de eliminación de la CPB de cada una de las unidades de acceso que siguen a la unidad de acceso CRA en orden de decodificación se conmuta antes por el desplazamiento de retardo de eliminación de la CPB.

5 **[0031]** Las técnicas de HRD que se describen en el WD9 de HEVC presentan los siguientes problemas. Primero, supongamos que el conjunto elegido de retardo de eliminación de la CPB inicial y de desplazamiento de retardo se aplica al caso en donde todas las unidades de acceso CRA o BLA, si están presentes, después de la unidad de acceso 0 en orden de decodificación en el flujo de bits, tienen sus unidades de acceso RASL asociadas presentes. Las técnicas actuales del WD9 de HEVC sólo funcionarán correctamente si no hay una unidad de acceso CRA o BLA que
10 siga a la unidad de acceso 0 en el orden de decodificación en el flujo de bits, y para las cuales las unidades de acceso RASL asociadas no estén presentes. Sin embargo, si hay dicha unidad de acceso CRA o BLA presente en el flujo de bits, el CPB puede desbordarse después de la primera dicha unidad de acceso CRA o BLA, y como consecuencia pueden producirse resultados de decodificación inesperados.

15 **[0032]** De manera similar, supongamos que el conjunto elegido de retardo de eliminación de la CPB inicial y de desplazamiento de retardo se aplica al caso en donde todas las unidades de acceso CRA o BLA, si están presentes, después de la unidad de acceso 0 en orden de decodificación en el flujo de bits, no tienen sus unidades de acceso RASL asociadas presentes. Las técnicas actuales del WD9 de HEVC solo funcionarán correctamente si no hay una unidad de acceso CRA o BLA que siga a la unidad de acceso 0 en orden de decodificación en el flujo de bits, y las
20 unidades de acceso RASL asociadas para la unidad de acceso 0 están presentes, y no funcionarían correctamente si existe dicha unidad de acceso CRA o BLA presente en el flujo de bits. Bajo las técnicas propuestas para el WD9 de HEVC, esto se aplica a cualquier unidad de acceso CRA o BLA cuando el conjunto elegido de retardo de eliminación de la CPB inicial y de desplazamiento de retardo se aplica al caso en donde todas las unidades de acceso CRA o BLA, si están presentes, siguen a la unidad de acceso 0 en orden de decodificación en el flujo de bits, no tienen sus
25 unidades de acceso RASL asociadas presentes

[0033] Otras suposiciones sobre un conjunto elegido de retardo de eliminación de la CPB inicial y de desplazamiento de retardo también pueden dar como resultado una decodificación incorrecta. Por ejemplo, una situación donde algunas unidades de acceso CRA o BLA que siguen a la unidad de acceso 0 tienen sus unidades de acceso RASL asociadas presentes, y otras unidades de acceso CRA o BLA que siguen a la unidad de acceso 0 no tienen sus
30 unidades de acceso RASL asociadas presentes, también pueden no decodificarse correctamente tampoco, siempre que no se cumpla la suposición sobre las presencias y ausencias de las unidades de acceso RASL asociadas para las siguientes unidades de acceso CRA o BLA.

35 **[0034]** En algunos casos, el desplazamiento de retardo de eliminación de la CPB solo puede compensar los tiempos de eliminación de la CPB para las unidades de acceso que siguen a la unidad de acceso 0 debido al descarte de las unidades de acceso RASL asociadas con la unidad de acceso 0. Si hay una o más de una unidad de acceso CRA o BLA que sigan a la unidad de acceso 0 en el orden de decodificación presente en el flujo de bits y para las cuales se han descartado las unidades de acceso RASL asociadas, el CPB puede desbordarse después de la dicha primera
40 unidad de acceso CRA o BLA, y, como consecuencia, puede producirse un resultado de decodificación inesperado.

[0035] Otro problema asociado con el procedimiento en el WD9 de HEVC como sigue. Actualmente, cuando la unidad de acceso (AU) 0 es una AU de CRA o BLA, si están presentes las AU de RASL asociadas, se usan el retardo de eliminación de la CPB inicial predeterminado y el desplazamiento de retardo; de lo contrario, se usan el retardo de
45 eliminación de la CPB inicial alternativo y el desplazamiento de retardo. Sin embargo, el cambio del retardo de eliminación de la CPB inicial y el desplazamiento de retardo no cambian la duración del tiempo entre los tiempos de decodificación (nominales) de AU0 y la primera AU que no es RASL después de AU0 en orden de decodificación. Por ejemplo, supongamos que hubiese 10 imágenes RASL inmediatamente después de AU0 en orden de decodificación e inmediatamente seguidas por imágenes secundarias en el flujo de bits original. En esta situación, el
50 tiempo de eliminación de la CPB es igual al tiempo de eliminación de la CPB nominal para cada AU, y la duración de tiempo entre los tiempos de decodificación de cualquiera de dos AU consecutivas es una marca de reloj cuando se decodifica el flujo de bits original. De acuerdo con el WD9 de HEVC, la duración del tiempo entre los tiempos de decodificación de AU0 y la primera imagen secundaria sería la misma (11 marcas de reloj) independientemente de si las imágenes RASL se descartan. Por tanto, la decodificación es continua para el flujo de bits original, pero no cuando
55 las imágenes RASL se descartan.

[0036] En vista de estos problemas, esta divulgación proporciona diversos procedimientos y técnicas para mejorar el almacenamiento en memoria intermedia de vídeo (por ejemplo, mejorar el HRD del WD9 de HEVC). Se describirán diversos ejemplos con referencia al WD9 de HEVC. Se puede suponer que cualquier técnica de HRD que no se
60 mencione específicamente es la misma que se especifica actualmente en el WD9 de HEVC u otras en otras memorias descriptivas de HEVC.

[0037] En un ejemplo de la divulgación, un codificador de vídeo para cada unidad de acceso CRA o BLA señala un desplazamiento de retardo de eliminación de la CPB. Para cada unidad de acceso CRA o BLA para la cual las
65 imágenes RASL asociadas no están presentes, independientemente de si el HRD (por ejemplo, el decodificador de vídeo) se inicializa en la unidad de acceso CRA o BLA, el decodificador de vídeo conmuta el tiempo de eliminación de

la CPB de cada una de las unidades de acceso que siguen a la unidad de acceso CRA o BLA en el orden de decodificación antes por el desplazamiento de retardo de eliminación de la CPB. En un ejemplo, el desplazamiento puede ser un desplazamiento acumulativo. Dicho desplazamiento se puede aplicar para conmutar el tiempo de eliminación de la CPB de cada una de las unidades de acceso que siguen a la unidad de acceso CRA o BLA en el orden de decodificación anterior por la cantidad acumulada. Adicionalmente, cuando el desplazamiento debido a cada "imagen RASL restante" es idéntico, el desplazamiento puede ser un desplazamiento por imagen. En dicho ejemplo, el desplazamiento se puede multiplicar por el número de imágenes restantes. La selección entre el retardo de eliminación de la CPB inicial alternativo y predeterminado y el desplazamiento de retardo para la unidad de acceso 0 se puede realizar de la misma manera que en el WD9 de HEVC. En otro ejemplo, el retardo de eliminación de la CPB inicial predeterminado y el desplazamiento de retardo para la unidad de acceso 0 siempre se eligen (es decir, siempre se usan por el decodificador de vídeo), y como consecuencia el retardo de eliminación de la CPB inicial alternativo y el desplazamiento de retardo no se señalan por el codificador para las unidades de acceso CRA o BLA a menos que estén presentes los parámetros de la CPB de nivel de subimágenes y luego el retardo de eliminación de la CPB inicial alternativo y el desplazamiento de retardo son para los propósitos de operaciones de HRD de nivel de subimágenes.

[0038] En un procedimiento de ejemplo para procesar datos de vídeo, un decodificador puede recibir una primera unidad de acceso que tenga una imagen de punto de acceso aleatorio (RAP) en un flujo de bits de vídeo. El decodificador también puede recibir una o más unidades de acceso subsiguientes que tengan una imagen RAP después de la unidad de acceso que tiene la imagen RAP. Por ejemplo, el decodificador puede recibir una o más imágenes CRA o BLA. En el caso de que una o más imágenes RASL para una unidad de acceso subsiguiente de la una o más unidades de acceso subsiguientes no estén presentes en el flujo de bits recibido, el decodificador puede conmutar un tiempo de eliminación de la memoria intermedia de imágenes para una memoria intermedia de imágenes codificadas (CPG) basada anteriormente en un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes. Como se mencionó anteriormente, el desplazamiento de retardo de eliminación de la memoria intermedia de imágenes puede ser [describamos nuevamente cómo se obtiene o selecciona]

[0039] En otro procedimiento de ejemplo para procesar datos de vídeo, un decodificador puede recibir una unidad de acceso que tenga una imagen RAP, en donde las unidades de acceso asociadas que contienen imágenes RASL no se reciban. El decodificador puede inicializar un tiempo de eliminación de la memoria intermedia de imágenes y un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes en respuesta a la recepción de la unidad de acceso y a la no recepción de las unidades de acceso asociadas que contienen imágenes RASL.

[0040] En otro ejemplo de la divulgación, el HRD especifica que un decodificador de vídeo se inicializa o reinicializa en cada unidad de acceso CRA o BLA para la cual las unidades de acceso RASL asociadas no están presentes. En los ejemplos anteriores relacionados con conmutar el tiempo de eliminación de la memoria intermedia de imágenes antes basándose en un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes, tal vez no sea necesaria una inicialización o una reinicialización.

[0041] La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación y decodificación de vídeo 10 de ejemplo que puede utilizar las técnicas descritas en esta divulgación. Como se muestra en la FIG. 1, el sistema 10 incluye un dispositivo de origen 12 que genera datos de vídeo codificados que un dispositivo de destino 14 va a decodificar en un momento posterior. El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender cualquiera de entre una amplia gama de dispositivos, incluidos ordenadores de escritorio, ordenadores plegables (es decir, portátiles), ordenadores de tablet, decodificadores, teléfonos tales como los denominados teléfonos «inteligentes», los denominados teclados «inteligentes», televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, un dispositivo 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.

[0042] El dispositivo de destino 14 puede recibir los datos de vídeo codificados que se van a decodificar mediante un enlace 16. El enlace 16 puede comprender cualquier tipo de medio o dispositivo capaz de desplazar los datos de vídeo codificados desde el dispositivo de origen 12 hasta el dispositivo de destino 14. En un ejemplo, el enlace 16 puede comprender un medio de comunicación para permitir al dispositivo de origen 12 transmitir datos de vídeo codificados directamente al dispositivo de destino 14 en tiempo real. Los datos de vídeo codificados se pueden modular de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y transmitir al dispositivo de destino 14. El medio de comunicación puede comprender cualquier medio de comunicación, inalámbrica o alámbrica, 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 amplia o una red global tal como Internet. El medio de comunicación puede incluir enrutadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 hasta el dispositivo de destino 14.

[0043] De manera alternativa, los datos codificados pueden facilitarse desde la interfaz de salida 22 hasta un dispositivo de almacenamiento 34. De manera similar, se puede acceder a los datos codificados desde el dispositivo de almacenamiento 34 mediante una interfaz de entrada 28. El dispositivo de almacenamiento 34 puede incluir cualquiera de una variedad de medios de almacenamiento de datos de acceso distribuido o local, tales como una

unidad de disco duro, unos discos Blu-ray, unos DVD, unos CD-ROM, una memoria flash, una memoria volátil o no volátil o cualquier otro medio de almacenamiento digital adecuado para almacenar datos de vídeo codificados. En otro ejemplo, el dispositivo de almacenamiento 34 puede corresponder a un servidor de archivos o a otro dispositivo de almacenamiento intermedio que pueda contener el vídeo codificado generado por el dispositivo de origen 12. El dispositivo de destino 14 puede acceder a los datos de vídeo almacenados del dispositivo de almacenamiento 34 por medio de transmisión o de descarga. El servidor de archivos puede ser cualquier tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir esos datos de vídeo codificados al dispositivo de destino 14. Los servidores de archivos de ejemplo incluyen un servidor web (por ejemplo, para una página web), un servidor FTP, dispositivos de almacenamiento conectado en 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 WiFi), una conexión alámbrica (por ejemplo, DSL, módem de cable, etc.) o una combinación de ambos que sea adecuada para acceder a datos de vídeo codificados almacenados en un servidor de archivos. La transmisión de datos de vídeo codificados desde el dispositivo de almacenamiento 34 puede ser una transmisión en continuo, una transmisión de descarga o una combinación de ambas.

[0044] Las técnicas de esta divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas se pueden aplicar a la codificación de vídeo como soporte a cualquiera de una variedad de aplicaciones de multimedia, tales como radiodifusiones de televisión por aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones de vídeo en continuo, por ejemplo, mediante Internet, codificación de vídeo digital para su almacenamiento en un medio de almacenamiento de datos, decodificación de vídeo digital almacenado en un medio de almacenamiento de datos, u otras aplicaciones. En algunos ejemplos, el sistema 10 se puede configurar para soportar transmisión de vídeo unidireccional o bidireccional para apoyar aplicaciones tales como la transmisión de vídeo, la reproducción de vídeo, la radiodifusión de vídeo y/o la videotelefonía.

[0045] En el ejemplo de la FIG. 1, el dispositivo de origen 12 incluye una fuente de vídeo 18, un codificador de vídeo 20 y una interfaz de salida 22. En algunos casos, la interfaz de salida 22 puede incluir un modulador/demodulador (módem) y/o un transmisor. En el dispositivo de origen 12, la fuente de vídeo 18 puede incluir una fuente tal como un dispositivo de captura de vídeo, por ejemplo, una videocámara, un archivo de vídeo que contenga vídeo previamente capturado, una interfaz de vídeo en tiempo real para recibir vídeo desde un proveedor de contenido de vídeo y/o un sistema de gráficos de ordenador para generar datos de gráficos de ordenador como el vídeo de origen, o una combinación de dichas fuentes. En un ejemplo, si la fuente 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, las técnicas descritas en esta divulgación pueden ser aplicables a la codificación de vídeo en general, y se pueden aplicar a aplicaciones inalámbricas y/o alámbricas.

[0046] El vídeo capturado, precapturado o generado por ordenador puede codificarse por el codificador de vídeo 20. Los datos de vídeo codificados se pueden transmitir directamente al dispositivo de destino 14 por medio de la interfaz de salida 22 del dispositivo de origen 12. Los datos de vídeo codificados se pueden almacenar también (o de forma alternativa) en el dispositivo de almacenamiento 34 para un posterior acceso por el dispositivo de destino 14 u otros dispositivos, para su decodificación y/o reproducción.

[0047] En la AVC y en la HEVC, la conformidad del flujo de bits y la conformidad del decodificador se especifican como partes de la memoria descriptiva del HRD. Típicamente, se necesita el HRD en el lado del codificador de vídeo 20 para garantizar la conformidad del flujo de bits, mientras que no es típicamente necesario en el lado del decodificador. Se especifican dos tipos de conformidad de flujo de bits o de HRD, concretamente, Tipo I y Tipo II. Además, se especifican dos tipos de conformidad de decodificador, de conformidad de decodificador de temporización de salida y de conformidad de decodificador de orden de salida.

[0048] El dispositivo de destino 14 incluye una interfaz de entrada 28, un decodificador de vídeo 30 y un dispositivo de visualización 32. En algunos casos, la interfaz de entrada 28 puede incluir un receptor y/o un módem. La interfaz de entrada 28 del dispositivo de destino 14 recibe los datos de vídeo codificados por el enlace 16. Los datos de vídeo codificados comunicados por el enlace 16, o proporcionados en el dispositivo de almacenamiento 34, pueden incluir una variedad de elementos sintácticos generados por el codificador de vídeo 20, para su uso por un decodificador de vídeo, tal como el decodificador de vídeo 30, en la decodificación de los datos de vídeo. Dichos elementos sintácticos se pueden incluir con los datos de vídeo codificados transmitidos en un medio de comunicación, almacenados en un medio de almacenamiento o almacenados en un servidor de archivos.

[0049] El dispositivo de visualización 32 puede estar integrado en, o ser externo a, el dispositivo de destino 14. En algunos ejemplos, el dispositivo de destino 14 puede incluir un dispositivo de visualización integrado y también estar configurado para interconectarse con un dispositivo de visualización externo. En otros ejemplos, el dispositivo de destino 14 puede ser un dispositivo de visualización. En general, el dispositivo de visualización 32 visualiza los datos de vídeo decodificados ante un usuario y puede comprender cualquiera de una variedad de dispositivos de visualización, tales como una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos emisores de luz (OLED) u otro tipo de dispositivo de visualización.

[0050] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar de acuerdo con una norma de compresión de vídeo, tal como la norma de codificación de vídeo de alta eficiencia (HEVC), actualmente en fase de elaboración, y pueden ajustarse al modelo de prueba de HEVC (HM). De manera alternativa, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar de acuerdo a otras normas de propiedad o industriales, tales como la norma ITU-T H.264, de manera alternativa llamada MPEG-4, parte 10, codificación avanzada de vídeo (AVC), o ampliaciones de dichas normas. Sin embargo, las técnicas de esta divulgación no están limitadas a ninguna norma de codificación particular. Otros ejemplos de normas de compresión de vídeo incluyen MPEG-2 e ITU-T H.263.

[0051] Aunque no se muestra en la FIG. 1, en algunos aspectos, tanto el codificador de vídeo 20 como el decodificador de vídeo 30 pueden estar integrados en un codificador y en un decodificador de audio, y pueden incluir unidades MUX-DEMUX adecuadas, u otro tipo de hardware y software, para ocuparse de la codificación tanto de audio como de vídeo en un flujo de datos común o en flujos de datos independientes. Si procede, en algunos ejemplos, las unidades MUX-DEMUX pueden ser conformes al protocolo de multiplexador ITU H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

[0052] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden implementarse cada uno como cualquiera entre una variedad de circuitos codificadores adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables por campo (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de estos. Cuando las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio adecuado no transitorio legible por ordenador, y ejecutar las instrucciones en hardware mediante uno o más procesadores para realizar las técnicas de esta divulgación. Cada uno de entre el codificador de vídeo 20 y el decodificador de vídeo 30 se puede incluir en uno o más codificadores o decodificadores, cualquiera de los cuales se puede integrar como parte de un codificador/decodificador (CÓDEC) combinado en un respectivo dispositivo.

[0053] Como se analizó anteriormente, el JCT-VC ha desarrollado la norma de HEVC. Los trabajos de normalización de HEVC se basan en un modelo de dispositivo de codificación de vídeo llamado modelo de prueba de HEVC (HM). El HM supone varias capacidades adicionales de los dispositivos de codificación de vídeo respecto a dispositivos existentes de acuerdo con, por ejemplo, la norma ITU-T H.264/AVC. Por ejemplo, mientras que la norma H.264 proporciona nueve modos de codificación mediante intrapredicción, el HM puede proporcionar hasta treinta y tres modos de codificación mediante intrapredicción.

[0054] En general, el modelo de funcionamiento del HM describe que una trama o imagen de vídeo puede dividirse en una secuencia de bloques de árbol o unidades de codificación de máximo tamaño (LCU), que incluyen muestras tanto de luma como de croma. Un bloque de árbol tiene un propósito similar al de un macrobloque de la norma H.264. Un fragmento incluye un número de bloques de árbol consecutivos en orden de codificación. Una trama o imagen de vídeo puede partitionarse en uno o más fragmentos. Cada bloque de árbol puede dividirse en unidades de codificación (CU) de acuerdo con un árbol cuaternario. Por ejemplo, un bloque de árbol, como un nodo raíz del árbol cuaternario, puede dividirse en cuatro nodos hijo, y cada nodo hijo puede, a su vez, ser un nodo padre y dividirse en otros cuatro nodos hijo. Un nodo hijo final no dividido, como un nodo hoja del árbol cuaternario, incluye un nodo de codificación, es decir, un bloque de vídeo codificado. Los datos sintácticos asociados a un flujo de bits codificado pueden definir un número máximo de veces que puede dividirse un bloque de árbol, y también pueden definir un tamaño mínimo de los nodos de codificación.

[0055] Una CU incluye un nodo de codificación y unidades de predicción (PU) y unidades de transformada (TU) asociadas con el nodo de codificación. Un tamaño de la CU corresponde en general a un tamaño del nodo de codificación y debe ser típicamente de forma cuadrada. El tamaño de la CU puede variar desde 8x8 píxeles hasta el tamaño del bloque de árbol, con un máximo de 64x64 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 partición de la CU en una o más PU. Los modos de partición pueden diferir dependiendo de si la CU está codificada en modo de salto o directo, codificada en modo de intrapredicción o codificada en modo de interpredicción. Las PU pueden partitionarse para tener forma no cuadrada. Los datos sintácticos asociados con una CU también pueden describir, por ejemplo, la partición de la CU en una o más TU de acuerdo con un árbol cuaternario. Una TU puede tener forma cuadrada o no cuadrada.

[0056] La norma de HEVC admite transformaciones de acuerdo con las TU, que pueden ser diferentes para diferentes CU. El tamaño de las TU se basa típicamente en el tamaño de las PU de una CU dada, definida para una LCU particionada, aunque puede que no sea siempre así. Las TU son típicamente del mismo tamaño o de un tamaño más pequeño que las PU. En algunos ejemplos, las muestras residuales correspondientes a una CU pueden subdividirse en unidades más pequeñas mediante una estructura de árbol cuaternario conocida como «árbol cuaternario residual» (RQT). Los nodos hoja del RQT pueden llamarse unidades de transformada (TU). Los valores de diferencia de píxeles asociados con las TU pueden transformarse para producir coeficientes de transformada, que pueden cuantizarse.

- 5 [0057] En general, una PU incluye datos relacionados con el proceso de predicción. Por ejemplo, cuando la PU se codifica mediante el intramodo, la PU puede incluir datos que describan un modo de intrapredicción para la PU. Como otro ejemplo, cuando la PU está codificada en modo de interpredicción, la PU puede incluir datos que definan un vector 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, lista 0, lista 1 o lista C) para el vector de movimiento.
- 10 [0058] En general, se usa una TU para los procesos de transformada y cuantización. Una CU determinada que tenga una o más PU también puede incluir una o más unidades de transformada (TU). Después de la predicción, el codificador de vídeo 20 puede calcular unos valores residuales a partir del bloque de vídeo identificado por el nodo de codificación de acuerdo con la PU. El nodo de codificación se actualiza luego para hacer referencia a los valores residuales, en lugar del bloque de vídeo original. Los valores residuales comprenden valores de diferencias de píxeles que pueden transformarse en coeficientes de transformada, cuantizarse y escanearse usando las transformadas y otra información de transformada especificada en las TU, para producir coeficientes de transformada serializados para la codificación por entropía. El nodo de codificación puede, una vez más, actualizarse para referirse a estos coeficientes de transformada en serie. Esta divulgación usa típicamente el término «bloque de vídeo» para referirse a un nodo de codificación de una CU. En algunos casos específicos, esta divulgación también puede usar el término «bloque de vídeo» para referirse a un bloque de árbol, es decir, una LCU o una CU, que incluya un nodo de codificación y unas PU y TU.
- 15 [0059] Una secuencia de vídeo incluye típicamente una serie de tramas o imágenes de vídeo. Un grupo de imágenes (GOP) comprende en general una serie de una o más de las imágenes de vídeo. Un GOP puede incluir datos sintácticos en una cabecera del GOP, en una cabecera de una o más de las imágenes, o en otras ubicaciones, que describan un número de imágenes incluidas en el GOP. Cada fragmento de una imagen puede incluir datos sintácticos de fragmento que describan un modo de codificación para el respectivo fragmento. Un codificador de vídeo 20 funciona típicamente sobre bloques de vídeo dentro de 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 de acuerdo con una norma de codificación especificada.
- 20 [0060] Como ejemplo, el HM soporta predicción en diversos tamaños de PU. Si se supone que el tamaño de una CU particular es de $2N \times 2N$, el HM soporta intrapredicción en tamaños de PU de $2N \times 2N$ o $N \times N$ e interpredicción en tamaños de PU simétricos de $2N \times 2N$, $2N \times N$, $N \times 2N$ o $N \times N$. El HM también soporta la división asimétrica para la interpredicción 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 en el 25 % y el 75 %. La porción de la CU correspondiente a la partición del 25 % está indicada por una «n» seguida de una indicación de «arriba», «abajo», «izquierda» o «derecha». Por tanto, por ejemplo, « $2N \times nU$ » se refiere a una CU $2N \times 2N$ que está particionada horizontalmente con una PU $2N \times 0,5N$ encima y una PU $2N \times 1,5N$ debajo.
- 25 [0061] En esta divulgación, « $N \times N$ » y «N por N» pueden usarse indistintamente para hacer referencia a las dimensiones de píxeles de un bloque de vídeo en términos de dimensiones verticales y horizontales, por ejemplo, 16×16 píxeles o 16 por 16 píxeles. En general, un bloque de 16×16 tendrá 16 píxeles en una dirección vertical ($y = 16$) y 16 píxeles en una dirección horizontal ($x = 16$). Asimismo, un bloque de $N \times N$ tiene en general N píxeles en una dirección vertical y N píxeles en una dirección horizontal, donde N representa un valor entero no negativo. Los píxeles de un bloque se pueden disponer en filas y columnas. Además, no es necesario que los bloques tengan 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.
- 30 [0062] Tras la codificación intrapredictiva o interpredictiva, usando las PU de una CU, el codificador de vídeo 20 puede calcular datos residuales a los que se apliquen las transformadas especificadas por las TU de la CU. Los datos residuales pueden corresponder a diferencias de píxeles entre píxeles de la imagen no codificada y valores de predicción correspondientes a las CU. El codificador de vídeo 20 puede formar los datos residuales para la CU, y a continuación transformar los datos residuales para producir coeficientes de transformada.
- 35 [0063] Tras cualquier transformada para generar coeficientes de transformada, el codificador de vídeo 20 puede realizar la cuantización de los coeficientes de transformada. La cuantización se refiere en general a un proceso en el que los coeficientes de transformada se cuantizan para reducir posiblemente la cantidad de datos usados para representar los coeficientes, proporcionando compresión adicional. El proceso de cuantización puede reducir la profundidad de bits asociada a algunos, o a la totalidad, de los coeficientes. Por ejemplo, un valor de n bits puede redondearse hacia abajo hasta un valor de m bits durante la cuantización, donde n es mayor que m .
- 40 [0064] En algunos ejemplos, el codificador de vídeo 20 puede utilizar un orden de escaneo predefinido para examinar los coeficientes de transformada cuantizados para producir un vector en serie que se pueda someter a codificación por entropía. En otros ejemplos, el codificador de vídeo 20 puede realizar un escaneo adaptativo. Después

de escanear los coeficientes de transformada cuantizados para formar un vector unidimensional, el codificador de vídeo 20 puede realizar la codificación de entropía del vector unidimensional, por ejemplo, de acuerdo con la codificación de longitud variable adaptativa de acuerdo con el contexto (CAVLC), la codificación aritmética binaria adaptativa de acuerdo con el contexto (CABAC), la codificación aritmética binaria adaptativa de acuerdo con el contexto basada en la sintaxis (SBAC), la codificación de entropía de división de intervalo de probabilidad (PIPE) u otros procedimientos de codificación de entropía. El codificador de vídeo 20 también puede realizar la codificación por entropía de los 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.

[0065] Para realizar la CABAC, el codificador de vídeo 20 puede asignar un contexto de un modelo de contexto a un símbolo que se vaya a transmitir. El contexto se puede referir, por ejemplo, a si los valores vecinos 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 que se vaya a transmitir. Las contraseñas en la VLC se pueden construir 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 forma, el uso de la VLC puede lograr un ahorro en bits con respecto, por ejemplo, al uso de contraseña de igual longitud para cada símbolo que se vaya a transmitir. La determinación de la probabilidad se puede basar en un contexto asignado al símbolo.

[0066] En algunos ejemplos, el dispositivo de destino 14 puede implementar un procedimiento de procesamiento de datos de vídeo. Por ejemplo, el decodificador de vídeo 30 puede recibir una unidad de acceso que tenga una imagen RAP. El decodificador de vídeo también puede recibir una o más unidades de acceso subsiguientes después de la unidad de acceso que tenga la imagen RAP. En el caso de que una unidad de acceso subsiguiente de una o más unidades de acceso subsiguientes no tenga imágenes RASL asociadas, el decodificador de vídeo 30 conmuta el tiempo de eliminación de la memoria intermedia de imágenes antes basándose en un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes. Por ejemplo, un mensaje SEI del período de almacenamiento en memoria intermedia puede incluir un valor de desplazamiento de retardo de la CPB, por ejemplo, `cpb_delay_offset [i]` que especifica, para el *i*-ésimo CPB, un desplazamiento que se vaya a usar. El desplazamiento se puede usar, por ejemplo, en la derivación de los tiempos de eliminación de la CPB nominales de las unidades de acceso siguiendo, en orden de decodificación, la unidad de acceso CRA o BLA asociada con el mensaje SEI del período de almacenamiento en memoria intermedia cuando las unidades de acceso RASL asociadas con las unidades de acceso CRA o BLA no estén presentes. El elemento sintáctico puede tener una longitud en bits dada por `au_cpbJemoval_delay_length_minus1 + 1` y puede estar, por ejemplo, en unidades de un reloj de 90 KHz.

[0067] En otros ejemplos, el dispositivo de destino 14 puede implementar otro procedimiento de procesamiento de datos de vídeo. Por ejemplo, el decodificador de vídeo 30 puede recibir una unidad de acceso que tenga una imagen RAP, en donde no se reciban las unidades de acceso asociadas que contienen imágenes principales omitidas de acceso aleatorio (RASL). El decodificador de vídeo 30 puede inicializar un tiempo de eliminación de la memoria intermedia de imágenes y un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes en respuesta a la recepción de la unidad de acceso y a la no recepción de las unidades de acceso asociadas que contienen imágenes RASL.

[0068] En otro ejemplo, el decodificador de vídeo 30 puede recibir una unidad de acceso que tenga una imagen RAP, en donde las unidades de acceso asociadas que contienen imágenes RASL no se reciben. El decodificador de vídeo 30 también puede inicializar un tiempo de eliminación de la memoria intermedia de imágenes y un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes en respuesta a la recepción de la unidad de acceso y a la no recepción de las unidades de acceso asociadas que contienen imágenes RASL.

[0069] La FIG. 2 es un diagrama de bloques que ilustra un ejemplo de codificador de vídeo 20 que puede implementar las técnicas descritas en esta divulgación. El codificador de vídeo 20 puede realizar intracodificación e intercodificación de bloques de vídeo dentro de fragmentos de vídeo. La intracodificación se basa en la predicción espacial para reducir o eliminar la redundancia espacial en el vídeo dentro de una trama o imagen de vídeo dada. La intercodificación se basa 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. El intramodo (modo I) puede referirse a cualquiera de varios modos de compresión espacial. Los intermodos, tales como la predicción unidireccional (modo P) o la predicción bidireccional (modo B), pueden referirse a cualquiera de varios modos de compresión temporal.

[0070] En el ejemplo de la FIG. 2, el codificador de vídeo 20 incluye una unidad de partición 35, una unidad de procesamiento de predicción 41, una memoria de imágenes de referencia 64, un sumador 50, una unidad de procesamiento de transformada 52, una unidad de cuantización 54 y una unidad de codificación por entropía 56. La unidad de procesamiento de predicción 41 incluye la unidad de estimación de movimiento 42, la unidad de compensación de movimiento 44 y la unidad de procesamiento de intrapredicción 46. Para la reconstrucción de bloques de vídeo, el codificador de vídeo 20 incluye también una unidad de cuantización inversa 58, una unidad de procesamiento de transformada inversa 60 y un sumador 62. También se puede incluir un filtro de desbloqueo (no mostrado en la FIG. 2) para filtrar límites de bloques para eliminar distorsiones de efecto pixelado del vídeo reconstruido. Si se desea, el filtro de desbloqueo filtrará típicamente la salida del sumador 62. También se pueden usar filtros de bucle adicionales (en bucle o tras el bucle), además del filtro de desbloqueo.

- 5 **[0071]** Como se muestra en la FIG. 2, el codificador de vídeo 20 recibe datos de vídeo, y la unidad de partición 35 particiona los datos en bloques de vídeo. Esta partición también puede incluir la partición en fragmentos, mosaicos u otras unidades mayores, así como la partición de bloques de vídeo, por ejemplo, de acuerdo con una estructura de árbol cuaternario de LCU y CU. El codificador de vídeo 20 ilustra en general los componentes que codifican bloques de vídeo dentro de un fragmento de vídeo que se vaya a codificar. El fragmento se puede dividir en múltiples bloques de vídeo (y posiblemente en conjuntos de bloques de vídeo denominados mosaicos). La unidad de procesamiento de predicción 41 puede seleccionar uno entre una pluralidad de posibles modos de codificación, tal como uno entre una pluralidad de modos de intracodificación, o uno de una pluralidad de modos de intercodificación, para el bloque de vídeo actual basándose en resultados de errores (por ejemplo, la velocidad de codificación y el nivel de distorsión). La unidad de procesamiento de predicción 41 puede proporcionar el bloque intra o intercodificado resultante al sumador 50 para generar datos de bloques residuales y al sumador 62 para reconstruir el bloque codificado para su uso como una imagen de referencia.
- 15 **[0072]** La unidad de procesamiento de intrapredicción 46, dentro de la unidad de procesamiento de predicción 41, puede realizar la codificación intrapredictiva del bloque de vídeo actual con respecto a uno o más bloques contiguos en la misma trama o fragmento que el bloque que vaya a codificarse, para proporcionar compresión espacial. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 dentro de la unidad de procesamiento de predicción 41 realizan la codificación interpredictiva del bloque de vídeo actual con respecto a uno o más bloques predictivos de una o más imágenes de referencia para proporcionar una compresión temporal.
- 20 **[0073]** La unidad de estimación de movimiento 42 puede estar configurada para determinar el modo de interpredicción para un fragmento de vídeo de acuerdo con un patrón por defecto para una secuencia de vídeo. El patrón predeterminado puede designar segmentos de vídeo de la secuencia como fragmentos P o fragmentos B. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar sumamente integradas, pero se ilustran por separado para propósitos conceptuales. La estimación del movimiento, realizada por la unidad de estimación de movimiento 42, es el proceso de generación de vectores de movimiento, que estiman el movimiento de los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una PU de un bloque de vídeo de una trama o imagen de vídeo actual a un bloque predictivo de una imagen de referencia.
- 25 **[0074]** Un bloque predictivo es un bloque que se observa que coincide estrechamente con la PU del bloque de vídeo que se va a codificar en términos de diferencia de píxel, que puede determinarse mediante una suma de una diferencia absoluta (SAD), una suma de diferencia al cuadrado (SSD) u otras métricas de diferencia. En algunos ejemplos, el codificador de vídeo 20 puede calcular valores para posiciones de píxel de subentero 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 de píxel fraccionario de la imagen de referencia. Por lo tanto, la unidad de estimación del movimiento 42 puede realizar una búsqueda de movimiento con respecto a las posiciones de píxeles completas y a las posiciones de píxeles fraccionarias, y generar un vector de movimiento con una precisión de píxel fraccionaria.
- 30 **[0075]** La unidad de estimación de movimiento 42 calcula un vector de movimiento para una PU de un bloque de vídeo en un fragmento intercodificado comparando la posición de la PU con la posición de un bloque predictivo de una imagen de referencia. La imagen de referencia puede seleccionarse de una primera lista de imágenes de referencia (lista 0) o de 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.
- 35 **[0076]** La compensación de movimiento, realizada por la unidad de compensación de movimiento 44, puede implicar extraer o generar el bloque predictivo basándose en el vector de movimiento determinado por la estimación de movimiento, realizando posiblemente interpolaciones hasta la precisión de subpíxel. 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 apunta el vector de movimiento en una de las listas de imágenes de referencia. El codificador de vídeo 20 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 se está codificando, formando valores de diferencia de píxel. Los valores de diferencia de píxel forman datos residuales para el bloque, y pueden incluir componentes de diferencia tanto de luma como de croma. El sumador 50 representa el componente o los componentes que realizan esta operación de resta. La unidad de compensación del movimiento 44 también puede generar elementos sintácticos asociados a los bloques de vídeo y al fragmento de vídeo para su uso por el decodificador de vídeo 30 en la decodificación de los bloques de vídeo del fragmento de vídeo.
- 40 **[0077]** La unidad de intrapredicción 46 puede intrapredicir un bloque actual, como alternativa a la interpredicción realizada 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 procesamiento de intrapredicción 46 puede determinar un modo de intrapredicción que se vaya a usar para codificar un bloque actual. En algunos ejemplos, la unidad de procesamiento de intrapredicción 46 puede codificar un bloque actual usando diversos modos de intrapredicción, por ejemplo, durante
- 45 **[0077]** La unidad de intrapredicción 46 puede intrapredicir un bloque actual, como alternativa a la interpredicción realizada 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 procesamiento de intrapredicción 46 puede determinar un modo de intrapredicción que se vaya a usar para codificar un bloque actual. En algunos ejemplos, la unidad de procesamiento de intrapredicción 46 puede codificar un bloque actual usando diversos modos de intrapredicción, por ejemplo, durante
- 50 **[0077]** La unidad de intrapredicción 46 puede intrapredicir un bloque actual, como alternativa a la interpredicción realizada 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 procesamiento de intrapredicción 46 puede determinar un modo de intrapredicción que se vaya a usar para codificar un bloque actual. En algunos ejemplos, la unidad de procesamiento de intrapredicción 46 puede codificar un bloque actual usando diversos modos de intrapredicción, por ejemplo, durante
- 55 **[0077]** La unidad de intrapredicción 46 puede intrapredicir un bloque actual, como alternativa a la interpredicción realizada 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 procesamiento de intrapredicción 46 puede determinar un modo de intrapredicción que se vaya a usar para codificar un bloque actual. En algunos ejemplos, la unidad de procesamiento de intrapredicción 46 puede codificar un bloque actual usando diversos modos de intrapredicción, por ejemplo, durante
- 60 **[0077]** La unidad de intrapredicción 46 puede intrapredicir un bloque actual, como alternativa a la interpredicción realizada 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 procesamiento de intrapredicción 46 puede determinar un modo de intrapredicción que se vaya a usar para codificar un bloque actual. En algunos ejemplos, la unidad de procesamiento de intrapredicción 46 puede codificar un bloque actual usando diversos modos de intrapredicción, por ejemplo, durante
- 65 **[0077]** La unidad de intrapredicción 46 puede intrapredicir un bloque actual, como alternativa a la interpredicción realizada 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 procesamiento de intrapredicción 46 puede determinar un modo de intrapredicción que se vaya a usar para codificar un bloque actual. En algunos ejemplos, la unidad de procesamiento de intrapredicción 46 puede codificar un bloque actual usando diversos modos de intrapredicción, por ejemplo, durante

diferentes pases de codificación, y la unidad de procesamiento de intrapredicción 46 (o la unidad de selección de modo 40, en algunos ejemplos) puede seleccionar un modo adecuado de intrapredicción para su uso a partir de los modos probados. Por ejemplo, la unidad de procesamiento de intrapredicción 46 puede calcular valores de velocidad-distorsión usando un análisis de velocidad-distorsión para los diversos modos de intrapredicción probados, y seleccionar el modo de intrapredicción que tenga las mejores características de velocidad-distorsión entre los modos probados. El análisis de velocidad-distorsión determina en general una cantidad de distorsión (o error) entre un bloque codificado y un bloque original no codificado que se codificó para producir el bloque codificado, así como una velocidad de bits (es decir, un número de bits) usada para producir el bloque codificado. La unidad de procesamiento de intrapredicción 46 puede calcular razones a partir de las distorsiones y velocidades para los diversos bloques codificados, para determinar qué modo de intrapredicción presenta el mejor valor de distorsión de velocidad para el bloque.

[0078] En cualquier caso, tras seleccionar un modo de intrapredicción para un bloque, la unidad de procesamiento de intrapredicción 46 puede proporcionar información que indique el modo de intrapredicción seleccionado para el bloque a la unidad de codificación por entropía 56. La unidad de codificación por entropía 56 puede codificar la información que indique el modo de intrapredicción seleccionado de acuerdo con las técnicas de esta divulgación. 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 modos de intrapredicción y una pluralidad de tablas de índices de modos de intrapredicción modificadas (también denominadas tablas de correlación de contraseñas), definiciones de contextos de codificación para diversos bloques e indicaciones de un modo de intrapredicción más probable, una tabla de índices de modos de intrapredicción y una tabla de índices de modos de intrapredicción modificadas para su uso para cada uno de los contextos.

[0079] Después de que la unidad de procesamiento de predicción 41 genera el bloque predictivo para el bloque de vídeo actual, ya sea mediante la interpredicción o la intrapredicción, el codificador de vídeo 20 forma un bloque de vídeo residual restando el bloque predictivo del bloque de vídeo actual. Los datos de vídeo residuales del bloque residual pueden incluirse en una o más TU y aplicarse a la unidad de procesamiento de transformada 52. La unidad de procesamiento de transformada 52 transforma los datos de vídeo residuales en coeficientes de transformada residuales usando una transformada, tal como una transformada de coseno discreta (DCT) o una transformada conceptualmente similar. La unidad de procesamiento de transformada 52 puede convertir los datos de vídeo residuales de un dominio de píxeles en un dominio de transformada, tal como un dominio de frecuencia.

[0080] La unidad de procesamiento de transformada 52 puede enviar los coeficientes de transformada resultantes a la unidad de cuantización 54. La unidad de cuantización 54 cuantiza los coeficientes de transformada para reducir aún más la velocidad de bits. El proceso de cuantización puede reducir la profundidad de bits asociada a algunos, o a la totalidad, de los coeficientes. El grado de cuantización se puede modificar ajustando un parámetro de cuantización. En algunos ejemplos, la unidad de cuantización 54 puede realizar luego un escaneo de la matriz que incluya los coeficientes de transformada cuantizados. De manera alternativa, la unidad de codificación por entropía 56 puede realizar el escaneo.

[0081] Tras la cuantización, la unidad de codificación por entropía 56 codifica por entropía los coeficientes de transformada cuantizados. Por ejemplo, la unidad de codificación por entropía 56 puede realizar una codificación de longitud variable adaptativa al contexto (CAVLC), una codificación aritmética binaria adaptativa al contexto (CABAC), una codificación aritmética binaria adaptativa al contexto basada en la sintaxis (SBAC), una codificación por entropía por división de intervalos de probabilidad (PIPE) u otros procedimientos o técnicas de codificación por entropía. Tras la codificación por entropía por la unidad de codificación por entropía 56, el flujo de bits codificado se puede transmitir al decodificador de vídeo 30, o archivar para su posterior transmisión o recuperación por el decodificador de vídeo 30. La unidad de codificación por entropía 56 también puede codificar por entropía los vectores de movimiento y los otros elementos sintácticos para el fragmento de vídeo actual que se esté codificando.

[0082] La unidad de cuantización inversa 58 y la unidad de procesamiento de transformada inversa 60 aplican una cuantización inversa y una transformada inversa, respectivamente, para reconstruir el bloque residual en el dominio de píxel, para su posterior uso como un bloque de referencia de una imagen 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 imágenes de referencia dentro de una de las listas de imágenes de referencia. La unidad de compensación de movimiento 44 también puede aplicar uno o más filtros de interpolación al bloque residual reconstruido para calcular valores de píxeles de subenteros para su uso en la estimación de movimiento. El sumador 62 añade el bloque residual reconstruido al bloque de predicción compensado por movimiento producido por la unidad de compensación de movimiento 44 para producir un bloque de referencia para su almacenamiento en la memoria de imágenes de referencia 64 (a menudo denominado memoria intermedia de imágenes decodificadas (DPB)). La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden usar el bloque de referencia como un bloque de referencia para interpredecir un bloque en una imagen o trama de vídeo subsiguiente.

[0083] Como se analizó anteriormente, en la AVC y en la HEVC, la conformidad del flujo de bits y la conformidad del decodificador se especifican como partes de la memoria descriptiva de HRD. Aunque el HRD se denomina como algún tipo de decodificador, el HRD se necesita en general en el codificador de vídeo 20 para garantizar la conformidad

del flujo de bits, mientras que no es necesario típicamente en el lado del decodificador. Se especifican dos tipos de conformidad de flujo de bits o de HRD, concretamente, Tipo I y Tipo II. Además, se especifican dos tipos de conformidad de decodificador, de conformidad de decodificador de temporización de salida y de conformidad de decodificador de orden de salida.

5 **[0084]** El HRD se inicializa típicamente en una imagen de punto de acceso aleatorio (RAP). Tanto en la HEVC como en la AVC, una vez que se inicializa el HRD, continúa funcionando hasta el final de un flujo de bits sin reinicialización.

10 **[0085]** En el WD9 de HEVC, si el HRD se inicializa en una unidad de acceso CRA o BLA, esta unidad de acceso se denomina unidad de acceso 0. En algunos ejemplos, el codificador de vídeo 20 puede transmitir o almacenar tanto el retardo de eliminación de la CPB inicial predeterminado como el alternativo y el desplazamiento de retardo asociado con la unidad de acceso 0 para su uso en las operaciones de HRD. El codificador de vídeo 20 puede señalar luego el retardo de eliminación de la CPB inicial predeterminado y alternativo y el desplazamiento de retardo asociado con la unidad de acceso 0. Si las unidades de acceso RASL asociadas con la unidad de acceso 0 no están presentes, se eligen el retardo de eliminación de la CPB inicial alternativo y el desplazamiento de retardo; de lo contrario (las unidades de acceso RASL asociadas con la unidad de acceso 0 están presentes), se eligen el retardo de eliminación de la CPB inicial predeterminado y el desplazamiento de retardo. El conjunto elegido de retardo de eliminación de la CPB inicial y el desplazamiento de retardo se pueden señalar en el flujo de bits. El conjunto elegido del retardo de eliminación de la CPB inicial y el desplazamiento de retardo también se pueden usar hasta el final del flujo de bits. Esta información puede señalarse, por ejemplo, desde el codificador de vídeo 20, usando un mensaje SEI del período de almacenamiento en memoria intermedia que incluye un valor de desplazamiento de retardo de la CPB, por ejemplo, `cpb_delay_offset [i]` que especifica, para la *i*-ésima CPB, un desplazamiento que se va a usar.

25 **[0086]** En un ejemplo, el codificador de vídeo 20 puede señalar un desplazamiento de retardo de eliminación de la CPB para cada unidad de acceso CRA o BLA. Para cada unidad de acceso CRA o BLA para la que no están presentes las imágenes RASL asociadas, independientemente de si el HRD se inicializa en la unidad de acceso CRA o BLA, el tiempo de eliminación de CBP de cada una de las unidades de acceso después de la unidad de acceso CRA o BLA en el orden de decodificación se conmuta, por ejemplo, en el decodificador de vídeo 30, antes por el desplazamiento de retardo de eliminación de la CPB. La selección entre el retardo de eliminación de la CPB inicial alternativo y predeterminado y el desplazamiento de retardo para la unidad de acceso 0 se realiza de la misma manera que en el WD9 de HEVC. De manera alternativa, el retardo de eliminación de la CPB inicial predeterminado y el desplazamiento de retardo para la unidad de acceso 0 siempre se eligen y, como consecuencia, el retardo de eliminación de la CPB inicial alternativo y, como consecuencia, el retardo de eliminación de la CPB inicial y el desplazamiento de retardo pueden no señalarse para las unidades de acceso CRA o BLA a menos que los parámetros de la CPB de nivel de imagen estén presentes y luego se usen el retardo de eliminación de la CPB inicial alternativo y el desplazamiento de retardo, por ejemplo, por el decodificador de vídeo 30, para los propósitos de realizar operaciones HRD de nivel de subimágenes.

40 **[0087]** En algunos ejemplos, el codificador de vídeo 20 puede codificar un mensaje de información de mejora suplementaria (SEI) del período de almacenamiento en memoria intermedia, por ejemplo, con sintaxis y semántica como se describe en la subcláusula C.2.3 del WD9 de HEVC (tiempo de la unidad de decodificación y decodificación de la unidad de decodificación). El codificador de vídeo 20 puede implementar los cambios descritos en el presente documento junto con, por ejemplo, el WD9 de HEVC. Otras partes no mencionadas pueden ser las mismas que en el WD9 de HEVC u en otras normas de HEVC, tales como las normas de HEVC subsiguientes.

45 **[0088]** La FIG. 3 es un diagrama de bloques que ilustra un decodificador de vídeo 30 de ejemplo que puede implementar las técnicas descritas en esta divulgación. En el ejemplo de la FIG. 3, el decodificador de vídeo 30 incluye la memoria intermedia de imágenes codificadas (CPB) 78, la unidad de decodificación por entropía 80, la unidad de procesamiento de predicción 81, la unidad de cuantización inversa 86, la unidad de procesamiento de transformada inversa 88, el sumador 90 y la memoria intermedia de imágenes decodificadas (DPB) 92. La unidad de procesamiento de predicción 81 incluye la unidad de compensación de movimiento 82 y la unidad de procesamiento de intrapredicción 84. En algunos ejemplos, el decodificador de vídeo 30 puede realizar una pasada de decodificación que en general es recíproca a la pasada de codificación descrita con respecto al codificador de vídeo 20 de la FIG. 2.

55 **[0089]** El CPB 78 almacena imágenes codificadas del flujo de bits de imágenes codificadas. En un ejemplo, la CPB 78 es una memoria intermedia del tipo 'primera en entrar, primera en salir' que contiene unidades de acceso (AU) en orden de decodificación. Una AU es un conjunto de unidades de capa de abstracción de red (NAL), que están asociadas entre sí de acuerdo con una regla de clasificación especificada, son consecutivas en el orden de decodificación y contienen exactamente una imagen codificada. El orden de decodificación es el orden en el que se decodifican las imágenes, y puede ser diferente del orden en que se muestran las imágenes (es decir, el orden de visualización). El funcionamiento de la CPB puede especificarse por un decodificador hipotético de referencia (HRD) en el WD9 de HEVC.

65 **[0090]** 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 80 del decodificador de vídeo 30 decodifica por

entropía el flujo de bits para generar coeficientes cuantizados, vectores de movimiento y otros elementos sintácticos. La unidad de decodificación por entropía 80 envía los vectores de movimiento y otros elementos sintácticos a la unidad de procesamiento de predicción 81. El decodificador de vídeo 30 puede recibir los elementos sintácticos en el nivel del fragmento de vídeo y/o en el nivel del bloque de vídeo.

5 **[0091]** Cuando el fragmento de vídeo se codifica como un fragmento intracodificado (I), la unidad de procesamiento de intrapredicción 84 de la unidad de procesamiento de predicción 81 puede generar datos de predicción para un bloque de vídeo del fragmento de vídeo actual basándose en un modo de intrapredicción señalado, y datos de bloques previamente decodificados de la trama o imagen actual. Cuando la trama de vídeo se codifica como un fragmento intercodificado (es decir, B, P o GPB), la unidad de compensación de movimiento 82 de la unidad de procesamiento de predicción 81 produce bloques predictivos para un bloque de vídeo del fragmento de vídeo actual basándose en los vectores de movimiento y en otros elementos sintácticos recibidos desde la unidad de decodificación por entropía 80. Los bloques predictivos se pueden producir 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 basándose en las imágenes de referencia almacenadas en la DPB 92.

20 **[0092]** La unidad de compensación de movimiento 82 determina la información de predicción para un bloque de vídeo del fragmento de vídeo actual, analizando los vectores de movimiento y otros elementos sintácticos, y usa la información de predicción para generar los bloques predictivos del bloque de vídeo actual que se esté decodificando. Por ejemplo, la unidad de compensación de movimiento 82 usa algunos de los elementos sintácticos recibidos para determinar un modo de predicción (por ejemplo, intrapredicción o interpredicción) usada para codificar los bloques de vídeo del fragmento de vídeo, un tipo de fragmento de interpredicción (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 con intercodificación del fragmento, el estado de interpredicción para cada bloque de vídeo con intercodificación del fragmento y otra información para decodificar los bloques de vídeo en el fragmento de vídeo actual.

30 **[0093]** La unidad de compensación de movimiento 82 también puede realizar la interpolación basándose en filtros de interpolación. La unidad de compensación de movimiento 82 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 píxeles de subenteros de bloques de referencia. En este caso, la unidad de compensación de movimiento 82 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 producir bloques predictivos.

35 **[0094]** La unidad de cuantización inversa 86 cuantiza de manera inversa, es decir, descuantiza, los coeficientes de transformada cuantizados, proporcionados en el flujo de bits y decodificados por la unidad de decodificación por entropía 80. El proceso de cuantización inversa puede incluir el uso de un parámetro de cuantización calculado por el codificador de vídeo 20 para cada bloque de vídeo del fragmento de vídeo para determinar un grado de cuantización y, asimismo, un grado de cuantización inversa que se debería aplicar. La unidad de procesamiento de transformada inversa 88 aplica una transformada inversa, por ejemplo una DCT inversa, una transformada inversa de enteros o un proceso de transformada inversa conceptualmente similar, a los coeficientes de transformada, con el fin de generar bloques residuales en el dominio de píxeles.

45 **[0095]** Después de que la unidad de compensación de movimiento 82 genera el bloque predictivo para el bloque de vídeo actual, basándose en los vectores de movimiento y en otros elementos sintácticos, el decodificador de vídeo 30 forma un bloque de vídeo decodificado sumando los bloques residuales procedentes de la unidad de procesamiento de transformada inversa 88 a los correspondientes bloques predictivos generados por la unidad de compensación de movimiento 82. El sumador 90 representa el componente o los componentes que realizan esta operación de suma. Si se desea, también se puede aplicar un filtro de desbloqueo para filtrar los bloques decodificados con el fin de eliminar distorsiones de efecto pixelado. También se pueden usar otros filtros de bucle (en el bucle de codificación o bien después del bucle de codificación) 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 se almacenan luego en la DPB 92, que almacena imágenes de referencia usadas para la compensación de movimiento subsiguiente. La DPB 92 almacena también vídeo decodificado para su presentación posterior en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la FIG. 1. Al igual que la CPB 78, en un ejemplo, el HRD puede especificar el funcionamiento de la DPB 92 en el WD9 de HVEC. La CPB 78 y la DPB 92 pueden comprender módulos de unidades de memoria o la misma memoria particionada en diferentes regiones para definir la CPB 78 y la DPB 92.

60 **[0096]** En la siguiente sección, la divulgación analizará las características de HEVC de acceso aleatorio, la conmutación del flujo de bits y las imágenes de puntos de acceso aleatorio (RAP).

65 **[0097]** El acceso aleatorio se refiere a una decodificación de un flujo de bits de vídeo empezando por una imagen codificada que no es la primera imagen codificada del flujo de bits. Se necesita acceso aleatorio a un flujo de bits en muchas aplicaciones de vídeo, tales como radiodifusión y transmisión, por ejemplo, para que los usuarios sintonicen un programa en cualquier momento, para conmutar entre diferentes canales, para saltar a partes específicas del vídeo

o para conmutar a un flujo de bits diferente para la adaptación de flujo (por ejemplo, la adaptación de la velocidad de transmisión de bits, de la velocidad de tramas, de la resolución espacial y similares). El acceso aleatorio se activa al insertar imágenes de acceso aleatorio o puntos de acceso aleatorio (RAP), muchas veces en intervalos regulares, en el flujo de bits de vídeo.

5 **[0098]** El empalme del flujo 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 adjuntarse por un segundo flujo de bits, posiblemente con algunas modificaciones a uno o ambos flujos de bits para generar un flujo de bits empalmado. La primera imagen codificada en el segundo flujo de bits también se denomina 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.

10 **[0099]** El empalme de los flujos de bits se realiza mediante empalmadores de flujos de bits. Los empalmadores de flujos de bits suelen ser ligeros (es decir, menos complejos o menos capaces de procesar) y mucho menos inteligentes que los codificadores de vídeo 20. Por ejemplo, pueden no estar equipados con capacidades de codificación y decodificación por entropía.

15 **[0100]** La conmutación de flujo de bits se puede usar en entornos de transmisión adaptativa. Una operación de conmutación de flujo de bits a una imagen determinada en el flujo de bits conmutado es efectivamente una operación de empalme de flujo de bits en donde el punto de empalme es el punto de conmutación de flujo de bits, es decir, la primera imagen del flujo de bits conmutado.

20 **[0101]** La siguiente sección analizará las imágenes RAP. Un ejemplo de una imagen RAP es una imagen de actualización de decodificación instantánea (IDR). En la imagen IDR, como se especifica en la AVC o en la HEVC, se puede usar para el acceso aleatorio. Sin embargo, dado que las imágenes que siguen a una imagen IDR en orden de decodificación no pueden usar imágenes decodificadas antes de la imagen IDR como referencias (por ejemplo, para interpredicciones), los flujos de bits que dependen de imágenes IDR para acceso aleatorio pueden tener una eficiencia de codificación significativamente menor.

25 **[0102]** Para mejorar la eficiencia de codificación, el concepto de imágenes de acceso aleatorio limpio (CRA) se introdujo en la HEVC para permitir a imágenes que sigan a una imagen CRA en orden de decodificación, pero la preceden en orden de salida para usar imágenes decodificadas antes de la imagen CRA como imágenes de referencia. Las imágenes que siguen a una imagen CRA en orden de decodificación, pero preceden a la imagen CRA en orden de salida, se denominan imágenes principales asociadas con la imagen CRA (o imágenes principales de la imagen CRA). Las imágenes principales de una imagen CRA se pueden decodificar correctamente si la decodificación comienza desde una imagen IDR o CRA antes de la imagen CRA actual. Sin embargo, las imágenes principales de una imagen CRA pueden no ser decodificables cuando se produzca un acceso aleatorio desde la imagen CRA; por lo tanto, las imágenes principales se descartan típicamente durante la decodificación de acceso aleatorio. Para evitar la propagación de errores a partir de imágenes de referencia que pueden no estar disponibles, dependiendo de dónde comience la decodificación, una propuesta para la HEVC especifica que todas las imágenes que siguen a una imagen CRA, tanto en orden de decodificación como en orden de salida, no usarán ninguna imagen que preceda a la imagen CRA en orden de decodificación o en orden de salida (que incluye las imágenes principales) como referencia.

30 **[0103]** El concepto de una imagen de acceso de enlace roto (BLA) se introdujo aún más en la HEVC después de la introducción de imágenes CRA, y se basa en el concepto de imágenes CRA. Una imagen BLA se origina típicamente a partir de un empalme de flujo de bits en la posición de una imagen CRA y, en el flujo de bits empalmado, la imagen CRA del punto de empalme se cambia a una imagen BLA.

35 **[0104]** Las imágenes IDR, las imágenes CRA y las imágenes BLA se denominan colectivamente imágenes de punto de acceso aleatorio (RAP). Una diferencia significativa entre las imágenes BLA y las imágenes CRA es la siguiente. Para una imagen CRA, las imágenes principales asociadas se pueden decodificar correctamente si la decodificación comienza desde una imagen RAP antes de la imagen CRA en orden de decodificación, y pueden decodificarse no correctamente cuando se produzca un acceso aleatorio desde la imagen CRA (es decir, cuando comience la decodificación desde la imagen CRA, o en otras palabras, cuando la imagen CRA sea la primera imagen en el flujo de bits). Para una imagen BLA, las imágenes principales asociadas pueden ser no decodificables en todos los casos, incluso cuando la decodificación comience desde una imagen RAP antes de la imagen BLA en orden de decodificación.

40 **[0105]** Para una imagen CRA o BLA particular, algunas de las imágenes principales asociadas son correctamente decodificables incluso cuando la imagen CRA o BLA sea la primera imagen en el flujo de bits. Estas imágenes principales se conocen como imágenes principales decodificables (DLP) o imágenes principales decodificables de acceso aleatorio (RADL). Otras imágenes principales se conocen como imágenes principales no decodificables (NLP), etiquetadas para imágenes de descarte (TFD) o imágenes principales omitidas de acceso aleatorio (RASL).

45 **[0106]** El WD9 de HEVC especifica los conceptos de un flujo de bits como una secuencia de bits que forma la representación de imágenes codificadas y los datos asociados que forman una o más secuencias de vídeo

codificadas. El flujo de bits es un término colectivo usado para referirse a un flujo de unidad NAL o un flujo de bytes. El WD9 de HEVC especifica los conceptos de un flujo elemental (ES) como una secuencia de uno o más flujos de bits.

- 5 **[0107]** Un flujo elemental que consta de dos o más flujos de bits se habría formado típicamente al empalmar juntos dos o más flujos de bits (o partes del mismo). Si un flujo elemental contiene múltiples flujos de bits, a excepción del último flujo de bits, al final de cada uno de los otros flujos de bits, hay en general una unidad NAL de final de flujo de bits (EOS).
- 10 **[0108]** Las normas de codificación de vídeo incluyen típicamente una memoria descriptiva de un modelo de almacenamiento en memoria intermedia de vídeo. En la AVC y en la HEVC, el modelo de almacenamiento en memoria intermedia se denomina decodificador hipotético de referencia (HRD), que incluye un modelo de almacenamiento en memoria intermedia de tanto la memoria intermedia de imagen codificada (CPB) como la memoria intermedia de imagen decodificada (DPB). Los comportamientos de la CPB y de la DPB se especifican matemáticamente. El HRD impone directamente restricciones sobre temporización, tamaños de almacenamiento en memoria intermedia y velocidades de bits diferentes, 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 de eliminación de la CPB inicial, el tamaño de la CPB, la velocidad de bits, el retardo de salida de DPB inicial y el tamaño de la DPB.
- 15
- 20 **[0109]** En la AVC y en la HEVC, la conformidad del flujo de bits y la conformidad del decodificador se especifican como partes de la memoria descriptiva del HRD. Aunque se denomina decodificador, la memoria descriptiva de HRD se necesita típicamente en el lado del codificador para garantizar la conformidad del flujo de bits. Se especifican dos tipos de conformidad de flujo de bits o de HRD, concretamente, Tipo I y Tipo II. Además, se especifican dos tipos de conformidad del decodificador (conformidad del decodificador de temporización de salida y conformidad del decodificador de orden de salida).
- 25
- [0110]** El HRD se inicializa típicamente en una imagen de punto de acceso aleatorio. Tanto en la HEVC como en la AVC, una vez que se inicializa el HRD, continúa funcionando hasta el final de un flujo de bits sin reinicialización.
- 30 **[0111]** En el WD9 de HEVC, si el HRD se inicializa en una unidad de acceso CRA o BLA (esta unidad de acceso se denomina unidad de acceso 0), se elige el retardo de eliminación de la CPB inicial predeterminado o alternativo y la compensación de retardo asociada con la unidad de acceso 0 para su uso en las operaciones de HRD, dependiendo de si las unidades de acceso RASL asociadas con la unidad de acceso 0 están presentes. Si las unidades de acceso RASL asociadas con la unidad de acceso 0 no están presentes, se eligen el retardo de eliminación de la CPB alternativo inicial y el desplazamiento de retardo; de lo contrario (es decir, las unidades de acceso RASL asociadas con la unidad de acceso 0 están presentes), se eligen el retardo de eliminación de la CPB inicial predeterminado y el desplazamiento de retardo. El decodificador de vídeo 30 usa el conjunto elegido de retardo de eliminación de la CPB inicial y de desplazamiento de retardo hasta el final del flujo de bits.
- 35
- 40 **[0112]** En la Solicitud de Patente de los EE. UU. N° 13/664.279, presentada el 30 de octubre de 2012, cuyo contenido completo se incluye en el presente documento como referencia, se describe el siguiente procedimiento. Se puede señalar un desplazamiento de eliminación de la CPB para cada unidad de acceso CRA. Si el HRD se inicializa en dicha unidad de acceso CRA (también denominada unidad de acceso 0), y las imágenes RASL asociadas no están presentes, el tiempo de eliminación de la CPB de cada una de las unidades de acceso que siguen a la unidad de acceso CRA en orden de decodificación se conmuta antes por el desplazamiento de retardo de eliminación de la CPB.
- 45
- [0113]** Las técnicas de HRD en el WD9 de HEVC presentan los siguientes problemas. Primero, supongamos que el conjunto elegido de retardo de eliminación de la CPB inicial y de desplazamiento de retardo se aplica al caso en el que todas las unidades de acceso CRA o BLA, si están presentes, después de la unidad de acceso 0 en orden de decodificación en el flujo de bits, tienen sus unidades de acceso RASL asociadas presentes. Las técnicas actuales del WD9 de HEVC solo funcionarán correctamente si no hay una unidad de acceso CRA o BLA que siga a la unidad de acceso 0 en el orden de decodificación en el flujo de bits, y para las cuales las unidades de acceso RASL asociadas no estén presentes. Sin embargo, si hay dicha unidad de acceso CRA o BLA presente en el flujo de bits, el CPB puede desbordarse después de la primera dicha unidad de acceso CRA o BLA, y como consecuencia pueden producirse resultados de decodificación inesperados.
- 50
- 55 **[0114]** De manera similar, supongamos que el conjunto elegido de retardo de eliminación de la CPB inicial y de desplazamiento de retardo se aplica al caso en donde todas las unidades de acceso CRA o BLA, si están presentes, después de la unidad de acceso 0 en orden de decodificación en el flujo de bits, no tienen sus unidades de acceso RASL asociadas presentes. Las técnicas actuales del WD9 de HEVC solo funcionarán correctamente si no hay una unidad de acceso CRA o BLA que siga a la unidad de acceso 0 en orden de decodificación en el flujo de bits, y para la cual estén presentes las unidades de acceso RASL asociadas, y no funcionaría correctamente si hubiera dicha unidad de acceso CRA o BLA presente en el flujo de bits.
- 60
- 65 **[0115]** Otras suposiciones sobre un conjunto elegido de retardo de eliminación de la CPB inicial y de desplazamiento de retardo también pueden dar como resultado una decodificación incorrecta. Por ejemplo, una situación en la que

algunas unidades de acceso CRA o BLA que siguen a la unidad de acceso 0 tienen sus unidades de acceso RASL asociadas presentes, y otras unidades de acceso CRA o BLA que siguen a la unidad de acceso 0 no tienen sus unidades de acceso RASL asociadas presentes, también pueden no decodificar correctamente tampoco, siempre que no se cumpla la suposición sobre las presencias y ausencias de las unidades de acceso RASL asociadas para las siguientes unidades de acceso CRA o BLA.

[0116] El desplazamiento de retardo de eliminación de la CPB sólo puede compensar los tiempos de eliminación de la CPB para las unidades de acceso que siguen a la unidad de acceso 0 debido al descarte de las unidades de acceso RASL asociadas con la unidad de acceso 0. Si hay una o más de una unidad de acceso CRA o BLA que sigan a la unidad de acceso 0 en el orden de decodificación presente en el flujo de bits y para las cuales se han descartado las unidades de acceso RASL asociadas, la CPB puede desbordarse después de la dicha primera unidad de acceso CRA o BLA, y como consecuencia puede producirse un resultado de decodificación inesperado.

[0117] Otro problema asociado con el procedimiento en el WD9 de HEVC es el siguiente. Actualmente, cuando la unidad de acceso (AU) 0 es una AU de CRA o BLA, si están presentes las AU de RASL asociadas, se usan el retardo de eliminación de la CPB inicial predeterminado y el desplazamiento de retardo; de lo contrario, se usan el retardo de eliminación de la CPB inicial alternativo y el desplazamiento de retardo. Sin embargo, el cambio del retardo de eliminación de la CPB inicial y el desplazamiento de retardo no cambian la duración del tiempo entre los tiempos de decodificación (nominales) de AUO y la primera AU que no es RASL después de la AUO en orden de decodificación. Por ejemplo, supongamos que hubiera 10 imágenes RASL inmediatamente después de la AUO en orden de decodificación e inmediatamente seguidas por imágenes secundarias en el flujo de bits original. En esta situación, el tiempo de eliminación de la CPB es igual al tiempo de eliminación de la CPB nominal para cada AU, y la duración de tiempo entre los tiempos de decodificación de cualquiera de dos AU consecutivas es una marca de reloj cuando se decodifica el flujo de bits original. De acuerdo con el WD9 de HEVC, la duración del tiempo entre los tiempos de decodificación de la AUO y la primera imagen secundaria sería la misma (11 marcas de reloj) independientemente de si las imágenes RASL se descartan. Por tanto, la decodificación es continua para el flujo de bits original, pero no cuando las imágenes RASL se descartan.

[0118] En vista de estos problemas, esta divulgación proporciona los siguientes procedimientos y técnicas para mejorar el almacenamiento en memoria intermedia de vídeo (por ejemplo, mejorar el HRD del WD9 de HEVC). Los siguientes ejemplos se describirán con referencia al WD9 de HEVC. Cualquier técnica de HRD que no se mencione específicamente puede ser la misma que la que se especifica actualmente en el WD9 HEVC, u otras normas de HEVC, tales como las normas de HEVC subsiguientes.

[0119] En un primer ejemplo de la divulgación, el codificador de vídeo 20 señala un desplazamiento de retardo de eliminación de la CPB para cada unidad de acceso CRA o BLA. Para cada unidad de acceso CRA o BLA para la cual las imágenes RASL asociadas no están presentes, independientemente de si el HRD (por ejemplo, el decodificador de vídeo 30) se inicializa en la unidad de acceso CRA o BLA, el decodificador de vídeo 30 conmuta el tiempo de eliminación de la CPB de cada una de las unidades de acceso que siguen a la unidad de acceso CRA o BLA en el orden de decodificación antes por el desplazamiento de retardo de eliminación de la CPB. Como se describió anteriormente, un mensaje SEI del período de almacenamiento en memoria intermedia puede incluir un valor de desplazamiento de retardo de la CPB, por ejemplo, `cpb_delay_offset [i]` que especifica, para la *i*-ésima CPB, un desplazamiento que se va a usar. El desplazamiento se puede usar, por ejemplo, en la derivación de los tiempos de eliminación de la CPB nominales de las unidades de acceso que siguen, en orden de decodificación, a la unidad de acceso CRA o BLA asociada con el mensaje SEI del período de almacenamiento en memoria intermedia cuando las unidades de acceso RASL asociadas con las unidades de acceso CRA o BLA no estén presentes. El elemento sintáctico puede tener una longitud en bits dada por $\text{au_cpb_removal_delay_length_minus1} + 1$ y puede estar, por ejemplo, en unidades de un reloj de 90 kHz.

[0120] La selección entre el retardo de eliminación de la CPB inicial alternativo y predeterminado y el desplazamiento de retardo para la unidad de acceso 0 se puede realizar de la misma manera que en, por ejemplo, el WD9 de HEVC. En otro ejemplo, el retardo de eliminación de la CPB inicial predeterminado y el desplazamiento de retardo para la unidad de acceso 0 siempre se eligen (por ejemplo, se eligen en el codificador de vídeo 20 para su uso por el decodificador de vídeo 30). Como consecuencia, el codificador de vídeo 20 no señala el retardo de eliminación de la CPB inicial alternativo y el desplazamiento de retardo para las unidades de acceso CRA o BLA a menos que estén presentes los parámetros de la CPB de nivel de subimágenes y luego el retardo de eliminación de la CPB inicial alternativo y el desplazamiento de retardo sean para los propósitos de las operaciones de HRD de nivel de subimágenes. El retardo de eliminación de la CPB inicial alternativo y el desplazamiento de retardo se pueden señalar usando un mensaje SEI del período de almacenamiento en memoria intermedia que tiene la sintaxis descrita en el presente documento.

[0121] En un segundo ejemplo de la divulgación, el HRD especifica que un decodificador de vídeo (por ejemplo, el decodificador de vídeo 30) se inicializa o reinicializa en cada unidad de acceso CRA o BLA para las cuales las unidades de acceso RASL asociadas no están presentes, por ejemplo, como se identifica por cualquiera de las siguientes condiciones siendo verdadera:

- a. nal_unit_type es igual a CRA_NUT o BLA_W_LP, y UseAltCpbParamsFlag es igual a 1;
- b. nal_unit_type es igual a BLA_W_DLP o BLA_N_LP.

5 El elemento sintáctico nal_unit_type especifica el tipo de una estructura de datos de la carga útil de la secuencia de bytes sin formato (RBSP) contenida en la unidad de capa abstracta de red (NAL). CRA_NUT es un tipo NAL para un segmento de fragmento codificado de una imagen CRA. BLA_W_LP es un tipo NAL para un segmento de fragmento codificado de una imagen BLA. Una imagen BLA que tenga nal_unit_type igual a BLA_W_LP puede tener imágenes RASL o RADL asociadas presentes en el flujo de bits. Una imagen BLA que tiene nal_unit_type igual a BLA_W_DLP
 10 no tiene imágenes RASL asociadas presentes en el flujo de bits, pero puede tener imágenes RADL asociadas en el flujo de bits. Por consiguiente, cuando una imagen BLA tiene un nal_unit_type igual a BLA_W_LP (que nuevamente puede tener imágenes RASL o RADL asociadas presentes en el flujo de bits), el decodificador de vídeo 30 puede inicializarse o reinicializarse en cada acceso CRA o BLA. Una imagen BLA que tiene nal_unit_type igual a BLA_N_LP
 15 no tiene imágenes principales asociadas presentes en el flujo de bits. Por consiguiente, cuando una imagen BLA tiene un nal_unit_type igual a BLA_N_LP, el decodificador de vídeo 30 no se inicializa o reinicializa en general en cada acceso CRA o BLA. UseAltCpbParamsFlag indica si se usan parámetros CPB alternativos o no. Si se utilizan parámetros de CPG alternativos, el decodificador de vídeo 30 puede usar un desplazamiento especificado por un valor de desplazamiento de retardo de la CPB, por ejemplo, cpb_delay_offset [i] que especifica, para la i-ésima CPB, el desplazamiento que se va a usar.

20 **[0122]** En un tercer ejemplo de la divulgación, cada unidad de acceso CRA o BLA para la cual no están presentes las unidades de acceso RASL asociadas, por ejemplo, como se identifica por cualquiera de las siguientes condiciones que son verdaderas, se considera, por ejemplo, por el decodificador de vídeo 30, como primera unidad de acceso en un flujo de bits (es decir, una unidad de acceso de este tipo inicializa un nuevo flujo de bits), y por tanto no es necesario
 25 que haya una unidad NAL de final de flujo de bits al final de cada flujo de bits que no sea el último flujo de bits en un flujo elemental:

- a. nal_unit_type es igual a CRA_NUT o BLA_W_LP, y UseAltCpbParamsFlag es igual a 1;
- b. nal_unit_type es igual a BLA_W_DLP o BLA_N_LP.

30 Como consecuencia, al implementar el modelo HRD, el decodificador de vídeo 30 inicializa o reinicializa el HRD en la unidad de acceso de acuerdo con la norma de HEVC, por ejemplo. En este tercer ejemplo, como se explicó anteriormente, cada unidad de acceso CRA o BLA para la cual las unidades de acceso RASL asociadas no están presentes se considera como la primera unidad de acceso en un flujo de bits.

35 **[0123]** En un cuarto ejemplo de la divulgación, para cada unidad de acceso CRA o BLA para la cual las unidades de acceso RASL asociadas no están presentes, por ejemplo, como se identifica si cualquiera de las siguientes condiciones es verdadera, se requiere que haya una unidad NAL de final de flujo de bits que preceda inmediatamente, en orden de decodificación, la primera unidad NAL en la unidad de acceso:

- a. nal_unit_type es igual a CRA_NUT o BLA_W_LP, y UseAltCpbParamsFlag es igual a 1;
- b. nal_unit_type es igual a BLA_W_DLP o BLA_N_LP.

40 Como consecuencia, al implementar el modelo HRD, el decodificador de vídeo 30 puede inicializar o reinicializar el HRD en la unidad de acceso de acuerdo con la norma de HEVC, por ejemplo.

45 **[0124]** Común a los segundo, tercer y cuarto ejemplos anteriores, cuando el decodificador de vídeo 30 se inicializa o reinicializa en una unidad de acceso CRA o BLA, en algunos ejemplos, tanto el retardo de eliminación de la CPB inicial predeterminado como el alternativo y el desplazamiento de retardo asociado con la unidad de acceso puede transmitirse o almacenarse por el decodificador de vídeo 30 para su uso en las operaciones HRD, dependiendo de si las unidades de acceso RASL asociadas con la unidad de acceso están presentes. Si las unidades de acceso RASL asociadas con la unidad de acceso no están presentes (por ejemplo, cuando cualquiera de las dos condiciones a o b anteriores son verdaderas), se seleccionan el retardo de eliminación de la CPB inicial alternativo y el desplazamiento de retardo; de lo contrario (las unidades de acceso RASL asociadas con la unidad de acceso están presentes), se seleccionan el retardo de eliminación de la CPB inicial predeterminado y el desplazamiento de retardo. El decodificador de vídeo 30 usa luego el conjunto elegido de retardo de eliminación de la CPB inicial y de desplazamiento de retardo hasta que se reinicializa el HRD o el final del flujo de bits. En un ejemplo, cuando dicha unidad de acceso CRA o BLA no es la primera unidad de acceso en el flujo de bits (es decir, no la unidad de acceso 0), solo se reinicializa la CPB, la DPB no se reinicializa. En otro ejemplo, cuando dicha unidad de acceso CRA o BLA no es la primera unidad de
 50 acceso en el flujo de bits (es decir, no la unidad de acceso 0), ni la CPB ni la DPB se reinicializan, sino que se hace una elección entre el retardo de eliminación de la CPB inicial alternativo o el predeterminado y el desplazamiento de retardo de manera similar a la descripción anterior. Por ejemplo, el decodificador de vídeo 30 puede seleccionar entre el retardo de eliminación de la CPB inicial alternativo y el predeterminado y el desplazamiento de retardo y luego el alternativo seleccionado puede usarse, por ejemplo, por el decodificador de vídeo 30 hasta que se reinicie el HRD o
 55 el final del flujo de bits.
 60
 65

[0125] A continuación se proporciona un análisis de la primera técnica de ejemplo. Cualquier referencia a cláusulas y subcláusulas a continuación se refiere al WD9 de HEVC. A menos que se especifique lo contrario, la operación y el significado de todos los elementos sintácticos se definen en el WD9 de HEVC y en general se pueden aplicar a las extensiones de HEVC subsiguientes.

5 **[0126]** De acuerdo con los sistemas y procedimientos descritos en el presente documento, la sintaxis y la semántica del mensaje SEI del período de almacenamiento en memoria intermedia en la subcláusula C.2.3 del WD9 de HEVC (tiempo de eliminación de la unidad de decodificación y decodificación de la unidad de decodificación) en, por ejemplo, el WD9, se puede cambiar como sigue. El decodificador de vídeo 30 puede, por ejemplo, implementar la decodificación usando algunos o todos los aspectos de la sintaxis y la semántica del mensaje SEI del período de almacenamiento en memoria intermedia de la subcláusula C.2.3 del WD9 de HEVC. Los elementos sintácticos actualizados están en **negrita**. Otras partes no mencionadas pueden ser las mismas que en el WD9 de HEVC o en las normas de HEVC posteriores.

15 **Tabla 1 - Sintaxis de mensajes SEI del período de almacenamiento en memoria intermedia**

buffering_period(payloadSize) {	Descriptor
bp_seq_parameter_set_id	ue(v)
si(!sub_pic_cpb_params_present_flag	
rap_cpb_params_present_flag	u(1)
si(NalHrdBpPresentFlag) {	
para(i = 0; i <= CpbCnt; i++) {	
initial_cpb_removal_delay [i]	u(v)
initial_cpb_removal_offset [i]	u(v)
si((sub_pic_cpb_params_present_flag rap_cpb_params_present_flag) {	
initial_alt_cpb_removal_delay [i]	u(v)
initial_alt_cpb_removal_offset [i]	u(v)
}	
si(rap_cpb_params_present_flag)	
cpb_delay_offset [i]	u(v)
}	
}	
si(VclHrdBpPresentFlag) {	
para(i = 0; i <= CpbCnt; i++) {	
initial_cpb_removal_delay [i]	u(v)
initial_cpb_removal_offset [i]	u(v)
si(sub_pic_cpb_params_present_flag rap_cpb_params_present_flag) {	
initial_alt_cpb_removal_delay [i]	u(v)
initial_alt_cpb_removal_offset [i]	u(v)
}	
si(rap_cpb_params_present_flag)	
cpb_delay_offset[i]	u(v)
}	
}	
}	

[0127] Un mensaje de información de mejora suplementaria (SEI) del período de almacenamiento en memoria intermedia proporciona el retardo de eliminación de la CPB inicial y la información de desplazamiento de retardo de

eliminación de la CPB inicial para la inicialización del HRD en la posición de la unidad de acceso asociada en orden de decodificación.

5 **[0128]** Lo siguiente se aplica a la sintaxis y a la semántica de los mensajes SEI del período de almacenamiento en memoria intermedia. Primero, los elementos sintácticos `initial_cpb_removal_delay_length_minus1`, `au_cpb_removal_delay_length_minus1`, `and_sub_pic_cpb_params_present_flag`, y los variables `NalHrdBpPresentFlag` and `VclHrdBpPresentFlag` se encuentra en o se derivan, por ejemplo, el decodificador 30, de elementos sintácticos en la estructura sintáctica `hrd_parameters()` que es aplicable a al menos uno de los puntos de operación a los que se aplica el mensaje SEI del período de almacenamiento en memoria intermedia. Segundo, las variables `CpbSize [i]`, `BitRate [i]` y `CpbCnt` se derivan, por ejemplo, del decodificador de vídeo 30, de elementos sintácticos encontrados en la estructura sintáctica `sub_layer_hrd_parameters()` que es aplicable a al menos uno de los puntos de operación se aplica el mensaje SEI del período de almacenamiento en memoria intermedia. Tercero, cualquiera de los dos puntos de operación a los que se aplique el mensaje SEI del período de almacenamiento en memoria intermedia con diferentes valores de `OpTid tIdA` y `tIdB` indican que los valores de `cpb_cnt_minus1 [tIdA]` y `cpb_cnt_minus1 [tIdB]` codificados en la(s) estructura(s) sintáctica(s) de `hrd_parameters()` aplicable(s) a los dos puntos de operación son idénticos. Cuarto, cualquiera de los dos puntos de operación a los que se aplica el mensaje SEI del período de almacenamiento en memoria intermedia que tiene diferentes valores de `OpLayerIdSet layerIdSetA` y `layerIdSetB` indica que los valores de `nal_hrd_parameters_present_flag` y `vcl_hrd_parameters_present_flag`, respectivamente, para las dos estructuras sintácticas de `hrd_parameters()` aplicables a los dos puntos de operación son idénticos. Finalmente, el flujo de bits (o una parte del mismo) se refiere al subconjunto del flujo de bits (o una parte del mismo) asociado con cualquiera de los puntos de operación a los que se aplica el mensaje SEI del período de almacenamiento en memoria intermedia.

25 **[0129]** Si los elementos sintácticos `NalHrdBpPresentFlag` o `VclHrdBpPresentFlag` son iguales a 1, se puede asociar un mensaje SEI del período de almacenamiento en memoria intermedia aplicable a los puntos de operación especificados en la secuencia de vídeo codificada, y un mensaje SEI del período de almacenamiento en memoria intermedia aplicable a los puntos de operación especificados asociado con cada unidad de acceso RAP, con cada unidad de acceso asociada con un mensaje SEI del punto de recuperación. De lo contrario (`NalHrdBpPresentFlag` y `VclHrdBpPresentFlag` son iguales a 0), en este ejemplo, ninguna unidad de acceso en la secuencia de vídeo codificada se asociará con un mensaje SEI del período de almacenamiento en memoria intermedia aplicable a los puntos de operación especificados.

35 **[0130]** Para algunas aplicaciones, la presencia frecuente de mensajes SEI del período de almacenamiento en memoria intermedia puede ser deseable.

[0131] Cuando está presente una unidad NAL de SEI que contiene un mensaje SEI del período de almacenamiento en memoria intermedia y tiene `nuh_reserved_zero_6bits` igual a 0, la unidad NAL SEI deberá preceder, en orden de decodificación, a la primera unidad NAL VCL en la unidad de acceso.

40 **[0132]** El elemento sintáctico `bp_seq_parámetro_set_id` especifica el `sps_seq_parameter_set_id` para el conjunto de parámetros de secuencia que está activo para la imagen codificada asociada con el mensaje SEI del período de almacenamiento en memoria intermedia. El valor de `bp_seq_parameter_set_id` será igual al valor de `pps_seq_parameter-set-id` en el conjunto de parámetros de imagen referenciado por `slice_pic_parameter_set_id` de los encabezados de segmento de fragmento de la imagen codificada asociada con el mensaje SEI del período de almacenamiento en memoria intermedia. El valor de `pic_order_cnt_delta` deberá estar dentro del intervalo de 0 y 15, inclusive.

50 **[0133]** El elemento sintáctico `rap_cpb_params_present_flag` igual a 1 especifica la presencia de los elementos sintácticos `initial_alt_cpb_removal_delay [i]` y `initial_alt_cpb_removal_offset [i]`. Cuando no está presente, se deduce que el valor de `rap_cpb_params_present_flag` es igual a 0. Cuando la imagen asociada no es una imagen CRA ni una imagen BLA, el valor de `rap_cpb_params_present_flag` será igual a 0.

55 **[0134]** El elemento sintáctico `initial_cpb_removal_delay [i]` y `initial_alt_cpb_removal_delay [i]` especifican los retardos de eliminación predeterminados de CPB iniciales alternativos, respectivamente, para la i-ésima CPB. Los elementos sintácticos tienen una longitud en bits dada por `initial_cpb_removal_delay_length_minus1 + 1`, y están en unidades de un reloj de 90 kHz. Los valores de los elementos sintácticos no serán iguales a 0 y serán menores o iguales a $90000 * (CpbSize [i] \div BitRate [i])$, el equivalente en tiempo del tamaño del CPB en unidades de reloj de 90 kHz.

60 **[0135]** Los elementos sintácticos `initial_cpb_removal_offset[i]` y `initial_alt_cpb_removal_offset[i]` especifican los desplazamientos de eliminación de la CPB iniciales predeterminados y alternativos, respectivamente, para la i-ésima CPB para especificar el tiempo de entrega inicial de las unidades de datos codificadas a la CPB. Los elementos sintácticos tienen una longitud en bits dada por `initial_cpb_removal_delay_length_minus1 + 1` y están en unidades de un reloj de 90 kHz.

65

[0136] A lo largo de toda la secuencia de vídeo codificada, la suma de `initial_cpb_removal_delay[i]` y `initial_cpb_removal_offset[i]` deberá ser constante para cada valor de `i`, y la suma de `initial_alt_cpb_removal_delay[i]` y `initial_alt_cpb_removal_offset[i]` deberá ser constante para cada valor de `i`.

5 [0137] El elemento sintáctico `cpb_delay_offset[i]` especifica, para la `i`-ésima CPB, un desplazamiento para su uso en la derivación de los tiempos de eliminación de la CPB nominales de las unidades de acceso que siguen, en orden de decodificación, a la unidad de acceso CRA o BLA asociada con el mensaje SEI del período de almacenamiento en memoria intermedia cuando las unidades de acceso RASL asociadas con la unidad de acceso CRA o BLA no están presentes. El elemento sintáctico tiene una longitud en bits dada por `au_cpb_removal_delay_length_minus1 + 1` y está en unidades de un reloj de 90 kHz.

10 [0138] En algunos ejemplos, se recomienda que los codificadores de vídeo 20 no incluyan `initial_alt_cpb_removal_delay[i]`, `initial_alt_cpb_removal_offset[i]`, y `cpb_delay_offset [i]` en los mensajes SEI del período de almacenamiento en memoria intermedia asociados con las imágenes de CRA o BLA que tienen imágenes RASL y RADL asociadas que se intercalan en orden de decodificación.

15 [0139] La siguiente sección analizará el tiempo de decodificación de la eliminación de unidades y la decodificación de unidades de decodificación. Las variables `InitCpbRemovalDelay [SchedSelIdx]`, `InitCpbRemovalDelayOffset [SchedSelIdx]` y `cpbDelayOffset [SchedSelIdx]` se derivan como sigue. Primero, si una o más de las condiciones (A) y (B) presentadas a continuación son verdaderas, `InitCpbRemovalDelay[SchedSelIdx]`, `InitCpbRemovalDelayOffset[SchedSelIdx]` y `cpbDelayOffset[SchedSelIdx]` se establecen iguales a los valores del mensaje SEI del período de almacenamiento en memoria intermedia `initial_alt_cpb_removal_delay [SchedSelIdx]`, `initial_alt_cpb_removal_offset [SchedSelIdx]`, y `cpb_delay_offset [SchedSelIdx]`, respectivamente, que se seleccionan dependiendo de `NalHrdModeFlag` como se especifica en la subclase C.1 del WD9 HVEC. La condición (A) es que la unidad de acceso 0 es una unidad de acceso BLA para la cual la imagen codificada tiene `nal_unit_type` igual a `BLA_W_DLP` o `BLA_N_LP`, y el valor de `rap_cpb_params_present_flag` del mensaje del período de almacenamiento en memoria intermedia es igual a 1. La condición (B) es que la unidad de acceso 0 es una unidad de acceso BLA para la cual la imagen codificada tiene `nal_unit_type` igual a `BLA_W_LP` o es una unidad de acceso CRA, y el valor de `rap_cpb_params_present_flag` del mensaje SEI del período de almacenamiento en memoria intermedia es igual a 1, y una más de las siguientes condiciones se cumplen: (1) `UseAltCpbParamsFlag` para la unidad de acceso 0 es igual a 1. `DefaultInitCpbParamsFlag` es igual a 0, (2) en caso contrario, `InitCpbRemovalDelay [SchedSelIdx]` y `InitCpbRemovalDelayOffset [SchedSelIdx]` se establecen iguales a los valores de los elementos sintácticos del mensaje SEI del período de almacenamiento en memoria intermedia `initial_cpb_removal_delay [SchedSelIdx]` y `initial_cpb_removal_offset [SchedSelIdx]`, respectivamente, que se seleccionan dependiendo en `NalHrdModeFlag` como se especifica en la subcláusula C.1, y `cpbDelayOffset [SchedSelIdx]` se establece igual a 0.

20 [0140] El tiempo nominal de eliminación de la unidad de acceso `n` del CPB se especifica como sigue. Primero, si la unidad de acceso `n` es la unidad de acceso con `n` igual a 0 (la unidad de acceso que inicializa el HRD), el tiempo de eliminación nominal de la unidad de acceso del CPB se especifica mediante:

40
$$\text{Tiempo de eliminación nominal}(0) = \text{InitCpbRemovalDelay}[\text{SchcdSelIdx}] / 90000 \text{ (C-9)}$$

De lo contrario, cuando la unidad de acceso `n` es la primera unidad de acceso de un período de almacenamiento en memoria intermedia que no inicializa el HRD, se aplica lo siguiente. El tiempo de eliminación nominal de la unidad de acceso `n` de la CPB se especifica mediante:

45
$$\text{NominalRemovalTime}(n) = \text{NominalRemovalTime}(\text{firstPicInPrevBuffPeriod}) + \text{ClockTick} * ((\text{AuCpbRemovalDelayVal}(n) - \text{cpbDclayOffset}[\text{SchedSelIdx}]) \text{ (C -10)})$$

50 donde `NominalRemovalTime(firstPicInPrevBuffPeriod)` es el tiempo de eliminación nominal de la primera unidad de acceso del período de almacenamiento en memoria intermedia anterior, y `AuCpbRemovalDelayVal(n)` es el valor de `AuCpbRemovalDelayVal` derivado de acuerdo con `au_cpb_removal_delay_minus1` en el mensaje SEI de temporización de imágenes, seleccionado como se especifica en la subcláusula C.1, asociada con la unidad de acceso `n`.

55 [0141] Después de la derivación del tiempo de eliminación de la CPB nominal de la unidad de acceso `n`, el valor de `cpbDelayOffset[SchedSelIdx]` se actualiza de la siguiente manera. Si una o más de las siguientes condiciones son verdaderas, `cpbDelayOffset[SchedSelIdx]` se establece igual al valor de la sintaxis del mensaje SEI del período de almacenamiento en memoria intermedia `elementcpb_removal_delay_offset[SchedSelIdx]`, seleccionado dependiendo de cómo `NalHrdModeFlag` se especifica en la cláusula C.1 del WD9 de HVEC. La unidad de acceso `n` es una unidad de acceso BLA para la cual la imagen codificada tiene `nal_unit_type` igual a `BLA_W_DLP` o `BLA_N_LP`, y el valor de `rap_cpb_params_present_flag` del mensaje SEI del período de almacenamiento en memoria intermedia asociado es igual a 1. La unidad de acceso `n` es una unidad de acceso BLA para la cual la imagen codificada tiene `nal_unit_type` igual a `BLA_W_LP` o es una unidad de acceso CRA, y el valor de `rap_cpb_params_present_flag` del mensaje SEI del período de almacenamiento en memoria intermedia es igual a 1, y `UseAltCpbParamsFlag` para la unidad de acceso `n`

es igual a 1. De lo contrario, `cpbDelayOffset [SchedSelIdx]` se establece igual a 0. Cuando la unidad de acceso `n` no es la primera unidad de acceso de un período de almacenamiento en memoria intermedia, el tiempo nominal de eliminación de la unidad de acceso `n` de la CPB se especifica mediante:

$$\begin{aligned}
 &5 \quad \text{NominalRemovalTime}(n) = \text{NominalRemovalTime}(\text{firstPicInCurrBuffPeriod}) \\
 &\quad + \\
 &\quad \text{ClockTick} * ((\text{AuCpbRemovalDelayVal}(n)) - \\
 &\quad \text{cpbDelayOffset}[\text{SchedSelIdx}]) \quad (\text{C-11})
 \end{aligned}$$

10 donde `NominalRemovalTime(firstPicInCurrBuffPeriod)` es el tiempo nominal de eliminación de la primera unidad de acceso del período de almacenamiento en memoria intermedia actual.

15 **[0142]** Cuando `sub_pic_cpb_params_present_flag` es igual a 1, se aplica lo siguiente. La variable `cpbRemovalDelayInc(m)` se deriva como sigue. Si `sub_pic_cpb_params_in_pic_timing_sei_flag` es igual a 0, la variable `cpbRemovalDelayInc(m)` se establece en el valor de `du_spt_cpb_removal_delay_increment` en el mensaje SEI de la información de la unidad de decodificación, seleccionado como especificado en la subcláusula C.1. De lo contrario, si `du_common_cpb_removal_delay_flag` es igual a 0, la variable `cpbRemovalDelayInc(m)` se establece en el valor de: `du_cpb_removal_delay_increment_minus1 [i] + 1` para la unidad de decodificación `m` en el mensaje SEI de la temporización de imágenes, seleccionado como se especifica en la subcláusula C.1., asociada con la unidad de acceso `n`, donde el valor de `i` es 0 para el primer `num_nalus_in_du_minus1 [0] + 1` unidades NAL consecutivas en la unidad de acceso que contiene la unidad de decodificación `m`, 1 para las `num_nalus_in_du_minus1 [1] + 1` unidades NAL subsiguientes en la misma unidad de acceso, etc. De lo contrario, la variable `cpbRemovalDelayInc(m)` se establece en el valor de `du_common_cpb_removal_delay_increment_minus1 + 1` en el mensaje SEI de la temporización de imágenes, seleccionado como se especifica en la subcláusula C.1, asociado a la unidad de acceso `n`. El tiempo de eliminación nominal de la unidad de decodificación `m` del CPB se especifica como sigue, donde `NominalRemovalTime(n)` es el tiempo de eliminación nominal de la unidad de acceso `n`. Si la unidad de decodificación `m` es la última unidad de decodificación en la unidad de acceso `n`, el tiempo de eliminación nominal de la unidad de decodificación `m` `NominalRemovalTime(m)` se establece en `NominalRemovalTime(n)`. De lo contrario, (es decir, la unidad de decodificación `m` no es la última unidad de decodificación en la unidad de acceso `n`), el tiempo de eliminación nominal de la unidad de decodificación `m` `NominalRemovalTime(m)` se deriva de la siguiente manera.

```

If(sub_pic_cpb_params_in_pic_timing_sei_flag); NominalRemovalTime(m) =
  NominalRemovalTime(m + 1) - ClockSubTick * cpbRemovalDelayInc(m) (C-12)
else
  NominalRemovalTime(m) = NominalRemovalTime(n) -
  ClockSubTick * cpbRemovalDelayInc(m)

```

35 **[0143]** El tiempo de eliminación de la unidad de acceso `n` del CPB se especifica como sigue, donde `FinalArrivalTime(m)` y `NominalRemovalTime(m)` son el tiempo de llegada final y el tiempo de eliminación nominal, respectivamente, de la última unidad de decodificación en la unidad de acceso `n`, y `FinalArrivalTime(n)` y `NominalRemovalTime(n)` son el tiempo de llegada final y el tiempo de eliminación nominal, respectivamente, de la unidad de acceso `n`.

```

if(!low_delay_hrd_flag[HighestTid] || NominalRemovalTime(n) >=
  FinalArrivalTime(n))
  CpbRemovalTime(n) = NominalRemovalTime(n)
else if(sub_pic_cpb_params_present_flag) (C-13)
  CpbRemovalTime(n) = NominalRemovalTime(n) +
    Max((ClockSubTick * Ceil((FinalArrivalTime(m) -
    NominalRemovalTime(m))/ClockSubTick)),
    (ClockTick * Ceil((FinalArrivalTime(n) - NominalRemovalTime(n)) / ClockTick
    )))
else
  CpbRemovalTime(n) = NominalRemovalTime(n) + ClockTick *
    Ceil((FinalArrivalTime(n) - NominalRemovalTime(n)) / ClockTick)

```

40 **[0144]** Cuando `SubPicCpbFlag` es igual a 1, la variable `lastDulnAuFlag` se establece en 1 si la unidad de decodificación `m` es la última unidad de decodificación de la unidad de acceso `n` y 0 de otro modo, y el tiempo de eliminación de la unidad de decodificación `m` del CPB se especifica como sigue.

```

45 if(!low_delay_hrd_flag[HighestTid] || NominalRemovalTime(m) >=

```

```

FinalArrivalTime (m) )
    CpbRemovalTime (m) = NominalRemovalTime (m)
else if (!lastDuInAuFlag) (C-14)
    CpbRemovalTime (m) = NominalRemovalTime (m) + ClockSubTick *
        Ceil ((FinalArrivalTime (m) - NominalRemovalTime (m)) /
ClockSubTick)
else
    CpbRemovalTime (m) = CpbRemovalTime (n)

```

5 **[0145]** Cuando `low_delay_hrd_flag` [HighestTid] es igual a 1 y `NominalRemovalTime (m) < FinalArrivalTime(m)`, el tamaño de la unidad de decodificación `m` es tan grande que impide la eliminación en el tiempo de eliminación nominal. En el momento de la eliminación de la CPB de la unidad de decodificación `m`, la unidad de decodificación se decodifica instantáneamente. La imagen `n` se considera decodificada cuando se decodifica la última unidad de decodificación de la imagen. Los cambios descritos anteriormente también pueden implementarse mediante un codificador de vídeo 20, por ejemplo, para codificar un flujo de bits usando algunos o todos los aspectos de la sintaxis y la semántica de los mensajes SEI del período de almacenamiento en memoria intermedia y la subcláusula C.2.3 del WD9 de HEVC, de modo que el flujo de bits pueda decodificarse por el decodificador de vídeo 30.

15 **[0146]** La FIG. 4 es un diagrama de flujo que ilustra un procedimiento de ejemplo de acuerdo con uno o más ejemplos descritos en esta divulgación. El dispositivo de destino 14 puede implementar un procedimiento de procesamiento de datos de vídeo. Por ejemplo, el decodificador de vídeo 30 puede recibir una unidad de acceso que tenga una imagen RAP en un flujo de bits de vídeo (400). El decodificador de vídeo también puede recibir en el flujo de bits, después de la unidad de acceso, una unidad de acceso subsiguiente que tenga una imagen RAP (402). En el caso de que una o más imágenes principales omitidas de acceso aleatorio (RASL) para la unidad de acceso subsiguiente no estén presentes en el flujo de bits recibido, se conmuta el tiempo de eliminación de la memoria intermedia de imágenes antes basándose en un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes (404).

25 **[0147]** En algunos ejemplos, la memoria intermedia de imágenes incluye una memoria intermedia de imágenes codificadas (CPB) y el desplazamiento de retardo de eliminación de la memoria intermedia de imágenes incluye un desplazamiento de retardo de eliminación de la CPB. En otros ejemplos, la memoria intermedia de imágenes incluye una memoria intermedia de imágenes decodificadas (DPB) y el desplazamiento de retardo de eliminación de la memoria intermedia de imágenes incluye un desplazamiento de retardo de eliminación de DPB.

30 **[0148]** El decodificador de vídeo 30 puede recibir el desplazamiento de retardo de eliminación de la memoria intermedia de imágenes para la unidad de acceso. En algunos ejemplos, el decodificador de vídeo 30 puede recibir el desplazamiento de retardo de eliminación de la memoria intermedia de imágenes en un mensaje SEI del período de almacenamiento en memoria intermedia. En algunos ejemplos, la imagen RAP incluye una imagen de acceso aleatorio limpio (CRA) y una imagen de acceso de enlace roto (BLA).

35 **[0149]** La FIG. 5 es un diagrama de flujo que ilustra un procedimiento de ejemplo de acuerdo con uno o más ejemplos descritos en esta divulgación. El dispositivo de destino 14 puede implementar un procedimiento de procesamiento de datos de vídeo. Por ejemplo, el decodificador de vídeo 30 puede recibir una unidad de acceso, por ejemplo, después de una inicialización anterior, teniendo la unidad de acceso una imagen de punto de acceso aleatorio (RAP). Esto puede ser, por ejemplo, después de una unidad de acceso RAP que inicialice el HRD. Las unidades de acceso asociadas pueden contener imágenes principales omitidas de acceso aleatorio (RASL) no recibidas (500). El decodificador de vídeo 30 puede inicializar un tiempo de eliminación de la memoria intermedia de imágenes y un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes en respuesta a la recepción de la unidad de acceso y a la no recepción de las unidades de acceso asociadas que contienen imágenes RASL (502).

45 **[0150]** El tiempo de eliminación de la memoria intermedia de imágenes incluye un tiempo de eliminación de la memoria intermedia de imágenes codificadas (CPB) y el desplazamiento de retardo de eliminación de la memoria intermedia de imágenes incluye un desplazamiento de retardo de eliminación de la CPB. El tiempo de eliminación de la memoria intermedia de imágenes incluye un tiempo de eliminación de la memoria intermedia de imágenes decodificadas (DPB) y el desplazamiento de retardo de eliminación de la memoria intermedia de imágenes incluye un desplazamiento de retardo de eliminación de DPB. En un ejemplo, el decodificador de vídeo 30 puede recibir el desplazamiento de retardo de eliminación de la memoria intermedia de imágenes en un mensaje SEI del período de almacenamiento en memoria intermedia.

55 **[0151]** En un ejemplo, la imagen RAP es una imagen de acceso aleatorio limpio (CRA). En otro ejemplo, la imagen RAP es una imagen de acceso de enlace roto (BLA) que puede tener imágenes RASL asociadas o imágenes decodificables omitidas de acceso aleatorio (RADL). La imagen RAP es una de una imagen de acceso de enlace roto (BLA) que no tiene imágenes RASL asociadas, pero puede tener imágenes RADL y una imagen BLA que no tenga

imágenes iniciales asociadas. Un ejemplo puede incluir además designar la unidad de acceso recibida como la primera unidad de acceso en un flujo de bits.

5 **[0152]** En otro procedimiento de ejemplo para procesar datos de vídeo, un codificador de vídeo tal como el codificador de vídeo 20 o el decodificador de vídeo 30 puede señalar un desplazamiento de retardo de eliminación de la CPB para cada unidad de acceso CRA o BLA. El codificador de vídeo también puede conmutar el tiempo de eliminación de CBP de cada una de las unidades de acceso que siguen a la unidad de acceso CRA o BLA en el orden de decodificación anterior por el desplazamiento de retardo de eliminación de la CPB para cada unidad de acceso CRA o BLA para la cual no están presentes las imágenes RASL asociadas independientemente de si el HRD se
10 inicializa en la unidad de acceso CRA o BLA.

[0153] En otro procedimiento de ejemplo de procesamiento de datos de vídeo, el codificador de vídeo puede indicar un desplazamiento de retardo de eliminación de la CPB para cada unidad de acceso CRA o BLA. El codificador de vídeo también puede seleccionar el retardo inicial de eliminación de la CPB predeterminado y el desplazamiento de retardo para la unidad de acceso 0 de modo que el retardo de eliminación de la CPB inicial alternativo y el desplazamiento de retardo no se señalen para las unidades de acceso CRA o BLA a menos que estén presentes los parámetros de la CPB de nivel de subimágenes. En un ejemplo, el retardo de eliminación de la CPB inicial alternativo y el desplazamiento de retardo pueden ser para operaciones HRD de nivel de subimágenes.

20 **[0154]** En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o en cualquier combinación de los mismos. Si se implementan en software, las funciones, como una o más instrucciones o código, se pueden almacenar en, o transmitir 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, que correspondan a un medio tangible tal como medios de almacenamiento de datos o medios de comunicación que incluyan cualquier medio que facilite la transferencia de un programa informático desde un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder en general a (1) medios de almacenamiento tangibles 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 cualquier medio disponible al 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.

[0155] A modo de ejemplo, y no de limitación, dichos 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 llamarse de manera apropiada medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde una página web, un servidor u otra fuente remota 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, se debería entender 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 un disco compacto (CD), un disco láser, un disco óptico, un disco versátil digital (DVD), un disco flexible y un disco Blu-ray, donde algunos discos reproducen usualmente los datos magnéticamente, mientras que otros discos reproducen los datos ópticamente con láseres. Las combinaciones de lo anterior también deberían incluirse dentro del alcance de los medios legibles por ordenador.

50 **[0156]** Las instrucciones pueden ejecutarse por uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de uso general, circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables por campo (FPGA) u otros circuitos lógicos integrados o discretos equivalentes. Por consiguiente, el término "procesador", como se usa en el presente documento, se puede referir a cualquiera de las estructuras anteriores 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 se puede proporcionar dentro de módulos de hardware y/o software dedicados, configurados para la codificación y la decodificación, o incorporados en un códec combinado. También, las técnicas se podrían implementar totalmente en uno o más circuitos o elementos lógicos.

60 **[0157]** Las técnicas de esta divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluidos un equipo manual inalámbrico, un circuito integrado (CI) o un conjunto de CI (por ejemplo, un conjunto de chips). Diversos 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 su realización mediante diferentes unidades de hardware. En cambio, como se ha descrito anteriormente, diversas unidades se pueden combinar en una unidad de hardware de códec o proporcionar mediante un grupo de unidades
65

de hardware interoperativas, incluyendo uno o más procesadores, como se ha descrito anteriormente, conjuntamente con software y/o firmware adecuados.

5 **[0158]** Se han descrito diversos ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un procedimiento para procesar datos de vídeo, comprendiendo el procedimiento: recibir, en un flujo de bits de vídeo, una unidad de acceso inicial que tenga una imagen de punto de acceso aleatorio (RAP), en donde un decodificador hipotético de referencia (HRD) se inicializa en la unidad de acceso inicial;

5 recibir una unidad de acceso subsiguiente que tenga una imagen RAP;

recibir, en un mensaje de información de mejora suplementaria (SEI) del período de almacenamiento en memoria intermedia, un desplazamiento de retardo de eliminación de la memoria intermedia para la unidad de acceso subsiguiente; y

10 conmutar el tiempo de eliminación de la memoria intermedia de imágenes para una memoria intermedia de imágenes antes, mediante el desplazamiento de retardo de eliminación de la memoria intermedia de imágenes, independientemente de si el HRD se inicializa en la unidad de acceso subsiguiente, si:

15 la unidad de acceso subsiguiente tiene un tipo de unidad de capa de abstracción de red (NAL) que indica una unidad de acceso del acceso de enlace roto (BLA) para la cual la imagen codificada no tiene imágenes principales omitidas de acceso aleatorio (RASL) presentes en el flujo de bits, pero puede tener imágenes principales decodificables de acceso aleatorio asociadas (RADL) o un tipo de unidad de capa de abstracción de red (NAL) que indique una unidad de acceso BLA para la cual la imagen codificada no tenga imágenes principales, y un valor de un elemento sintáctico indica que un retardo de eliminación de la memoria intermedia de imágenes codificadas (CPB) inicial alternativo y un desplazamiento de eliminación de la CPB inicial alternativo se especifican en el mensaje SEI del período de almacenamiento en memoria intermedia;

20 o

la unidad de acceso subsiguiente tiene un tipo de unidad NAL que indica una unidad de acceso BLA para la cual la imagen codificada puede tener imágenes RASL o RADL asociadas presentes en el flujo de bits, o que indica una unidad de acceso de acceso aleatorio limpio (CRA), un valor de un elemento sintáctico indica que un retardo de eliminación de la memoria intermedia de imágenes codificadas (CPB) inicial alternativo y un desplazamiento de eliminación de la CPB inicial alternativo se especifican en un mensaje SEI del periodo de almacenamiento en memoria intermedia, y un valor de una variable UseAltCpbParamsFlag para la unidad de acceso subsiguiente indica que se usan parámetros de la CPB alternativos.

30
2. El procedimiento según la reivindicación 1, que comprende además conmutar el tiempo de eliminación de la memoria intermedia de imágenes para la memoria intermedia de imágenes antes basándose en una acumulación de los respectivos desplazamientos de retardo de eliminación de la memoria intermedia de imágenes para las unidades de acceso, subsiguientes a la unidad de acceso inicial, que tienen imágenes RAP.

40
3. El procedimiento según la reivindicación 1, que comprende además conmutar el tiempo de eliminación de la memoria intermedia de imágenes para la memoria intermedia de imágenes antes basándose en los respectivos desplazamientos de retardo de eliminación de la memoria intermedia de imágenes tanto para la unidad de acceso subsiguiente que tiene el RAP como para la unidad de acceso inicial que tiene la imagen RAP que inicializa el HRD.

45
4. El procedimientos según la reivindicación 1, en donde el tiempo de eliminación de la memoria intermedia de imágenes comprende una memoria intermedia de imágenes decodificadas (DPB) y el desplazamiento de retardo de eliminación de la memoria intermedia de imágenes comprende un desplazamiento de retardo de eliminación de DPB.

50
5. Un dispositivo para procesar datos de vídeo que comprende:

55 medios para recibir, en un flujo de bits de vídeo, una unidad de acceso inicial que tenga una imagen de punto de acceso aleatorio (RAP), en donde un decodificador hipotético de referencia (HRD) se inicializa en la unidad de acceso inicial;

medios para recibir una unidad de acceso subsiguiente que tenga una imagen RAP;

60 medios para recibir, en un mensaje de información de mejora suplementaria (SEI) del período de almacenamiento en memoria intermedia, un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes para la unidad de acceso subsiguiente; y

medios para conmutar un tiempo de eliminación de la memoria intermedia de imágenes para una memoria intermedia de imágenes antes por un desplazamiento de retardo de eliminación de la memoria intermedia de imágenes, independientemente de si el HRD se inicializa en la unidad de acceso, si:

65

la unidad de acceso subsiguiente tiene un tipo de unidad de capa de abstracción de red (NAL) que indica una unidad de acceso del acceso de enlace roto (BLA) para la cual la imagen codificada no tiene imágenes principales omitidas de acceso aleatorio (RASL) asociadas presentes en el flujo de bits, pero puede tener imágenes principales decodificables de acceso aleatorio asociadas (RADL) asociadas o un tipo de unidad de capa de abstracción de red (NAL) que indique una unidad de acceso BLA para la cual la imagen codificada no tenga imágenes principales, y un valor de un elemento sintáctico indica que un retardo de eliminación de la memoria intermedia de imágenes codificadas (CPB) inicial alternativo y un desplazamiento de eliminación de la CPB inicial alternativo se especifican en el mensaje SEI del período de almacenamiento en memoria intermedia; o

la unidad de acceso subsiguiente tiene un tipo de unidad NAL que indica una unidad de acceso BLA para la cual la imagen codificada puede tener imágenes RASL o RADL asociadas presentes en el flujo de bits, o que indica una unidad de acceso de acceso aleatorio limpio (CRA), un valor de un elemento sintáctico indica que un retardo de eliminación de la memoria intermedia de imágenes codificadas (CPB) inicial alternativo y un desplazamiento de eliminación de la CPB inicial alternativo se especifican en un mensaje SEI del período de almacenamiento en memoria intermedia, y un valor de una variable UseAltCpbParamsFlag para la unidad de acceso subsiguiente indica que se usan parámetros de la CPB alternativos.

6. Un medio de almacenamiento no transitorio legible por ordenador que almacena instrucciones que, tras su ejecución por uno o más procesadores, causan que el uno o más procesadores realicen el procedimiento de una cualquiera de las reivindicaciones 1-4.

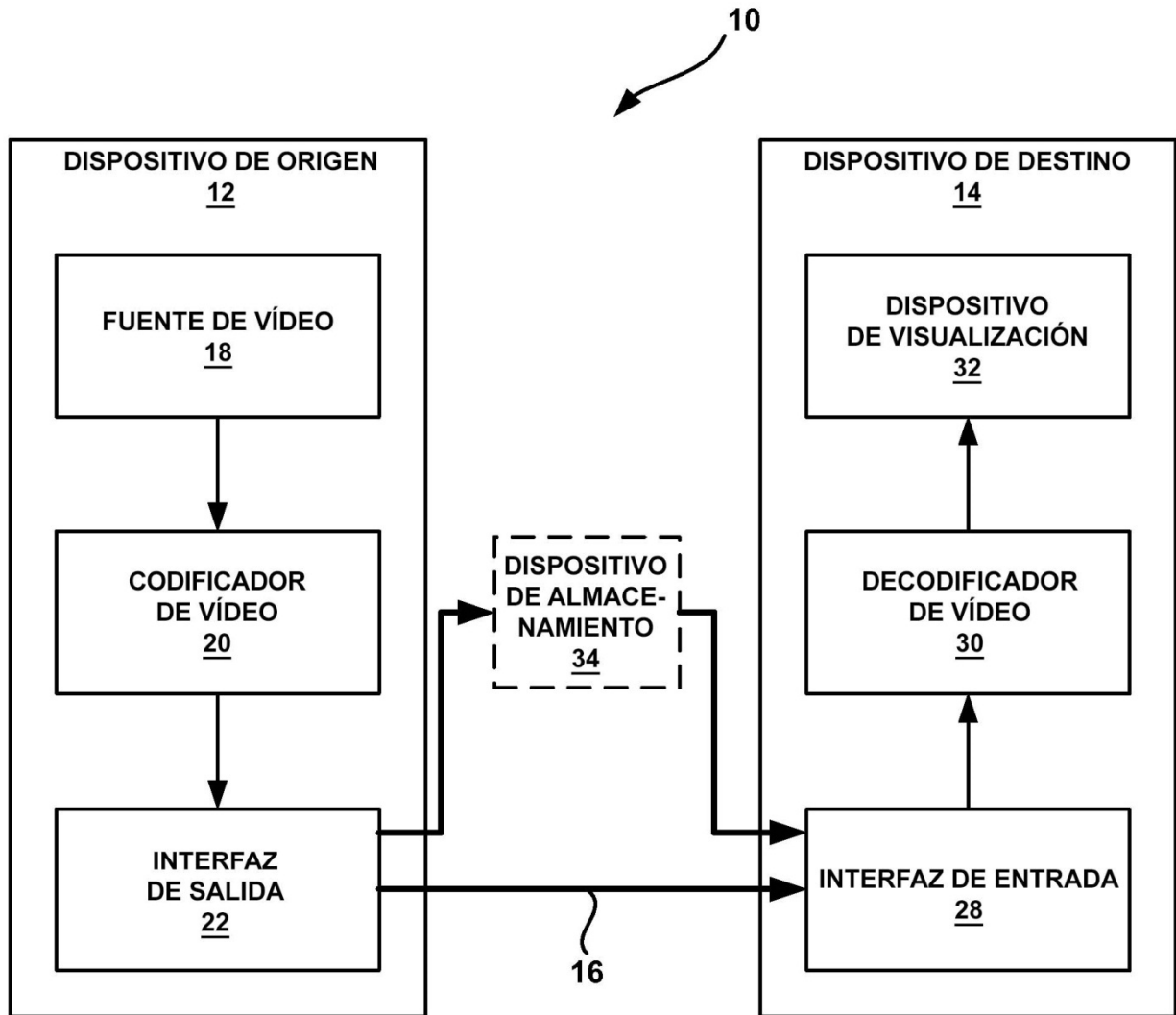


FIG. 1

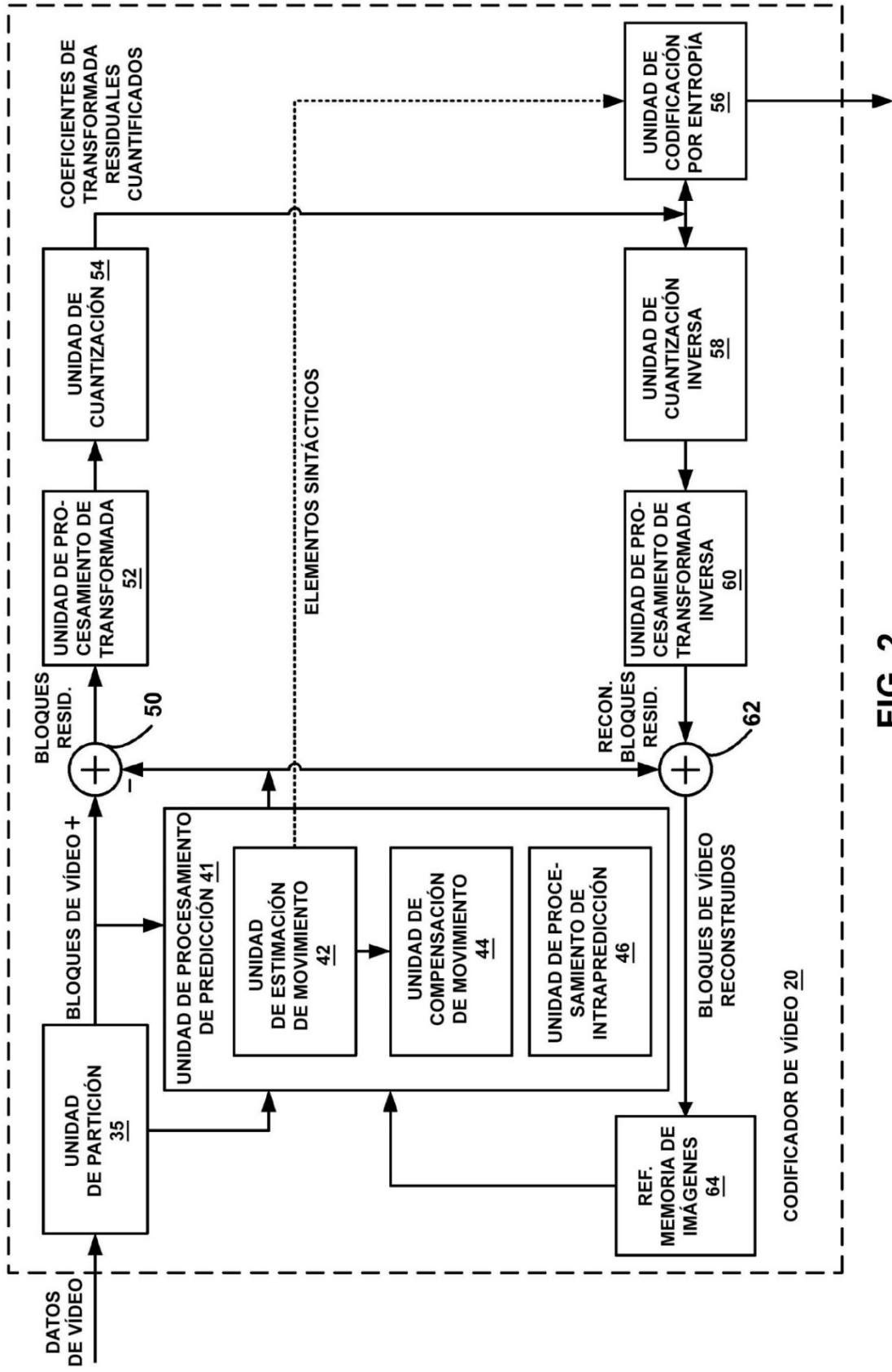


FIG. 2

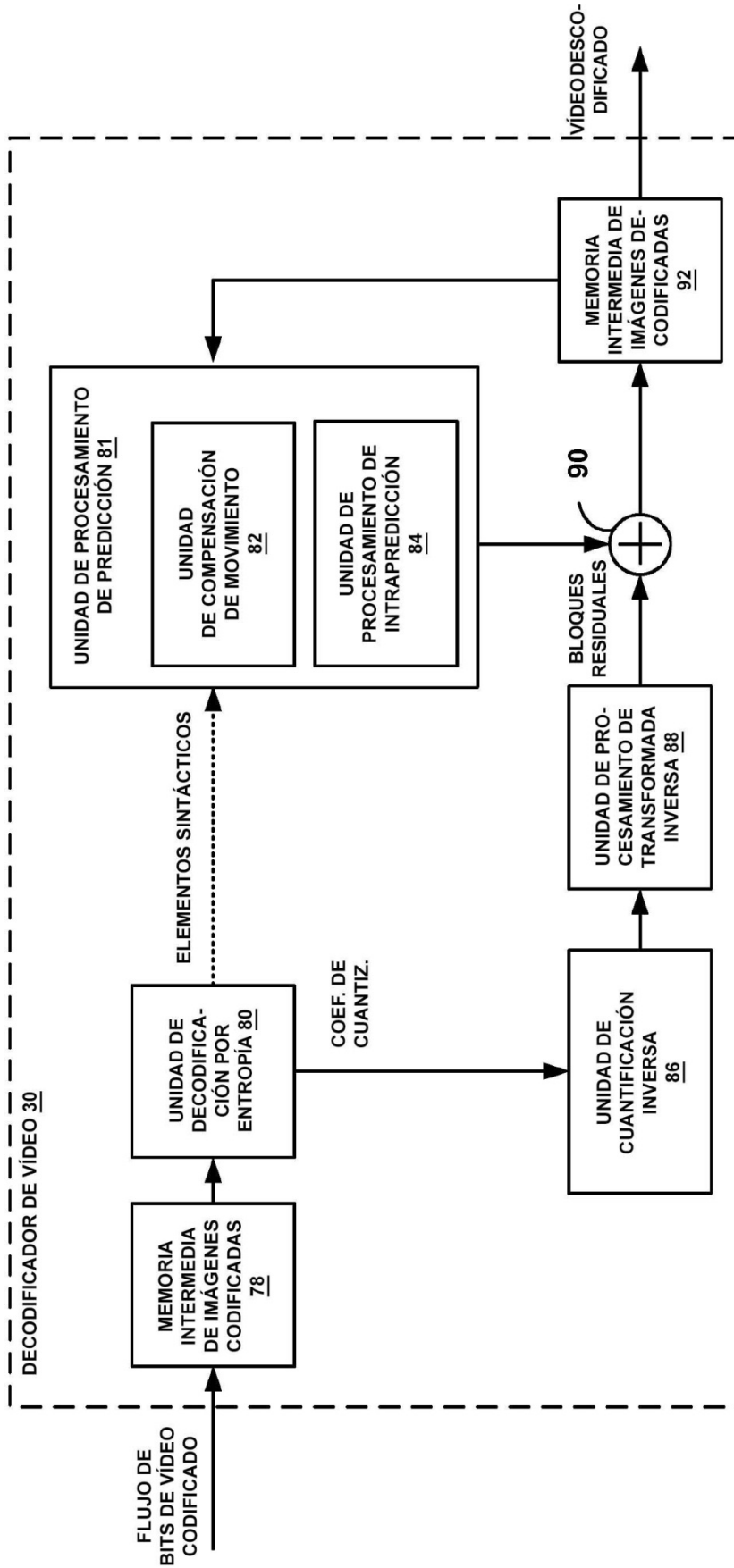


FIG. 3

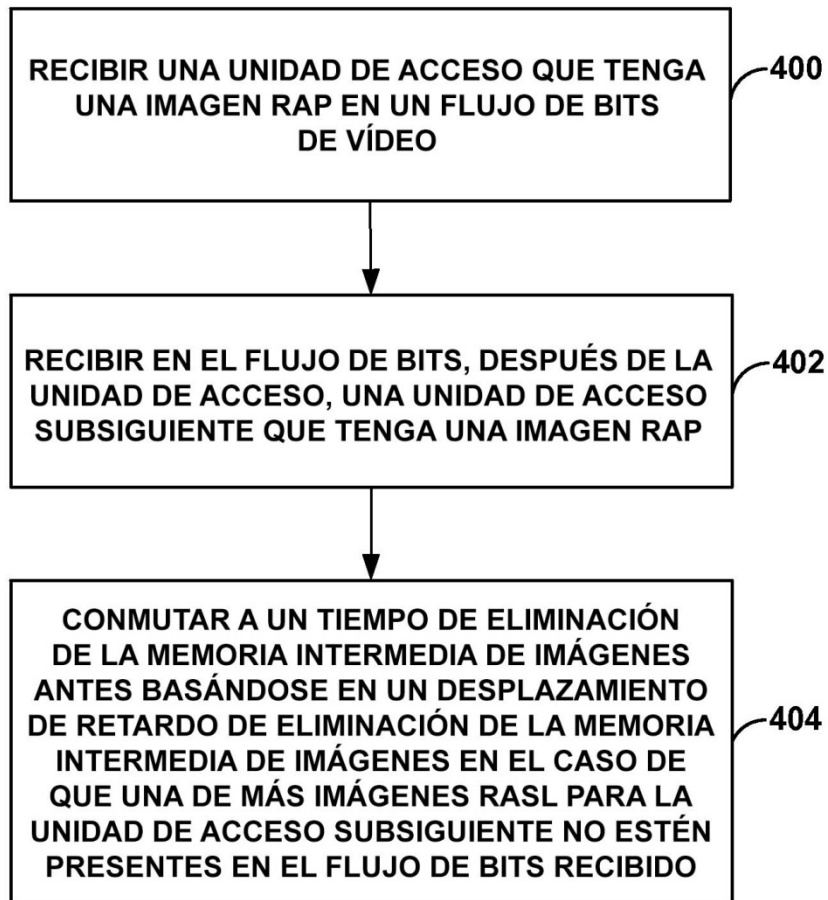


FIG. 4

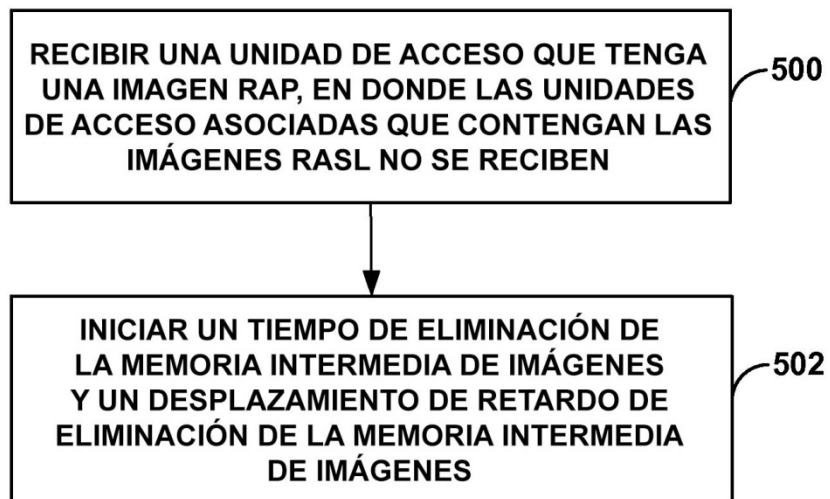


FIG. 5