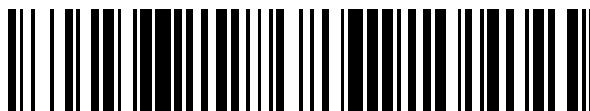


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 765 038**

51 Int. Cl.:

H04N 19/70	(2014.01)
H04N 19/115	(2014.01)
H04N 19/146	(2014.01)
H04N 19/174	(2014.01)
H04N 19/42	(2014.01)
H04N 19/433	(2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **10.12.2013 PCT/US2013/074104**

87 Fecha y número de publicación internacional: **26.06.2014 WO14099489**

96 Fecha de presentación y número de la solicitud europea: **10.12.2013 E 13818050 (0)**

97 Fecha y número de publicación de la concesión europea: **09.10.2019 EP 2936818**

54 Título: **Modelo de almacenamiento intermedio de bajo retardo en codificación de vídeo**

30 Prioridad:

19.12.2012 US 201261739632 P
21.12.2012 US 201261745423 P
25.09.2013 US 201314036615

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
05.06.2020

73 Titular/es:

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

72 Inventor/es:

WANG, YE-KUI

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 765 038 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

Modelo de almacenamiento intermedio de bajo retardo en codificación de vídeo

5 **CAMPO TÉCNICO**

[0001] La presente divulgación se refiere a la codificación de vídeo y, más particularmente, a técnicas para el almacenamiento intermedio de bajo retardo en un proceso de codificación de vídeo.

10 **ANTECEDENTES**

[0002] Las capacidades del vídeo digital pueden incorporarse a una amplia gama de dispositivos, incluyendo televisores digitales, sistemas de radiodifusión digital directa, sistemas de radiodifusión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, ordenadores de tableta, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos celulares o de radio por satélite, los denominados "teléfonos inteligentes", dispositivos de videoconferencia, dispositivos de transmisión continua de vídeo y similares. Los dispositivos de vídeo digital 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 eficacia (HEVC) actualmente en desarrollo y las ampliaciones de dichas normas. Los dispositivos de vídeo pueden transmitir, recibir, codificar, descodificar y/o almacenar información de vídeo digital con más eficacia implementando dichas técnicas de compresión de vídeo.

[0003] Las técnicas de compresión de vídeo llevan a cabo la predicción espacial (intraimagen) y/o la predicción temporal (interimagen) para reducir o eliminar la redundancia intrínseca 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 parte de una trama de vídeo) puede dividirse en bloques de vídeo, que también pueden denominarse 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 de un fragmento intercodificado (P o B) de una imagen pueden usar predicción espacial con respecto a unas muestras de referencia de bloques contiguos de la misma imagen, o predicción temporal con respecto a unas muestras de referencia de otras imágenes de referencia. Las imágenes pueden denominarse tramas y las imágenes de referencia pueden denominarse tramas de referencia.

[0004] La predicción espacial o temporal utiliza un bloque predictivo. Los datos residuales representan diferencias de píxeles entre el bloque original 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 de acuerdo con datos residuales. Para una mayor compresión, los datos residuales pueden transformarse desde el dominio del píxel a un dominio de transformada, dando como resultado coeficientes de transformada residuales, que a continuación se pueden cuantificar. Los coeficientes de transformada cuantificados, dispuestos inicialmente en una formación bidimensional, pueden explorarse con el fin de generar un vector unidimensional de coeficientes de transformada, y puede aplicarse codificación de entropía para lograr aún más compresión.

[0005] El documento de "BROSS B y otros: "High Efficiency Video Coding (HEVC) text specification draft 9 (SoDIS)", divulga el manejo de la memoria intermedia de imágenes descodificadas (DPB) HEVC y el cálculo del tiempo de salida de la imagen, que depende de si se calcula la eliminación de la memoria intermedia de imágenes codificadas (CPB) a nivel de subimagen o a nivel de imagen.

[0006] El documento "US 2010/246662 A1", divulga un sistema de codificación de vídeo, que usa un modelo de memoria intermedia virtual que puede funcionar tanto a nivel de imagen como a nivel de fragmento. Se divulga que la señalización de los valores de slice_dpb_output_delay a nivel de fragmento permite que la memoria intermedia de visualización de imágenes descodificadas funcione a nivel de fragmento, de modo que se logra la visualización de datos descodificados con bajo retardo.

RESUMEN

[0007] En general, esta divulgación describe las técnicas para la codificación de vídeo y, más particularmente, técnicas para el almacenamiento intermedio de bajo retardo en un proceso de codificación de vídeo. En uno o más ejemplos, esta divulgación propone técnicas para señalar retardos de salida de la memoria intermedia de imágenes descodificadas (DPB) para usar cuando un descodificador de vídeo está funcionando a un nivel de subimagen, para mejorar el retardo de la memoria intermedia de vídeo.

[0008] En un ejemplo de la divulgación, un procedimiento de descodificación de vídeo comprende recibir un primer retardo de salida de la memoria intermedia de imágenes descodificadas (DPB) y un segundo retardo de la salida de

la DPB para una imagen descodificada, determinar, para la imagen descodificada, un primer tiempo de salida de la DPB usando el primer retardo de salida de la DPB en el caso de que un ajuste del descodificador de referencia hipotético (HRD) para un descodificador de vídeo indique una operación a nivel de imagen, y determinar, para la imagen descodificada, un segundo tiempo de salida de la DPB usando el segundo retardo de salida de la DPB en el caso que el ajuste de HRD para el descodificador de vídeo indique una operación a nivel de subimagen.

[0009] En otro ejemplo de la divulgación, un procedimiento de codificación de vídeo comprende determinar un primer tiempo de salida de la DPB usando un primer retardo de la salida de la DPB en el caso de que un ajuste del HRD para un descodificador de vídeo indique una operación a nivel de imagen, determinar un segundo tiempo de salida de la DPB usando un segundo retardo de salida de la DPB en el caso de que el ajuste del HRD para el descodificador de vídeo indique una operación a nivel de subimagen, y señalar el primer retardo de salida de la memoria intermedia de imágenes descodificadas (DPB) y el segundo retardo de salida de la DPB.

[0010] En otro ejemplo de la divulgación, un aparato configurado para descodificar datos de vídeo comprende un descodificador de vídeo configurado para recibir un primer retardo de salida de la DPB y un segundo retardo de salida de la DPB para una imagen descodificada, determinar, para la imagen descodificada, un primer tiempo de salida de la DPB usando el primer retardo de salida de la DPB en el caso de que un ajuste del HRD para un descodificador de vídeo indique una operación a nivel de imagen, y determinar, para la imagen descodificada, un segundo tiempo de salida de la DPB usando el segundo retardo de salida de la DPB en el caso de que el ajuste del HRD para el descodificador de vídeo indique una operación a nivel de subimagen.

[0011] En otro ejemplo de la divulgación, un aparato configurado para codificar datos de vídeo comprende un codificador de vídeo configurado para determinar un primer tiempo de salida de la DPB usando un primer retardo de salida de la DPB en el caso de que un ajuste del HRD para un descodificador de vídeo indique una operación a nivel de imagen, determinar un segundo tiempo de salida de la DPB usando un segundo retardo de salida de la DPB en el caso de que el ajuste del HRD para el descodificador de vídeo indique una operación a nivel de subimagen, y señalar el primer retardo de salida de la DPB y el segundo retardo de salida de la DPB.

[0012] En otro ejemplo de la divulgación, un aparato configurado para descodificar datos de vídeo comprende medios para recibir un primer retardo de salida de la DPB y un segundo retardo de salida de la DPB para una imagen descodificada, medios para determinar, para la imagen descodificada, un primer tiempo de salida de la DPB usando el primer retardo de salida de la DPB en el caso de que un ajuste del HRD para un descodificador de vídeo indique una operación a nivel de imagen, y medios para determinar, para la imagen descodificada, un segundo tiempo de salida de la DPB usando el segundo retardo de salida de la DPB en el caso de que el ajuste del HRD para el descodificador de vídeo indique una operación a nivel de subimagen.

[0013] En otro ejemplo de la divulgación, un aparato configurado para codificar datos de vídeo comprende medios para determinar un primer tiempo de salida de la DPB usando un primer retardo de salida de la DPB en el caso de que un ajuste del HRD para un descodificador de vídeo indique una operación a nivel de imagen, medios para determinar un segundo tiempo de salida de la DPB usando un segundo retardo de salida de la DPB en el caso de que el ajuste del HRD para el descodificador de vídeo indique una operación a nivel de subimagen, medios para señalar el primer retardo de salida de la DPB y el segundo retardo de salida de la DPB.

[0014] En otro ejemplo, esta divulgación divulga un medio de almacenamiento legible por ordenador que almacena instrucciones que, al ejecutarse, hacen que uno o más procesadores de un dispositivo configurado para descodificar los datos de vídeo reciban un primer retardo de salida de la DPB y un segundo retardo de salida de la DPB para una imagen descodificada, determinen, para la imagen descodificada, un primer tiempo de salida de la DPB utilizando el primer retardo de salida de la DPB en el caso de que un ajuste del HRD para un descodificador de vídeo indique una operación a nivel de imagen, y determinen, para la imagen descodificada, un segundo tiempo de salida de la DPB usando el segundo retardo de salida de la DPB en el caso de que el ajuste del HRD para el descodificador de vídeo indique una operación a nivel de subimagen.

[0015] En otro ejemplo, esta divulgación divulga un medio de almacenamiento legible por ordenador que almacena instrucciones que, al ejecutarse, hacen que uno o más procesadores de un dispositivo configurado para codificar datos de vídeo determinen un primer tiempo de salida de la DPB usando un primer retardo de salida de la DPB en el caso de que un ajuste del HRD para un descodificador de vídeo indique una operación a nivel de imagen, determinen un segundo tiempo de salida de la DPB usando un segundo retardo de salida de la DPB en el caso de que el ajuste del HRD para el descodificador de vídeo indique una operación a nivel de subimagen, y señalicen el primer retardo de salida de la DPB y el segundo retardo de salida de la DPB.

[0016] Los detalles de uno o más ejemplos se exponen en los dibujos adjuntos y en la descripción siguiente. La invención se expone en las reivindicaciones adjuntas.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0017]

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

5 La FIG. 2 es un diagrama de bloques que ilustra un modelo de memoria intermedia para un descodificador de referencia hipotético (HRD).

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

10 La FIG. 4 es un diagrama de bloques que ilustra un descodificador de vídeo de ejemplo que puede implementar las técnicas descritas en esta divulgación.

15 La FIG. 5 es un diagrama de flujo que muestra un procedimiento de codificación a modo de ejemplo de acuerdo con las técnicas de la presente divulgación.

La FIG. 6 es un diagrama de flujo que muestra un procedimiento de descodificación de ejemplo de acuerdo con las técnicas de la presente divulgación.

20 DESCRIPCIÓN DETALLADA

[0018] Esta divulgación describe diversos procedimientos y técnicas para lograr un retardo de codec (codificador/descodificador) reducido de una forma interoperable, a través de un modelo de descodificador de referencia hipotético (HRD) genérico basado en subimágenes que incluye tanto operaciones de memoria intermedia de imágenes codificadas (CPB) basadas en subimágenes como operaciones de memoria intermedia de imágenes descodificadas (DPB) basadas en temporización de subimágenes.

[0019] Los enfoques actuales para minimizar el tiempo de retardo de la DPB y/o CPB presenta los siguientes inconvenientes. El tiempo de salida de una imagen descodificada es igual al tiempo de descodificación (es decir, el tiempo de eliminación de la CPB) de la última unidad de descodificación (es decir, la propia unidad de acceso para la operación de acceso a nivel de unidad) más el retardo de salida de la DPB señalado. Por lo tanto, se utilizan, en general, dos enfoques para reducir el retardo. Uno es adelantar el tiempo de descodificación. El otro es reducir el valor del retardo de salida de la DPB señalado (en relación con el tiempo de eliminación de la CPB). Sin embargo, las soluciones existentes para un modelo de memoria intermedia de retardo ultrabajo solo implican operaciones de la CPB basadas en subimágenes y solo aprovechan el primer enfoque para reducir el retardo.

[0020] En vista de estos inconvenientes, la presente divulgación propone técnicas para reducir más el retardo de descodificación a través de la señalización y el uso de valores reducidos del retardo de salida de la DPB señalado en relación con el tiempo de eliminación de la CPB.

[0021] La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación y descodificación 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 descodificar en un momento posterior. El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender cualquiera entre una amplia gama de dispositivos, incluidos ordenadores de escritorio, ordenadores plegables (es decir, portátiles), ordenadores de tableta, descodificadores, equipos telefónicos de mano tales como los denominados teléfonos "inteligentes", los denominados paneles "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, dispositivos de transmisión continua 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.

[0022] El dispositivo de destino 14 puede recibir los datos de vídeo codificados que se van a descodificar, por medio de un enlace 16. El enlace 16 puede comprender cualquier tipo de medio o dispositivo capaz de transferir 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 que el dispositivo de origen 12 transmita datos de vídeo codificados directamente a un 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 inalámbrico o cableado, tal como un espectro de radiofrecuencia (RF) o una o más líneas físicas de transmisión. El medio de comunicación puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área extensa o una red global tal como Internet. El medio de comunicación puede incluir encaminadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 hasta el dispositivo de destino 14.

[0023] De forma alternativa, los datos codificados se pueden transmitir desde la interfaz de salida 22 a un dispositivo de almacenamiento 33. De forma similar, una interfaz de entrada puede acceder a los datos codificados del dispositivo

de almacenamiento 33. El dispositivo de almacenamiento 33 puede incluir cualquiera de una variedad de medios de almacenamiento de datos de acceso distribuido o local, tales como una unidad de disco duro, discos Blu-ray, DVD, 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 33 puede corresponder a un servidor de archivos o a otro dispositivo de almacenamiento intermedio que puede contener el vídeo codificado generado por el dispositivo de origen 12. El dispositivo de destino 14 puede acceder a datos de vídeo almacenados en el dispositivo de almacenamiento 33 a través de transmisión en tiempo real o 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. Servidores de archivos de ejemplo incluyen un servidor web (por ejemplo, para un sitio web), un servidor FTP, dispositivos de almacenamiento conectados 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, incluida una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión Wi-Fi), una conexión cableada (por ejemplo, DSL, módem por cable, etc.) o una combinación de ambas 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 33 puede ser una transmisión continua, una transmisión de descarga o una combinación de ambas cosas.

[0024] Las técnicas de la presente divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas se pueden aplicar a la codificación de vídeo como apoyo a cualquiera de una variedad de aplicaciones multimedia, tales como radiodifusiones de televisión por aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones continuas de vídeo, por ejemplo, a través de Internet, codificación de vídeo digital para su almacenamiento en un medio de almacenamiento de datos, descodificación de vídeo digital almacenado en un medio de almacenamiento de datos, u otras aplicaciones. En algunos ejemplos, el sistema 10 puede estar configurado para prestar soporte a una transmisión de vídeo unidireccional o bidireccional, a fin de prestar soporte a aplicaciones tales como la transmisión continua de vídeo, la reproducción de vídeo, la difusión de vídeo y/o la videotelefonía.

[0025] 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/desmodulador (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 captación de vídeo, por ejemplo, una videocámara, un archivo de vídeo que contiene vídeo captado previamente, una interfaz de alimentación de vídeo para recibir vídeo desde un proveedor de contenido de vídeo y/o un sistema de gráficos por ordenador para generar datos de gráficos por 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 cableadas.

[0026] El vídeo capturado, precapturado o generado por ordenador se puede codificar mediante 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 de forma adicional (o alternativa) en el dispositivo de almacenamiento 33 para un posterior acceso por el dispositivo de destino 14 u otros dispositivos, para su descodificación y/o reproducción.

[0027] El dispositivo de destino 14 incluye una interfaz de entrada 28, un descodificador 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 a través del enlace 16. Los datos de vídeo codificados transmitidos a través del enlace 16, o proporcionados en el dispositivo de almacenamiento 33, pueden incluir una diversidad de elementos sintácticos generados por el codificador de vídeo 20 para su uso por un descodificador de vídeo, tal como el descodificador de vídeo 30, en la descodificación 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.

[0028] El dispositivo de visualización 32 se puede integrar con, 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 descodificados 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.

[0029] El codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con una norma de compresión de vídeo, tal como la norma de codificación de vídeo de alta eficacia (HEVC), actualmente en fase de desarrollo, y se pueden ajustar al modelo de prueba de la HEVC (HM). La HEVC está siendo desarrollada por el Equipo de Colaboración Conjunta sobre Codificación de Vídeo (JCT-VC) del Grupo de Expertos en Codificación de Vídeo (VCEG) de ITU-T y el Grupo de Expertos en Imágenes en Movimiento (MPEG) de la ISO/IEC. Un borrador de

trabajo (WD) de HEVC, Bross y otros, "High Efficiency Video Coding (HEVC) text specification draft 9", y denominado WD9 de HEVC en adelante, está disponible desde el 5 de julio de 2013 en http://phenix.int-evry.fr/jct/doc_end_user/documents/11_Shanghai/wg11/JCTVC-K1003-v13.zip.

5 **[0030]** De forma alternativa, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con otras normas patentadas o industriales, tales como la norma ITU-T H.264, denominada de forma alternativa 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.

10 **[0031]** Aunque no se muestra en la FIG. 1, en algunos aspectos, el codificador de vídeo 20 y el descodificador de vídeo 30 se pueden integrar, cada uno, en un codificador y descodificador de audio, y pueden incluir unidades MUX-DEMUX apropiadas, u otro hardware y software, para tratar la codificación tanto de audio como de vídeo en un flujo de datos común o en flujos de datos separados. Si procede, en algunos ejemplos, las unidades MUX-DEMUX pueden ajustarse al protocolo de multiplexor ITU H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

15 **[0032]** El codificador de vídeo 20 y el descodificador de vídeo 30 se pueden implementar, 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 in situ (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. 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 usando uno o más procesadores para realizar las técnicas de esta divulgación. Tanto el codificador de vídeo 20 como el descodificador de vídeo 30 se pueden incluir en uno o más codificadores o descodificadores, cualquiera de los cuales se puede integrar como parte de un codificador/descodificador (CÓDEC) combinado en un dispositivo respectivo.

20 **[0033]** El equipo JCT-VC está trabajando en el desarrollo del estándar HEVC. Los esfuerzos de normalización de la HEVC se basan en un modelo en evolución de un dispositivo de codificación de vídeo, denominado modelo de prueba de la HEVC (HM). El HM supone varias capacidades adicionales de los dispositivos de codificación de vídeo respecto a los 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.

25 **[0034]** 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 dividirse 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, tal 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, tal como un nodo hoja del árbol cuaternario, comprende 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.

30 **[0035]** Una CU incluye un nodo de codificación y unidades de predicción (PU) y unidades de transformada (TU) asociadas al 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 división de la CU en una o más PU. Los modos de división pueden diferir entre si la CU está codificada en modo directo o de salto, codificada en modo de intrapredicción o codificada en modo de interpretación. Las PU pueden dividirse para tener una forma no cuadrada. Los datos sintácticos asociados a una CU también pueden describir, por ejemplo, la división de la CU en una o más TU de acuerdo con un árbol cuaternario. Una TU puede tener una forma cuadrada o no cuadrada.

35 **[0036]** La norma HEVC permite 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 dentro de una CU dada definida para una LCU dividida, 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 usando una estructura de árbol cuaternario conocida como "árbol cuaternario residual" (RQT). Los nodos hoja del RQT pueden denominarse unidades de transformada (TU). Los valores de diferencias de píxeles asociados a las TU pueden transformarse para generar coeficientes de transformada, que pueden cuantificarse.

65

- 5 **[0037]** En general, una PU incluye datos relacionados con el proceso de predicción. Por ejemplo, cuando la PU se codifica en intramodo, la PU puede incluir datos que describen un modo de intrapredicción para la PU. Como otro ejemplo, cuando la PU se codifica en intermodo, la PU puede incluir datos que definen un vector de movimiento para la PU. Los datos que definen el vector de movimiento para una PU pueden describir, por ejemplo, una componente horizontal del vector de movimiento, una 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 **[0038]** En general, se usa una TU para los procesos de transformada y de cuantificación. Una CU dada que tiene una o más PU también puede incluir una o más unidades de transformada (TU). Después de la predicción, el codificador 20 de vídeo 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, cuantificarse y explorarse usando las transformadas y otra información de transformada especificada en las TU, para producir coeficientes de transformada en serie 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 incluye un nodo de codificación y unas PU y TU.
- 15 **[0039]** 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, una cabecera de una o más de las imágenes, o en otras ubicaciones, que describen una serie de imágenes incluidas en el GOP. Cada fragmento de una imagen puede incluir datos sintácticos de fragmento que describen un modo de codificación para el fragmento respectivo. El codificador de vídeo 20 actúa típicamente en bloques de vídeo dentro de fragmentos de vídeo individuales a 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 **[0040]** Como ejemplo, el HM presta soporte a la predicción en diversos tamaños de PU. Si se supone que el tamaño de una CU particular es $2N \times 2N$, el HM admite la intrapredicción en tamaños de PU de $2N \times 2N$ o $N \times N$, así como la interpredicción en tamaños de PU simétricas de $2N \times 2N$, $2N \times N$, $N \times 2N$ o $N \times N$. El HM también admite 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 entre un 25 % y un 75 %. La parte de la CU correspondiente a la división del 25 % está indicada por una "n" seguida de una indicación de "arriba", "abajo", "izquierda" o "derecha". Así pues, por ejemplo, "2NxnU" se refiere a una CU de tamaño $2N \times 2N$ que está dividida horizontalmente, con una PU de tamaño $2N \times 0,5N$ encima y una PU de tamaño $2N \times 1,5N$ debajo.
- 25 **[0041]** En esta divulgación, "NxN" 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 tamaño NxN 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 que en la dirección vertical. Por ejemplo, los bloques pueden comprender NxM píxeles, donde M no es necesariamente igual a N.
- 30 **[0042]** Tras la codificación intrapredictiva o interpredictiva, usando las PU de una CU, el codificador 20 de vídeo 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 20 de vídeo puede formar los datos residuales para la CU, y a continuación transformar los datos residuales para producir coeficientes de transformada.
- 35 **[0043]** Tras cualquier transformada para generar coeficientes de transformada, el codificador de vídeo 20 puede realizar la cuantificación de los coeficientes de transformada. La cuantificación se refiere, en general, a un proceso en el que unos coeficientes de transformada se cuantifican para reducir posiblemente la cantidad de datos usados para representar los coeficientes, proporcionando una compresión adicional. El proceso de cuantificación 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 a la baja hasta un valor de m bits durante la cuantificación, donde n es mayor que m .
- 40 **[0044]** En algunos ejemplos, el codificador de vídeo 20 puede usar un orden de exploración predefinido para explorar los coeficientes de transformada cuantificados, para generar un vector en serie que pueda ser codificado por entropía. En otros ejemplos, el codificador de vídeo 20 puede realizar una exploración adaptativa. Después de explorar los coeficientes de transformada cuantificados para formar un vector unidimensional, el codificador de vídeo 20 puede realizar la codificación por entropía del vector unidimensional, por ejemplo, de acuerdo con la codificación de longitud
- 45 **[0041]** En esta divulgación, "NxN" 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 tamaño NxN 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 que en la dirección vertical. Por ejemplo, los bloques pueden comprender NxM píxeles, donde M no es necesariamente igual a N.
- 50 **[0042]** Tras la codificación intrapredictiva o interpredictiva, usando las PU de una CU, el codificador 20 de vídeo 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 20 de vídeo puede formar los datos residuales para la CU, y a continuación transformar los datos residuales para producir coeficientes de transformada.
- 55 **[0043]** Tras cualquier transformada para generar coeficientes de transformada, el codificador de vídeo 20 puede realizar la cuantificación de los coeficientes de transformada. La cuantificación se refiere, en general, a un proceso en el que unos coeficientes de transformada se cuantifican para reducir posiblemente la cantidad de datos usados para representar los coeficientes, proporcionando una compresión adicional. El proceso de cuantificación 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 a la baja hasta un valor de m bits durante la cuantificación, donde n es mayor que m .
- 60 **[0044]** En algunos ejemplos, el codificador de vídeo 20 puede usar un orden de exploración predefinido para explorar los coeficientes de transformada cuantificados, para generar un vector en serie que pueda ser codificado por entropía. En otros ejemplos, el codificador de vídeo 20 puede realizar una exploración adaptativa. Después de explorar los coeficientes de transformada cuantificados para formar un vector unidimensional, el codificador de vídeo 20 puede realizar la codificación por entropía del vector unidimensional, por ejemplo, de acuerdo con la codificación de longitud
- 65

variable adaptativa al contexto (CAVLC), la codificación aritmética binaria adaptativa al contexto (CABAC), la codificación aritmética binaria adaptativa al contexto basada en la sintaxis (SBAC), la codificación por entropía de división de intervalos de probabilidad (PIPE) u otra metodología de codificación por 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 descodificador de vídeo 30 en la descodificación de los datos de vídeo.

[0045] Para realizar la CABAC, el codificador de vídeo 20 puede asignar un contexto dentro de un modelo contextual a un símbolo que se va 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 codificación CAVLC, el codificador de vídeo 20 puede seleccionar un código de longitud variable para un símbolo a transmitir. Las palabras de código en la VLC 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 palabras de código de igual longitud para cada símbolo que se va a transmitir. La determinación de la probabilidad se puede basar en un contexto asignado al símbolo.

[0046] Las aplicaciones de vídeo que pueden hacer uso del codificador de vídeo 20 y el descodificador de vídeo 30 pueden incluir la reproducción local, la reproducción continua, la radiodifusión/multidifusión y aplicaciones conversacionales. Las aplicaciones conversacionales incluyen videotelefonía y videoconferencia. Las aplicaciones conversacionales también se conocen como aplicaciones de bajo retardo, ya que dichas aplicaciones en tiempo real no son tolerantes a retardos significativos. Para una buena experiencia de usuario, las aplicaciones conversacionales requieren un retardo relativamente bajo de extremo a extremo de la totalidad de los sistemas, es decir, el retardo entre el momento en que se captura una trama de vídeo en un dispositivo de origen y el momento en que se muestra la trama de vídeo en un dispositivo de destino. Típicamente, un retardo de extremo a extremo aceptable para las aplicaciones conversacionales debe ser inferior a 400 ms. Un retardo de extremo a extremo de alrededor de 150 ms se considera muy bueno.

[0047] Cada etapa de procesamiento de una aplicación conversacional puede contribuir al retardo global de extremo a extremo. Retardos de ejemplo de las etapas de procesamiento incluyen retardo de captura, retardo de preprocesamiento, retardo de codificación, retardo de transmisión, retardo de almacenamiento en memoria intermedia de recepción (para antifluctuación), retardo de descodificación, retardo de la salida de imagen descodificada, retardo postprocesamiento y retardo de visualización. Típicamente, el retardo de códec (retardo de codificación, retardo de descodificación y retardo de salida de la imagen descodificada) debe reducirse al mínimo en las aplicaciones conversacionales. En particular, la estructura de codificación debe garantizar que el orden de descodificación de las imágenes y el orden de salida son idénticos, de tal forma que el retardo de salida de la imagen descodificada es igual o cercano a cero.

[0048] Las normas de codificación de vídeo típicamente incluyen una especificación 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 descodificador hipotético de referencia (HRD), que incluye un modelo de almacenamiento en memoria intermedia de tanto la memoria intermedia de imágenes codificadas (CPB) como la memoria intermedia de imágenes descodificadas (DPB). Una CPB es una memoria intermedia de primero en entrar, primero en salir que contiene imágenes codificadas para descodificar. Una DPB es una memoria intermedia que contiene imágenes descodificadas para su uso en referencia (por ejemplo, interpretaciones), reordenamiento de salida, retardo de salida y visualización final. Los comportamientos de la CPB y de la DPB son especificados matemáticamente por el HRD. El HRD impone directamente restricciones sobre temporización, tamaños de almacenamiento en memoria intermedia y velocidades de bits diferentes, e impone restricciones indirectamente sobre características y estadísticas de los flujos de bits. Un conjunto completo de parámetros del HRD incluye cinco parámetros básicos: el retardo inicial de eliminación de la CPB, el tamaño de la CPB, la velocidad de transmisión de bits, el retardo inicial de salida de la DPB y el tamaño de la DPB.

[0049] En AVC y HEVC, la conformidad del flujo de bits y la conformidad del descodificador se especifican como partes de la especificación del HRD. Aunque el HRD se denomina como un descodificador, algunas técnicas especificadas por el HRD también son necesarias típicamente en el lado del codificador para garantizar la conformidad del flujo de bits, mientras que no es necesario típicamente en el lado del descodificador. 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 descodificador (es decir, la conformidad de descodificador de la temporización de salida y la conformidad de descodificador del orden de salida).

[0050] El flujo de bits Tipo I es un flujo de unidades de capa de abstracción de red (NAL) que contiene solo las unidades NAL de capa de codificación de vídeo (VCL) y las unidades NAL con `nal_unit_type` igual a `FD_NUT` (unidades NAL de datos de relleno) para todas las unidades de acceso en el flujo de bits. Un flujo de bits de Tipo II contiene, además de las unidades VCL NAL y las unidades NAL de datos de relleno para todas las unidades de acceso en el flujo de bits, al menos uno de los siguientes: unidades NAL no VCL adicionales distintas de las unidades NAL de datos de relleno; todos los elementos sintácticos `leading_zero_8bits`, `zero_byte`, `start_code_prefix_one_3bytes` y `trailing_zero_8bits` que forman un flujo de bytes a partir del flujo de unidades NAL.

[0051] La FIG. 2 es un diagrama de bloques que ilustra un modelo de memoria intermedia para un descodificador de referencia hipotético (HRD). El HRD funciona de la forma siguiente. Los datos asociados con las unidades de descodificación que fluyen hacia la CPB 102 de acuerdo con un programa de llegada especificado son entregados por el planificador de flujo hipotético (HSS) 100. Los flujos entregados por el HSS 100 pueden ser flujos de bits Tipo I o Tipo II, como se definió anteriormente. Los datos asociados a cada unidad de descodificación se eliminan y descodifican mediante el proceso de descodificación 104 (por ejemplo, en el descodificador de vídeo 30) en el momento de eliminación de la CPB de la unidad de descodificación. El proceso de descodificación 104 es realizado por el descodificador de vídeo 30. Cada imagen descodificada producida por el proceso de descodificación 104 se coloca en la DPB 106. Las imágenes descodificadas se pueden usar como imágenes de referencia durante el proceso de descodificación 104 (por ejemplo, durante la interpredicción). Se elimina una imagen descodificada de la DPB 106 cuando ya no es necesaria para la referencia de interpredicción y ya no es necesaria para la salida. En algunos ejemplos, las imágenes descodificadas en la DPB 106 pueden recortarse mediante la unidad de recorte de salida 108 antes de mostrarse. La unidad de recorte de salida 108 puede ser parte del descodificador de vídeo 30 o puede ser parte de un procesador externo (por ejemplo, un procesador de visualización) configurado para procesar aún más la salida de un descodificador de vídeo.

[0052] En los modelos del HRD de la AVC y la HEVC, la descodificación o la eliminación de la CPB está basada en unidades de acceso (AU), y se supone que la descodificación de imágenes es instantánea (por ejemplo, el proceso de descodificación 104 en la FIG. 2 se supone que es instantáneo). Una unidad de acceso es un conjunto de unidades de capa abstracta de red (NAL) y contiene una imagen codificada. En aplicaciones prácticas, si un descodificador en conformidad sigue estrictamente los tiempos de descodificación señalizados, por ejemplo, en los mensajes de información de mejora complementaria (SEI) de temporización de imágenes generados por el codificador de vídeo 20, para iniciar la descodificación de las AU, entonces el momento más temprano posible para emitir una imagen descodificada específica es igual al momento de descodificación de esa imagen específica (es decir, el momento en el que una imagen comienza a ser descodificada) más el tiempo necesario para descodificar esa imagen específica. El tiempo necesario para descodificar una imagen en aplicaciones prácticas no puede ser igual a cero.

[0053] El WD9 de HEVC incluye el soporte de operaciones de la CPB basadas en subimágenes para permitir un retardo de códec reducido, a veces denominado retardo ultrabajo. La CPB puede funcionar a nivel de AU (es decir, a nivel de imagen) o a nivel de subimagen (es decir, menos de una imagen completa), dependiendo de si el descodificador prefiere una operación de la CPB a nivel de subimagen (que puede ser especificado por un medio externo no especificado en la especificación de HEVC) y si los parámetros de la CPB de la subimagen están presentes (en el flujo de bits o mediante medios externos no especificados en la especificación de HEVC). Cuando ambas condiciones son verdaderas, la CPB funciona a nivel de subimagen (y en este caso cada unidad de descodificación se define como un subconjunto de una AU). Una unidad de descodificación (DU) es la unidad en la que opera el descodificador. De lo contrario, la CPB funciona a nivel de AU (y en este caso cada unidad de descodificación se define como una AU). Una DU es igual a una AU si el elemento de sintaxis SubPicCpbFlag es igual a 0. De lo contrario, una DU es un subconjunto de una UA.

[0054] La sintaxis de HEVC para los parámetros de la CPB a nivel de subimagen incluye lo siguiente:

- La siguiente sintaxis se encuentra en la parte de información de usabilidad de vídeo (VUI) del conjunto de parámetros de secuencia (SPS)

- Si los parámetros de la CPB a nivel de subimagen están presentes

- Un divisor de ciclos, para la obtención del reloj de subciclos

- Duración del retardo de eliminación de la CPB

- Si los valores de retardo de eliminación de la CPB de la unidad de descodificación se señalizan en mensajes SEI de temporización de imagen o mensajes SEI de información de la unidad de descodificación

- Longitud de los valores de tamaño de la CPB para operaciones de la CPB a nivel de subimagen

- La siguiente sintaxis está en los mensajes SEI del período de almacenamiento en memoria intermedia

- Un conjunto de retardo inicial de eliminación de la CPB y el desplazamiento de retardo para operaciones de la CPB de nivel de subimagen

- La siguiente sintaxis está en los mensajes SEI de temporización de imagen

- El número de unidades de descodificación en una unidad de acceso

- El número de unidades NAL en cada unidad de descodificación

- Los valores de retardo de eliminación de la CPB de la unidad de descodificación para las unidades de descodificación

- La siguiente sintaxis está en los mensajes SEI de temporización de imagen

- El índice de cada unidad de descodificación a la lista de unidades de descodificación en una unidad de acceso
- El valor de retardo de eliminación de la CPB de la unidad de descodificación para cada unidad de descodificación

[0055] Los enfoques actuales para reducir al mínimo el tiempo de retardo de la CPB y/o la DPB presentan los siguientes inconvenientes. El tiempo de salida de una imagen descodificada es igual al tiempo de descodificación (es decir, el tiempo de eliminación de la CPB) de la última DU (es decir, la propia AU para operación a nivel de AU) más el retardo de salida de la DPB señalado. Por lo tanto, se utilizan, en general, dos enfoques para reducir el retardo. Uno es adelantar el tiempo de descodificación. El otro es reducir el valor del retardo de salida de la DPB señalado (en relación con el tiempo de eliminación de la CPB). Sin embargo, las soluciones existentes para un modelo de memoria intermedia de retardo ultrabajo solo implican operaciones de la CPB basadas en subimágenes y solo aprovechan el primer enfoque para reducir el retardo.

[0056] En vista de estos inconvenientes, la presente divulgación propone técnicas para reducir más el retardo de descodificación a través de la señalización y el uso de valores reducidos del retardo de salida de la DPB señalado en relación con el tiempo de eliminación de la CPB.

[0057] Específicamente, en un ejemplo de la divulgación, un valor señalado adicional del retardo de salida de la DPB en relación con el tiempo de eliminación de la CPB de cada AU es señalado por un codificador, por ejemplo, en el mensaje SEI de temporización de imagen. Este retardo de salida de la DPB señalado adicional se utiliza en la obtención del tiempo de salida de la DPB para operaciones del HRD basadas en subimágenes. En otro ejemplo, además del retardo de salida de la DPB señalado adicionalmente, los tiempos de salida de la DPB se obtienen usando un reloj de subciclos en lugar del reloj de ciclos.

[0058] Algunos ejemplos detallados se proporcionan a continuación. Si no se menciona específicamente, los aspectos de los siguientes ejemplos pueden funcionar como se define en el WD9 de HEVC.

[0059] Una sintaxis y la semántica del mensaje SEI de temporización de imagen de ejemplo, de acuerdo con un ejemplo de esta divulgación son las siguientes. Los elementos de sintaxis alterados o introducidos por esta divulgación se muestran en **negrita**.

	Descriptor
pic_timing(payloadSize) {	
if(frame_field_info_present_flag) {	
pic_struct	u(4)
progressive_source_idc	u(2)
duplicate_flag	u(1)
}	
au_cpb_removal_delay_minus1	u(v)
pic_dpb_output_delay	u(v)
if(sub_pic_cpb_params_present_flag)	
pic_dpb_output_du_delay	u(v)
if(sub_pic_cpb_params_present_flag && sub_pic_cpb_params_in_pic_timing_sei_flag) {	
num_decoding_units_minus1	ue(v)
du_common_cpb_removal_delay_flag	u(1)
if(du_common_cpb_removal_delay_flag)	
du_common_cpb_removal_delay_minus1	u(v)
for(i = 0; i <= num_decoding_units_minus1; i++) {	
num_nalus_in_du_minus1[i]	ue(v)
if(ldu_common_cpb_removal_delay_flag && i < num_decoding_units_minus1)	
du_cpb_removal_delay_minus1[i]	u(v)
}	
}	
}	
}	

[0060] En este ejemplo de la divulgación, los elementos de sintaxis en negrita pueden operar como sigue. El elemento de sintaxis **pic_dpb_output_du_delay** se usa para calcular el tiempo de salida de la DPB de la imagen cuando el HRD funciona a nivel de subimagen (es decir, cuando SubPicCpbFlag es igual a 1). El elemento de sintaxis **pic_dpb_output_du_delay** especifica el número de subciclos de reloj a esperar después de la eliminación de la última unidad de descodificación en una unidad de acceso de la CPB antes de que la imagen descodificada se transmita desde la DPB.

[0061] En un ejemplo, la longitud del elemento de sintaxis **pic_dpb_output_du_delay** viene dada en bits por $\text{dpb_output_delay_length_minus1} + 1$. En otro ejemplo, la longitud del elemento de sintaxis **pic_dpb_output_du_delay** se da en bits mediante el valor de otro elemento de sintaxis más 1, donde, por ejemplo, el elemento de sintaxis se denomina **dpb_output_delay_length_du_minus1** y se señala en la parte VUI del conjunto de parámetros de secuencia.

[0062] El tiempo de salida obtenido de **pic_dpb_output_du_delay** de cualquier imagen que se transmite desde un descodificador en conformidad con la temporización de salida precederá al tiempo de salida obtenido de **pic_dpb_output_du_delay** de todas las imágenes en cualquier secuencia de vídeo codificada posterior en el orden de descodificación. En un ejemplo, el orden de salida de imagen establecido por los valores de este elemento de sintaxis será el mismo orden que se establece por los valores de PicOrderCntVal, como se especifica por el WD9 de HEVC. El elemento de sintaxis PicOrderCntVal indica el recuento de orden de imagen (POC) de la imagen actual. Un valor POC es una variable que está asociada con cada imagen que transmitirá la DPB que indica la posición de la imagen asociada en orden de salida en relación con las posiciones del orden de salida de las otras imágenes que se transmitirán de la DPB en la misma secuencia de vídeo codificada.

[0063] Para las imágenes que no se transmiten mediante el proceso de "desplazamiento" (es decir, el proceso mediante el cual las imágenes son eliminadas de la DPB) porque preceden, en orden de descodificación, a una imagen de actualización de descodificación instantánea (IDR) o imagen de acceso de enlace roto (BLA) con **no_output_of_prior_pics_flag** igual a 1 o que se deduce como igual a 1, los tiempos de salida obtenidos de **pic_dpb_output_du_delay** aumentarán con el valor en aumento de PicOrderCntVal con respecto a todas las imágenes dentro de la misma secuencia de vídeo codificada. El elemento de sintaxis **no_output_of_prior_pics_flag** especifica cómo se tratan las imágenes previamente descodificadas en la DPB después de la descodificación de una imagen IDR o BLA. Si **no_output_of_prior_pics_flag** es igual o se deduce que es 1, después de descodificar una imagen IDR o BLA, esas imágenes descodificadas previamente no se emitirán, sino que se vaciarán/eliminarán directamente de la memoria intermedia de imágenes descodificadas (DPB).

[0064] El proceso de "desplazamiento" se invoca en los siguientes casos.

- La imagen actual es una imagen IDR o BLA y **no_output_of_prior_pics_flag** no es igual a 1 y no se deduce que sea igual a 1.
- La imagen actual no es una imagen IDR ni una imagen BLA, y la cantidad de imágenes en la DPB que están marcadas como "necesarias para transmitir" es mayor que la cantidad máxima de imágenes permitidas antes de cualquier imagen en orden de descodificación ($\text{sps_max_num_reorder_pics}[\text{HighestTid}]$).
- La imagen actual no es una imagen IDR ni una imagen BLA, y el número de imágenes en la DPB es igual al tamaño máximo requerido de la DPB en la unidad de memorias intermedias de almacenamiento de imágenes ($\text{sps_max_dec_pic_buffering}[\text{HighestTid}]$).

[0065] El proceso de "desplazamiento" puede incluir las siguientes etapas ordenadas:

1. La imagen que está primero para la salida puede seleccionarse como la única que tiene el valor más pequeño de PicOrderCntVal de todas las imágenes en la DPB marcada como "necesaria para transmitir".
2. La imagen se recorta, usando la ventana de recorte de conformidad especificada en el conjunto de parámetros de secuencia activa para la imagen, la imagen recortada es transmitida, y la imagen se marca como "no necesaria para transmitir".
3. Si la memoria intermedia de almacenamiento de imágenes que incluía la imagen que se recortó y se transmitió contiene una imagen marcada como "no usada para referencia", la memoria intermedia de almacenamiento de imágenes se puede vaciar. Es decir, si se ha transmitido una imagen para su visualización y ya no es necesaria para la interpredicción, puede ser "desplazada", es decir, eliminada de la DPB.

[0066] Para dos imágenes cualesquiera en la secuencia de vídeo codificada, la diferencia entre los tiempos de salida de las dos imágenes obtenidas en operaciones del HRD a nivel de subimagen será idéntica a la misma diferencia obtenida en operaciones del HDR a nivel de AU (es decir, a nivel de imagen).

[0067] Un proceso de salida de imagen de ejemplo es el siguiente. Lo siguiente puede suceder instantáneamente en el tiempo de eliminación de la CPB de la unidad de acceso n, $t_r(n)$. Según si una imagen n tiene o no PicOutputFlag igual a 1 (es decir, se utiliza el HRD de subimagen), su tiempo de salida de la DPB $t_{o,dpb}(n)$ se obtiene de la siguiente ecuación:

5

```

if( !SubPicCpbFlag ) // es decir, el HRD funciona a nivel de AU
     $t_{o,dpb}(n) = t_r(n) + t_c * pic\_dpb\_output\_delay(n)$ 
else // es decir, el HRD funciona a nivel de subimagen
     $t_{o,dpb}(n) = t_r(n) + t_{c\_sub} * pic\_dpb\_output\_du\_delay(n)$ 

```

donde $pic_dpb_output_delay(n)$ y $pic_dpb_output_du_delay(n)$ son los valores de $pic_dpb_output_delay$ y $pic_dpb_output_du_delay$, respectivamente, especificados en el mensaje SEI de temporización de imagen asociado con la unidad de acceso n. La variable t_c se obtiene de la siguiente manera y se denomina ciclo de reloj:

10

$$t_c = \text{num_units_in_tick} \div \text{time_scale}$$

La variable t_{c_sub} se puede obtener como se indica a continuación y puede denominarse ciclo de reloj de subimagen:

15

$$t_{c_sub} = t_c \div (\text{tick_divisor_minus2} + 2)$$

[0068] De acuerdo con un ejemplo de esta divulgación, la salida de la imagen actual se especifica como sigue:

20

- Si PicOutputFlag es igual a 1 y $t_{o,dpb}(n) = t_r(n)$, la imagen actual se transmite.
- De otro modo, si PicOutputFlag es igual a 0, la imagen actual no se transmite, sino que se almacenará en la DPB como se especifica mediante el proceso de "desplazamiento" expuesto anteriormente.
- De otro modo (PicOutputFlag es igual a 1 y $t_{o,dpb}(n) > t_r(n)$), la imagen actual se transmite más tarde y se almacenará en la DPB (como se especifica por el proceso de "desplazamiento") y se transmite en el tiempo $t_{o,dpb}(n)$, a menos que se indique que no se transmite por la descodificación u obtención de $no_output_of_prior_pics_flag$ igual a 1 en un tiempo que antecede a $t_{o,dpb}(n)$. Cuando se transmite, la imagen se recortará, utilizando la ventana de recorte de conformidad especificada en el conjunto de parámetros de secuencia activa.

25

30

[0069] Cuando la imagen n es una imagen que se transmite y no es la última imagen del flujo de bits que se transmite, el valor de $\Delta t_{o,dpb}(n)$ (es decir, el tiempo de salida de la DPB entre imágenes) se define como:

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

35

donde n_n indica la imagen que sigue a la imagen n en orden de salida y tiene PicOutputFlag igual a 1.

[0070] La FIG. 3 es un diagrama de bloques que ilustra un codificador de vídeo 20 de ejemplo que puede implementar las técnicas descritas en esta divulgación. El codificador de vídeo 20 puede realizar la intracodificación y la 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 de un 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 de un 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.

40

45

[0071] En el ejemplo de la FIG. 3, el codificador de vídeo 20 incluye la 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 cuantificación 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 cuantificación inversa 58, una unidad de procesamiento de transformada inversa 60 y un sumador 62. También se puede incluir un filtro de eliminación de bloques (no mostrado en la FIG. 3) para filtrar los límites de bloque, para eliminar distorsiones de efecto pixelado del vídeo reconstruido. Si se desea, el filtro de eliminación de bloques filtrará típicamente la salida del sumador 62. También se pueden usar filtros de bucle adicionales (en el bucle o tras el bucle), además del filtro de eliminación de bloques.

50

55

[0072] Como se muestra en la FIG. 3, el codificador de vídeo 20 recibe datos de vídeo, y la unidad de procesamiento de predicción 41 puede dividir los datos en bloques de vídeo. Esta división también puede incluir la división en fragmentos, mosaicos u otras unidades mayores, así como la división en 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 va 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 entre una pluralidad de modos de intercodificación, para el bloque de vídeo actual basándose en resultados de error (por ejemplo, la velocidad de codificación y el nivel de distorsión). La unidad de procesamiento de predicción 41 puede proporcionar el bloque intracodificado o intercodificado resultante al sumador 50 para generar datos de bloque residuales y al sumador 62 para reconstruir el bloque codificado para su uso como imagen de referencia.

[0073] 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 actual que va a codificarse, para proporcionar una 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 en una o más imágenes de referencia, para proporcionar compresión temporal.

[0074] La unidad de estimación del movimiento 42 se puede configurar para determinar el modo de interpredicción para un fragmento de vídeo de acuerdo con un patrón predeterminado para una secuencia de vídeo. El patrón predeterminado puede designar fragmentos de vídeo en la secuencia como fragmentos P, fragmentos B o fragmentos GPB. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar sumamente integradas, pero se ilustran por separado con fines conceptuales. La estimación del movimiento, realizada por la unidad de estimación de movimiento 42, es el proceso de generación de vectores de movimiento, que estiman el movimiento para los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una PU de un bloque de vídeo de una trama o imagen de vídeo actual en relación con un bloque predictivo de una imagen de referencia.

[0075] Un bloque predictivo es un bloque que se descubre que se corresponde estrechamente con la PU del bloque de vídeo que se va a codificar en términos de diferencia de píxeles, que puede determinarse mediante la suma de diferencias absolutas (SAD), la suma de las diferencias al cuadrado (SSD) u otras métricas de diferencia. En algunos ejemplos, el codificador de vídeo 20 puede calcular valores para posiciones fraccionarias de píxeles de imágenes de referencia almacenadas en la memoria de imágenes de referencia 64. Por ejemplo, el codificador de vídeo 20 puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones fraccionarias de píxel de la imagen de referencia. Por lo tanto, la unidad de estimación de movimiento 42 puede realizar una búsqueda de movimiento relativa a las posiciones de píxel completo y las posiciones fraccionarias de píxel, y proporcionar un vector de movimiento con una precisión de píxel fraccionaria.

[0076] 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 ser seleccionada a partir de una primera lista (lista 0) de imágenes de referencia o de una segunda lista (lista 1) de imágenes de referencia, cada una de las cuales identifica una o más imágenes de referencia almacenadas en una 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.

[0077] La compensación del movimiento, realizada por la unidad de compensación del movimiento 44, puede implicar obtener o generar el bloque predictivo en base al vector de movimiento determinado mediante estimación de movimiento, posiblemente realizando interpolaciones hasta 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 luminancia como de crominancia. El sumador 50 representa el componente o los componentes que realizan esta operación de resta. La unidad de compensación de movimiento 44 también puede generar elementos sintácticos asociados a los bloques de vídeo y al fragmento de vídeo para su uso por el descodificador de vídeo 30 en la descodificación de los bloques de vídeo del fragmento de vídeo.

[0078] La unidad de procesamiento de intrapredicción 46 puede intrapredicir un bloque actual, de forma alternativa a la interpredicción llevada a cabo por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como se ha descrito anteriormente. En particular, la unidad de procesamiento de intrapredicción 46 puede determinar una modalidad de intrapredicción 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.

5 **[0079]** 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
10 bloque original no codificado que fue codificado para generar el bloque codificado, así como una velocidad de transferencia de bits (es decir, un número de bits) usada para generar el bloque codificado. La unidad de procesamiento de intrapredicción 46 puede calcular proporciones a partir de las distorsiones y velocidades para los diversos bloques codificados, para determinar qué modalidad de intrapredicción presenta el mejor valor de velocidad-distorsión para el bloque.

15 **[0080]** En cualquier caso, tras seleccionar un modo de intrapredicción para un bloque, la unidad de procesamiento de intrapredicción 46 puede proporcionar información que indica 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 indica el modo de intrapredicción seleccionado de acuerdo con las técnicas de esta divulgación. El codificador de vídeo 20 puede incluir, en el flujo de bits transmitido, datos de configuración, que pueden incluir una pluralidad de
20 tablas de índices del modo de intrapredicción y una pluralidad de tablas de índices del modo de intrapredicción modificadas (también denominadas tablas de correlación de palabras de código), definiciones de contextos de codificación para diversos bloques e indicaciones de un modo de intrapredicción más probable, una tabla de índices del modo de intrapredicción y una tabla de índices del modo de intrapredicción modificada, a usar para cada uno de los contextos.

25 **[0081]** Después de que la unidad de procesamiento de predicción 41 genera el bloque predictivo para el bloque de vídeo actual, por medio de interpredicción o bien intrapredicción, el codificador de vídeo 20 forma un bloque de vídeo residual restando el bloque predictivo al bloque de vídeo actual. Los datos de vídeo residuales del bloque residual pueden estar incluidos en una o más TU y aplicarse a la unidad de procesamiento de transformada 52. La unidad de
30 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 píxel a un dominio de transformada, tal como un dominio de frecuencia.

35 **[0082]** La unidad de procesamiento de transformada 52 puede enviar los coeficientes de transformada resultantes a la unidad de cuantificación 54. La unidad de cuantificación 54 cuantifica los coeficientes de transformada para reducir más la velocidad de bits. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos, o a la totalidad, de los coeficientes. El grado de cuantificación se puede modificar ajustando un parámetro de cuantificación. En algunos ejemplos, la unidad de cuantificación 54 puede realizar, a continuación, una exploración de la matriz que
40 incluye los coeficientes de transformada cuantificados. De forma alternativa, la unidad de codificación por entropía 56 puede realizar la exploración.

[0083] Tras la cuantificación, la unidad de codificación por entropía 56 codifica por entropía los coeficientes de transformada cuantificados. 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 otra metodología o técnicas de codificación de entropía. Tras la codificación por entropía mediante la unidad de codificación por entropía 56, el flujo de bits codificado se puede transmitir al descodificador de vídeo 30, o archivarse para su posterior transmisión o recuperación mediante el descodificador de vídeo 30. La unidad de codificación por entropía 56 también puede realizar una codificación por entropía de los vectores de movimiento y los otros elementos sintácticos para el fragmento de vídeo actual que se está codificando.

45 **[0084]** La unidad de cuantificación inversa 58 y la unidad de procesamiento de transformada inversa 60 aplican una cuantificación inversa y una transformada inversa, respectivamente, para reconstruir el bloque residual en el dominio de los píxeles, para su posterior uso como 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 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 fraccionarios de píxeles, 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 denominada memoria intermedia de imágenes descodificadas (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 realizar una interpredicción de un bloque en una trama o imagen de vídeo subsiguiente.

[0085] El codificador de vídeo 20 puede configurarse para implementar las técnicas de esta divulgación. En un ejemplo, el codificador de vídeo 20 puede configurarse para determinar un primer tiempo de salida de la DPB usando un primer retardo de salida de la DPB en el caso de que un ajuste del HRD para un descodificador de vídeo indique una operación a nivel de imagen, determinar un segundo tiempo de salida de la DPB usando un segundo retardo de salida de la DPB en el caso de que el ajuste del HRD para el descodificador de vídeo indique una operación a nivel de subimagen, señalar el primer retardo de salida de la DPB y el segundo retardo de salida de la DPB en un flujo de bits de vídeo codificado. Otros ejemplos del funcionamiento del codificador de vídeo 20 de acuerdo con las técnicas de esta divulgación se analizarán a continuación con respecto a la FIG. 5.

[0086] La FIG. 4 es un diagrama de bloques que ilustra un descodificador de vídeo 30 de ejemplo que puede implementar las técnicas descritas en esta divulgación. En el ejemplo de la FIG. 4, el descodificador de vídeo 30 incluye la memoria intermedia de imágenes codificadas (CPB) 78, la unidad de descodificación por entropía 80, la unidad de procesamiento de predicción 81, la unidad de cuantificación inversa 86, la unidad de procesamiento de transformada inversa 88, el sumador 90 y la memoria intermedia de imágenes descodificadas (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 descodificador de vídeo 30 realiza una pasada de descodificación que, en general, es recíproca a la pasada de codificación descrita con respecto al codificador de vídeo 20 de la FIG. 3.

[0087] La 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 descodificación. 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 descodificación y contienen exactamente una imagen codificada. El orden de descodificación es el orden en el que se descodifican 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 mediante un descodificador de referencia hipotético (HRD), como un HRD que funciona de acuerdo con las técnicas de esta divulgación.

[0088] Durante el proceso de descodificación, el descodificador de vídeo 30 recibe un flujo de bits de vídeo codificado, que representa los bloques de vídeo de un fragmento de vídeo codificado y elementos sintácticos asociados, desde el codificador de vídeo 20. La unidad de descodificación por entropía 80 del descodificador de vídeo 30 descodifica por entropía el flujo de bits para generar coeficientes cuantificados, vectores de movimiento y otros elementos sintácticos. La unidad de descodificación por entropía 80 reenvía los vectores de movimiento y otros elementos sintácticos a la unidad de procesamiento de predicción 81. El descodificador de vídeo 30 puede recibir los elementos sintácticos a nivel de fragmento de vídeo y/o a nivel de bloque de vídeo.

[0089] 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 los bloques previamente descodificados de la trama o imagen actual. Cuando la trama de vídeo se codifica como un fragmento intercodificado (es decir, B o P), 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, en base a los vectores de movimiento y otros elementos sintácticos recibidos desde la unidad de descodificación por entropía 80. Los bloques predictivos se pueden generar a partir de una de las imágenes de referencia dentro de una de las listas de imágenes de referencia. El descodificador de vídeo 30 puede construir las listas de tramas de referencia, la Lista 0 y la Lista 1, usando técnicas de construcción por defecto basándose en las imágenes de referencia almacenadas en la DPB 92.

[0090] 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 producir los bloques predictivos del bloque de vídeo actual que se está descodificando. Por ejemplo, la unidad de compensación de movimiento 82 usa algunos de los elementos de sintaxis recibidos para determinar un modo de predicción (por ejemplo, intrapredicción o interpredicción), usado para codificar los bloques de vídeo del fragmento de vídeo, un tipo de fragmento de interpredicción (por ejemplo, fragmento B o fragmento P), información de construcción para una o más de las listas de imágenes de referencia para el fragmento, vectores de movimiento para cada bloque de vídeo intercodificado del fragmento, el estado de interpredicción para cada bloque de vídeo intercodificado del fragmento y otra información para descodificar los bloques de vídeo en el fragmento de vídeo actual.

[0091] 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 fraccionarios de los 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.

- 5 **[0092]** La unidad de cuantificación inversa 86 cuantifica de manera inversa, es decir, descuantifica, los coeficientes de transformada cuantificados proporcionados en el flujo de bits y descodificados por la unidad de descodificación por entropía 80. El proceso de cuantificación inversa puede incluir el uso de un parámetro de cuantificación calculado por el codificador de vídeo 20 para cada bloque de vídeo del fragmento de vídeo para determinar un grado de cuantificación y, asimismo, un grado de cuantificación inversa que se debería aplicar. La unidad de procesamiento de transformada inversa 88 aplica una transformada inversa, por ejemplo, una DCT inversa, una transformada entera inversa o un proceso de transformada inversa conceptualmente similar, a los coeficientes de transformada a fin de generar bloques residuales en el dominio del píxel.
- 10 **[0093]** 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 otros elementos sintácticos, el descodificador de vídeo 30 forma un bloque de vídeo descodificado sumando los bloques residuales 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.
- 15 Se desea, también se puede aplicar un filtro de eliminación de bloques para filtrar los bloques descodificados, a fin de eliminar distorsiones de efecto pixelado. También se pueden usar otros filtros de bucle (bien en el bucle de codificación o bien después del bucle de codificación) para suavizar las transiciones de píxeles o mejorar de otro modo la calidad del vídeo. Los bloques de vídeo descodificados en una trama o imagen dada son a continuación almacenados en la DPB 92, que almacena imágenes de referencia usadas para la posterior compensación de movimiento. La DPB 92 almacena también vídeo descodificado 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, como se define en las técnicas de esta divulgación.
- 20 **[0094]** El descodificador de vídeo 30 puede configurarse para implementar las técnicas de esta divulgación. En un ejemplo, el descodificador de vídeo 30 puede configurarse para recibir un primer retardo de salida de la DPB y un segundo retardo de salida de la DPB para una imagen descodificada, determinar, para la imagen descodificada, un primer tiempo de salida de la DPB usando el primer retardo de salida de la DPB en el caso de que un ajuste del HRD para un descodificador de vídeo indique una operación a nivel de imagen y determinar, para la imagen descodificada, un segundo tiempo de salida de la DPB usando el segundo retardo de salida de la DPB en el caso de que el ajuste del HRD para el descodificador de vídeo indique una operación a nivel de subimagen. Otros ejemplos del funcionamiento del descodificador de vídeo 30 de acuerdo con las técnicas de esta divulgación se analizarán a continuación con respecto a la FIG. 6.
- 25 **[0095]** La FIG. 5 es un diagrama de flujo que muestra un procedimiento de codificación a modo de ejemplo de acuerdo con las técnicas de esta divulgación. Las técnicas de la FIG. 5 puede implementarse mediante una o más estructuras del codificador de vídeo 20.
- 30 **[0096]** En un ejemplo, el codificador de vídeo 20 puede estar configurado para determinar un primer tiempo de salida de la DPB usando un primer retardo de la salida de la DPB en el caso de que un ajuste del HRD para un descodificador de vídeo indique un funcionamiento a nivel de imagen (500), y determinar un segundo tiempo de salida de la DPB usando un segundo retardo de salida de la DPB en el caso de que el ajuste del HRD para el descodificador de vídeo indique la operación a nivel de subimagen (502). El codificador de vídeo 20 puede configurarse adicionalmente para señalar el primer retardo de salida de la DPB y el segundo retardo de salida de la DPB en un flujo de bits de vídeo codificado (504).
- 35 **[0097]** El codificador de vídeo 20 puede configurarse adicionalmente para señalar un indicador de la CPB de subimagen que indica si el ajuste del HRD para el descodificador de vídeo es a nivel de imagen o a nivel de subimagen (506), y codificar imágenes de vídeo basándose en el indicador de la CPB de subimagen (508).
- 40 **[0098]** En un ejemplo de la divulgación, la determinación del segundo tiempo de salida de la DPB comprende multiplicar el segundo retardo de salida de la DPB por un ciclo de reloj de subimagen y la adición de un valor resultante para un tiempo de eliminación de la CPB. En otro ejemplo de la divulgación, determinar el primer tiempo de salida de la DPB comprende multiplicar el primer retardo de salida de la DPB por un ciclo de reloj y agregar un valor resultante a un tiempo de eliminación de la CPB.
- 45 **[0099]** La FIG. 6 es un diagrama de flujo que muestra un procedimiento de descodificación de ejemplo de acuerdo con las técnicas de la presente divulgación. Las técnicas de la FIG. 6 puede implementarse mediante una o más estructuras del descodificador de vídeo 30.
- 50 **[0100]** En un ejemplo, el descodificador de vídeo 30 puede configurarse para recibir un indicador de la CPB de subimagen que indica si el ajuste del HRD para el descodificador de vídeo es a nivel de imagen o a nivel de subimagen (600), y descodificar imágenes de vídeo basándose en el indicador de la CPB de subimagen (602).
- 55 **[0101]** El descodificador de vídeo 30 puede configurarse adicionalmente para recibir un primer retardo de salida de la DPB y un segundo retardo de salida de la DPB para una imagen descodificada (604), y determinar, para la imagen descodificada, un primer tiempo de salida de la DPB usando el primer retardo de salida de la DPB en el caso de que

el ajuste del HRD para el descodificador de vídeo indique una operación a nivel de imagen (606), y determinar, para la imagen descodificada, un segundo tiempo de salida de la DPB usando el segundo retardo de salida de la DPB en el caso de que el ajuste del HRD para el descodificador de vídeo indique una operación a nivel de subimagen (608).

5 **[0102]** El descodificador de vídeo 30 puede configurarse adicionalmente para emitir imágenes desde una memoria intermedia de imágenes descodificadas basándose en el primer tiempo de salida de la DPB o el segundo tiempo de salida de la DPB basándose en el ajuste del HRD (610). El primer tiempo de salida de la DPB se usa si el indicador de CPB de subimagen indica que el ajuste del HRD para el descodificador de vídeo indica una operación a nivel de imagen, y el segundo tiempo de salida de la DPB se usa si el indicador de CPB de subimagen indica que el ajuste del HRD para el descodificador de vídeo indica una operación a nivel de subimagen.

15 **[0103]** En otro ejemplo de la divulgación, el descodificador de vídeo 30 está configurado para determinar el segundo tiempo de salida de la DPB multiplicando el segundo retardo de salida de la DPB por un ciclo de reloj de subimagen y agregando un valor resultante a un tiempo de eliminación de la CPB. En otro ejemplo de la divulgación, el descodificador de vídeo 30 está configurado para determinar el primer tiempo de salida de la DPB multiplicando el primer retardo de salida de la DPB por un ciclo de reloj y agregando un valor resultante a un tiempo de eliminación de la CPB.

20 **[0104]** En uno o más ejemplos, las funciones descritas se pueden implementar en hardware, software, firmware o cualquier combinación de los mismos. Si se implementan en software, las funciones se pueden almacenar en o transmitir por un medio legible por ordenador como una o más instrucciones o código, y ejecutarse mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que corresponden 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 y legibles por ordenador que sean no transitorios o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que se pueda acceder mediante 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.

35 **[0105]** 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 se pueda usar para almacenar un código de programa deseado en forma de instrucciones o estructuras de datos y al que se pueda acceder mediante un ordenador. Además, cualquier conexión recibe apropiadamente la denominación de medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde un sitio web, un servidor u otro origen remoto usando un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, la DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas se incluyen en la definición de medio. Sin embargo, debería entenderse que los medios de almacenamiento legibles por ordenador y los medios de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales ni otros medios transitorios, sino que, en cambio, se orientan a medios de almacenamiento tangibles no transitorios. El término disco, como se usa en el presente documento, incluye el disco compacto (CD), el disco láser, el disco óptico, el disco versátil digital (DVD), el disco flexible y el disco Blu-ray, donde algunos discos normalmente reproducen datos de manera magnética, mientras que otros discos reproducen datos ópticamente con láseres. Las combinaciones de lo anterior se deberían incluir también dentro del alcance de los medios legibles por ordenador.

50 **[0106]** Las instrucciones se pueden ejecutar por uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados específicos de la aplicación (ASIC), matrices lógicas programables in situ (FPGA) u otros circuitos lógicos integrados o discretos equivalentes. En consecuencia, 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 descodificación, o incorporarse en un códec combinado. Además, las técnicas se pueden implementar totalmente en uno o más circuitos o elementos lógicos.

60 **[0107]** Las técnicas de la presente divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluyendo un teléfono inalámbrico, un circuito integrado (IC) o un conjunto de IC (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 describe anteriormente, diversas unidades se pueden combinar en una unidad de hardware de códec o proporcionarse mediante un grupo de unidades de

hardware interoperativas, incluidos uno o más procesadores, como se describe anteriormente, junto con software y/o firmware adecuados.

5 **[0108]** Se han descrito diversos ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un aparato configurado para decodificar datos de vídeo, comprendiendo el aparato:

5 una memoria configurada para almacenar datos de vídeo; y
 un decodificador de vídeo en comunicación con la memoria, el decodificador de vídeo configurado para:
 10 recibir un primer retardo de salida de la memoria intermedia de imágenes decodificadas DPB y un segundo retardo de salida de la DPB en un mensaje SEI de temporización de imagen;
 recibir un indicador de memoria intermedia de imágenes codificadas, CPB, de subimagen;
 15 determinar si un ajuste de un decodificador de referencia hipotético, HRD, para un decodificador de vídeo indica una operación a nivel de imagen o una operación a nivel de subimagen basándose en el indicador de la CPB de subimagen; y
 basándose en una determinación de que el ajuste del HRD indica una operación a nivel de imagen, determinar, para la imagen decodificada, un primer tiempo de salida de la DPB multiplicando el primer retardo de salida de la DPB por un ciclo del reloj de imagen y agregando un valor resultante a un tiempo de eliminación de la CPB, y emitir imágenes desde una memoria intermedia de imágenes decodificadas basándose en el primer tiempo de salida de la DPB; y
 20 basándose en una determinación de que el ajuste del HRD indica una operación a nivel de subimagen, obtener un ciclo de reloj de subimagen basándose en el ciclo de reloj de imagen y un valor divisor de ciclo por el cual el ciclo de reloj de subimagen es igual al ciclo de reloj de imagen dividido por el valor divisor de ciclo;
 25 determinar, para la imagen decodificada, un segundo tiempo de salida de la DPB multiplicando el segundo retardo de salida de la DPB por un ciclo de reloj de subimagen y agregando un valor resultante a un tiempo de eliminación de la CPB y emitiendo imágenes desde una memoria intermedia de imágenes decodificadas basándose en el segundo tiempo de salida de la DPB.

35 2. El aparato según la reivindicación 1, en el que el decodificador de vídeo está configurado, además, para:
 decodificar imágenes de vídeo basándose en el indicador de CPB de subimagen.

40 3. Un aparato configurado para codificar datos de vídeo, comprendiendo el aparato:

una memoria configurada para almacenar datos de vídeo; y
 un codificador de vídeo en comunicación con la memoria, el codificador de vídeo configurado para:
 45 determinar si un ajuste de un decodificador de referencia hipotético HRD para un decodificador de vídeo indica una operación a nivel de imagen o una operación a nivel de subimagen;
 señalar un indicador de memoria intermedia de imágenes codificadas, CPB, de subimagen que indica si el ajuste del HDR para el decodificador de vídeo es a nivel de imagen o a nivel de subimagen;
 50 basándose en una determinación de que el ajuste del HDR indica una operación a nivel de imagen, determinar un primer tiempo de salida de la memoria intermedia de imágenes decodificadas DPB multiplicando el primer retardo de salida de la DPB por un ciclo del reloj de imagen y agregando un valor resultante a un tiempo de eliminación de la CPB; y
 55 basándose en una determinación de que el ajuste del HDR indica una operación a nivel de subimagen, obteniendo un ciclo de reloj de subimagen basándose en el ciclo de reloj de imagen y un valor divisor del ciclo por el cual el ciclo del reloj de subimagen es igual al ciclo del reloj de la imagen dividido por el valor divisor del ciclo; y
 60 determinar un segundo tiempo de salida de la DPB usando un segundo retardo de salida de la DPB y el ciclo de reloj de subimagen multiplicando el segundo retardo de salida de la DPB por un ciclo de reloj de subimagen y agregando un valor resultante a un tiempo de eliminación de la CPB;
 65 señalar el primer retardo de salida de la DPB y el segundo retardo de salida de la DPB en un mensaje SEI de temporización de imagen.

4. El aparato según la reivindicación 3, en el que el codificador de vídeo está configurado, además, para codificar imágenes de vídeo basándose en el indicador de la CPB de subimagen.
5. Un aparato configurado para descodificar datos de vídeo, comprendiendo el aparato:
- medios para recibir un primer retardo de salida de la memoria intermedia de imágenes descodificadas DPB y un segundo retardo de salida de la DPB en el mensaje SEI de temporización de imagen;
- medios para recibir un indicador de memoria intermedia de imágenes codificadas CPB de subimagen;
- medios de indicación para determinar si un ajuste del descodificador de referencia hipotético HRD para un descodificador de vídeo indica una operación a nivel de imagen o una operación a nivel de subimagen basándose en el indicador de la CPB de subimagen;
- medios, operables cuando los medios de indicación determinan una operación a nivel de imagen, para determinar, para la imagen descodificada, un primer tiempo de salida de la DPB multiplicando el primer retardo de salida de la DPB por un ciclo de reloj de imagen y agregando un valor resultante a un tiempo de eliminación de la CPB, y emitir imágenes desde una memoria intermedia de imágenes descodificadas basándose en el primer tiempo de salida de la DPB; y
- medios, operables cuando los medios de indicación determinan una operación a nivel de subimagen, para obtener un ciclo de reloj de subimagen basado en el ciclo de reloj de imagen y un valor divisor de ciclo por el cual el ciclo de reloj de subimagen es igual al ciclo de reloj de imagen dividido por el valor divisor del ciclo y determinar, para la imagen descodificada, un segundo tiempo de salida de la DPB multiplicando el segundo retardo de salida de la DPB por un ciclo de reloj de subimagen y agregando un valor resultante a un tiempo de eliminación de la CPB y emitir imágenes desde una memoria intermedia de imágenes descodificadas basándose en el segundo tiempo de salida de la DPB.
6. El aparato según la reivindicación 5, que comprende, además, medios para descodificar imágenes de vídeo basándose en el indicador de la CPB de subimagen.
7. Un aparato configurado para codificar datos de vídeo, comprendiendo el aparato:
- medios de indicación para determinar si un ajuste de un descodificador de referencia hipotético, HRD, para un descodificador de vídeo indica una operación a nivel de imagen o una operación a nivel de subimagen;
- medios para señalar un indicador de memoria intermedia de imágenes codificadas CPB de subimagen que indica si el ajuste de HDR para el descodificador de vídeo es a nivel de imagen o a nivel de subimagen;
- medios, operables cuando los medios de indicación determinan que el ajuste del HDR indica una operación a nivel de imagen, para determinar un primer tiempo de salida de la memoria intermedia de imágenes descodificadas DPB multiplicando el primer retardo de salida de la DPB por un ciclo de reloj de imagen y agregando un valor resultante a un tiempo de eliminación de la CPB; y
- medios, operables cuando los medios de indicación determinan que el ajuste del HDR indica una operación a nivel de subimagen, para obtener un ciclo de reloj de subimagen basándose en el ciclo de reloj de imagen y un valor divisor de ciclo por el cual el ciclo de reloj de subimagen es igual al ciclo de reloj de imagen dividido por el valor divisor de ciclo y para
- determinar un segundo tiempo de salida de la DPB usando un segundo retardo de salida de la DPB y el ciclo de reloj de subimagen multiplicando el segundo retardo de salida de la DPB por un ciclo del reloj de subimagen y agregando un valor resultante a un tiempo de eliminación de la CPB; y
- medios para señalar el primer retardo de salida de la DPB y el segundo retardo de salida de la DPB en un mensaje SEI de temporización de imagen.
8. El aparato según la reivindicación 7, que comprende, además, medios para codificar imágenes de vídeo basándose en el indicador de la CPB de subimagen.

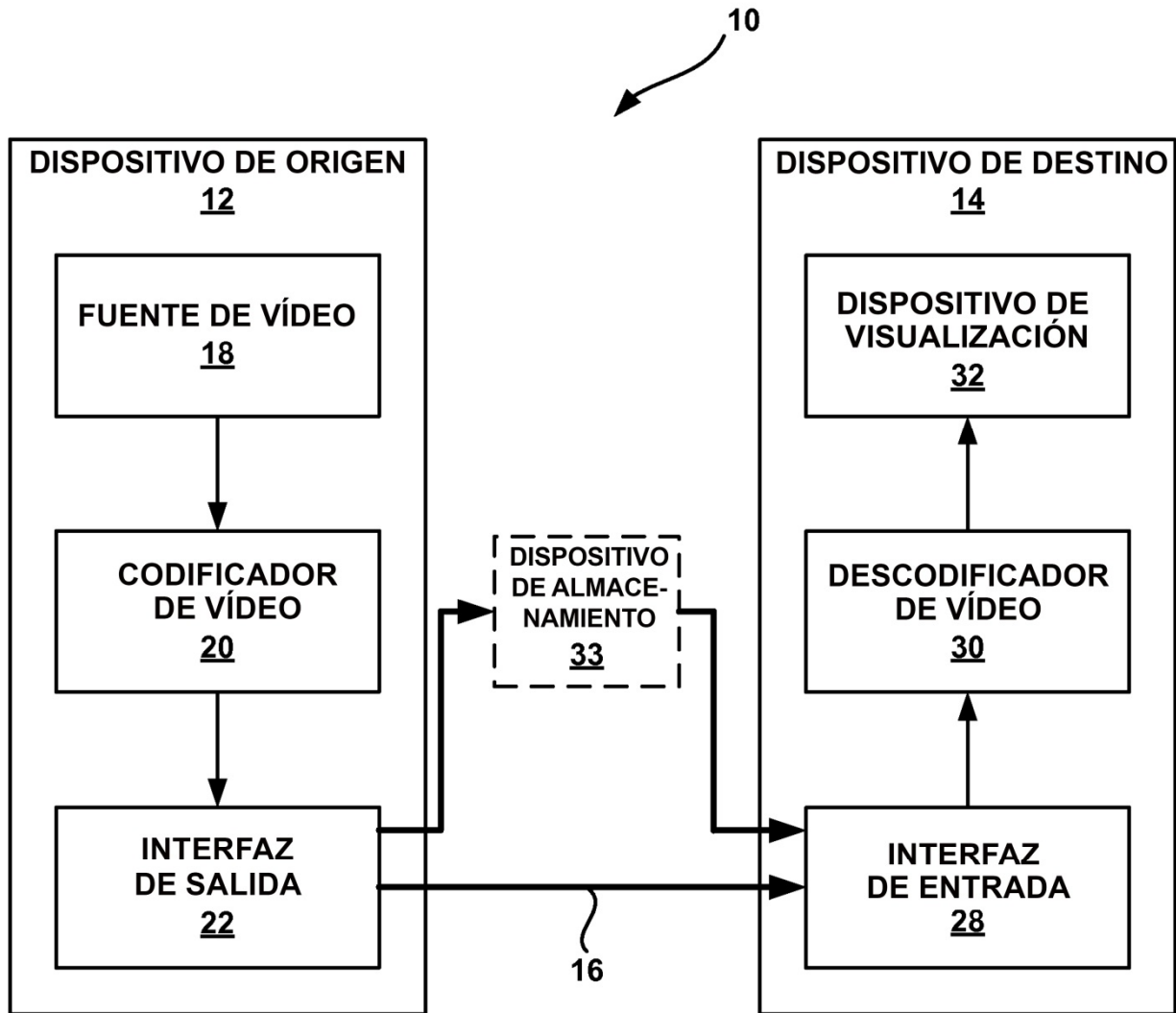


FIG. 1

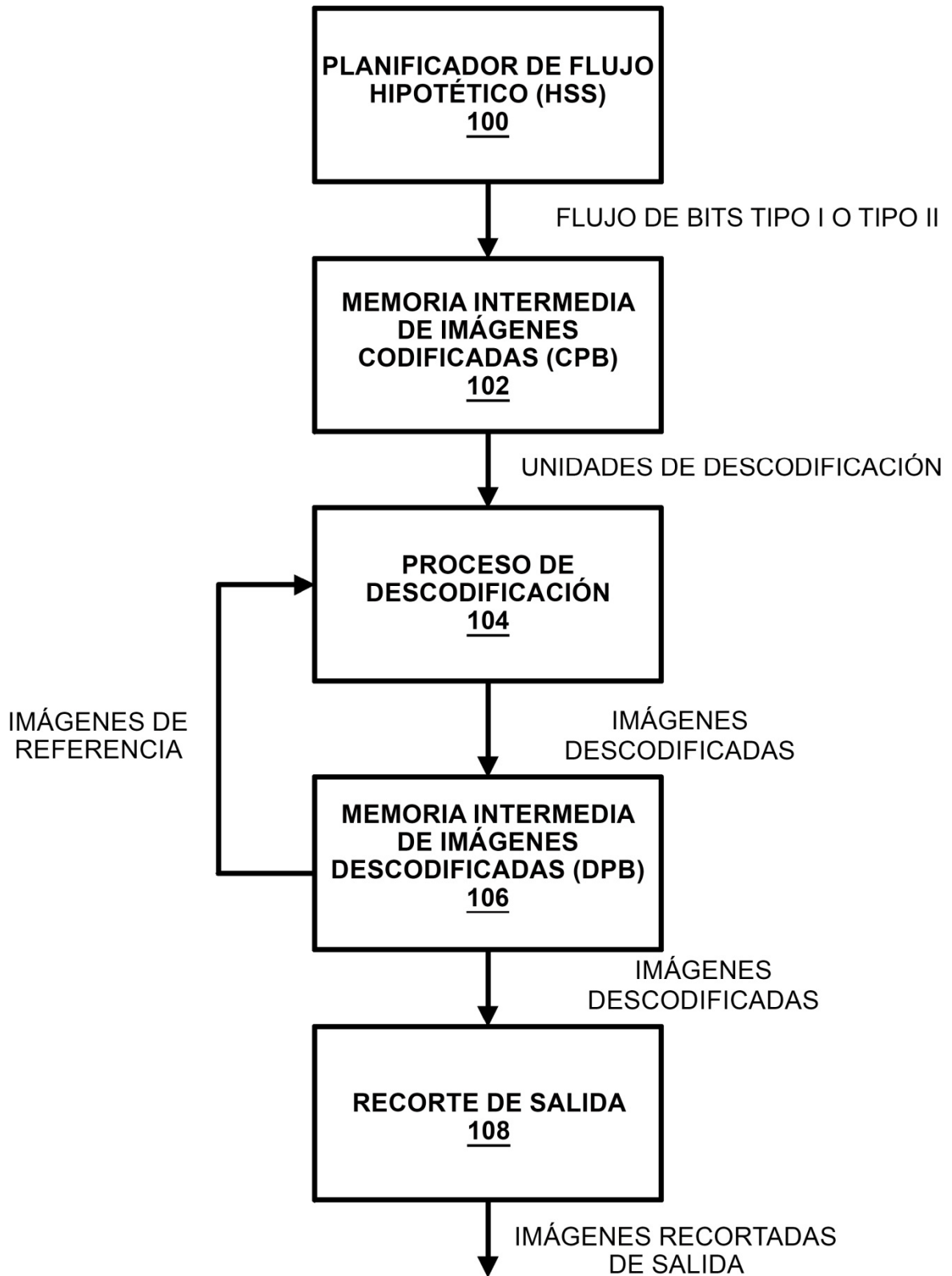


FIG. 2

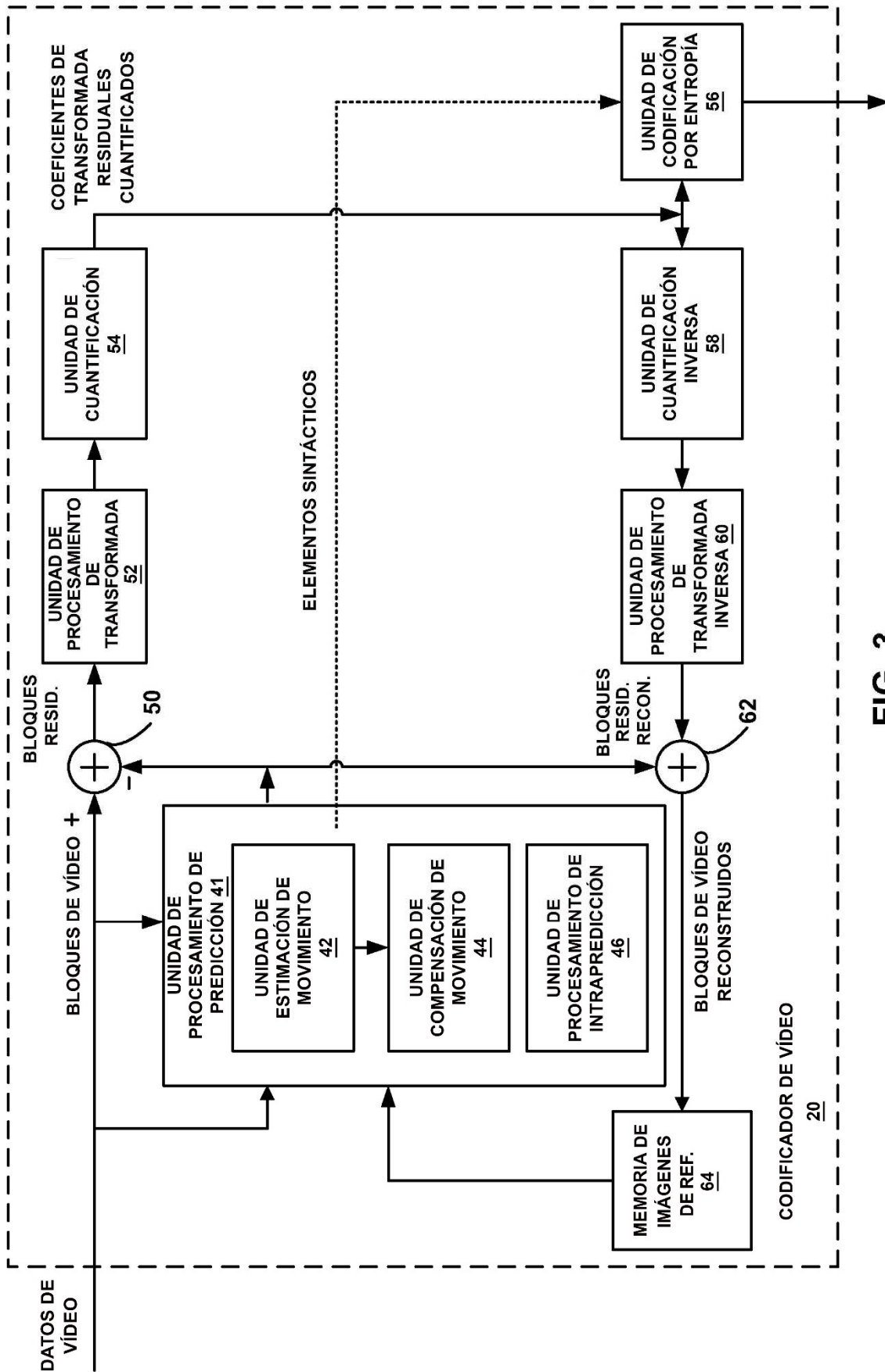


FIG. 3

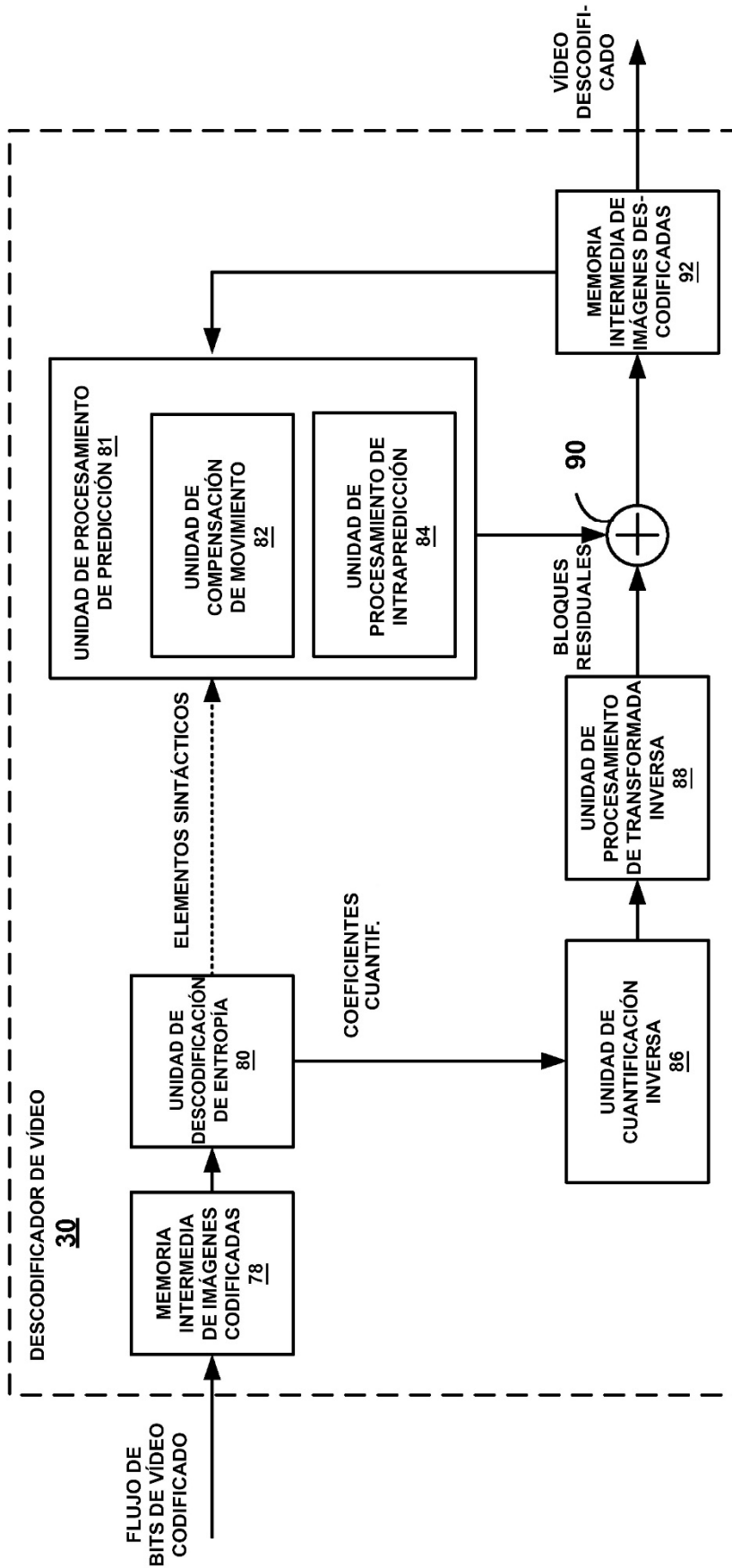


FIG. 4

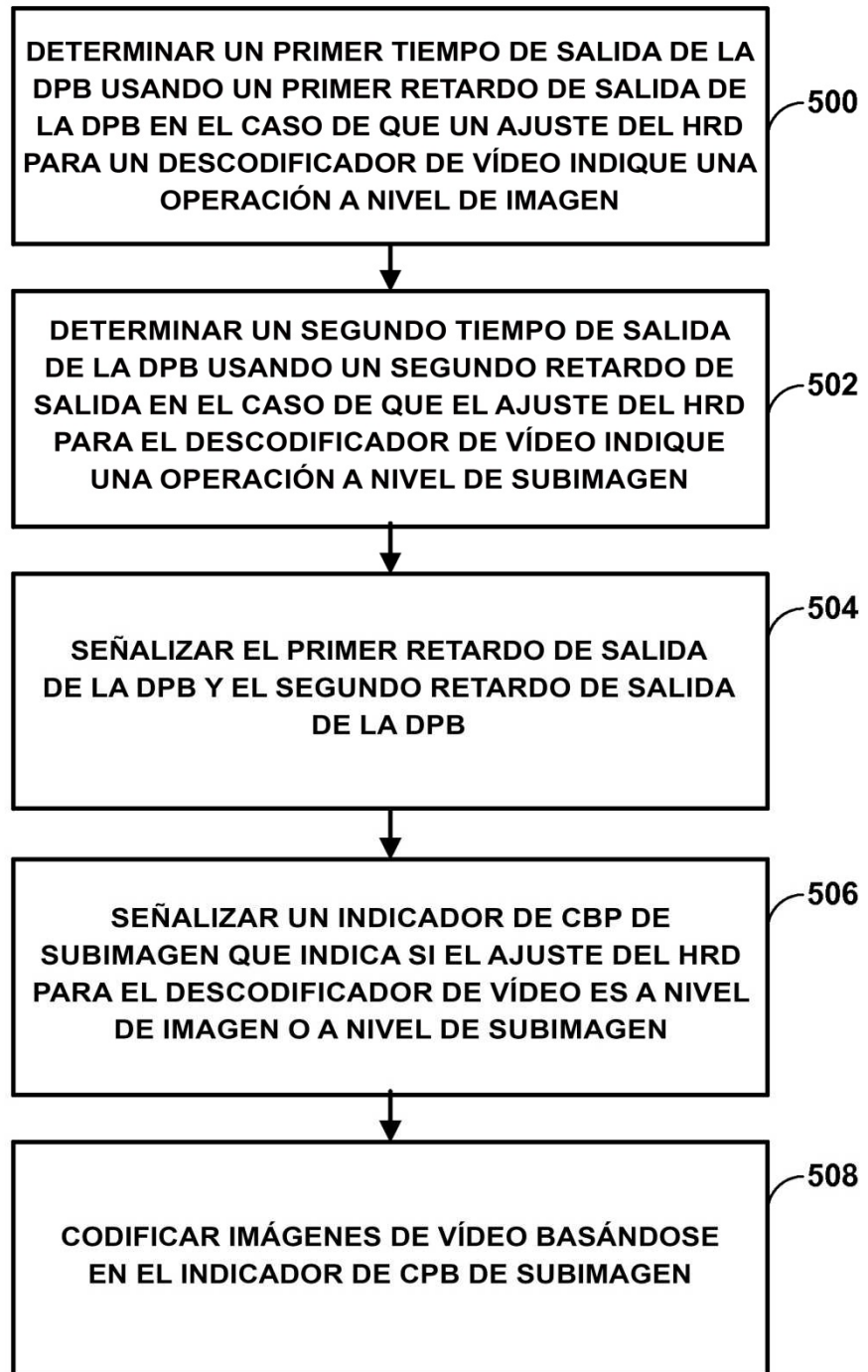


FIG. 5

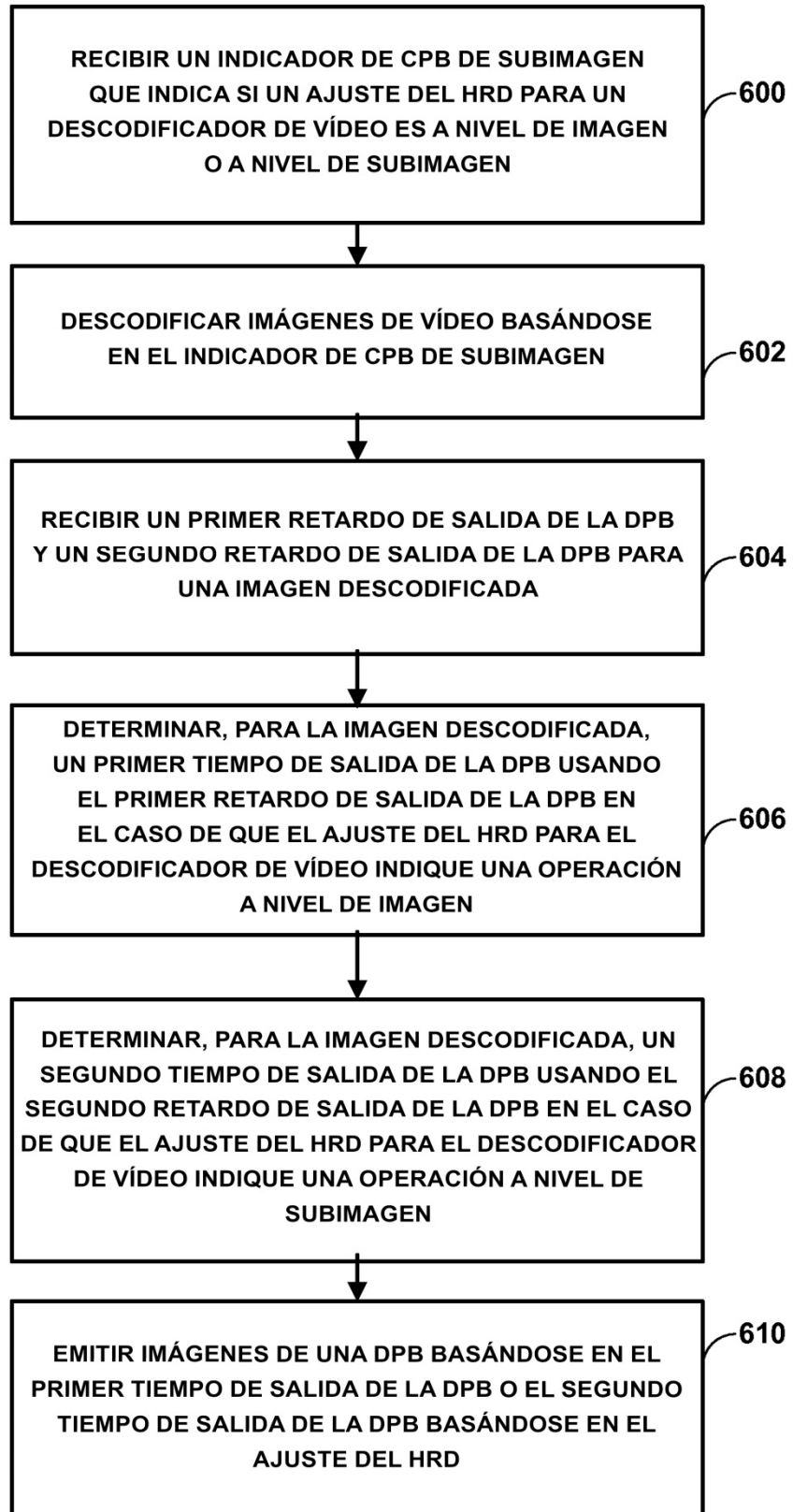


FIG. 6