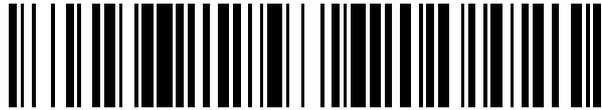


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 795 682**

51 Int. Cl.:

| | |
|--------------------|-----------|
| H04N 19/33 | (2014.01) |
| H04N 19/174 | (2014.01) |
| H04N 19/157 | (2014.01) |
| H04N 19/46 | (2014.01) |
| H04N 19/70 | (2014.01) |

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **18.07.2014 PCT/US2014/047250**
- 87 Fecha y número de publicación internacional: **29.01.2015 WO15013137**
- 96 Fecha de presentación y número de la solicitud europea: **18.07.2014 E 14747485 (2)**
- 97 Fecha y número de publicación de la concesión europea: **04.03.2020 EP 3025499**

54 Título: **Dispositivo y procedimiento para codificación escalable de información de vídeo**

30 Prioridad:

22.07.2013 US 201361857165 P
04.10.2013 US 201361886997 P
17.07.2014 US 201414334417

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
24.11.2020

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)
5775 Morehouse Drive
San Diego, CA 92121-1714, US

72 Inventor/es:

CHEN, JIANLE;
SEREGIN, VADIM;
LI, XIANG;
RAPAKA, KRISHNAKANTH y
KARCZEWICZ, MARTA

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 795 682 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

Dispositivo y procedimiento para codificación escalable de información de vídeo

5 **CAMPO TÉCNICO**

[0001] Esta divulgación se refiere al campo de la codificación y compresión de vídeo, en particular a la codificación de vídeo escalable (SVC) o la codificación de vídeo multivista (MVC, 3DV).

10 **ANTECEDENTES**

[0002] Las capacidades de vídeo digital se pueden incorporar a una amplia gama de dispositivos, incluyendo televisores digitales, sistemas de radiodifusión directa digital, sistemas de radiodifusión inalámbrica, asistentes personales digitales (PDA), ordenadores portátiles o de escritorio, cámaras digitales, dispositivos de grabación digitales, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos de radio celulares o por satélite, dispositivos de videoconferencia y similares. Los dispositivos de vídeo digital implementan técnicas de compresión de vídeo, tales como las descritas en los estándares definidos por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, parte 10, Codificación de vídeo avanzada (AVC), el estándar de codificación de vídeo de alta eficacia (HEVC) actualmente en desarrollo y las extensiones de dichos estándares. Los dispositivos de vídeo pueden transmitir, recibir, codificar, descodificar y/o almacenar información de vídeo digital más eficazmente, implementando dichas técnicas de codificación de vídeo.

[0003] Las técnicas de compresión de vídeo realizan predicción espacial (intraimagen) y/o predicción temporal (interimagen) para reducir o eliminar la redundancia intrínseca a las secuencias de vídeo. Para la codificación de vídeo basada en bloques, un fragmento de vídeo (por ejemplo, una trama de vídeo, una parte de una trama de vídeo, etc.) 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 en un fragmento intercodificado (P o B) de una imagen pueden usar predicción espacial con respecto a muestras de referencia de bloques vecinos en la misma imagen o predicción temporal con respecto a muestras de referencia en otras imágenes de referencia. Las imágenes se pueden denominar tramas, y las imágenes de referencia se pueden denominar tramas de referencia.

[0004] La predicción espacial o temporal da como resultado un bloque predictivo para un bloque que se va a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original que se va a codificar y el bloque predictivo. Un bloque intercodificado se codifica de acuerdo con un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo, y los datos residuales que indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intracodificado se codifica de acuerdo con un modo de intracodificación y los datos residuales. Para una mayor compresión, los datos residuales se pueden transformar desde el dominio de píxel en 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 matriz bidimensional, pueden escanearse con el fin de generar un vector unidimensional de coeficientes de transformada, y puede aplicarse codificación por entropía para lograr aún más compresión.

45 **BREVE EXPLICACIÓN**

[0005] La codificación de vídeo escalable (SVC) se refiere a la codificación de vídeo en la que se usan una capa base (BL), a veces denominada capa de referencia (RL), y una o más capas de mejora escalable (EL). Para SVC, la capa base puede transportar datos de vídeo con un nivel de calidad base. Las una o más capas de mejora pueden transportar datos de vídeo adicionales para admitir, por ejemplo, niveles espaciales, temporales y/o de señal a ruido (SNR) más altos. Las capas de mejora pueden definirse en relación con una capa previamente codificada. Por ejemplo, una capa inferior puede servir como una BL, mientras que una capa superior puede servir como una EL. Las capas intermedias pueden servir como EL o RL, o como ambas. Por ejemplo, una capa del medio puede ser una EL para las capas de debajo de esta, tales como la capa base o cualquier capa de mejora intermedia, y al mismo tiempo servir como RL para una o más capas de mejora encima de esta. De forma similar, en la ampliación de múltiples vistas o 3D de la norma HEVC, puede haber múltiples vistas, y la información de una vista puede utilizarse para codificar (por ejemplo, codificar o descodificar) la información de otra vista (por ejemplo, estimación de movimiento, predicción del vector de movimiento y/u otras redundancias).

[0006] En SVC, un bloque actual en la capa de mejora se puede codificar (por ejemplo, codificar o descodificar) usando la información derivada de una capa de referencia. Por ejemplo, un bloque actual en la capa de mejora se puede codificar usando la información (por ejemplo, información de textura o información de movimiento) de un bloque de localización conjunta en la capa de referencia (el término "localización conjunta" como se usa en la presente divulgación se puede referir a un bloque en otra capa que corresponde a la misma imagen que el

bloque actual, por ejemplo, el bloque que se está codificando actualmente). Sin embargo, para determinados esquemas de escalabilidad tales como la escalabilidad espacial puede ser necesario modificar la información de la capa de referencia antes de usarla para codificar la capa de mejora. Por ejemplo, puede ser necesario volver a muestrear la información de la capa de referencia (por ejemplo, de acuerdo con la relación de resolución entre la capa de referencia y la capa de mejora) antes de usarla para codificar la capa de mejora. Por ejemplo, en algunas implementaciones, una imagen de capa de referencia se puede volver a muestrear e insertar en la lista de imágenes de referencia de una imagen de capa de mejora y usarse para codificar la imagen de capa de mejora. La predicción entre capas mediante el muestreo ascendente de una imagen de capa de referencia es bien conocida, por ejemplo, de CHEN J Y COL.: "SHVC Test Model 2", 13.^a REUNIÓN DE JCT-VC; 104.^a REUNIÓN DE JCT-VC; 18-4-2013 - 26-4-2013; INCHEON; (EQUIPO COLABORATIVO CONJUNTO SOBRE CODIFICACIÓN DE VÍDEO DE ISO/CEI JTC1/SC29/WG11 E ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/, n.º JCTVC-M1007, 6 de junio de 2013 (06-06-2013); SCHWARZ H Y COL.: "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard", TRANSACCIONES IEEE EN CIRCUITOS Y SISTEMAS PARA TECNOLOGÍA DE VÍDEO, CENTRO DE SERVICIOS DEL IEEE, PISCATAWAY, NJ, EE. UU., vol. 17, n.º 9, 1 de septiembre de 2007 (01-09-2007), páginas 1103-1120, ISSN: 1051-8215, DOI: 10.1109/TCSVT.2007.905532; o CHEN J Y COL.: "SHVC Working Draft 2", 13.^a REUNIÓN DE JCT-VC; 104.^a REUNIÓN DE JCT-VC; 18-4-2013 - 26-4-2013; INCHEON; (EQUIPO COLABORATIVO CONJUNTO SOBRE CODIFICACIÓN DE VÍDEO DE ISO/CEI JTC1/SC29/WG11 E ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/, n.º JCTVC-M1008, 21 de mayo de 2013 (21-05-2013).

[0007] Sin embargo, volver a muestrear la imagen de la capa de referencia puede dar como resultado que la imagen muestreada resultante viole una o más restricciones de conformidad del flujo de bits. Por ejemplo, la imagen de la capa de referencia se puede dividir en múltiples segmentos, y algunas implementaciones pueden dictar que el límite del segmento coincida con los límites de la unidad de codificación más grande (LCU). Sin embargo, después de volver a muestrear la imagen de la capa de referencia, el límite de segmento de la imagen de la capa de referencia ya no puede coincidir con los límites de la LCU. En dicho caso, se puede desear una modificación del límite de segmento en la imagen de la capa de referencia muestreada nuevamente, a costa de un procesamiento y potencia de cálculo adicionales.

[0008] Por tanto, mediante el uso de un procedimiento más simple para el muestreo de imágenes de la capa de referencia que generará imágenes que cumplan con las restricciones de conformidad del flujo de bits, se puede reducir la complejidad computacional.

[0009] Cada uno de los sistemas, procedimientos y dispositivos de esta divulgación tiene varios aspectos innovadores, ninguno de los cuales es el único responsable de los atributos deseables divulgados en el presente documento.

[0010] Se divulga un aparato, un procedimiento y un medio legible por ordenador no transitorio que comprende código que, cuando se ejecuta, hace que un aparato realice un proceso de acuerdo con las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0011]

La FIG. 1A es un diagrama de bloques que ilustra un sistema de codificación y decodificación de vídeo de ejemplo que puede utilizar técnicas de acuerdo con aspectos descritos en esta divulgación.

La FIG. 1B es un diagrama de bloques que ilustra otro sistema de codificación y decodificación de vídeo de ejemplo que puede llevar a cabo técnicas de acuerdo con aspectos descritos en esta divulgación.

La FIG. 2A es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación.

La FIG. 2B es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación.

La FIG. 3A es un diagrama de bloques que ilustra un ejemplo de un decodificador de vídeo que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación.

La FIG. 3B es un diagrama de bloques que ilustra un ejemplo de un decodificador de vídeo que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación.

Las FIGS. 4A y 4B ilustran un procedimiento de ejemplo para generar una imagen de referencia entre capas, de acuerdo con un modo de realización de la presente descripción.

La FIG. 5 ilustra un diagrama de flujo que ilustra un procedimiento de codificación de información de vídeo, de

acuerdo con un modo de realización de la presente divulgación.

La **FIG. 6** ilustra un diagrama de flujo que ilustra un procedimiento de codificación de información de vídeo, de acuerdo con un modo de realización de la presente divulgación.

La **FIG. 7** ilustra un diagrama de flujo que ilustra un procedimiento de codificación de información de vídeo, de acuerdo con un modo de realización de la presente divulgación.

La **FIG. 8** ilustra un diagrama de flujo que ilustra un procedimiento de codificación de información de vídeo, de acuerdo con un modo de realización de la presente divulgación.

DESCRIPCIÓN DETALLADA

[0012] Determinados modos de realización descritos en el presente documento se refieren a la predicción entre capas para la codificación de vídeo escalable en el contexto de los códecs de vídeo avanzados, tal como la HEVC (codificación de vídeo de alta eficacia). Más específicamente, la presente divulgación se refiere a sistemas y procedimientos para mejorar el rendimiento de la predicción entre capas en la ampliación de codificación de vídeo escalable (SVC) de la HEVC.

[0013] En la descripción siguiente, se describen técnicas H.264/AVC relacionadas con determinados modos de realización y también se analizan la norma HEVC y técnicas relacionadas. Aunque ciertos modos de realización se describen en el presente documento en el contexto de las normas HEVC y/o H.264, un experto en la materia puede apreciar que los sistemas y procedimientos divulgados en el presente documento pueden aplicarse a cualquier norma de codificación de vídeo adecuada. Por ejemplo, los modos de realización divulgados en el presente documento pueden ser aplicables a una o más de las siguientes normas: ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual e ITU-T H.264 (también conocida como ISO/IEC MPEG-4 AVC), incluyendo sus ampliaciones de codificación de vídeo escalable (SVC) y de codificación de vídeo multivista (MVC).

[0014] HEVC, en general, sigue el marco de las normas de codificación de vídeo anteriores en muchos aspectos. La unidad de predicción en HEVC es diferente de la de determinadas normas de codificación de vídeo previas (por ejemplo, un macrobloque). De hecho, el concepto de macrobloque no existe en HEVC como se entiende en determinadas normas de codificación de vídeo previas. Un macrobloque se reemplaza por una estructura jerárquica basada en un sistema de árbol cuaternario, lo cual puede proporcionar una gran flexibilidad, entre otros beneficios posibles. Por ejemplo, dentro del esquema HEVC, se definen tres tipos de bloques, unidad de codificación (CU), unidad de predicción (PU) y unidad de transformada (TU). La CU puede referirse a la unidad básica de división de la región. La CU se puede considerar análoga al concepto de macrobloque, pero no limita el tamaño máximo y puede permitir una división recursiva en cuatro CU de igual tamaño para mejorar la adaptabilidad del contenido. La PU se puede considerar la unidad básica de inter/intrapredicción, y puede contener múltiples divisiones de forma arbitraria en una única PU para codificar eficazmente patrones de imagen irregulares. La TU se puede considerar como la unidad básica de transformada. Se puede definir independientemente de la PU; sin embargo, su tamaño puede estar limitado a la CU a la que pertenece la TU. Esta separación de la estructura de bloques en tres conceptos diferentes puede permitir optimizar cada uno de ellos de acuerdo con su función, lo cual puede dar como resultado una mejora de la eficacia de codificación.

[0015] Para fines de ilustración solamente, ciertos modos de realización divulgados en el presente documento se describen con ejemplos que incluyen solo dos capas (por ejemplo, una capa inferior tal como la capa base y una capa superior tal como la capa de mejora). Debe entenderse que dichos ejemplos pueden ser aplicables a configuraciones que incluyen múltiples capas base y/o de mejora. Además, para facilitar la explicación, la siguiente divulgación incluye los términos "tramas" o "bloques" con referencia a ciertos modos de realización. Sin embargo, estos términos no pretenden ser limitativos. Por ejemplo, las técnicas descritas a continuación se pueden usar con cualquier unidad de vídeo adecuada, como bloques (por ejemplo, CU, PU, TU, macrobloques, etc.), segmentos, tramas, etc.

Normas de codificación de vídeo

[0016] Una imagen digital, tal como una imagen de vídeo, una imagen de televisión, una imagen fija o una imagen generada por una grabadora de vídeo o un ordenador puede consistir en píxeles o muestras dispuestas en líneas horizontales y verticales. La cantidad de píxeles en una sola imagen suele ser de decenas de miles. Cada píxel contiene típicamente información de luminancia y crominancia. Sin compresión, la cantidad de información que se va a transmitir desde un codificador de imágenes hasta un descodificador de imágenes es tan enorme que haría imposible la transmisión de imágenes en tiempo real. Para reducir la cantidad de información a transmitir, se han desarrollado una serie de procedimientos de compresión diferentes, tales como las normas JPEG, MPEG y H.263.

[0017] Las normas de codificación de vídeo incluyen ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o

ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual e ITU-T H.264 (también conocida como ISO/IEC MPEG-4 AVC), incluyendo sus ampliaciones de codificación de vídeo escalable (SVC) y de codificación de vídeo multivista (MVC).

5 **[0018]** Además, existe una nueva norma de codificación de vídeo, concretamente la codificación de vídeo de alta eficacia (HEVC), que está siendo desarrollada por el Equipo de colaboración conjunta en 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 ISO/IEC. La cita completa del borrador 10 de HEVC es el documento
10 JCTVC-L1003, de Bross y col., "High Efficiency Video Coding (HEVC) Text Specification Draft 10", Equipo de colaboración conjunta en codificación de vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 12.^a Reunión: Ginebra, Suiza, del 14 de enero de 2013 al 23 de enero de 2013. La ampliación multivista para HEVC, llamada MV-HEVC, y la ampliación escalable para HEVC, llamada SHVC, también están siendo desarrolladas por el JCT-3V (Equipo de colaboración conjunta de ITU-T/ISO/IEC para el desarrollo de la ampliación de codificación de vídeo 3D) y JCT-VC, respectivamente.

15 **[0019]** A continuación, en el presente documento se describen de forma más detallada diversos aspectos de los sistemas, aparatos y procedimientos novedosos, con referencia a los dibujos adjuntos. Sin embargo, la presente divulgación se puede realizar de muchas formas diferentes y no se debería interpretar que está limitada a ninguna estructura o función específica presentada a lo largo de la presente divulgación. En cambio, estos
20 aspectos se proporcionan para que la presente divulgación sea exhaustiva y completa, y transmita por completo el alcance de la divulgación a los expertos en la materia. Basándose en las enseñanzas del presente documento, un experto en la materia debería apreciar que el alcance de la divulgación pretende abarcar cualquier aspecto de los sistemas, aparatos y procedimientos novedosos divulgados en el presente documento, ya sean implementados de forma independiente de, o en combinación con, cualquier otro aspecto de la presente divulgación. Por ejemplo, un aparato se puede implementar o un procedimiento se puede llevar a la práctica usando un número cualquiera de los aspectos expuestos en el presente documento. Además, el alcance de la presente divulgación pretende abarcar dicho aparato o procedimiento que se lleva a la práctica usando otra estructura, funcionalidad o estructura y funcionalidad, además, o aparte, de los diversos aspectos de la presente divulgación expuestos en el presente documento. Se debe entender que cualquier aspecto divulgado en el
25 30 presente documento se puede realizar mediante uno o más elementos de una reivindicación.

[0020] Aunque en el presente documento se describen aspectos particulares, muchas variaciones y permutaciones de estos aspectos se encuentran dentro del alcance de la divulgación. Aunque se mencionan algunos beneficios y ventajas de los aspectos preferentes, el alcance de la divulgación no pretende estar limitado a beneficios, usos u objetivos particulares. En cambio, los aspectos de la divulgación pretenden ser ampliamente aplicables a diferentes tecnologías inalámbricas, configuraciones de sistema, redes y protocolos de transmisión, algunos de los cuales se ilustran a modo de ejemplo en las figuras y en la siguiente descripción de los aspectos preferentes. La descripción detallada y los dibujos son meramente ilustrativos de la divulgación, en lugar de limitantes, estando definido el alcance de la divulgación por las reivindicaciones adjuntas y equivalentes de las mismas.
35 40

[0021] Los dibujos adjuntos ilustran ejemplos. Los elementos indicados mediante números de referencia en los dibujos adjuntos corresponden a elementos indicados mediante números de referencia similares en la siguiente descripción. En esta divulgación, los elementos que tienen nombres que comienzan con palabras ordinales (por ejemplo, "primero", "segundo", "tercero", y así sucesivamente) no necesariamente implican que los elementos tienen un orden particular. Más bien, dichas palabras ordinales se usan simplemente para referirse a diferentes elementos de un mismo tipo o un tipo similar.
45

Sistema de codificación de vídeo

50 **[0022]** La **FIG. 1A** es un diagrama de bloques que ilustra un sistema de codificación de vídeo 10 de ejemplo que puede usar técnicas de acuerdo con aspectos descritos en esta divulgación. Como se usa y describe en el presente documento, el término "codificador de vídeo" se refiere, en general, tanto a codificadores de vídeo como a descodificadores de vídeo. En esta divulgación, los términos "codificación de vídeo" o "codificación" pueden referirse genéricamente a la codificación de vídeo y a la descodificación de vídeo.
55

[0023] Como se muestra en la **FIG. 1A**, el sistema de codificación de vídeo 10 incluye un módulo de origen 12 que genera datos de vídeo codificados para ser descodificados posteriormente por un módulo de destino 14. En el ejemplo de la **FIG. 1A**, el módulo de origen 12 y el módulo de destino 14 están en dispositivos separados; específicamente, el módulo de origen 12 es parte de un dispositivo de origen, y el módulo de destino 14 es parte de un dispositivo de destino. Sin embargo, se observa que los módulos de origen y de destino 12, 14 pueden estar en el mismo dispositivo, o ser parte del mismo, como se muestra en el ejemplo de la **FIG. 1B**.
60

[0024] Con referencia una vez más a la **FIG. 1A**, el módulo de origen 12 y el módulo de destino 14 pueden comprender cualquiera entre una amplia gama de dispositivos, incluidos los ordenadores de escritorio, los ordenadores plegables (por ejemplo, los ordenadores portátiles), los ordenadores de tableta, los
65

descodificadores, los equipos telefónicos tales como los llamados teléfonos "inteligentes", los llamados paneles "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, dispositivos de transmisión de flujo de vídeo o similares. En algunos casos, el módulo de origen 12 y el módulo de destino 14 pueden estar equipados para la comunicación inalámbrica.

[0025] El módulo de destino 14 puede recibir los datos de vídeo codificados que se van a descodificar a través de un enlace 16. El enlace 16 puede comprender cualquier tipo de medio o dispositivo que pueda desplazar los datos de vídeo codificados desde el módulo de origen 12 al módulo de destino 14. En el ejemplo de la **FIG. 1A**, el enlace 16 puede comprender un medio de comunicación para permitir que el módulo de origen 12 transmita datos de vídeo codificados directamente al módulo de destino 14 en tiempo real. Los datos de vídeo codificados pueden modularse de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y transmitirse al módulo de destino 14. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrica o por cable, 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 amplia 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 módulo de origen 12 al módulo de destino 14.

[0026] De forma alternativa, los datos codificados pueden emitirse desde una interfaz de salida 22 hasta un dispositivo de almacenamiento 31 optativo. De forma similar, se puede acceder a los datos codificados del dispositivo de almacenamiento 31 mediante una interfaz de entrada 28. El dispositivo de almacenamiento 31 puede incluir cualquiera de una diversidad de medios de almacenamiento de datos, de acceso distribuido o local, tales como una unidad de disco duro, memoria flash, memoria volátil o no volátil u otros medios cualesquiera de almacenamiento digital adecuados para almacenar datos de vídeo codificados. En otro ejemplo, el dispositivo de almacenamiento 31 puede corresponder a un servidor de ficheros o a otro dispositivo de almacenamiento intermedio que pueda guardar el vídeo codificado generado mediante el módulo de origen 12. El módulo de destino 14 puede acceder a datos de vídeo almacenados del dispositivo de almacenamiento 31, mediante transmisión continua o descarga. El servidor de ficheros puede ser cualquier tipo de servidor que pueda almacenar datos de vídeo codificados y transmitir esos datos de vídeo codificados al módulo de destino 14. Servidores de archivos de ejemplo incluyen un servidor web (por ejemplo, para una página web), un servidor FTP, dispositivos de almacenamiento conectados en red (NAS) o una unidad de disco local. El módulo de destino 14 puede acceder a los datos de vídeo codificados mediante cualquier conexión de datos estándar, incluyendo una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión wifi), una conexión por cable (por ejemplo, DSL, módem de 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 31 puede ser una transmisión continua, una transmisión de descarga o una combinación de ambas.

[0027] Las técnicas de esta divulgación no están limitadas a aplicaciones o configuraciones inalámbricas. Las técnicas se pueden aplicar a la codificación de vídeo para admitir 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 de vídeo en continuo, por ejemplo, por medio de Internet (por ejemplo, transmisión continua dinámica adaptativa por HTTP (DASH), etc.), 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 de codificación de vídeo 10 puede estar configurado para admitir transmisión de vídeo unidireccional o bidireccional para admitir aplicaciones tales como la transmisión continua de vídeo, la reproducción de vídeo, la radiodifusión de vídeo y/o la videotelefonía.

[0028] En el ejemplo de la **FIG. 1A**, el módulo de origen 12 incluye una fuente de vídeo 18, un codificador de vídeo 20 y una interfaz de salida 22. En algunos casos, la interfaz de salida 22 puede incluir un modulador/demodulador (módem) y/o un transmisor. En el módulo de origen 12, el origen de vídeo 18 puede incluir un origen tal como un dispositivo de captura de vídeo, por ejemplo, una videocámara, un archivo de vídeo que contiene vídeo previamente capturado, 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 dichos orígenes. Como 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 "videotéfonos", tal como se ilustra en el ejemplo de la **FIG. 1B**. Sin embargo, las técnicas descritas en la presente divulgación pueden ser aplicables a la codificación de vídeo en general, y se pueden aplicar a aplicaciones inalámbricas y/o por cable.

[0029] El vídeo capturado, precapturado o generado por ordenador puede ser codificado por el codificador de vídeo 20. Los datos de vídeo codificados pueden ser directamente transmitidos al módulo de destino 14 a través de la interfaz de salida 22 del módulo de origen 12. Los datos de vídeo codificados pueden almacenarse también (o de forma alternativa) en el dispositivo de almacenamiento 31 para un acceso posterior mediante el módulo de destino 14 u otros dispositivos, para su descodificación y/o reproducción.

[0030] En el ejemplo de la **FIG. 1A**, el módulo de destino 14 incluye una interfaz de entrada 28, un decodificador de vídeo 30 y un dispositivo de visualización 32. En algunos casos, la interfaz de entrada 28 puede incluir un receptor y/o un módem. La interfaz de entrada 28 del módulo de destino 14 puede recibir los datos de vídeo codificados a través del enlace 16. Los datos de vídeo codificados, comunicados a través del enlace 16, o proporcionados en el dispositivo de almacenamiento 31, pueden incluir una diversidad de elementos sintácticos generados mediante el codificador de vídeo 20, para su uso mediante un decodificador de vídeo, tal como el decodificador 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.

[0031] El dispositivo de visualización 32 puede estar integrado con, o ser externo a, el módulo de destino 14. En algunos ejemplos, el módulo 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 módulo de destino 14 puede ser un dispositivo de visualización. En general, el dispositivo de visualización 32 muestra los datos de vídeo descodificados a un usuario, y puede comprender cualquiera entre varios 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.

[0032] En aspectos relacionados, la **FIG. 1B** muestra un sistema de codificación y descodificación de vídeo 10' de ejemplo en el que los módulos de origen y de destino 12, 14 están en, o forman parte de, un dispositivo o un dispositivo de usuario 11. El dispositivo 11 puede ser un equipo telefónico de mano, tal como un teléfono "inteligente" o similar. El dispositivo 11 puede incluir un módulo controlador/procesador 13 optativo en comunicación operativa con los módulos de origen y de destino 12, 14. El sistema 10' de la **FIG. 1B** puede incluir, además, una unidad de procesamiento de vídeo 21 entre el codificador de vídeo 20 y la interfaz de salida 22. En algunas implementaciones, la unidad de procesamiento de vídeo 21 es una unidad separada, como se ilustra en la **FIG. 1B**; sin embargo, en otras implementaciones, la unidad de procesamiento de vídeo 21 puede implementarse como una porción del codificador de vídeo 20 y/o el módulo procesador/controlador 13. El sistema 10' también puede incluir un rastreador 29 optativo, que puede rastrear un objeto de interés en una secuencia de vídeo. El objeto o interés a rastrear puede ser segmentado por una técnica descrita en relación con uno o más aspectos de la presente divulgación. En aspectos relacionados, el rastreo puede ser realizado por el dispositivo de visualización 32, solo o junto con el rastreador 29. El sistema 10' de la **FIG. 1B**, y los componentes del mismo, son similares de otro modo al sistema 10 de la **FIG. 1A** y componentes del mismo.

[0033] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar de acuerdo con una norma de compresión de vídeo, tal como la norma de codificación de vídeo de alta eficacia (HEVC), actualmente en desarrollo, y se pueden adaptar a un modelo de prueba de HEVC (HM). De forma alternativa, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar de acuerdo con otras normas privadas 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 la presente 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.

[0034] Aunque no se muestra en los ejemplos de las **FIGS. 1A** y **FIG. 1B**, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden estar integrados, cada uno, con un codificador y decodificador de audio, y pueden incluir unidades MUX-DEMUX apropiadas, u otro hardware y software, para gestionar la codificación tanto de audio como de vídeo en un flujo de datos común o flujos de datos por separado. Si procede, en algunos ejemplos, las unidades MUX-DEMUX se pueden ajustar al protocolo de multiplexor ITU H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

[0035] Cada uno entre el codificador de vídeo 20 y el decodificador de vídeo 30 puede implementarse 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), formaciones de compuertas 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 la presente divulgación. Cada uno entre el codificador de vídeo 20 y el decodificador de vídeo 30 puede estar incluido en uno o más codificadores o decodificadores, cualquiera de los cuales puede estar integrado como parte de un codificador/decodificador (CÓDEC) combinado en un respectivo dispositivo.

Proceso de codificación de vídeo

[0036] Como se ha mencionado anteriormente de manera breve, el codificador de vídeo 20 codifica datos de vídeo. Los datos de vídeo pueden comprender una o más imágenes. Cada una de las imágenes es una imagen fija que forma parte de un vídeo. En algunos casos, una imagen puede denominarse una "trama" de vídeo.

Cuando el codificador de vídeo 20 codifica los datos de vídeo, el codificador de vídeo 20 puede generar un flujo de bits. El flujo de bits puede incluir una secuencia de bits que forman una representación codificada de los datos de vídeo. El flujo de bits puede incluir imágenes codificadas y datos asociados. Una imagen codificada es una representación codificada de una imagen.

5

[0037] Para generar el flujo de bits, el codificador de vídeo 20 puede realizar operaciones de codificación en cada imagen de los datos de vídeo. Cuando el codificador de vídeo 20 realiza operaciones de codificación en las imágenes, el codificador de vídeo 20 puede generar una serie de imágenes codificadas y datos asociados. Los datos asociados pueden incluir conjuntos de parámetros de vídeo (VPS), conjuntos de parámetros de secuencia, conjuntos de parámetros de imagen, conjuntos de parámetros de adaptación y otras estructuras sintácticas. Un conjunto de parámetros de secuencia (SPS) puede contener parámetros aplicables a cero o más secuencias de imágenes. Un conjunto de parámetros de imagen (PPS) puede contener parámetros aplicables a cero o más imágenes. Un conjunto de parámetros de adaptación (APS) puede contener parámetros aplicables a cero o más imágenes. Los parámetros de un APS pueden ser parámetros que tienen más probabilidades de cambiar que los parámetros de un PPS.

10

15

[0038] Para generar una imagen codificada, el codificador de vídeo 20 puede dividir una imagen en bloques de vídeo de igual tamaño. Un bloque de vídeo puede ser una matriz bidimensional de muestras. Cada uno de los bloques de vídeo está asociado a un bloque de árbol. En algunos casos, un bloque de árbol también puede denominarse unidad de codificación más grande (LCU). Los bloques de árbol de HEVC pueden ser aproximadamente análogos a los macrobloques de normas anteriores, tales como la H.264/AVC. Sin embargo, un bloque de árbol no está limitado necesariamente a un tamaño particular y puede incluir una o más unidades de codificación (CU). El codificador de vídeo 20 puede usar una división en árbol cuaternario para dividir los bloques de vídeo de bloques de árbol en bloques de vídeo asociados a las CU, de ahí el nombre "bloques de árbol".

20

25

[0039] En algunos ejemplos, el codificador de vídeo 20 puede dividir una imagen en una pluralidad de segmentos. Cada uno de los segmentos puede incluir un número entero de CU. En algunos casos, un segmento comprende un número entero de bloques de árbol. En otros casos, un límite de un segmento puede estar dentro de un bloque de árbol.

30

[0040] Como parte del modo de realización de una operación de codificación en una imagen, el codificador de vídeo 20 puede realizar operaciones de codificación en cada segmento de la imagen. Cuando el codificador de vídeo 20 realiza una operación de codificación en un segmento, el codificador de vídeo 20 puede generar datos codificados asociados al segmento. Los datos codificados asociados al segmento pueden denominarse "segmento codificado".

35

[0041] Para generar un segmento codificado, el codificador de vídeo 20 puede realizar unas operaciones de codificación en cada bloque de árbol de un segmento. Cuando el codificador de vídeo 20 realiza una operación de codificación en un bloque de árbol, el codificador de vídeo 20 puede generar un bloque de árbol codificado. El bloque de árbol codificado puede comprender datos que representan una versión codificada del bloque de árbol.

40

[0042] Cuando el codificador de vídeo 20 genera un sector codificado, el codificador de vídeo 20 puede realizar operaciones de codificación (es decir, codificar) en los bloques de árbol del segmento de acuerdo con un orden de escaneo de trama. Por ejemplo, el codificador de vídeo 20 puede codificar los bloques de árbol del segmento en un orden dirigido de izquierda a derecha en la fila más alta de los bloques de árbol del segmento, a continuación de izquierda a derecha en la siguiente fila inferior de los bloques de árbol, y así sucesivamente hasta que el codificador de vídeo 20 ha codificado cada uno de los bloques de árbol del segmento.

45

[0043] Como resultado de codificar los bloques de árbol de acuerdo con el orden de escaneo de trama, los bloques de árbol de arriba y a la izquierda de un bloque de árbol dado pueden haber sido codificados, pero los bloques de árbol de debajo y a la derecha del bloque de árbol dado aún no han sido codificados. En consecuencia, el codificador de vídeo 20 puede ser capaz de acceder a la información generada por la codificación de los bloques de árbol de encima y de la izquierda del bloque de árbol dado cuando se codifica el bloque de árbol dado. Sin embargo, el codificador de vídeo 20 puede no ser capaz de acceder a la información generada por la codificación de los bloques de árbol de debajo y de la derecha del bloque de árbol dado cuando se codifica el bloque de árbol dado.

50

55

[0044] Para generar un bloque de árbol codificado, el codificador de vídeo 20 puede realizar de forma recursiva una división en árbol cuaternario en el bloque de vídeo del bloque de árbol para dividir el bloque de vídeo en bloques de vídeo progresivamente más pequeños. Cada uno de los bloques de vídeo más pequeños puede estar asociado a una CU diferente. Por ejemplo, el codificador de vídeo 20 puede dividir el bloque de vídeo de un bloque de árbol en cuatro subbloques de igual tamaño, dividir uno o más de los subbloques en cuatro subsubbloques de igual tamaño, y así sucesivamente. Una CU dividida puede ser una CU cuyo bloque de vídeo está dividido en bloques de vídeo asociados a otras CU. Una CU no dividida puede ser una CU cuyo bloque de vídeo no esté dividido en bloques de vídeo asociados a otras CU.

60

65

[0045] Uno o más elementos sintácticos del flujo de bits pueden indicar un número máximo de veces que el codificador de vídeo 20 puede dividir el bloque de vídeo de un bloque de árbol. Un bloque de vídeo de una CU puede tener forma cuadrada. El tamaño del bloque de vídeo de una CU (es decir, el tamaño de la CU) puede variar desde 8x8 píxeles hasta el tamaño de un bloque de vídeo de un bloque de árbol (es decir, el tamaño del bloque de árbol) con un máximo de 64x64 píxeles o mayor.

[0046] El codificador de vídeo 20 puede realizar operaciones de codificación (es decir, codificar) en cada CU de un bloque de árbol de acuerdo con un orden de escaneo en z. En otras palabras, el codificador de vídeo 20 puede codificar una CU superior izquierda, una CU superior derecha, una CU inferior izquierda y a continuación una CU inferior derecha, en ese orden. Cuando el codificador de vídeo 20 realiza una operación de codificación en una CU dividida, el codificador de vídeo 20 puede codificar las CU asociadas a subbloques del bloque de vídeo de la CU dividida de acuerdo con el orden de escaneo en z. En otras palabras, el codificador de vídeo 20 puede codificar una CU asociada a un subbloque superior izquierdo, una CU asociada a un subbloque superior derecho, una CU asociada a un subbloque inferior izquierdo y a continuación una CU asociada a un subbloque inferior derecho, en ese orden.

[0047] Como resultado de la codificación de las CU de un bloque de árbol de acuerdo con un orden de escaneo en z, las CU de arriba, de arriba y a la izquierda, de arriba y a la derecha, a la izquierda y de abajo y a la izquierda de una CU dada pueden haber sido codificadas. Las CU de abajo y a la derecha de la CU determinada todavía no han sido codificadas. En consecuencia, el codificador de vídeo 20 puede ser capaz de acceder a información generada por la codificación de algunas CU vecinas de la CU dada cuando se codifica la CU dada. Sin embargo, el codificador de vídeo 20 puede no ser capaz de acceder a información generada por la codificación de otras CU vecinas de la CU dada cuando se codifica la CU dada.

[0048] Cuando el codificador de vídeo 20 codifica una CU no dividida, el codificador de vídeo 20 puede generar una o más unidades de predicción (PU) para la CU. Cada una de las PU de la CU puede estar asociada a un bloque de vídeo diferente dentro del bloque de vídeo de la CU. El codificador de vídeo 20 puede generar un bloque de vídeo predicho para cada PU de la CU. El bloque de vídeo predicho de una PU puede ser un bloque de muestras. El codificador de vídeo 20 puede usar intrapredicción o interpredicción para generar el bloque de vídeo predicho para una PU.

[0049] Cuando el codificador de vídeo 20 usa intrapredicción para generar el bloque de vídeo predicho de una PU, el codificador de vídeo 20 puede generar el bloque de vídeo predicho de la PU en base a unas muestras descodificadas de la imagen asociada a la PU. Si el codificador de vídeo 20 usa intrapredicción para generar bloques de vídeo predichos de las PU de una CU, la CU es una CU intrapredicha. Cuando el codificador de vídeo 20 usa interpredicción para generar el bloque de vídeo predicho de la PU, el codificador de vídeo 20 puede generar el bloque de vídeo predicho de la PU en base a unas muestras descodificadas de una o más imágenes distintas a la imagen asociada a la PU. Si el codificador de vídeo 20 usa interpredicción para generar bloques de vídeo predichos de las PU de una CU, la CU es una CU interpredicha.

[0050] Además, cuando el codificador de vídeo 20 usa interpredicción para generar un bloque de vídeo predicho para una PU, el codificador de vídeo 20 puede generar información de movimiento para la PU. La información de movimiento para una PU puede indicar uno o más bloques de referencia de la PU. Cada bloque de referencia de la PU puede ser un bloque de vídeo dentro de una imagen de referencia. La imagen de referencia puede ser una imagen distinta de la imagen asociada a la PU. En algunos casos, un bloque de referencia de una PU también puede denominarse "muestra de referencia" de la PU. El codificador de vídeo 20 puede generar el bloque de vídeo predicho para la PU en base a los bloques de referencia de la PU.

[0051] Después de que el codificador de vídeo 20 genere bloques de vídeo predichos para una o más PU de una CU, el codificador de vídeo 20 puede generar datos residuales para la CU en base a los bloques de vídeo predichos para las PU de la CU. Los datos residuales para la CU pueden indicar diferencias entre muestras en los bloques de vídeo predichos para las PU de la CU y el bloque de vídeo original de la CU.

[0052] Además, como parte del modo de realización de una operación de codificación en una CU no dividida, el codificador de vídeo 20 puede realizar una división recursiva en árbol cuaternario en los datos residuales de la CU para dividir los datos residuales de la CU en uno o más bloques de datos residuales (es decir, bloques de vídeo residuales) asociados a las unidades de transformada (TU) de la CU. Cada TU de una CU puede estar asociada a un bloque de vídeo residual diferente.

[0053] El codificador de vídeo 20 puede aplicar una o más transformadas a los bloques de vídeo residuales asociados con las TU para generar bloques de coeficientes de transformada (por ejemplo, bloques de coeficientes de transformada) asociados con las TU. Conceptualmente, un bloque de coeficientes de transformada puede ser una matriz bidimensional (2D) de coeficientes de transformada.

[0054] Después de generar un bloque de coeficientes de transformada, el codificador de vídeo 20 puede

realizar un proceso de cuantificación en el bloque de coeficientes de transformada. La cuantificación se refiere, en general, a un proceso en el que coeficientes de transformada se cuantifican para reducir, posiblemente, la cantidad de datos usados para representar los coeficientes de transformada, proporcionando más compresión. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos, o a la totalidad, de los coeficientes de transformada. Por ejemplo, un coeficiente de transformada de n bits se puede redondear a la baja hasta un coeficiente de transformada de m bits durante la cuantificación, donde n es mayor que m .

[0055] El codificador de vídeo 20 puede asociar cada CU a un valor de parámetro de cuantificación (QP). El valor de QP asociado a una CU puede determinar cómo cuantifica el codificador de vídeo 20 los bloques de coeficientes de transformada asociados a la CU. El codificador de vídeo 20 puede ajustar el grado de cuantificación aplicado a los bloques de coeficientes de transformada asociados a una CU ajustando el valor de QP asociado a la CU.

[0056] Después de que el codificador de vídeo 20 cuantifique un bloque de coeficientes de transformada, el codificador de vídeo 20 puede generar conjuntos de elementos sintácticos que representan los coeficientes de transformada en el bloque de coeficientes de transformada cuantificados. El codificador de vídeo 20 puede aplicar operaciones de codificación por entropía, tales como operaciones de codificación aritmética binaria adaptativa al contexto (CABAC), a algunos de estos elementos sintácticos. También podrían usarse otras técnicas de codificación por entropía, tales como la codificación de longitud variable adaptable al contexto (CAVLC), la codificación por entropía de división de intervalo de probabilidad (PIPE) u otra codificación aritmética binaria.

[0057] El flujo de bits generado por el codificador de vídeo 20 puede incluir una serie de unidades de capa de abstracción de red (NAL). Cada una de las unidades NAL puede ser una estructura sintáctica que contiene una indicación de un tipo de datos en la unidad NAL y octetos que contienen los datos. Por ejemplo, una unidad NAL puede contener datos que representan un conjunto de parámetros de vídeo, un conjunto de parámetros de secuencia, un conjunto de parámetros de imagen, un segmento codificado, información de mejora complementaria (SEI), un delimitador de unidad de acceso, datos de relleno u otro tipo de datos. Los datos de una unidad NAL pueden incluir diversas estructuras sintácticas.

[0058] El descodificador de vídeo 30 puede recibir el flujo de bits generado por el codificador de vídeo 20. El flujo de bits puede incluir una representación codificada de los datos de vídeo codificados por el codificador de vídeo 20. Cuando el descodificador de vídeo 30 recibe el flujo de bits, el descodificador de vídeo 30 puede realizar una operación de análisis sintáctico en el flujo de bits. Cuando el descodificador de vídeo 30 realiza la operación de análisis sintáctico, el descodificador de vídeo 30 puede extraer elementos sintácticos del flujo de bits. El descodificador de vídeo 30 puede reconstruir las imágenes de los datos de vídeo en base a los elementos sintácticos extraídos del flujo de bits. El proceso para reconstruir los datos de vídeo en base a los elementos sintácticos puede ser, en general, recíproco al proceso realizado por el codificador de vídeo 20 para generar los elementos sintácticos.

[0059] Después de que el descodificador de vídeo 30 extraiga los elementos sintácticos asociados a una CU, el descodificador de vídeo 30 puede generar bloques de vídeo predichos para las PU de la CU en base a los elementos sintácticos. Además, el descodificador de vídeo 30 puede realizar una cuantificación inversa de unos bloques de coeficientes de transformada asociados a las TU de la CU. El descodificador de vídeo 30 puede realizar transformadas inversas en los bloques de coeficientes de transformada para reconstruir bloques de vídeo residuales asociados a las TU de la CU. Después de generar los bloques de vídeo predichos y reconstruir los bloques de vídeo residuales, el descodificador de vídeo 30 puede reconstruir el bloque de vídeo de la CU en base a los bloques de vídeo predichos y los bloques de vídeo residuales. De esta manera, el descodificador de vídeo 30 puede reconstruir los bloques de vídeo de las CU en base a los elementos sintácticos del flujo de bits.

Codificador de vídeo

[0060] La **FIG. 2A** es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación. El codificador de vídeo 20 puede estar configurado para procesar una sola capa de una trama de vídeo, tal como para HEVC. Además, el codificador de vídeo 20 puede estar configurado para realizar cualquiera o la totalidad de las técnicas de esta divulgación. Como ejemplo, la unidad de procesamiento de predicción 100 puede configurarse para realizar cualquiera o todas las técnicas descritas en esta divulgación. En otro modo de realización, el codificador de vídeo 20 incluye una unidad 128 de predicción entre capas opcional que está configurada para realizar cualquiera o todas las técnicas descritas en esta divulgación. En otros modos de realización, la predicción entre capas se puede realizar mediante la unidad de procesamiento de predicción 100 (por ejemplo, la unidad de interpredicción 121 y/o la unidad de intrapredicción 126), en cuyo caso la unidad 128 de predicción entre capas puede omitirse. Sin embargo, los aspectos de esta divulgación no se limitan a lo anterior. En algunos ejemplos, las técnicas descritas en esta divulgación se pueden compartir entre los diversos componentes del codificador de vídeo 20. En algunos ejemplos, adicionalmente o de forma alternativa, un procesador (no mostrado) puede configurarse para realizar cualquiera o todas las técnicas descritas en esta divulgación.

[0061] Con propósitos explicativos, esta divulgación describe un codificador de vídeo 20 en el contexto de la codificación HEVC. Sin embargo, las técnicas de esta divulgación pueden ser aplicables a otras normas o procedimientos de codificación. El ejemplo representado en la **FIG. 2A** es para un códec de una sola capa. Sin embargo, como se describirá en mayor detalle con respecto a la **FIG. 2B**, una parte o la totalidad del codificador de vídeo 20 se puede duplicar para procesar un códec multicapa.

[0062] El codificador de vídeo 20 puede realizar la intracodificación y la intercodificación de bloques de vídeo dentro de segmentos de vídeo. La intracodificación se basa en la predicción espacial para reducir o eliminar la redundancia espacial en el vídeo dentro de una trama o imagen de vídeo determinada. 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. Intramodo (modo I) se puede referir a cualquiera de varios modos de codificación de base espacial. Los intermodos, tales como la predicción unidireccional (modo P) o la predicción bidireccional (modo B), pueden referirse a cualquiera de varios modos de codificación de base temporal.

[0063] En el ejemplo de la **FIG. 2A**, el codificador de vídeo 20 incluye una pluralidad de componentes funcionales. Los componentes funcionales del codificador de vídeo 20 incluyen una unidad de procesamiento de predicción 100, una unidad de generación residual 102, una unidad de procesamiento de transformada 104, una unidad de cuantificación 106, una unidad de cuantificación inversa 108, una unidad de transformada inversa 110, una unidad de reconstrucción 112, una unidad de filtro 113, una memoria intermedia de imágenes descodificadas 114 y una unidad de codificación por entropía 116. La unidad de procesamiento de predicción 100 incluye una unidad de interpredicción 121, una unidad de estimación de movimiento 122, una unidad de compensación de movimiento 124, una unidad de intrapredicción 126 y una unidad de predicción entre capas 128. En otros ejemplos, el codificador de vídeo 20 puede incluir más, menos o diferentes componentes funcionales. Además, la unidad de estimación de movimiento 122 y la unidad de compensación de movimiento 124 pueden estar altamente integradas, pero están representadas en el ejemplo de la **FIG. 2A** de forma separada con fines explicativos.

[0064] El codificador de vídeo 20 puede recibir datos de vídeo. El codificador de vídeo 20 puede recibir los datos de vídeo desde diversas fuentes. Por ejemplo, el codificador de vídeo 20 puede recibir los datos de vídeo de la fuente de vídeo 18 (por ejemplo, mostrada en la **FIG. 1A** o **1B**) u otra fuente. Los datos de vídeo pueden representar una serie de imágenes. Para codificar los datos de vídeo, el codificador de vídeo 20 puede realizar una operación de codificación en cada una de las imágenes. Como parte del modo de realización de la operación de codificación en una imagen, el codificador de vídeo 20 puede realizar operaciones de codificación en cada segmento de la imagen. Como parte del modo de realización de una operación de codificación en un segmento, el codificador de vídeo 20 puede realizar operaciones de codificación en bloques de árbol en el segmento.

[0065] Como parte de realizar una operación de codificación en un bloque de árbol, la unidad de procesamiento de predicción 100 puede realizar la división en árbol cuaternario en el bloque de vídeo del bloque de árbol para dividir el bloque de vídeo en bloques de vídeo progresivamente más pequeños. Cada uno de los bloques de vídeo más pequeños puede estar asociado a una CU diferente. Por ejemplo, la unidad de procesamiento de predicción 100 puede dividir un bloque de vídeo de un bloque de árbol en cuatro subbloques de igual tamaño, dividir uno o más de los subbloques en cuatro subbloques de igual tamaño, etc.

[0066] Los tamaños de los bloques de vídeo asociados a las CU pueden variar desde muestras de 8x8 hasta el tamaño del bloque de árbol, con un máximo de muestras de 64x64 o mayores. En esta divulgación, "NxN" y "N por N" pueden usarse indistintamente para hacer referencia a las dimensiones de muestras de un bloque de vídeo en lo que respecta a dimensiones verticales y horizontales, por ejemplo, muestras de 16x16 o muestras de 16 por 16. En general, un bloque de vídeo de 16x16 tiene dieciséis muestras en una dirección vertical ($y = 16$) y dieciséis muestras en una dirección horizontal ($x = 16$). Asimismo, un bloque NxN presenta, en general, N muestras en una dirección vertical y N muestras en una dirección horizontal, donde N representa un valor entero no negativo.

[0067] Además, como parte de realizar la operación de codificación en un bloque de árbol, la unidad de procesamiento de predicción 100 puede generar una estructura de datos jerárquica en árbol cuaternario para el bloque de árbol. Por ejemplo, un bloque de árbol puede corresponder a un nodo raíz de la estructura de datos en árbol cuaternario. Si la unidad de procesamiento de predicción 100 divide el bloque de vídeo del bloque de árbol en cuatro subbloques, el nodo raíz tiene cuatro nodos secundarios en la estructura de datos en árbol cuaternario. Cada uno de los nodos secundarios corresponde a una CU asociada con uno de los subbloques. Si la unidad de procesamiento de predicción 100 divide uno de los subbloques en cuatro subbloques, el nodo correspondiente a la CU asociada con el subbloque puede tener cuatro nodos secundarios, cada uno de los cuales corresponde a una CU asociada con uno de los subbloques.

[0068] Cada nodo de la estructura de datos en árbol cuaternario puede contener datos sintácticos (por ejemplo, elementos sintácticos) para el bloque de árbol o CU correspondiente. Por ejemplo, un nodo en el árbol

cuaternario puede incluir un indicador de división, que indica si el bloque de vídeo de la CU correspondiente al nodo está dividido (es decir, partido) en cuatro subbloques. Los elementos sintácticos para una CU pueden definirse de manera recursiva y pueden depender de si el bloque de vídeo de la CU está dividido en subbloques. Una CU cuyo bloque de vídeo no está dividido puede corresponder a un nodo hoja en la estructura de datos en árbol cuaternario. Un bloque de árbol codificado puede incluir datos basados en la estructura de datos en árbol cuaternario para un bloque de árbol correspondiente.

[0069] El codificador de vídeo 20 puede realizar operaciones de codificación en cada CU no dividida de un bloque de árbol. Cuando el codificador de vídeo 20 realiza una operación de codificación en una CU no dividida, el codificador de vídeo 20 genera datos que representan una representación codificada de la CU no dividida.

[0070] Como parte de realizar una operación de codificación en una CU, la unidad de procesamiento de predicción 100 puede dividir el bloque de vídeo de la CU entre una o más PU de la CU. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden admitir diversos tamaños de PU. Suponiendo que el tamaño de una CU en particular sea $2N \times 2N$, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden admitir tamaños de PU de $2N \times 2N$ o $N \times N$ e interpredicción en tamaños de PU simétricos de $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$, $2N \times nU$, $nL \times 2N$, $nR \times 2N$ o similares. El codificador de vídeo 20 y el descodificador de vídeo 30 también pueden admitir la división asimétrica para tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$. En algunos ejemplos, la unidad de procesamiento de predicción 100 puede realizar una división geométrica para dividir el bloque de vídeo de una CU entre las PU de la CU a lo largo de un límite que no coincide con los lados del bloque de vídeo de la CU en ángulos rectos.

[0071] La unidad de interpredicción 121 puede realizar la interpredicción en cada PU de la CU. La interpredicción puede proporcionar compresión temporal. Para realizar la interpredicción en una PU, la unidad de estimación de movimiento 122 puede generar información de movimiento para la PU. La unidad de compensación de movimiento 124 puede generar un bloque de vídeo predicho para la PU basándose en la información de movimiento y las muestras descodificadas de imágenes distintas a la imagen asociada a la CU (por ejemplo, imágenes de referencia). En esta divulgación, un bloque de vídeo predicho generado por la unidad de compensación de movimiento 124 se puede denominar un bloque de vídeo interpredicho.

[0072] Los segmentos pueden ser segmentos I, segmentos P o segmentos B. El módulo de estimación de movimiento 122 y la unidad de compensación de movimiento 124 pueden realizar diferentes operaciones para una PU de una CU dependiendo de si la PU está en un segmento I, un segmento P o un segmento B. En un segmento I, todas las PU se intrapredicen. Por lo tanto, si la PU está en un segmento I, la unidad de estimación de movimiento 122 y la unidad de compensación de movimiento 124 no realizan la interpredicción en la PU.

[0073] Si la PU está en un segmento P, la imagen que contiene la PU está asociada a una lista de imágenes de referencia denominada "lista 0". Cada una de las imágenes de referencia de la lista 0 contiene muestras que pueden usarse para la interpredicción de otras imágenes. Cuando la unidad de estimación de movimiento 122 realiza la operación de estimación de movimiento con respecto a una PU en un segmento P, la unidad de estimación de movimiento 122 puede buscar las imágenes de referencia en la lista 0 para un bloque de referencia para la PU. El bloque de referencia de la PU puede ser un conjunto de muestras, por ejemplo, un bloque de muestras, que se corresponde más estrechamente con las muestras del bloque de vídeo de la PU. La unidad de estimación de movimiento 122 puede usar una variedad de métricas para determinar cuán estrechamente un conjunto de muestras en una imagen de referencia se corresponde con las muestras del bloque de vídeo de una PU. Por ejemplo, la unidad de estimación de movimiento 122 puede determinar cuán estrechamente un conjunto de muestras en una imagen de referencia se corresponde con las muestras en el bloque de vídeo de una PU por la suma de la diferencia absoluta (SAD), la suma de diferencias al cuadrado (SSD) u otras métricas de diferencia.

[0074] Después de identificar un bloque de referencia de una PU en un segmento P, la unidad de estimación de movimiento 122 puede generar un índice de referencia que indica la imagen de referencia de la lista 0 que contiene el bloque de referencia y un vector de movimiento que indica un desplazamiento espacial entre la PU y el bloque de referencia. En diversos ejemplos, la unidad de estimación de movimiento 122 puede generar vectores de movimiento con diferentes grados de precisión. Por ejemplo, la unidad de estimación de movimiento 122 puede generar vectores de movimiento con una precisión de un cuarto de muestra, una precisión de un octavo de muestra o una precisión de otra fracción de muestra. En el caso de la precisión de fracción de muestra, los valores de bloque de referencia pueden interpolarse a partir de valores enteros de posición de muestra en la imagen de referencia. La unidad de estimación de movimiento 122 puede proporcionar el índice de referencia y el vector de movimiento como información de movimiento de la PU. La unidad de compensación de movimiento 124 puede generar el bloque de vídeo predicho de la PU basándose en el bloque de referencia identificado por la información de movimiento de la PU.

[0075] Si la PU está en un segmento B, la imagen que contiene la PU puede estar asociada a dos listas de imágenes de referencia, denominadas "lista 0" y "lista 1". En algunos ejemplos, una imagen que contiene un segmento B puede estar asociada a una combinación de listas que es una combinación de la lista 0 y la lista 1.

- 5 **[0076]** Además, si la PU está en un segmento B, la unidad de estimación de movimiento 122 puede llevar a cabo la predicción unidireccional o la predicción bidireccional para la PU. Cuando la unidad de estimación de movimiento 122 realiza la predicción unidireccional para la PU, la unidad de estimación de movimiento 122 puede buscar las imágenes de referencia de la lista 0 o la lista 1 para un bloque de referencia para la PU. La unidad de estimación de movimiento 122 puede entonces generar un índice de referencia que indica la imagen de referencia en la lista 0 o la lista 1 que contiene el bloque de referencia y un vector de movimiento que indica un desplazamiento espacial entre la PU y el bloque de referencia. La unidad de estimación de movimiento 122 puede emitir el índice de referencia, un indicador de la dirección de predicción y el vector de movimiento como la información de movimiento de la PU. El indicador de la dirección de predicción puede indicar si el índice de referencia indica una imagen de referencia en la lista 0 o la lista 1. La unidad de compensación de movimiento 124 puede generar el bloque de vídeo predicho de la PU basándose en el bloque de referencia indicado por la información de movimiento de la PU.
- 10
- 15 **[0077]** Cuando la unidad de estimación de movimiento 122 realiza la predicción bidireccional para una PU, la unidad de estimación de movimiento 122 puede buscar las imágenes de referencia de la lista 0 para un bloque de referencia para la PU y también puede buscar las imágenes de referencia de la lista 1 para otro bloque de referencia para la PU. La unidad de estimación de movimiento 122 puede entonces generar índices de referencia que indican las imágenes de referencia en la lista 0 y la lista 1 que contienen los bloques de referencia y los vectores de movimiento que indican desplazamientos espaciales entre los bloques de referencia y la PU. La unidad de estimación de movimiento 122 puede emitir los índices de referencia y los vectores de movimiento de la PU como la información de movimiento de la PU. La unidad de compensación de movimiento 124 puede generar el bloque de vídeo predicho de la PU basándose en los bloques de referencia indicados por la información de movimiento de la PU.
- 20
- 25 **[0078]** En algunos casos, la unidad de estimación de movimiento 122 no emite un conjunto completo de información de movimiento para una PU a la unidad de codificación por entropía 116. En lugar de eso, la unidad de estimación de movimiento 122 puede señalar la información de movimiento de una PU con referencia a la información de movimiento de otra PU. Por ejemplo, la unidad de estimación de movimiento 122 puede determinar que la información de movimiento de la PU es suficientemente similar a la información de movimiento de una PU próxima. En este ejemplo, la unidad de estimación de movimiento 122 puede indicar, en una estructura sintáctica asociada a la PU, un valor que indica al descodificador de vídeo 30 que la PU tiene la misma información de movimiento que la PU vecina. En otro ejemplo, la unidad de estimación de movimiento 122 puede identificar, en una estructura sintáctica asociada a la PU, una PU vecina y una diferencia de vectores de movimiento (MVD). La diferencia de vectores de movimiento indica una diferencia entre el vector de movimiento de la PU y el vector de movimiento de la PU próxima indicada. El descodificador de vídeo 30 puede usar el vector de movimiento de la PU vecina indicada y la diferencia de vectores de movimiento para determinar el vector de movimiento de la PU. Con referencia a la información de movimiento de una primera PU cuando se señala la información de movimiento de una segunda PU, el codificador de vídeo 20 puede ser capaz de señalar la información de movimiento de la segunda PU usando menos bits.
- 30
- 35
- 40 **[0079]** Como se analizará más adelante a continuación con referencia a las **FIGS. 5-8**, la unidad de procesamiento de predicción 100 puede configurarse para codificar (por ejemplo, codificar o descodificar) la PU (o cualquier otra capa de referencia y/o bloques de capa de mejora o unidades de vídeo) realizando los procedimientos ilustrados en las **FIGS. 5-8**. Por ejemplo, la unidad de interpredicción 121 (por ejemplo, a través de la unidad de estimación de movimiento 122 y/o la unidad de compensación de movimiento 124), la unidad de intrapredicción 126, o la unidad de predicción entre capas 128 pueden configurarse para realizar los procedimientos ilustrados en las **FIGS. 5-8** ya sea juntos o por separado.
- 45
- 50 **[0080]** Como parte de realizar una operación de codificación en una CU, la unidad de intrapredicción 126 puede llevar a cabo la intrapredicción en las PU de la CU. La intrapredicción puede proporcionar compresión espacial. Cuando la unidad de intrapredicción 126 realiza la intrapredicción en una PU, la unidad de intrapredicción 126 puede generar datos de predicción para la PU basados en muestras descodificadas de otras PU en la misma imagen. Los datos de predicción para la PU pueden incluir un bloque de vídeo predicho y varios elementos sintácticos. La unidad de intrapredicción 126 puede realizar la intrapredicción en PU en segmentos I, segmentos P y segmentos B.
- 55
- 60 **[0081]** Para realizar la intrapredicción en una PU, la unidad de intrapredicción 126 puede usar múltiples modos de intrapredicción para generar múltiples conjuntos de datos de predicción para la PU. Cuando la unidad de intrapredicción 126 usa un modo de intrapredicción para generar un conjunto de datos de predicción para la PU, la unidad de intrapredicción 126 puede extender muestras de bloques de vídeo de PU vecinas a través del bloque de vídeo de la PU en una dirección y/o gradiente asociados con el modo de intrapredicción. Las PU vecinas pueden estar arriba, arriba y a la derecha, arriba y a la izquierda o a la izquierda de la PU, suponiendo un orden de codificación de izquierda a derecha, de arriba a abajo para las PU, CU y bloques de árbol. La unidad de intrapredicción 126 puede usar diversos números de modos de intrapredicción, por ejemplo, 33 modos de intrapredicción direccional, dependiendo del tamaño de la PU.
- 65

5 **[0082]** La unidad de procesamiento de predicción 100 puede seleccionar los datos de predicción para una PU entre los datos de predicción generados por la unidad de compensación de movimiento 124 para la PU o los datos de predicción generados por la unidad de intrapredicción 126 para la PU. En algunos ejemplos, la unidad de procesamiento de predicción 100 selecciona los datos de predicción para la PU basándose en las métricas de velocidad/distorsión de los conjuntos de datos de predicción.

10 **[0083]** Si la unidad de procesamiento de predicción 100 selecciona datos de predicción generados por la unidad de intrapredicción 126, la unidad de procesamiento de predicción 100 puede señalar el modo de intrapredicción que se usó para generar los datos de predicción para las PU, es decir, el modo de intrapredicción seleccionado. La unidad de procesamiento de predicción 100 puede señalar el modo de intrapredicción seleccionado de varias maneras. Por ejemplo, es probable que el modo de intrapredicción seleccionado sea el mismo que el modo de intrapredicción de una PU vecina. En otras palabras, el modo de intrapredicción de la PU vecina puede ser el modo más probable para la PU actual. De este modo, la unidad de procesamiento de predicción 100 puede generar un elemento sintáctico para indicar que el modo de intrapredicción seleccionado es el mismo que el modo de intrapredicción de la PU vecina.

20 **[0084]** Como se analiza anteriormente, el codificador de vídeo 20 puede incluir la unidad de predicción entre capas 128. La unidad de predicción entre capas 128 está configurada para predecir un bloque actual (por ejemplo, un bloque actual en la EL) usando una o más capas diferentes que están disponibles en SVC (por ejemplo, una capa base o de referencia). Dicha predicción se puede denominar predicción entre capas. La unidad de predicción entre capas 128 utiliza procedimientos de predicción para reducir la redundancia entre capas, mejorando así la eficacia de codificación y reduciendo los requisitos de recursos informáticos. Algunos ejemplos de predicción entre capas incluyen la intrapredicción entre capas, la predicción de movimiento entre capas y la predicción residual entre capas. La intrapredicción entre capas usa la reconstrucción de bloques ubicados conjuntamente en la capa base para predecir el bloque actual en la capa de mejora. La predicción de movimiento entre capas usa la información de movimiento de la capa base para predecir el movimiento en la capa de mejora. La predicción residual entre capas usa el residuo de la capa base para predecir el residuo de la capa de mejora. Cada uno de los esquemas de predicción entre capas se analiza a continuación con mayor detalle.

35 **[0085]** Después de que la unidad de procesamiento de predicción 100 seleccione los datos de predicción para las PU de una CU, la unidad de generación residual 102 puede generar datos residuales para la CU restando (por ejemplo, indicado por el signo menos) los bloques de vídeo predichos de las PU de la CU del bloque de vídeo de la CU. Los datos residuales de una CU pueden incluir bloques de vídeo residuales 2D que corresponden a diferentes componentes de muestra de las muestras en el bloque de vídeo de la CU. Por ejemplo, los datos residuales pueden incluir un bloque de vídeo residual que corresponde a diferencias entre componentes de luminancia de muestras en los bloques de vídeo predichos de las PU de la CU y componentes de luminancia de muestras en el bloque de vídeo original de la CU. Además, los datos residuales de la CU pueden incluir bloques de vídeo residuales que corresponden a las diferencias entre componentes de crominancia de muestras en los bloques de vídeo predichos de las PU de la CU y los componentes de crominancia de las muestras en el bloque de vídeo original de la CU.

45 **[0086]** La unidad de procesamiento de predicción 100 puede realizar la división en árbol cuaternario para dividir los bloques de vídeo residuales de una CU en subbloques. Cada bloque de vídeo residual no dividido puede estar asociado a una TU diferente de la CU. Los tamaños y posiciones de los bloques de vídeo residuales asociados a las TU de una CU pueden basarse, o no, en los tamaños y posiciones de bloques de vídeo asociados a las PU de la CU. Una estructura en árbol cuaternario conocida como "árbol cuaternario residual" (RQT) puede incluir nodos asociados a cada uno de los bloques de vídeo residuales. Las TU de una CU pueden corresponder a nodos hoja del RQT.

55 **[0087]** La unidad de procesamiento de transformada 104 puede generar uno o más bloques de coeficientes de transformada para cada TU de una CU aplicando una o más transformadas a un bloque de vídeo residual asociado con la TU. Cada uno de los bloques de coeficientes de transformada puede ser una matriz 2D de coeficientes de transformada. La unidad de procesamiento de transformada 104 puede aplicar varias transformadas al bloque de vídeo residual asociado con una TU. Por ejemplo, la unidad de procesamiento de transformada 104 puede aplicar una transformada de coseno discreta (DCT), una transformada direccional o una transformada conceptualmente similar al bloque de vídeo residual asociado con una TU.

60 **[0088]** Después de que la unidad de procesamiento de transformada 104 genere un bloque de coeficientes de transformada asociado con una TU, la unidad de cuantificación 106 puede cuantificar los coeficientes de transformada en el bloque de coeficientes de transformada. La unidad de cuantificación 106 puede cuantificar un bloque de coeficientes de transformada asociado con una TU de una CU basándose en un valor de QP asociado con la CU.

65 **[0089]** El codificador de vídeo 20 puede asociar un valor de QP a una CU de diversas maneras. Por ejemplo, el

codificador de vídeo 20 puede realizar un análisis de velocidad-distorsión en un bloque de árbol asociado a la CU. En el análisis de velocidad-distorsión, el codificador de vídeo 20 puede generar múltiples representaciones codificadas del bloque de árbol realizando una operación de codificación múltiples veces en el bloque de árbol. El codificador de vídeo 20 puede asociar diferentes valores de QP a la CU cuando el codificador de vídeo 20 genera diferentes representaciones codificadas del bloque de árbol. El codificador de vídeo 20 puede señalar que un valor de QP dado está asociado a la CU cuando el valor de QP dado está asociado a la CU en una representación codificada del bloque de árbol que tiene una velocidad de bits y una métrica de distorsión más bajas.

[0090] La unidad de cuantificación inversa 108 y la unidad de transformada inversa 110 pueden aplicar la cuantificación inversa y las transformadas inversas al bloque de coeficientes de transformada, respectivamente, para reconstruir un bloque de vídeo residual a partir del bloque de coeficientes de transformada. La unidad de reconstrucción 112 puede añadir el bloque de vídeo residual reconstruido a las muestras correspondientes de uno o más bloques de vídeo predichos generados por la unidad de procesamiento de predicción 100 para producir un bloque de vídeo reconstruido asociado con una TU. Reconstruyendo bloques de vídeo para cada TU de una CU de esta manera, el codificador de vídeo 20 puede reconstruir el bloque de vídeo de la CU.

[0091] Después de que la unidad de reconstrucción 112 reconstruya el bloque de vídeo de una CU, la unidad de filtro 113 puede realizar una operación de desbloqueo para reducir los artefactos de bloqueo en el bloque de vídeo asociado con la CU. Después de realizar las una o más operaciones de desbloqueo, la unidad de filtro 113 puede almacenar el bloque de vídeo reconstruido de la CU en la memoria intermedia de imágenes descodificadas 114. La unidad de estimación de movimiento 122 y la unidad de compensación de movimiento 124 pueden usar una imagen de referencia que contiene el bloque de vídeo reconstruido para realizar la interpredicción en las PU de las imágenes posteriores. Además, la unidad de intrapredicción 126 puede usar bloques de vídeo reconstruidos en la memoria intermedia de imágenes descodificadas 114 para realizar la intrapredicción en otras PU en la misma imagen que la CU.

[0092] La unidad de codificación por entropía 116 puede recibir datos desde otros componentes funcionales del codificador de vídeo 20. Por ejemplo, la unidad de codificación por entropía 116 puede recibir bloques de coeficientes de la unidad de cuantificación 106 y puede recibir elementos sintácticos de la unidad de procesamiento de predicción 100. Cuando la unidad de codificación por entropía 116 recibe los datos, la unidad de codificación por entropía 116 puede realizar una o más operaciones de codificación por entropía para generar datos codificados por entropía. Por ejemplo, el codificador de vídeo 20 puede realizar una operación de codificación de longitud variable adaptativa al contexto (CAVLC), una operación CABAC, una operación de codificación de longitud de variable a variable (V2V), una operación de codificación aritmética binaria adaptativa al contexto basada en sintaxis (SBAC), una operación de codificación por entropía de división de intervalo de probabilidad (PIPE) u otro tipo de operación de codificación por entropía en los datos. La unidad de codificación por entropía 116 puede emitir un flujo de bits que incluye los datos codificados por entropía.

[0093] Como parte de realizar una operación de codificación por entropía en los datos, la unidad de codificación por entropía 116 puede seleccionar un modelo de contexto. Si la unidad de codificación por entropía 116 está realizando una operación CABAC, el modelo de contexto puede indicar las estimaciones de las probabilidades de que un bin particular tenga unos valores particulares. En el contexto de CABAC, el término "bin" se usa para hacer referencia a un bit de una versión binarizada de un elemento sintáctico.

Codificador de vídeo multicapa

[0094] La FIG. 2B es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo multicapa que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación. El codificador de vídeo 23 puede configurarse para procesar tramas de vídeo multicapa, tal como para codificación SHVC y de múltiples visualizaciones. Además, el codificador de vídeo 23 puede configurarse para realizar cualquiera o todas las técnicas de esta divulgación.

[0095] El codificador de vídeo 23 incluye un codificador de vídeo 20A y un codificador de vídeo 20B, cada uno de los cuales puede configurarse como el codificador de vídeo 20 y puede realizar las funciones descritas anteriormente con respecto al codificador de vídeo 20. Además, como se indica por la reutilización de los números de referencia, los codificadores de vídeo 20A y 20B pueden incluir, al menos, algunos de los sistemas y subsistemas como el codificador de vídeo 20. Aunque el codificador de vídeo 23 se ilustra como que incluye dos codificadores de vídeo 20A y 20B, el codificador de vídeo 23 no está limitado como tal y puede incluir cualquier número de capas de codificador de vídeo 20. En algunos modos de realización, el codificador de vídeo 23 puede incluir un codificador de vídeo 20 para cada imagen o trama en una unidad de acceso. Por ejemplo, una unidad de acceso que incluye cinco imágenes puede ser procesada o codificada por un codificador de vídeo que incluye cinco capas de codificador. En algunos modos de realización, el codificador de vídeo 23 puede incluir más capas de codificador que tramas en una unidad de acceso. En algunos de dichos casos, algunas de las capas del codificador de vídeo pueden estar inactivas al procesar algunas unidades de acceso.

[0096] Además de los codificadores de vídeo 20A y 20B, el codificador de vídeo 23 puede incluir una unidad de remuestreo 90. La unidad de remuestreo 90 puede, en algunos casos, realizar muestreo ascendente de una capa base de una trama de vídeo recibida para, por ejemplo, crear una capa de mejora. La unidad de remuestreo 90 puede realizar muestreo ascendente de la información particular asociada con la capa base recibida de una trama, pero no otra información. Por ejemplo, la unidad de remuestreo 90 puede realizar muestreo ascendente del tamaño espacial o el número de píxeles de la capa base, pero el número de segmentos o el recuento de orden de imágenes puede permanecer constante. En algunos casos, la unidad de remuestreo 90 puede no procesar el vídeo recibido y/o puede ser opcional. Por ejemplo, en algunos casos, la unidad de procesamiento de predicción 100 puede realizar un muestreo ascendente. En algunos modos de realización, la unidad de remuestreo 90 está configurada para realizar muestreo ascendente de una capa y reorganizar, redefinir, modificar o ajustar uno o más segmentos para cumplir con un conjunto de reglas de límite de segmento y/o reglas de escaneo de trama. Aunque se describe principalmente como un muestreo ascendente de una capa base o de una capa inferior en una unidad de acceso, en algunos casos, la unidad de remuestreo 90 puede realizar muestreo descendente de una capa. Por ejemplo, si durante la transmisión de un vídeo en tiempo real se reduce el ancho de banda, una trama puede sufrir muestreo descendente en lugar de muestreo ascendente.

[0097] La unidad de remuestreo 90 puede configurarse para recibir una imagen o trama (o información de imagen asociada con la imagen) desde la memoria intermedia de imágenes descodificadas 114 del codificador de capa inferior (por ejemplo, el codificador de vídeo 20A) y para realizar muestreo ascendente de la imagen (o la información de imagen recibida). Esta imagen muestreada ascendentemente puede entonces proporcionarse a la unidad de procesamiento de predicción 100 de un codificador de capa superior (por ejemplo, el codificador de vídeo 20B) configurado para codificar una imagen en la misma unidad de acceso que el codificador de capa inferior. En algunos casos, el codificador de capa superior es una capa eliminada del codificador de capa inferior. En otros casos, puede haber uno o más codificadores de capa superior entre el codificador de vídeo de capa 0 y el codificador de capa 1 de la **FIG. 2B**.

[0098] En algunos casos, la unidad de remuestreo 90 se puede omitir o anular. En dichos casos, la imagen de la memoria intermedia de imágenes descodificadas 114 del codificador de vídeo 20A se puede proporcionar directamente, o al menos sin proporcionarse a la unidad de remuestreo 90, a la unidad de procesamiento de predicción 100 del codificador de vídeo 20B. Por ejemplo, si los datos de vídeo proporcionados al codificador de vídeo 20B y la imagen de referencia de la memoria intermedia de imágenes descodificadas 114 del codificador de vídeo 20A son del mismo tamaño o resolución, la imagen de referencia puede proporcionarse al codificador de vídeo 20B sin ningún remuestreo.

[0099] En algunos modos de realización, el codificador de vídeo 23 realiza muestreo descendente de los datos de vídeo a proporcionar al codificador de capa inferior usando la unidad de muestreo descendente 94 antes de proporcionar los datos de vídeo al codificador de vídeo 20A. De forma alternativa, la unidad de muestreo descendente 94 puede ser una unidad de remuestreo 90 capaz de realizar un muestreo ascendente o muestreo descendente de los datos de vídeo. En otros modos de realización más, la unidad de muestreo descendente 94 puede omitirse.

[0100] Como se ilustra en la **FIG. 2B**, el codificador de vídeo 23 puede incluir además un multiplexor 98 o mux. El mux 98 puede enviar un flujo de bits combinado desde el codificador de vídeo 23. El flujo de bits combinado se puede crear tomando un flujo de bits de cada uno de los codificadores de vídeo 20A y 20B y alternando qué flujo de bits se envía en un momento determinado. Mientras que en algunos casos los bits de los dos (o más en el caso de más de dos capas de codificador de vídeo) flujos de bits pueden alternarse un bit a la vez, en muchos casos los flujos de bits se combinan de manera diferente. Por ejemplo, el flujo de bits de salida puede crearse alternando el flujo de bits seleccionado un bloque a la vez. En otro ejemplo, el flujo de bits de salida se puede crear enviando una proporción de bloques diferente a 1:1 de cada uno de los codificadores de vídeo 20A y 20B. Por ejemplo, pueden enviarse dos bloques desde el codificador de vídeo 20B para cada salida de bloque del codificador de vídeo 20A. En algunos modos de realización, el flujo de salida del mux 98 puede preprogramarse. En otros modos de realización, el mux 98 puede combinar los flujos de bits de los codificadores de vídeo 20A, 20B en base a una señal de control recibida desde un sistema externo al codificador de vídeo 23, tal como desde un procesador en un dispositivo de origen que incluye el módulo de origen 12. La señal de control puede generarse en base a la resolución o velocidad de bits de un vídeo desde la fuente de vídeo 18, en base al ancho de banda del enlace 16, en base a una suscripción asociada a un usuario (por ejemplo, una suscripción de pago frente a una suscripción gratuita) o en base a cualquier otro factor para determinar una salida de resolución deseada del codificador de vídeo 23.

Descodificador de vídeo

[0101] La **FIG. 3A** es un diagrama de bloques que ilustra un ejemplo de un descodificador de vídeo que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación. El descodificador de vídeo 30 puede configurarse para procesar una sola capa de una trama de vídeo, tal como para HEVC. Además, el descodificador de vídeo 30 puede estar configurado para realizar una cualquiera o la totalidad de las técnicas de esta divulgación. Como ejemplo, la unidad de compensación de movimiento 162 y/o la unidad de intrapredicción

164 pueden configurarse para realizar cualquiera o todas las técnicas descritas en esta divulgación. En un modo de realización, el descodificador de vídeo 30 puede incluir opcionalmente una unidad de predicción entre capas 166 que está configurada para realizar una cualquiera o la totalidad de las técnicas descritas en esta divulgación. En otros modos de realización, la predicción entre capas puede realizarse mediante la unidad de procesamiento de predicción 152 (por ejemplo, la unidad de compensación de movimiento 162 y/o la unidad de intrapredicción 164), en cuyo caso la unidad de predicción entre capas 166 puede omitirse. Sin embargo, los aspectos de esta divulgación no se limitan a lo anterior. En algunos ejemplos, las técnicas descritas en esta divulgación se pueden compartir entre los diversos componentes del descodificador de vídeo 30. En algunos ejemplos, adicionalmente o de forma alternativa, un procesador (no mostrado) puede configurarse para realizar cualquiera o todas las técnicas descritas en esta divulgación.

[0102] Con propósitos explicativos, esta divulgación describe un descodificador de vídeo 30 en el contexto de la codificación HEVC. Sin embargo, las técnicas de esta divulgación pueden ser aplicables a otras normas o procedimientos de codificación. El ejemplo representado en la **FIG. 3A** es para un códec de una sola capa. Sin embargo, como se describe en mayor detalle con respecto a la **FIG. 3B**, una parte o la totalidad del descodificador de vídeo 30 se puede duplicar para el procesamiento de un códec multicapa.

[0103] En el ejemplo de la **FIG. 3A**, el descodificador de vídeo 30 incluye una pluralidad de componentes funcionales. Los componentes funcionales del descodificador de vídeo 30 incluyen una unidad de descodificación por entropía 150, una unidad de procesamiento de predicción 152, una unidad de cuantificación inversa 154, una unidad de transformada inversa 156, una unidad de reconstrucción 158, una unidad de filtro 159 y una memoria intermedia de imágenes descodificadas 160. La unidad de procesamiento de predicción 152 incluye una unidad de compensación de movimiento 162, una unidad de intrapredicción 164 y una unidad de predicción entre capas 166. En algunos ejemplos, el descodificador de vídeo 30 puede realizar una pasada de descodificación, en general, recíproca a la pasada de codificación descrita con respecto al codificador de vídeo 20 de la **FIG. 2A**. En otros ejemplos, el descodificador de vídeo 30 puede incluir más, menos o diferentes componentes funcionales.

[0104] El descodificador de vídeo 30 puede recibir un flujo de bits que comprende datos de vídeo codificados. El flujo de bits puede incluir una pluralidad de elementos sintácticos. Cuando el descodificador de vídeo 30 recibe el flujo de bits, la unidad de descodificación por entropía 150 puede realizar una operación de análisis sintáctico en el flujo de bits. Como resultado de realizar la operación de análisis sintáctico en el flujo de bits, la unidad de descodificación por entropía 150 puede extraer elementos sintácticos del flujo de bits. Como parte de realizar la operación de análisis sintáctico, la unidad de descodificación por entropía 150 puede realizar la descodificación por entropía de elementos sintácticos sometidos a codificación por entropía en el flujo de bits. La unidad de procesamiento de predicción 152, la unidad de cuantificación inversa 154, la unidad de transformada inversa 156, la unidad de reconstrucción 158 y la unidad de filtro 159 pueden realizar una operación de reconstrucción que genera datos de vídeo descodificados basados en los elementos sintácticos extraídos del flujo de bits.

[0105] Como se analiza anteriormente, el flujo de bits puede comprender una serie de unidades NAL. Las unidades NAL del flujo de bits pueden incluir unidades NAL de conjunto de parámetros de vídeo, unidades NAL de conjunto de parámetros de secuencia, unidades NAL de conjunto de parámetros de imagen, unidades NAL SEI, etc. Como parte de realizar la operación de análisis sintáctico en el flujo de bits, la unidad de descodificación por entropía 150 puede realizar operaciones de análisis sintáctico que extraen y descodifican por entropía conjuntos de parámetros de secuencia a partir de unidades NAL de conjuntos de parámetros de secuencia, conjuntos de parámetros de imagen a partir de unidades NAL de conjuntos de parámetros de imagen, datos SEI a partir de unidades NAL SEI, etc.

[0106] Además, las unidades NAL del flujo de bits pueden incluir unidades NAL de segmentos codificados. Como parte de realizar la operación de análisis sintáctico en el flujo de bits, la unidad de descodificación por entropía 150 puede realizar operaciones de análisis sintáctico que extraen y descodifican por entropía segmentos codificados a partir de las unidades NAL de segmentos codificados. Cada uno de los segmentos codificados puede incluir una cabecera de segmento y datos de segmento. La cabecera de segmento puede contener elementos sintácticos pertenecientes a un segmento. Los elementos sintácticos de la cabecera de segmento pueden incluir un elemento sintáctico que identifica un conjunto de parámetros de imagen asociado a una imagen que contiene el segmento. La unidad de descodificación por entropía 150 puede realizar operaciones de descodificación por entropía, tales como operaciones de descodificación de CABAC, en los elementos sintácticos de la cabecera del segmento codificado para recuperar la cabecera del segmento.

[0107] Como parte de extraer los datos del segmento de unidades NAL de segmentos codificados, la unidad de descodificación por entropía 150 puede realizar operaciones de análisis sintáctico que extraen los elementos sintácticos de las CU codificadas en los datos del segmento. Los elementos sintácticos extraídos pueden incluir elementos sintácticos asociados a bloques de coeficientes de transformada. La unidad de descodificación por entropía 150 puede entonces realizar operaciones de descodificación de CABAC en algunos de los elementos sintácticos.

[0108] Después de que la unidad de descodificación por entropía 150 realice una operación de análisis

5 sintáctico en una CU no dividida, el descodificador de vídeo 30 puede realizar una operación de reconstrucción en la CU no dividida. Para realizar la operación de reconstrucción en una CU no dividida, el descodificador de vídeo 30 puede realizar una operación de reconstrucción en cada TU de la CU. Al realizar la operación de reconstrucción para cada TU de la CU, el descodificador de vídeo 30 puede reconstruir un bloque de vídeo residual asociado a la CU.

10 **[0109]** Como parte de realizar una operación de reconstrucción en una TU, la unidad de cuantificación inversa 154 puede cuantificar de forma inversa, es decir, descuantificar, un bloque de coeficientes de transformada asociado a la TU. La unidad de cuantificación inversa 154 puede cuantificar de forma inversa el bloque de coeficientes de transformada de una manera similar a los procesos de cuantificación inversa propuestos para HEVC o definidos por la norma de descodificación H.264. La unidad de cuantificación inversa 154 puede usar un parámetro de cuantificación QP calculado por el codificador de vídeo 20 para una CU del bloque de coeficientes de transformada para determinar un grado de cuantificación y, del mismo modo, un grado de cuantificación inversa para que la unidad de cuantificación inversa 154 lo aplique.

15 **[0110]** Después de que la unidad de cuantificación inversa 154 cuantifique de forma inversa un bloque de coeficientes de transformada, la unidad de transformada inversa 156 puede generar un bloque de vídeo residual para la TU asociada con el bloque de coeficientes de transformada. La unidad de transformada inversa 156 puede aplicar una transformada inversa al bloque de coeficientes de transformada para generar el bloque de vídeo residual para la TU. Por ejemplo, la unidad de transformada inversa 156 puede aplicar una DCT inversa, una transformada de número entero inversa, una transformada de Karhunen-Loeve (KLT) inversa, una transformada de rotación inversa, una transformada direccional inversa u otra transformada inversa al bloque de coeficientes de transformada. En algunos ejemplos, la unidad de transformada inversa 156 puede determinar una transformada inversa que se va a aplicar al bloque de coeficientes de transformada en base a la señalización del codificador de vídeo 20. En dichos ejemplos, la unidad de transformada inversa 156 puede determinar la transformada inversa basándose en una transformada señalizada en el nodo raíz de un árbol cuaternario para un bloque de árbol asociado al bloque de coeficientes de transformada. En otros ejemplos, la unidad de transformada inversa 156 puede inferir la transformada inversa a partir de una o más características de codificación, tales como tamaño de bloque, modo de codificación o similares. En algunos ejemplos, la unidad de transformada inversa 156 puede aplicar una transformada inversa en cascada.

20 **[0111]** En algunos ejemplos, la unidad de compensación de movimiento 162 puede refinar el bloque de vídeo predicho de una PU mediante interpolación basándose en filtros de interpolación. Los identificadores para los filtros de interpolación que van a usarse para la compensación de movimiento con una precisión de submuestra pueden incluirse en los elementos sintácticos. La unidad de compensación de movimiento 162 puede usar los mismos filtros de interpolación usados por el codificador de vídeo 20 durante la generación del bloque de vídeo predicho de la PU para calcular valores interpolados para muestras de subenteros de un bloque de referencia. La unidad de compensación de movimiento 162 puede determinar los filtros de interpolación usados por el codificador de vídeo 20 de acuerdo con la información sintáctica recibida y usar los filtros de interpolación para generar el bloque de vídeo predicho.

25 **[0112]** Como se analiza adicionalmente más adelante con referencia a las **FIGS. 5-8**, la unidad de procesamiento de predicción 152 puede codificar (por ejemplo, codificar o descodificar) la PU (o cualquier otra capa de referencia y/o bloques de capa de mejora o unidades de vídeo) realizando los procedimientos ilustrados en las **FIGS. 5-8**. Por ejemplo, la unidad de compensación de movimiento 162, la unidad de intrapredicción 164 o la unidad de predicción entre capas 166 pueden configurarse para realizar los procedimientos ilustrados en las **FIGS. 5-8**, ya sea juntos o por separado.

30 **[0113]** Si se codifica una PU usando intrapredicción, la unidad de intrapredicción 164 puede realizar la intrapredicción para generar un bloque de vídeo predicho para la PU. Por ejemplo, la unidad de intrapredicción 164 puede determinar un modo de intrapredicción para la PU basándose en los elementos sintácticos del flujo de bits. El flujo de bits puede incluir elementos sintácticos que la unidad de intrapredicción 164 puede usar para determinar el modo de intrapredicción de la PU.

35 **[0114]** En algunos casos, los elementos sintácticos pueden indicar que la unidad de intrapredicción 164 tiene que usar el modo de intrapredicción de otra PU para determinar el modo de intrapredicción de la PU actual. Por ejemplo, puede ser probable que el modo de intrapredicción de la PU actual sea el mismo que el modo de intrapredicción de una PU próxima. En otras palabras, el modo de intrapredicción de la PU vecina puede ser el modo más probable para la PU actual. Por lo tanto, en este ejemplo, el flujo de bits puede incluir un pequeño elemento sintáctico que indica que el modo de intrapredicción de la PU es el mismo que el modo de intrapredicción de la PU próxima. La unidad de intrapredicción 164 puede entonces usar el modo de intrapredicción para generar datos de predicción (por ejemplo, muestras predichas) para la PU basándose en los bloques de vídeo de las PU espacialmente vecinas.

40 **[0115]** Como se analiza anteriormente, el descodificador de vídeo 30 también puede incluir una unidad de predicción entre capas 166. La unidad de predicción entre capas 166 está configurada para predecir un bloque

actual (por ejemplo, un bloque actual en la EL) usando una o más capas diferentes que están disponibles en SVC (por ejemplo, una capa base o de referencia). Dicha predicción se puede denominar predicción entre capas. La unidad de predicción entre capas 166 utiliza procedimientos de predicción para reducir la redundancia entre capas, mejorando así la eficacia de codificación y reduciendo los requisitos de recursos informáticos. Algunos ejemplos de predicción entre capas incluyen la intrapredicción entre capas, la predicción de movimiento entre capas y la predicción residual entre capas. La intrapredicción entre capas usa la reconstrucción de bloques ubicados conjuntamente en la capa base para predecir el bloque actual en la capa de mejora. La predicción de movimiento entre capas usa la información de movimiento de la capa base para predecir el movimiento en la capa de mejora. La predicción residual entre capas usa el residuo de la capa base para predecir el residuo de la capa de mejora. Cada uno de los esquemas de predicción entre capas se analiza a continuación con mayor detalle.

[0116] La unidad de reconstrucción 158 puede usar los bloques de vídeo residuales asociados a las TU de una CU y los bloques de vídeo predichos de las PU de la CU, es decir, datos de intrapredicción o datos de interpredicción, según corresponda, para reconstruir el bloque de vídeo de la CU. Por tanto, el descodificador de vídeo 30 puede generar un bloque de vídeo predicho y un bloque de vídeo residual en base a unos elementos sintácticos del flujo de bits y puede generar un bloque de vídeo en base al bloque de vídeo predicho y el bloque de vídeo residual.

[0117] Después de que la unidad de reconstrucción 158 reconstruya el bloque de vídeo de la CU, la unidad de filtro 159 puede realizar una operación de desbloqueo para reducir los artefactos de bloqueo asociados a la CU. Después de que la unidad de filtro 159 realice una operación de eliminación de bloques para reducir los artefactos de bloque asociados a la CU, el descodificador de vídeo 30 puede almacenar el bloque de vídeo de la CU en la memoria intermedia de imágenes descodificadas 160. La memoria intermedia de imágenes descodificadas 160 puede proporcionar imágenes de referencia para una posterior compensación de movimiento, intrapredicción y presentación en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la **FIG. 1A** o **1B**. Por ejemplo, el descodificador de vídeo 30 puede realizar, en base a los bloques de vídeo de la memoria intermedia de imágenes descodificadas 160, operaciones de intrapredicción o de interpredicción en las PU de otras CU.

Descodificador multicapa

[0118] La **FIG. 3B** es un diagrama de bloques que ilustra un ejemplo de descodificador de vídeo multicapa 33 que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación. El descodificador de vídeo 33 puede configurarse para procesar tramas de vídeo multicapa, tal como para codificación SHVC y de múltiples visualizaciones. Además, el descodificador de vídeo 33 se puede configurar para realizar cualquiera o todas las técnicas de esta divulgación.

[0119] El descodificador de vídeo 33 incluye un descodificador de vídeo 30A y un descodificador de vídeo 30B, cada uno de los cuales puede configurarse como el descodificador de vídeo 30 y puede realizar las funciones descritas anteriormente con respecto al descodificador de vídeo 30. Además, como se indica por la reutilización de los números de referencia, los descodificadores de vídeo 30A y 30B pueden incluir al menos algunos de los sistemas y subsistemas como el descodificador de vídeo 30. Aunque el descodificador de vídeo 33 se ilustra como que incluye dos descodificadores de vídeo 30A y 30B, el descodificador de vídeo 33 no está limitado como tal y puede incluir cualquier número de capas de descodificador de vídeo 30. En algunos modos de realización, el descodificador de vídeo 33 puede incluir un descodificador de vídeo 30 para cada imagen o trama en una unidad de acceso. Por ejemplo, una unidad de acceso que incluye cinco imágenes puede procesarse o descodificarse mediante un descodificador de vídeo que incluye cinco capas de descodificador. En algunos modos de realización, el descodificador de vídeo 33 puede incluir más capas de descodificador que tramas en una unidad de acceso. En algunos de dichos casos, algunas de las capas del descodificador de vídeo pueden estar inactivas al procesar algunas unidades de acceso.

[0120] Además de los descodificadores de vídeo 30A y 30B, el descodificador de vídeo 33 puede incluir una unidad de muestreo ascendente 92. En algunos modos de realización, la unidad de muestreo ascendente 92 puede realizar muestreo ascendente de una capa base de una trama de vídeo recibida para crear una capa mejorada para ser añadida a la lista de imágenes de referencia para la trama o la unidad de acceso. Esta capa mejorada se puede almacenar en la memoria intermedia de imágenes descodificadas 160. En algunos modos de realización, la unidad de muestreo ascendente 92 puede incluir algunos o todos los modos de realización descritos con respecto a la unidad de remuestreo 90 de la **FIG. 2A**. En algunos modos de realización, la unidad de muestreo ascendente 92 está configurada para realizar muestreo ascendente de una capa y reorganizar, redefinir, modificar o ajustar uno o más segmentos para cumplir con un conjunto de reglas de límite de segmento y/o reglas de escaneo de trama. En algunos casos, la unidad de muestreo ascendente 92 puede ser una unidad de remuestreo configurada para el muestreo ascendente y/o muestreo descendente de una capa de una trama de vídeo recibida.

[0121] La unidad de muestreo ascendente 92 puede estar configurada para recibir una imagen o trama (o

información de imagen asociada a la imagen) desde la memoria intermedia de imágenes descodificadas 160 del descodificador de capa inferior (por ejemplo, el descodificador de vídeo 30A) y para el muestreo ascendente de la imagen (o la información de imagen recibida). Esta imagen muestreada ascendentemente puede entonces proporcionarse a la unidad de procesamiento de predicción 152 de un descodificador de capa superior (por ejemplo, el descodificador de vídeo 30B) configurado para descodificar una imagen en la misma unidad de acceso que el descodificador de capa inferior. En algunos casos, el descodificador de capa superior es una capa eliminada del descodificador de capa inferior. En otros casos, puede haber uno o más descodificadores de capa superior entre el descodificador de capa 0 y el descodificador de capa 1 de la

10 **FIG. 3B.**

[0122] En algunos casos, la unidad de muestreo ascendente 92 se puede omitir o anular. En dichos casos, la imagen de la memoria intermedia de imágenes descodificadas 160 del descodificador de vídeo 30A se puede proporcionar directamente, o al menos sin proporcionarse a la unidad de muestreo ascendente 92, a la unidad de procesamiento de predicción 152 del descodificador de vídeo 30B. Por ejemplo, si los datos de vídeo proporcionados al descodificador de vídeo 30B y la imagen de referencia de la memoria intermedia de imágenes descodificadas 160 del descodificador de vídeo 30A son del mismo tamaño o resolución, la imagen de referencia puede proporcionarse al descodificador de vídeo 30B sin muestreo ascendente. Además, en algunos modos de realización, la unidad de muestreo ascendente 92 puede ser una unidad de remuestreo 90 configurada para realizar muestreo ascendente o muestreo descendente de la imagen de referencia recibida de la memoria intermedia de imágenes descodificadas 160 del descodificador de vídeo 30A.

[0123] Como se ilustra en la **FIG. 3B**, el descodificador de vídeo 33 puede incluir además un desmultiplexor 99 o demux. El demux 99 puede dividir un flujo de bits de vídeo codificado en múltiples flujos de bits con cada flujo de bits enviado por el demux 99 que se proporciona a un descodificador de vídeo diferente 30A y 30B. Los múltiples flujos de bits pueden crearse al recibir un flujo de bits y cada uno de los descodificadores de vídeo 30A y 30B recibe una parte del flujo de bits en un momento determinado. Mientras que en algunos casos los bits del flujo de bits recibidos en el demux 99 pueden alternarse un bit a la vez entre cada uno de los descodificadores de vídeo (por ejemplo, los descodificadores de vídeo 30A y 30B en el ejemplo de la **FIG. 3B**), en muchos casos el flujo de bits se divide de manera diferente. Por ejemplo, el flujo de bits se puede dividir alternando qué descodificador de vídeo recibe el flujo de bits un bloque a la vez. En otro ejemplo, el flujo de bits puede dividirse por una proporción de bloques diferente a 1:1 a cada uno de los descodificadores de vídeo 30A y 30B. Por ejemplo, pueden proporcionarse dos bloques al descodificador de vídeo 30B por cada bloque proporcionado al descodificador de vídeo 30A. En algunos modos de realización, la división del flujo de bits por el demux 99 puede preprogramarse. En otros modos de realización, el demux 99 puede dividir el flujo de bits en base a una señal de control recibida desde un sistema externo al descodificador de vídeo 33, tal como desde un procesador en un dispositivo de destino que incluye el módulo de destino 14. La señal de control puede generarse en base a la resolución o la velocidad de bits de un vídeo desde la interfaz de entrada 28, en base al ancho de banda del enlace 16, en base a una suscripción asociada a un usuario (por ejemplo, una suscripción de pago frente a una suscripción gratuita) o en base a cualquier otro factor para determinar una resolución obtenible por el descodificador de vídeo 33.

Codificación de vídeo de solo HLS escalable basada en HEVC

[0124] En algunas implementaciones (por ejemplo, la extensión escalable a HEVC), se puede desear la codificación de vídeo de sintaxis de alto nivel (HLS) solamente. Por ejemplo, la codificación de vídeo de solo HLS puede significar que no se introducen cambios a nivel de bloque o de unidad de codificación en dichas implementaciones. Los modos de codificación tal como IntraBL no se pueden usar en conexión con dichas implementaciones porque dichos modos de codificación pueden implicar cambios a nivel de bloque en el marco de HEVC. Por ejemplo, IntraBL implica acceder directamente a la información de textura del bloque en una capa de referencia que se localiza junto con el bloque actual en la capa de mejora que se está codificando.

[0125] Sin embargo, la codificación de vídeo de solo HLS se puede lograr tomando la imagen de la capa de referencia (por ejemplo, después de cualquier procesamiento aplicable tal como el remuestreo) e insertándola en la lista de imágenes de referencia de la imagen de la capa de mejora. En este caso, la imagen de referencia insertada en la lista de imágenes de referencia simplemente se puede tratar como otra imagen en la capa de mejora y se puede usar para la interpretación, sin requerir ningún cambio a nivel de bloque en el marco de HEVC. Si la capa de referencia y la capa de mejora tienen la misma resolución espacial, se puede usar un enfoque descrito en "MV-HEVC Working Draft 4 (JCT3V-D1004)", que está disponible en http://phenix.it-sudparis.eu/jct2/doc_end_user/documents/4_Incheon/wg11/JCT3V-D1004-v3.zip. En este enfoque, la imagen insertada en la lista de imágenes de referencia de la capa de mejora se puede usar como la imagen de localización conjunta (por ejemplo, la imagen en la misma capa desde la que se deriva la información de movimiento) para la derivación de TMVP sin procesamiento a nivel de bloque adicional.

65 **Interpredicción y TMVP**

[0126] En la interpredicción, un bloque actual en una imagen de capa de mejora (EL) se puede predecir usando la información de movimiento de una imagen temporalmente adyacente en la misma capa. Por ejemplo, un predictor de vector de movimiento temporal (TMVP) se puede derivar de un bloque de localización conjunta (por ejemplo, un bloque localizado en la misma posición en la imagen temporalmente adyacente que el bloque actual en la imagen actual) en una imagen de localización conjunta (por ejemplo, la imagen temporalmente adyacente en la misma capa) y agregada a la lista de candidatos del vector de movimiento (MV) del bloque actual en la EL.

[0127] El uso de TMVP puede mejorar la eficacia de codificación. Por ejemplo, un TMVP se puede usar como un predictor para el vector de movimiento del bloque que se está codificando actualmente (por ejemplo, el bloque actual), y una diferencia de vector de movimiento (MVD) entre el vector de movimiento del bloque actual y el TMVP se puede codificar en lugar de codificar todo el vector de movimiento del bloque actual.

Escalabilidad espacial

[0128] En un caso de escalabilidad espacial, la imagen de EL tiene una resolución espacial diferente a la imagen de la capa de referencia (RL). Por ejemplo, la relación de aspecto espacial entre la EL y la RL puede ser 1,5, 2,0 u otras relaciones. En este caso, la imagen de RL se puede volver a muestrear de modo que la imagen de RL muestreada coincida con la resolución espacial de la imagen de EL antes de insertar la imagen de RL remuestreada en las listas de referencia de EL como se describe en "SHVC Working Draft 2", que está disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/13_Incheon/ws11/JCTVC-M1008-v3.zip.

[0129] Para habilitar tanto la predicción de textura como la predicción de movimiento, se puede volver a muestrear información de píxeles y no píxeles (sintaxis, modos y parámetros de modo) asociados con la imagen de RL. Por ejemplo, el modo de predicción y la información de movimiento, tal como los vectores de movimiento, los índices de referencia, las interdirecciones se pueden volver a muestrear, de modo que se permita el uso de TMVP. Además de la información de nivel de bloque mencionada anteriormente, el proceso de derivación de TMVP también puede usar información de alto nivel de la imagen de localización conjunta (por ejemplo, imagen de RL remuestreada insertada en la lista de imágenes de referencia de la EL), tal como la información del tipo de imagen y la lista de imágenes de referencia de la misma.

Generación de una imagen de referencia entre capas (ILRP)

[0130] Las **FIGS. 4A y 4B** ilustran un procedimiento de ejemplo para generar una imagen de referencia entre capas usando una imagen de capa de referencia. Como se analiza anteriormente, en algunas implementaciones, la imagen de referencia entre capas generada se agrega a la lista de imágenes de referencia de la capa de mejora y se usa para codificar la imagen de la capa de mejora. La **FIG. 4A** ilustra una imagen de capa de referencia 402, que incluye una serie de segmentos 404A y 404B. Aunque solo se muestran dos segmentos en el ejemplo de la **FIG. 4A**, el número de segmentos no está limitado como tal, y puede ser cualquier número arbitrario. Como se ilustra en la **FIG. 4A**, la división entre los dos segmentos 404A y 404B se identifica por el borde de segmento 406. El borde de segmento 406 se distingue de las líneas de cuadrícula que dividen la imagen de capa de referencia 402 en unidades de codificación por el espesor aumentado del borde de segmento 406 en comparación con las líneas de cuadrícula.

[0131] Cada uno de los segmentos pueden estar asociados con información específica del segmento y/o información sintáctica específica del segmento. Esta información de segmento se puede incluir en una cabecera de segmento para cada segmento 404A y 404B o se puede almacenar en otro lugar y asociarse con el segmento, tal como a través de una tabla de mapeo u otra estructura de datos. La información del segmento puede incluir cualquier tipo de información que pueda asociarse con un segmento, tal como el valor de recuento de orden de imágenes (POC), el plano de color, la dirección del primer bloque de árbol de codificación en el segmento, la información de la lista de imágenes de referencia, etc. Esta información de segmento se puede definir por la norma HEVC. Sin embargo, la información de segmento no está limitada como tal y puede incluir otra información basada en normas y/o información específica de la aplicación, que puede o no estar asociada con una norma particular.

[0132] La imagen de capa de referencia 402 se puede dividir en varias unidades de codificación 408 o bloques de árbol de codificación (CTB) como se indica mediante las líneas de cuadrícula. Las unidades de codificación pueden ser de diferentes tamaños de píxeles. Por ejemplo, las unidades de codificación pueden ser de 16x16 píxeles o 64x64 píxeles. En algunos casos, las unidades de codificación pueden denominarse unidades de codificación más grandes (LCU). Los segmentos 404A y 404B pueden incluir varias LCU. En el ejemplo representado, el segmento 404A incluye 5 LCU y el segmento 404B incluye 11 LCU. Aunque cada bloque o CTB de la imagen de capa de referencia 402 se representa con un tamaño equivalente, en algunos modos de realización, los CTB de la capa 402 pueden variar en cuanto al tamaño. Aunque la imagen de capa de referencia 402 se ilustra en la **FIG. 4A**, la imagen de capa de referencia 402 puede ser cualquier imagen de capa base o una imagen de cualquier otra capa.

[0133] La **FIG. 4B** ilustra una imagen de capa de referencia remuestreada 412, que es una versión

remuestreada de la imagen de capa de referencia 402. La imagen de la capa de referencia remuestreada 412 también puede denominarse imagen de referencia entre capas 412. La imagen de la capa de referencia remuestreada 412 puede representar una imagen que se puede incluir en la lista de imágenes de referencia de la capa de mejora (o cualquier otra capa que tenga una resolución más alta que la capa de referencia). Esta imagen de la capa de referencia remuestreada 412 puede, en algunos casos, usarse para codificar (por ejemplo, codificar o decodificar) una imagen de capa de mejora. Como se ilustra en la **FIG. 4B**, la imagen de la capa de referencia remuestreada 412 representa un muestreo ascendente de la imagen de la capa de referencia 402 por 2x, o un muestreo ascendente diádico. En el ejemplo de la **FIG. 4B**, la imagen de la capa de referencia remuestreada 412 comprende unidades de codificación 418 que tienen el mismo tamaño que las de la imagen de capa de referencia 402, pero la imagen de capa de referencia 412 remuestreada comprende cuatro veces la misma cantidad de unidades de codificación 418 que de imágenes de capa de referencia 402. En otros modos de realización, el muestreo ascendente puede incluir ajuste a escala. Por ejemplo, cada unidad de codificación dentro de la imagen de la capa de referencia remuestreada 412 puede ser 2 veces tan larga y 2 veces tan ancha como cada unidad de codificación en la imagen de la capa de referencia 402 cuando se aplica un muestreo ascendente de 2x. En otros casos, cada unidad de codificación de la imagen de capa de referencia remuestreada 412 puede tener el mismo tamaño que las unidades de codificación en la imagen de capa de referencia 402. En algunos casos, el muestreo ascendente está restringido a una escala específica, tal como una escala entera (por ejemplo, 2x, 3x, 5x, etc.). En otros casos, el muestreo ascendente puede no estar restringido y puede incluir un muestreo ascendente no basado en enteros (por ejemplo, 1,5x, 3,3x, etc.). Además, aunque los modos de realización en el presente documento se describen principalmente usando muestreo ascendente, algunos modos de realización pueden incluir muestreo descendente (por ejemplo, 0,5x).

[0134] Como se ilustra en la **FIG. 4B**, la imagen de la capa de referencia remuestreada 412 incluye un solo segmento 414. Como se analiza en el presente documento, la información de segmento del segmento 414 se puede derivar del segmento 404A, el segmento 404B o ambos. En un modo de realización, el proceso de "derivar" información de segmento de otro segmento comprende copiar sobre la información de segmento de ese segmento. Por ejemplo, el segmento 414 puede tener la misma información de segmento que el segmento 404A. En otro ejemplo, el segmento 414 puede tener la misma información de segmento que el segmento 404B. Aún en otro ejemplo, el segmento 404 puede tener alguna información derivada del segmento 404A y otra información derivada del segmento 404B. Como la imagen de la capa de referencia remuestreada 412 tiene un segmento 414, no se ilustra ningún borde de segmento en la **FIG. 4B**. Como se analiza anteriormente, debido a que la imagen de la capa de referencia remuestreada 412 tiene un segmento y ningún borde de segmento, se elimina la necesidad de realizar un procesamiento adicional para garantizar la conformidad del flujo de bits de múltiples segmentos (por ejemplo, el procesamiento del orden de escaneo de trama).

Datos incluidos en la imagen de referencia entre capas

[0135] Por ejemplo, en algunos modos de realización, generar una imagen de referencia entre capas (por ejemplo, una imagen de referencia generada en base a una imagen de RL, que se usa para la interpredicción de la EL) que no implica cambios de proceso de bajo nivel (por ejemplo, nivel de bloque) en HEVC-SVC de solo HLS en el caso de escalabilidad espacial: (A) datos de YUV de muestra ascendente de la imagen de RL; (B) parámetros de movimiento de la muestra ascendente (por ejemplo, vectores de movimiento e índices de referencia) de la imagen de RL; y (C) obtener información de alto nivel (por ejemplo, imagen o nivel de segmento) de la imagen de RL. Por ejemplo, la imagen de referencia entre capas (ILRP) generada puede incluir los siguientes datos derivados de la imagen de RL: (A) imagen de textura remuestreada que tiene la misma resolución espacial que la imagen de EL; (B) parámetros de movimiento para cada unidad básica (por ejemplo, unidad de codificación o unidad de predicción); y (C) información de alto nivel (por ejemplo, imagen o nivel de segmento).

[0136] La información de alto nivel (por ejemplo, el elemento (C) anterior) de la imagen de referencia entre capas generada puede incluir el valor de POC de la imagen, el patrón de división de segmento, los tipos de segmento, la información de la lista de imágenes de referencia y cualquier otra información de alto nivel que pueda ser usada en el proceso de derivación de TMVP. La información de la lista de imágenes de referencia puede incluir el número de imagen de referencia en cada lista de imágenes de referencia, el tipo de imagen de referencia (por ejemplo, a corto plazo o a largo plazo) de todas las imágenes de referencia, los valores de POC de imagen de referencia de todas las imágenes de referencia. En algunas implementaciones, la información de la lista de imágenes de referencia puede incluir cualquiera de (1) el número de imágenes de referencia en la lista de imágenes de referencia 0, (2) la imagen de referencia correspondiente a cada índice de imagen de referencia de la lista de imágenes de referencia 0 (por ejemplo, la imagen de referencia asociada con un índice de referencia de 0, la imagen de referencia asociada con un índice de referencia de 1, etc.), (3) el número de imágenes de referencia en la lista de imágenes de referencia 1, y/o (4) la imagen de referencia correspondiente a cada índice de imagen de referencia de la lista de imágenes de referencia 1. Parte de la información de alto nivel de la imagen de referencia entre capas generada se puede copiar simplemente a partir de la de la imagen de RL. De forma alternativa, parte de la información de alto nivel de la imagen de referencia entre capas generada se puede establecer en un valor predefinido. Para la información a nivel de imagen, tal como el valor de POC de la imagen, se puede copiar directamente a partir de la imagen de BL relacionada. Sin embargo, parte de otra información,

tal como el tipo de segmento y la información de la imagen de la lista de referencia, está asociada con segmentos individuales (por ejemplo, cada segmento contiene su propio tipo de segmento y lista de imágenes de referencia). Dicha información de la imagen de referencia entre capas generada necesitaría derivarse al nivel de segmento.

[0137] En algunas implementaciones, la modificación del límite de segmento y/o la definición de segmento de la imagen de RL remuestreada al generar la información de alto nivel de la imagen de referencia entre capas en el caso de segmentos múltiples se realiza en base a la segmentación de segmento en la imagen de RL correspondiente. Por ejemplo, dependiendo de cómo se divide la imagen de RL correspondiente en múltiples segmentos y dónde se localizan los límites del segmento, en algunos casos es conveniente modificar los límites de segmento de la imagen de RL remuestreada de modo que la imagen de RL remuestreada resultante se ajuste a las normas y restricciones que pueden aplicarse.

[0138] En el caso de la SVC escalable MV-HEVC (multivista) o la relación señal/ruido (SNR), la información de segmento de la imagen de referencia generada se puede copiar directamente desde la capa de referencia correspondiente o la imagen de vista de referencia, ya que la imagen de referencia generada puede tener el mismo patrón de segmento con la imagen de la capa base/vista base (por ejemplo, si no se realiza un remuestreo espacial). Por otro lado, en el caso de la escalabilidad espacial, la derivación del patrón de división de segmento se puede hacer primero remuestreando el patrón de división de segmento de la imagen de RL correspondiente, y a continuación, ajustando el patrón de división de segmento de modo que la imagen de referencia resultante se ajuste a cualquiera de las normas o restricciones aplicables (por ejemplo, reglas de división de segmento de HEVC, tal como los segmentos que deben contener un conjunto de LCU continuas en orden de escaneo de trama).

ILRP de segmento individual

[0139] En algunos modos de realización, cuando se vuelve a muestrear una imagen de capa de referencia en base a la relación de escalabilidad para generar una imagen de referencia entre capas, se genera un segmento individual para la imagen de capa de referencia remuestreada incluso si la imagen de capa de referencia tiene múltiples segmentos definidos en la misma. Por ejemplo, todos los bloques de árbol de codificación (CTB) en la imagen de capa de referencia remuestreada están asociados con el segmento individual. La información de segmento del segmento individual de la imagen de capa de referencia remuestreada se genera usando la información de segmento de uno o más segmentos de la imagen de capa de referencia. Por ejemplo, la información de segmento del segmento individual de la imagen de capa de referencia remuestreada se establece igual a la información de segmento de uno de los segmentos de la imagen de capa de referencia. En un modo de realización, la información de segmento incluye un tipo de segmento y una lista de imágenes de referencia. Sin embargo, la información de segmento no se limita a las descritas en el presente documento, sino que puede incluir cualquier parámetro o variable que pueda señalizarse o recibirse a nivel del segmento (por ejemplo, en la cabecera del segmento).

Segmentos que tienen diferentes listas de imágenes de referencia

[0140] Si una imagen de capa de referencia tiene múltiples segmentos (por ejemplo, segmento # 1 y segmento # 2) y tienen diferentes listas de imágenes de referencia, puede haber un problema. Por ejemplo, si la imagen de la capa de referencia se vuelve a muestrear y se genera un segmento individual para la imagen de capa de referencia remuestreada, y la información del segmento individual se genera en base a la información del segmento # 1 de la imagen de capa de referencia, los bloques en la imagen de capa de referencia remuestreada que corresponde al segmento # 2 de la imagen de capa de referencia pueden tener un índice de referencia (por ejemplo, un valor que se usa para identificar una imagen en la lista de imágenes de referencia) que ya no es válido porque la lista de imágenes de referencia del segmento individual de la imagen de capa de referencia remuestreada es diferente de la lista de imágenes de referencia del segmento # 2 de la imagen de capa de referencia. En este ejemplo, la lista de imágenes de referencia de la imagen de capa de referencia remuestreada sería la misma que la del segmento # 1 de la imagen de capa de referencia, la cual es diferente de la lista de imágenes de referencia del segmento # 2.

[0141] En un ejemplo, el segmento # 1 tiene tres imágenes de referencia y el segmento # 2 tiene cinco imágenes de referencia, y la información de segmento del segmento individual de la imagen de capa de referencia remuestreada se deriva de la información de segmento del segmento # 1 (por ejemplo, la lista de imágenes de referencia del segmento individual es igual a la lista de imágenes de referencia del segmento # 1). Sin embargo, algunos bloques de la imagen de la capa de referencia remuestreada que corresponden al segmento # 2 pueden tener valores de índice de referencia de 3 o 4, aunque la lista de imágenes de referencia del segmento individual tiene tres imágenes de referencia.

[0142] Como se describe anteriormente, el proceso de generar o derivar la información de alto nivel de la imagen de referencia entre capas generada (por ejemplo, el patrón de división de segmento y otra información asociada con cada segmento) puede ser bastante complejo. Diversos modos de realización de la presente

solicitud pueden proporcionar un proceso simplificado para generar o derivar información a nivel de segmento (por ejemplo, evitando generar múltiples segmentos para la imagen de referencia entre capas remuestreada o generando un segmento individual para la imagen de referencia entre capas remuestreada).

5 **Enfoque # 1: Restricción de predicción de movimiento entre capas**

[0143] Una solución al problema que surge cuando los segmentos de la imagen de la capa de referencia tienen información de segmento diferente (por ejemplo, la lista de imágenes de referencia) es la restricción de la predicción de movimiento entre capas al caso en el que los múltiples segmentos de la imagen de la capa de referencia tienen listas de imágenes de referencia idénticas. Por ejemplo, cuando se usa una imagen de capa de referencia remuestreada como la imagen de localización conjunta para la derivación de TMVP, la predicción de movimiento entre capas se deshabilita a menos que todos los segmentos en la imagen de capa de referencia tengan la misma lista de imágenes de referencia 0 y la imagen de referencia idéntica 1. Además, esta restricción también se puede aplicar a casos de escalabilidad de relación señal/ruido (SNR), o específicamente al caso de la escalabilidad de SNR con diferentes tamaños de CTB en la capa de mejora y la capa de referencia, como se describe más adelante en la solicitud.

[0144] La FIG. 5 es un diagrama de flujo que ilustra un procedimiento 500 para codificar información de vídeo, de acuerdo con un modo de realización de la presente divulgación. Las etapas ilustradas en la FIG. 5 pueden ser realizadas por un codificador (por ejemplo, el codificador de vídeo como se muestra en la FIG. 2A o FIG. 2B) u otro componente descrito en el presente documento. Por conveniencia, el procedimiento 500 se describe como realizado por un codificador, que puede ser, por ejemplo, un codificador de vídeo.

[0145] El procedimiento 500 comienza en el bloque 501. En el bloque 505, el codificador determina si los segmentos de la imagen de capa de referencia tienen información de segmento idéntica. En un modo de realización, la información de segmento incluye la lista de imágenes de referencia asociada con el segmento correspondiente. Si el codificador determina que los segmentos de la imagen de capa de referencia tienen información de segmento idéntica, el codificador habilita la predicción de movimiento entre capas en el bloque 510. Por ejemplo, el término "habilitar" puede significar, además de su significado original, permitir que el codificador considere la predicción de movimiento entre capas como una de las opciones de codificación para codificar el bloque actual o la imagen actual. En otro ejemplo, el término "habilitar" puede significar simplemente abstenerse de restringir (o deshabilitar) el uso de la predicción de movimiento entre capas y no significa necesariamente que el bloque actual o la imagen actual se codifique usando la predicción de movimiento entre capas. Puede significar simplemente que si el codificador determina que la predicción de movimiento entre capas es el procedimiento deseado para codificar el bloque actual o la imagen actual, puede hacerlo.

[0146] Por otra parte, si el codificador determina que los segmentos de la imagen de capa de referencia tienen información de segmento idéntica, el codificador deshabilita la predicción de movimiento entre capas en el bloque 515. Al restringir la predicción de movimiento entre capas a aquellas situaciones en las que los segmentos de la imagen de capa de referencia tienen información de segmento idéntica (por ejemplo, tipo de segmento, lista de imágenes de referencia, etc.), se puede implementar la generación de imágenes de referencia entre capas de un segmento individual de forma más fluida y la complejidad computacional asociada con la gestión de múltiples segmentos en imágenes de referencia entre capas se puede reducir o eliminar. El procedimiento 500 termina en el bloque 520.

[0147] Como se analiza anteriormente, uno o más componentes del codificador de vídeo 20 de la FIG. 2A o del codificador de vídeo 23 de la FIG. 2B (por ejemplo, la unidad de predicción entre capas 128) se puede usar para implementar cualquiera de las técnicas analizadas en la presente divulgación, tal como determinar si los segmentos de la imagen de capa de referencia tienen información de segmento idéntica y habilitar/deshabilitar la predicción de movimiento entre capas.

[0148] La FIG. 6 es un diagrama de flujo que ilustra un procedimiento 600 para codificar información de vídeo, de acuerdo con otro modo de realización de la presente divulgación. Una restricción similar a la de la FIG. 5 se aplica en el ejemplo de la FIG. 6, pero en este ejemplo, la predicción de movimiento entre capas es realizada en realidad por el codificador. Una o más etapas ilustradas en la FIG. 6 se pueden realizar mediante un codificador (por ejemplo, el codificador de vídeo como se muestra en la FIG. 2A o la FIG. 2B), un descodificador (por ejemplo, el descodificador de vídeo como se muestra en la FIG. 3A o la FIG. 3B) o cualquier otro componente. Por conveniencia, el procedimiento 700 se describe como realizado por un codificador, que puede ser el codificador, el descodificador u otro componente.

[0149] El procedimiento 600 comienza en el bloque 601. En el bloque 605, el codificador determina si los segmentos de la imagen de capa de referencia tienen información de segmento idéntica. En un modo de realización, la información de segmento incluye la lista de imágenes de referencia asociada con el segmento correspondiente. Si el codificador determina que los segmentos de la imagen de la capa de referencia tienen información de segmento idéntica, el codificador genera una imagen de referencia entre capas que tiene un segmento individual al muestrear ascendientemente la imagen de capa de referencia en el bloque 610. En el

bloque 615, el codificador establece la información de segmento de la imagen de referencia entre capas igual a la información de segmento de uno de los segmentos de la imagen de capa de referencia. Por ejemplo, la información de segmento del primer segmento de la imagen de la capa de referencia se copia en el segmento individual de la imagen de referencia entre capas. En el bloque 620, el codificador usa la imagen de referencia entre capas para codificar (por ejemplo, codificar o descodificar) al menos una parte de la imagen de la capa de mejora (por ejemplo, la imagen actual en la capa de mejora que se está codificando). Por ejemplo, como se analiza anteriormente, la información de movimiento asociada con un bloque en la capa de referencia entre capas (por ejemplo, una versión remuestreada de la imagen de la capa de referencia) que se localiza junto con un bloque actual en la capa de mejora se puede usar como un predictor de vector de movimiento temporal (TMVP). Por otra parte, si el codificador determina que los segmentos de la imagen de capa de referencia no tienen información de segmento idéntica, el procedimiento 600 termina en el bloque 620.

[0150] Como se analiza anteriormente, se puede usar uno o más componentes del codificador de vídeo 20 de la FIG. 2A, del codificador de vídeo 23 de la FIG. 2B, del descodificador de vídeo 30 de la FIG. 3A o del descodificador de vídeo 33 de la FIG. 3B (por ejemplo, la unidad de predicción entre capas 128 y/o la unidad de predicción entre capas 166) para implementar cualquiera de las técnicas analizadas en la presente divulgación, tal como determinar si los segmentos de la imagen de capa de referencia tienen información de segmento idéntica, generar una imagen de referencia entre capas, configurar la información de segmento del segmento individual en la imagen de capa de referencia remuestreada igual a uno de los segmentos en las imágenes de capa de referencia, y usar la imagen de referencia entre capas generada para codificar la capa de mejora.

Enfoque # 2: Modificación de la información de segmento

[0151] De forma alternativa, en un modo de realización, para evitar que los valores de índice estén fuera de los límites, los valores de índice de referencia se pueden recortar en el intervalo de -1 al valor de índice de referencia máximo de la lista de imágenes de referencia de la imagen de capa de referencia remuestreada. En un modo de realización, el valor de -1 indica que no hay predicción en esta dirección de referencia particular (por ejemplo, lista 0 o lista 1). Por ejemplo, un bloque unipredicho tiene un vector de movimiento que apunta a una sola dirección, no a ambas. Por tanto, si hay dos listas de imágenes de referencia, el índice de referencia para una de las dos direcciones puede ser -1 para bloques unipredichos.

[0152] En otro modo de realización, después de generar la imagen de la capa de referencia remuestreada, los valores del índice de referencia de los bloques que corresponden a un segmento de la imagen de la capa de referencia que tiene una lista de imágenes de referencia que es diferente de la del segmento individual de la imagen de la capa de referencia remuestreada se modifican de modo que identifiquen las imágenes de referencia correctas. Por ejemplo, la lista de imágenes de referencia del segmento # 1 puede ser [imagen # 1, imagen # 2] y la lista de imágenes de referencia del segmento # 2 puede ser [imagen # 2, imagen # 1]. En dicho caso, las listas de imágenes de referencia de los dos segmentos son diferentes. Sin embargo, después de configurar la lista de imágenes de referencia del segmento individual de la imagen de la capa de referencia remuestreada a, por ejemplo, la lista de imágenes de referencia del segmento # 1, para cualquier bloque correspondiente al segmento # 2 de la imagen de la capa de referencia, el valor del índice de referencia de 0 se puede cambiar a 1, y el valor del índice de referencia de 1 se puede cambiar a 0 para corregir la inexactitud resultante de la generación del segmento individual.

[0153] Aún en otro modo de realización, el cambio en la lista de imágenes de referencia se aborda escalando temporalmente los vectores de movimiento de esos bloques de modo que los vectores de movimiento apunten a las partes pertinentes de la nueva imagen de referencia identificada por los valores de índice de referencia y la nueva lista de imágenes de referencia. En un ejemplo, los vectores de movimiento se pueden escalar por un factor determinado en base a la distancia temporal entre la imagen actual y la imagen de referencia original y la distancia temporal entre la imagen actual y la nueva imagen de referencia. En otro ejemplo, los vectores de movimiento se pueden escalar en base a los valores de recuento de orden de imágenes (POC) de las imágenes de referencia.

[0154] Con referencia a las FIGS. 7 y 8, se describen procedimientos de codificación de ejemplo que se pueden realizar en el caso de que los segmentos de la imagen de la capa de referencia no tengan información de segmento idéntica. La FIG. 7 es un diagrama de flujo que ilustra un procedimiento 700 para codificar información de vídeo, de acuerdo con un modo de realización de la presente divulgación. Una o más etapas ilustradas en la FIG. 7 se pueden realizar mediante un codificador (por ejemplo, el codificador de vídeo como se muestra en la FIG. 2A o la FIG. 2B), un descodificador (por ejemplo, el descodificador de vídeo como se muestra en la FIG. 3A o la FIG. 3B) o cualquier otro componente. Por conveniencia, el procedimiento 700 se describe como realizado por un codificador, que puede ser el codificador, el descodificador u otro componente.

[0155] El procedimiento 700 comienza en el bloque 701. En el bloque 705, el codificador determina si los segmentos de la imagen de capa de referencia tienen información de segmento idéntica. En un modo de realización, la información de segmento incluye la lista de imágenes de referencia asociada con el segmento correspondiente. Si el codificador determina que los segmentos de la imagen de la capa de referencia tienen

información de segmento idéntica, el codificador genera una imagen de referencia entre capas que tiene un segmento individual al muestrear ascendentemente la imagen de capa de referencia en el bloque 710. En el bloque 715, el codificador establece la información de segmento de la imagen de referencia entre capas igual a la información de segmento de uno de los segmentos de la imagen de capa de referencia que tiene la lista de imágenes de referencia más larga. Por ejemplo, la información de segmento puede incluir la información de la lista de imágenes de referencia, y el primer segmento de la imagen de la capa de referencia está asociado con una lista de imágenes de referencia que tiene una longitud de 5 (por ejemplo, cinco imágenes), y el segundo segmento de la imagen de la capa de referencia está asociado con una lista de imágenes de referencia que tiene una longitud de 7 (por ejemplo, siete imágenes), la información de segmento del segundo segmento se puede copiar en el segmento individual de la imagen de referencia entre capas. En el bloque 720, el codificador usa la imagen de referencia entre capas para codificar (por ejemplo, codificar o decodificar) al menos una parte de la imagen de la capa de mejora (por ejemplo, la imagen actual en la capa de mejora que se está codificando). Por ejemplo, como se analiza anteriormente, la información de movimiento asociada con un bloque en la capa de referencia entre capas (por ejemplo, una versión remuestreada de la imagen de la capa de referencia) que se localiza junto con un bloque actual en la capa de mejora se puede usar como un predictor de vector de movimiento temporal (TMVP). Por otra parte, si el codificador determina que los segmentos de la imagen de capa de referencia no tienen información de segmento idéntica, el procedimiento 700 termina en el bloque 720.

[0156] Como se analiza anteriormente, se puede usar uno o más componentes del codificador de vídeo 20 de la **FIG. 2A**, del codificador de vídeo 23 de la **FIG. 2B**, del decodificador de vídeo 30 de la **FIG. 3A** o del decodificador de vídeo 33 de la **FIG. 3B** (por ejemplo, la unidad de predicción entre capas 128 y/o la unidad de predicción entre capas 166) para implementar cualquiera de las técnicas analizadas en la presente divulgación, tal como determinar si los segmentos de la imagen de capa de referencia tienen información de segmento idéntica, generar una imagen de referencia entre capas, configurar la información de segmento del segmento individual en la imagen de capa de referencia remuestreada igual a uno de los segmentos en las imágenes de capa de referencia que tienen una lista de imágenes de referencia más larga, y usar la imagen de referencia entre capas generada para codificar la capa de mejora.

[0157] La **FIG. 8** es un diagrama de flujo que ilustra un procedimiento 800 para codificar información de vídeo, de acuerdo con otro modo de realización de la presente divulgación. Mientras que el ejemplo de la **FIG. 7** tiene como objetivo proporcionar una capa de referencia entre capas conforme a la restricción de flujo de bits, el ejemplo de la **FIG. 8** tiene como objetivo mejorar aún más la eficacia de la codificación resultante del mismo. Una o más etapas ilustradas en la **FIG. 8** se pueden realizar mediante un codificador (por ejemplo, el codificador de vídeo como se muestra en la **FIG. 2A** o la **FIG. 2B**), un decodificador (por ejemplo, el decodificador de vídeo como se muestra en la **FIG. 3A** o la **FIG. 3B**) o cualquier otro componente. Por conveniencia, el procedimiento 700 se describe como realizado por un codificador, que puede ser el codificador, el decodificador u otro componente.

[0158] El procedimiento 800 comienza en el bloque 801. En el bloque 805, el codificador determina si los segmentos de la imagen de capa de referencia tienen información de segmento idéntica. En un modo de realización, la información de segmento incluye la lista de imágenes de referencia asociada con el segmento correspondiente. Si el codificador determina que los segmentos de la imagen de la capa de referencia tienen información de segmento idéntica, el codificador genera una imagen de referencia entre capas que tiene un segmento individual al muestrear ascendentemente la imagen de capa de referencia en el bloque 810. En el bloque 815, el codificador establece la información de segmento de la imagen de referencia entre capas igual a la información de segmento de uno de los segmentos de la imagen de capa de referencia. Por ejemplo, la información de segmento del primer segmento de la imagen de la capa de referencia se copia en el segmento individual de la imagen de referencia entre capas. En el bloque 820, el codificador modifica la información de movimiento asociada con una parte de la imagen de referencia entre capas que corresponde a segmentos no seleccionados de la imagen de referencia para corregir cualquier inexactitud en la misma. Por ejemplo, en este ejemplo, dado que la información de segmento del segmento individual de la capa de referencia entre capas se copió desde el primer segmento de la imagen de la capa de referencia, la parte de la imagen de referencia entre capas que corresponde a cualquier otro segmento en la capa de la imagen de referencia puede sufrir un proceso de modificación. Como se describe en el presente documento, el proceso de modificación puede incluir el ajuste a escala temporal de los vectores de movimiento en dicha parte o reasignar el índice de referencia de modo que los índices de referencia usados en dicha parte correspondan a la imagen de referencia correcta en la lista de imágenes de referencia.

[0159] Al restringir la predicción de movimiento entre capas a aquellas situaciones en las que los segmentos de la imagen de capa de referencia tienen información de segmento idéntica (por ejemplo, tipo de segmento, lista de imágenes de referencia, etc.), se puede implementar la generación de imágenes de referencia entre capas de un segmento individual de forma más fluida y la complejidad computacional asociada con la gestión de múltiples segmentos en imágenes de referencia entre capas se puede reducir o eliminar. El procedimiento 800 termina en el bloque 820.

[0160] Como se analiza anteriormente, se puede usar uno o más componentes del codificador de vídeo 20 de

la **FIG. 2A**, del codificador de vídeo 23 de la **FIG. 2B**, del descodificador de vídeo 30 de la **FIG. 3A** o del descodificador de vídeo 33 de la **FIG. 3B** (por ejemplo, la unidad de predicción entre capas 128 y/o la unidad de predicción entre capas 166) para implementar cualquiera de las técnicas analizadas en la presente divulgación, tal como determinar si los segmentos de la imagen de capa de referencia tienen información de segmento idéntica, generar una imagen de referencia entre capas, configurar la información de segmento del segmento individual en la imagen de capa de referencia remuestreada igual a uno de los segmentos en las imágenes de capa de referencia que tienen una lista de imágenes de referencia más larga, modificar la información de movimiento asociada con la imagen de referencia entre capas, y usar la imagen de referencia entre capas generada para codificar la capa de mejora.

Derivación de información de nivel de segmento

[0161] En un modo de realización, todos los bloques de árbol de codificación (CTB) de la imagen de la capa de referencia remuestreada (también denominada imagen de referencia entre capas) están asociados con el segmento individual que se genera para la imagen de la capa de referencia remuestreada. La información de segmento del segmento individual se puede derivar de cualquier segmento en la imagen de capa de referencia correspondiente. En un modo de realización, el tipo de segmento y la información de la lista de imágenes de referencia del segmento generado se establece igual a los de un segmento particular en la imagen de la capa de referencia. Por ejemplo, el segmento particular puede ser el primer segmento en la imagen de la capa de referencia. El siguiente texto se puede usar en el borrador de trabajo de SHVC como la implementación detallada de este procedimiento:

[0162] *El slice_type, num_ref_idx_l0_active_minus1 y num_ref_idx_l1_active_minus1 de rsSlice se establecen iguales al valor de slice_type, num_ref_idx_l0_active_minus1 y num_ref_idx_l1_active_minus1 del primer segmento de rPic, respectivamente. Cuando rsSlice es un segmento P o B, para i en el intervalo de 0 a num_ref_idx_l0_active_minus1 de rsSlice, inclusive, la imagen de referencia con índice i en la lista de imágenes de referencia 0 de rsSlice se establece igual a la imagen de referencia con índice i en la lista de imágenes de referencia 0 del primer segmento de rPic. Cuando rsSlice es un segmento B, para i en el intervalo de 0 a num_ref_idx_l1_active_minus1 de rsSlice, inclusive, la imagen de referencia con índice i en la lista de imágenes de referencia 1 de rsSlice se establece igual a la imagen de referencia con índice i en la lista de imágenes de referencia 1 del primer segmento de rPic.*

[0163] *Donde, rsPic es la referencia de la capa intermedia remuestreada, rsSlice es el segmento de rsPic; rPic es la imagen de capa de referencia correspondiente.*

Modos de realización alternativos

[0164] En otros modos de realización, la generación de la capa de referencia entre capas de un segmento individual todavía se puede realizar incluso si no todos los segmentos de la imagen de la capa de referencia correspondiente tienen información de segmento idéntica (por ejemplo, listas de imágenes de referencia). En dichos modos de realización, el tipo de segmento, por ejemplo, del segmento individual se puede derivar como sigue: (1) si hay al menos un segmento B en la imagen de la capa de referencia correspondiente, el tipo de segmento del segmento generado de la imagen de la capa de referencia remuestreada se establece igual al segmento B; (2) si no hay un segmento B en la imagen de la capa de referencia, y si hay al menos un segmento P en la imagen de la capa de referencia correspondiente, el tipo de segmento del segmento generado de la imagen de la capa de referencia remuestreada es igual al segmento P; y (3) si ni un segmento B ni un segmento P está presente en la imagen de capa de referencia correspondiente, el tipo de segmento del segmento generado de la imagen de capa de referencia remuestreada se establece igual al segmento I.

[0165] Además, en dichos modos de realización, la información de la lista de imágenes de referencia se puede derivar como sigue: (1) num_ref_idx_l0_active_minus1, que indica que el número de imágenes de referencia en la lista de imágenes de referencia, del segmento generado de la imagen de la capa de referencia remuestreada es igual al del segmento que tiene el mayor valor de num_ref_idx_l0_active_minus1 entre todos los segmentos en la imagen de la capa de referencia correspondiente; la lista de imágenes de referencia 0 del segmento generado se establece igual al del mismo segmento que tiene el mayor valor de num_ref_idx_l0_active_minus1. Por ejemplo, la memoria intermedia de imagen descodificada puede tener cinco imágenes, y el segmento # 1 puede tener dos imágenes de referencia en su lista de imágenes de referencia y el segmento # 2 puede tener cuatro imágenes de referencia en su lista de imágenes de referencia. En ese caso, num_ref_idx_l0_active_minus1 del segmento # 2 es mayor, por lo que la lista de imágenes de referencia del segmento individual generado para la imagen de la capa de referencia remuestreada se establece igual a la lista de imágenes de referencia del segmento # 2 de la imagen de la capa de referencia.

[0166] En el caso de que múltiples segmentos tengan el mismo valor de num_ref_idx_l0_active_minus1, la lista de imágenes de referencia 0 del segmento generado de la imagen de la capa de referencia remuestreada se establece igual al del segmento con la slice_segment_address menor entre todos los segmentos que tienen el mayor valor de num_ref_idx_l0_active_minus1; (2) num_ref_idx_l1_active_minus1 del segmento generado de la

imagen de la capa de referencia remuestreada se establece igual al del segmento que tiene el mayor valor de num_ref_idx_l1_active_minus1 entre todos los segmentos en la imagen de la capa de referencia correspondiente; la lista de imágenes de referencia 1 del segmento generado de la imagen de la capa de referencia remuestreada se establece igual a la del mismo segmento que tiene el mayor valor de num_ref_idx_l1_active_minus1. En el caso de que múltiples segmentos tengan el mismo valor de num_ref_idx_l1_active_minus1, la lista de imágenes de referencia 1 del segmento generado se establece igual a la del segmento con la slice_segment_address menor entre todos los segmentos que tienen el mayor valor de num_ref_idx_l1_active_minus1.

Restricción normativa del codificador

[0167] Como se describe anteriormente, en algunas implementaciones, el tipo de segmento y la lista de imágenes de referencia de un segmento de la imagen de referencia entre capas se usan al derivar un TMVP para un bloque en la capa actual (por ejemplo, capa de mejora). Por tanto, si los tipos de segmento y las listas de imágenes de referencia de todos los segmentos en la imagen de capa de referencia correspondiente son idénticos, se puede generar un segmento individual para la imagen de capa de referencia remuestreada y la información de segmento del segmento individual se puede derivar de cualquiera de los segmentos en la capa de referencia correspondiente porque la información de segmento de esos segmentos es idéntica.

[0168] En consecuencia, puede ser deseable que la información de segmento de los segmentos en la imagen de la capa de referencia sea idéntica entre sí al derivar un TMVP a partir de la imagen de la capa de referencia remuestreada. Por tanto, en un modo de realización, se puede imponer una restricción sobre el elemento sintáctico collocated_ref_idx usado para identificar una imagen de localización conjunta. Por ejemplo, cuando la imagen de referencia entre capas remuestreada se usa como una imagen de localización conjunta para la derivación de TMVP, todos los segmentos en la imagen de capa de referencia correspondiente tendrán un tipo de segmento idéntico, una lista de imágenes de referencia idéntica 0 y una lista de imágenes de referencia idéntica 1. Si una cualquiera de esas condiciones no se cumple, la restricción puede dictar que dicha imagen de referencia entre capas no se use como una imagen de localización conjunta para la derivación de TMVP.

[0169] En otro modo de realización, la restricción puede ser que para la derivación de TMVP, todos los segmentos en la imagen de capa de referencia correspondiente tendrán una lista de imágenes de referencia idéntica 0 y una lista de imágenes de referencia idéntica 1 (si está presente). En este ejemplo, el tipo de segmento puede diferir entre los segmentos, y el tipo de segmento del segmento individual de la imagen de la capa de referencia remuestreada se puede determinar como se describe anteriormente. Si estas condiciones no se cumplen, la restricción puede dictar que dicha imagen de referencia entre capas no se use como una imagen de localización conjunta para la derivación de TMVP.

Escalabilidad de relación señal/ruido (SNR)

[0170] En el caso de la escalabilidad de SNR, la información del formato de imagen de la capa de referencia y la capa de mejora puede ser idéntica y, por tanto, el proceso de remuestreo de imagen puede no ser necesario o no puede realizarse. En dicho caso, la imagen de capa de referencia reconstruida o descodificada se puede usar directamente como una imagen de referencia de la capa de mejora. Cuando los tamaños de CTB de la imagen de la capa de referencia y la imagen de la capa de mejora son diferentes, y la imagen de la capa de referencia tiene múltiples segmentos, el límite de división de segmento de la imagen de referencia entre capas puede no coincidir con los bordes de CTB. Por tanto, puede surgir un problema similar cuando la imagen de capa de referencia reconstruida o descodificada se usa como una imagen de localización conjunta para la derivación de TMVP. Por tanto, en algunos modos de realización, la restricción que requiere información de segmento idéntica entre múltiples segmentos de la imagen de capa de referencia antes de que se habilite la predicción de movimiento entre capas (o se permita su realización) también se aplica al caso de la escalabilidad de SNR.

[0171] Como se describe anteriormente, se puede imponer una restricción al elemento sintáctico collocated_ref_idx usado para identificar la imagen de localización conjunta, que cuando la imagen de referencia entre capas (ya sea remuestreada o no) se usa como una imagen de localización conjunta para la derivación de TMVP, todos los segmentos en la imagen de capa de referencia correspondiente tendrán un tipo de segmento idéntico, una lista de imágenes de referencia idéntica 0 y una lista de imágenes de referencia idéntica 1. Si estas condiciones no se cumplen, la restricción puede dictar que dicha imagen de referencia entre capas no se use como una imagen de localización conjunta para la derivación de TMVP. En otro modo de realización, la restricción se aplica en el caso de escalabilidad de SNR solo cuando hay múltiples segmentos en la imagen de la capa de referencia y el tipo de segmento o las listas de imágenes de referencia de los múltiples segmentos no son idénticos, y los tamaños de CTB de la capa de mejora y la capa de referencia son diferentes. Aún en otro modo de realización, la restricción se aplica en el caso de escalabilidad de SNR solo cuando hay múltiples segmentos en la imagen de la capa de referencia y el tipo de segmento o las listas de imágenes de referencia de los múltiples segmentos no son idénticos, y el tamaño de CTB de la capa de referencia es más pequeño que el de la capa de mejora.

Otras Consideraciones

[0172] La información y las señales divulgadas en el presente documento se pueden representar usando cualquiera entre varias tecnologías y técnicas diferentes. Por ejemplo, los datos, instrucciones, comandos, información, señales, bits, símbolos y chips que pueden haberse referenciado a lo largo de la descripción anterior se pueden representar mediante tensiones, corrientes, ondas electromagnéticas, campos o partículas magnéticas, campos o partículas ópticos o cualquier combinación de los mismos.

[0173] Los diversos bloques lógicos, módulos, circuitos y etapas de algoritmo ilustrativos descritos en relación con los modos de realización divulgados en el presente documento se pueden implementar como hardware electrónico, software informático o combinaciones de ambos. Para ilustrar claramente esta intercambiabilidad de hardware y software, anteriormente se han descrito diversos componentes, bloques, módulos, circuitos y etapas ilustrativos, en general, en lo que respecta a su funcionalidad. Que dicha funcionalidad se implemente como hardware o software depende de las restricciones particulares de aplicación y de diseño impuestas al sistema global. Los expertos en la técnica pueden implementar la funcionalidad descrita de formas distintas para cada solicitud particular, pero no debería interpretarse que dichas decisiones de implementación suponen apartarse del alcance de la presente invención.

[0174] Las técnicas descritas en el presente documento pueden implementarse en hardware, software, firmware o en cualquier combinación de los mismos. Dichas técnicas pueden implementarse en cualquiera entre una variedad de dispositivos tales como ordenadores de propósito general, equipos manuales de dispositivos de comunicación inalámbrica o dispositivos de circuitos integrados que tienen múltiples usos, incluyendo su aplicación en equipos manuales de dispositivos de comunicación inalámbrica y otros dispositivos. Todos los rasgos característicos descritos como módulos o componentes pueden implementarse juntos en un dispositivo lógico integrado o por separado, como dispositivos lógicos discretos pero interoperables. Si se implementan en software, las técnicas pueden realizarse, al menos en parte, mediante un medio de almacenamiento de datos legible por ordenador que comprenda código de programa que incluye instrucciones que, cuando se ejecutan, realizan uno o más de los procedimientos descritos anteriormente. El medio de almacenamiento de datos legible por ordenador puede formar parte de un producto de programa informático, que puede incluir materiales de embalaje. El medio legible por ordenador puede comprender memoria o medios de almacenamiento de datos, tales como memoria de acceso aleatorio (RAM), tal como memoria de acceso aleatorio dinámica síncrona (SDRAM), memoria de solo lectura (ROM), memoria de acceso aleatorio no volátil (NVRAM), memoria de solo lectura programable y borrable eléctricamente (EEPROM), memoria FLASH, medios de almacenamiento de datos magnéticos u ópticos, y similares. Las técnicas se pueden realizar, adicionalmente o de forma alternativa, al menos en parte, por un medio de comunicación legible por ordenador que transporta o comunica código de programa en forma de instrucciones o estructuras de datos y a las que se puede acceder, leer y/o ejecutar por medio de un ordenador, tales como señales u ondas propagadas.

[0175] El código de programa puede ser ejecutado por un procesador, que puede incluir 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 de puertas programables en el terreno (FPGA) u otros circuitos lógicos equivalentes, integrados o discretos. Un procesador de este tipo puede estar configurado para realizar cualquiera de las técnicas descritas en esta divulgación. Un procesador de propósito general puede ser un microprocesador, pero, de forma alternativa, el procesador puede ser cualquier procesador, controlador, microcontrolador o máquina de estados convencional. Un procesador también se puede implementar como una combinación de dispositivos informáticos, por ejemplo, una combinación de un DSP y un microprocesador, una pluralidad de microprocesadores, uno o más microprocesadores junto con un núcleo de DSP o cualquier otra configuración de este tipo. Por consiguiente, el término "procesador", como se usa en el presente documento, puede referirse a cualquier estructura anterior, cualquier combinación de la estructura anterior, o cualquier otra estructura o aparato adecuados para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento puede proporcionarse dentro de módulos de software o módulos de hardware dedicados configurados para la codificación y la decodificación, o incorporados en un codificador-descodificador de vídeo combinado (CODEC). Además, las técnicas se podrían implementar por completo en uno o más circuitos o elementos lógicos.

[0176] 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). En la presente divulgación se describen diversos componentes, módulos o unidades para destacar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no se requiere necesariamente su realización mediante diferentes unidades de hardware. En su lugar, como se describe anteriormente, diversas unidades se pueden combinar en una unidad de hardware de códec o proporcionar mediante un grupo de unidades de hardware interoperativas, que incluya uno o más procesadores como se describe anteriormente, junto con software y/o firmware adecuados.

[0177] Se han descrito diversos modos de realización de la invención. Estos y otros modos de realización están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un aparato configurado para codificar o descodificar información de vídeo, comprendiendo el aparato:
- 5 una memoria configurada para almacenar información de vídeo asociada con una capa de referencia y una capa de mejora, comprendiendo la capa de referencia una imagen de capa de referencia (RL) que tiene múltiples segmentos, y comprendiendo la capa de mejora una imagen de capa de mejora (EL) correspondiente a la imagen de RL; y
- 10 un procesador en comunicación con la memoria, estando configurado el procesador para:
- generar una imagen de referencia entre capas (ILRP) mediante el muestreo ascendente de la imagen RL, teniendo la ILRP un segmento individual asociado a la misma;
- 15 establecer el tipo de segmento y la información de la lista de imágenes de referencia del segmento individual de la ILRP igual al tipo de segmento correspondiente y la información de la lista de imágenes de referencia de uno de los múltiples segmentos de la imagen de RL; y
- 20 usar la ILRP para codificar al menos una parte de la imagen de EL.
2. El aparato de la reivindicación 1, en el que la imagen de EL y la imagen de RL están localizadas en la misma unidad de acceso.
3. El aparato de la reivindicación 1, en el que el tipo de segmento y la información de la lista de imágenes de referencia del segmento individual comprende un tipo de segmento del segmento individual, una o más listas de imágenes de referencia, y un número de índices de referencia para cada lista de imágenes de referencia del segmento individual.
4. El aparato de la reivindicación 3, en el que la información de la lista de imágenes de referencia comprende:
- 30 un primer número de imágenes de referencia en la lista de imágenes de referencia 0;
- una primera asociación entre las imágenes de referencia en la lista de imágenes de referencia 0 y un primer conjunto de índices de imágenes de referencia;
- 35 un segundo número de imágenes de referencia en la lista de imágenes de referencia 1; y
- una segunda asociación entre las imágenes de referencia en la lista de imágenes de referencia 1 y un segundo conjunto de índices de imágenes de referencia.
- 40 5. El aparato de la reivindicación 1, en el que el aparato comprende un codificador, y en el que el procesador está configurado además para codificar la información de vídeo en un flujo de bits de vídeo.
- 45 6. El aparato de la reivindicación 1, en el que el aparato comprende un descodificador, y en el que el procesador está configurado además para descodificar la información de vídeo en un flujo de bits de vídeo.
7. Un procedimiento de codificación o descodificación de información de vídeo, comprendiendo el procedimiento:
- 50 generar una imagen de referencia entre capas (ILRP) mediante el muestreo ascendente de una imagen de capa de referencia (RL) en una capa de referencia que tiene múltiples segmentos, teniendo la ILRP un solo segmento asociado con la misma;
- 55 establecer el tipo de segmento y la información de la lista de imágenes de referencia del segmento individual de la ILRP igual al tipo de segmento correspondiente y la información de la lista de imágenes de referencia de uno de los múltiples segmentos de la imagen de RL; y
- usar la ILRP para codificar al menos una parte de una imagen de capa de mejora (EL) en una capa de mejora.
- 60 8. El procedimiento de la reivindicación 7, en el que la imagen de EL y la imagen de RL están localizadas en la misma unidad de acceso.
- 65 9. El procedimiento de la reivindicación 7, en el que el tipo de segmento y la información de la lista de imágenes de referencia del segmento individual comprende un tipo de segmento del segmento individual, una o más listas de imágenes de referencia del segmento individual, y un número de índices de referencia para cada

lista de imágenes de referencia del segmento individual.

10. El procedimiento de la reivindicación 9, en el que la información de la lista de imágenes de referencia comprende:

5

un primer número de imágenes de referencia en la lista de imágenes de referencia 0;

una primera asociación entre las imágenes de referencia en la lista de imágenes de referencia 0 y un primer conjunto de índices de imágenes de referencia;

10

un segundo número de imágenes de referencia en la lista de imágenes de referencia 1; y

una segunda asociación entre las imágenes de referencia en la lista de imágenes de referencia 1 y un segundo conjunto de índices de imágenes de referencia.

15

11. Un medio no transitorio legible por ordenador que comprende código que, cuando se ejecuta, hace que un aparato realice un proceso de acuerdo con el procedimiento de las reivindicaciones 7 a 10.

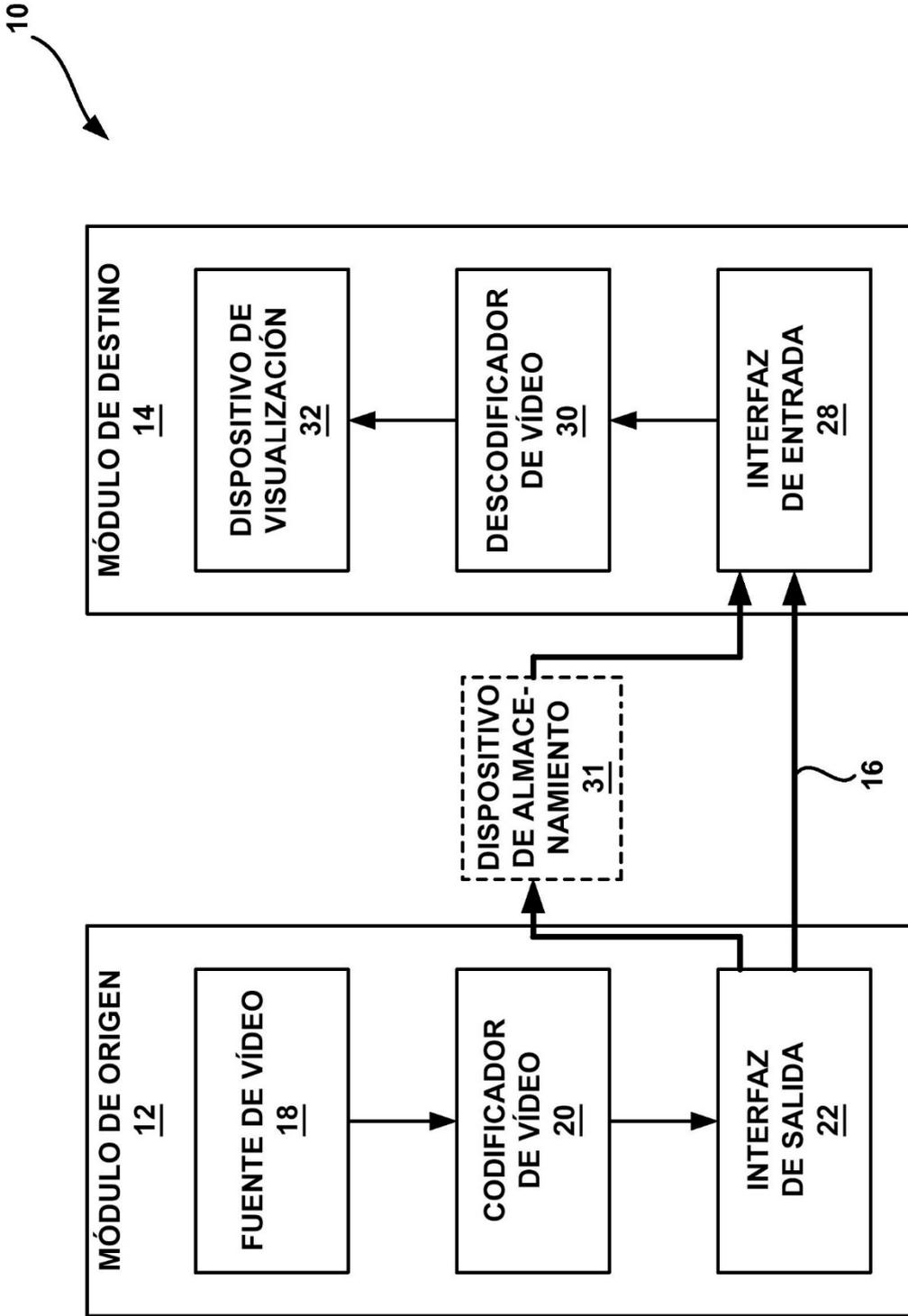


FIG. 1A

10'

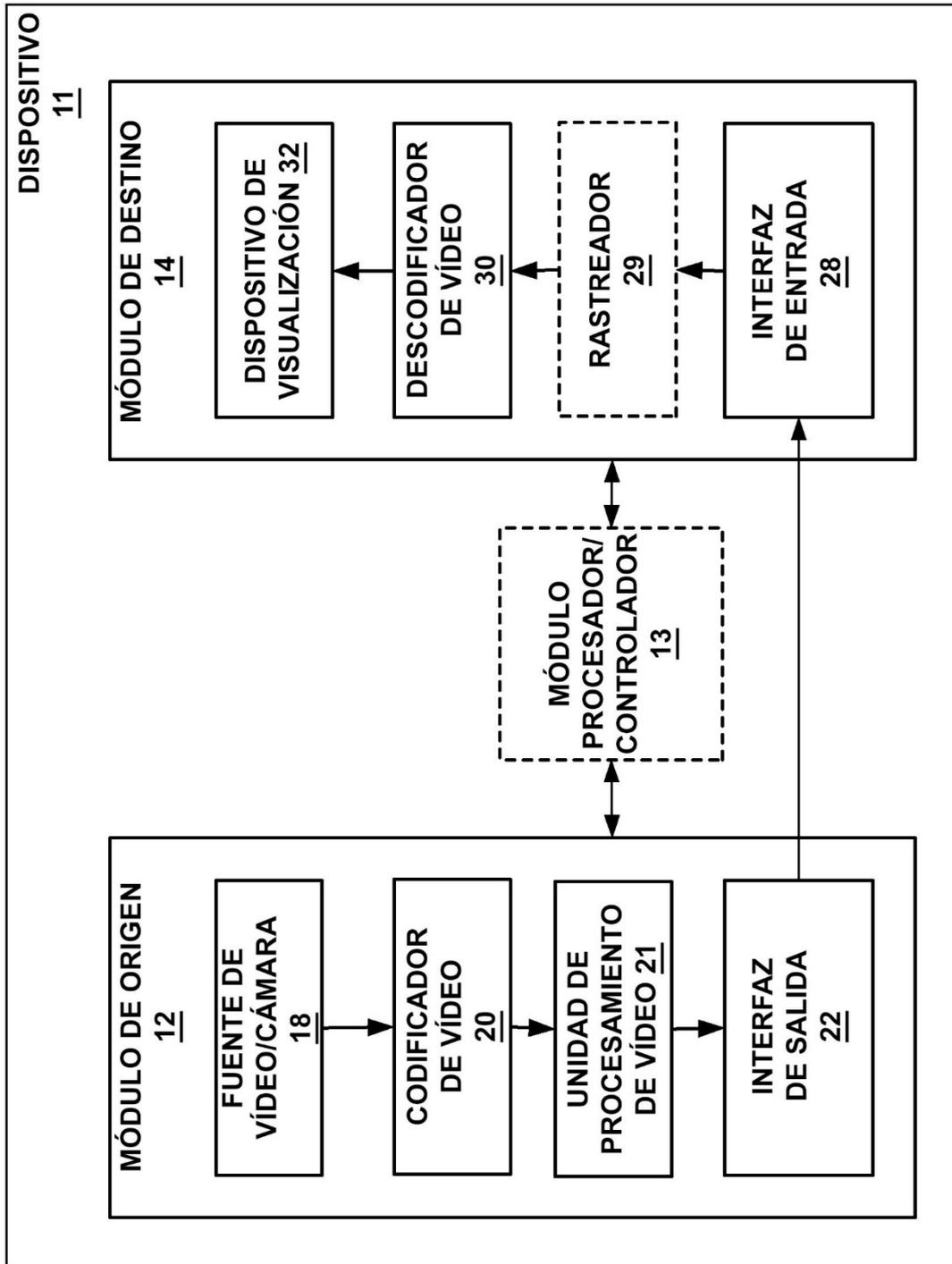


FIG. 1B

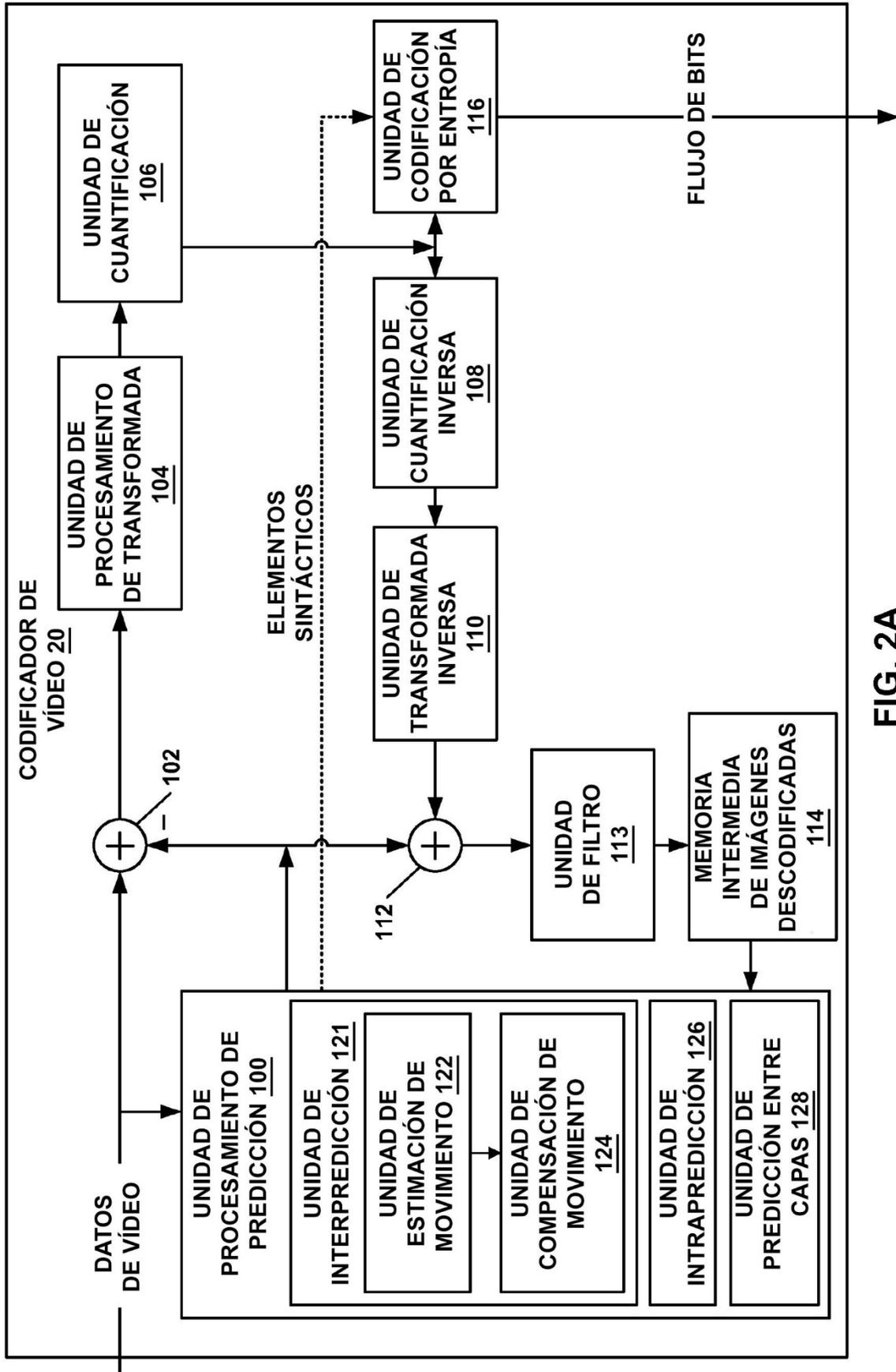


FIG. 2A

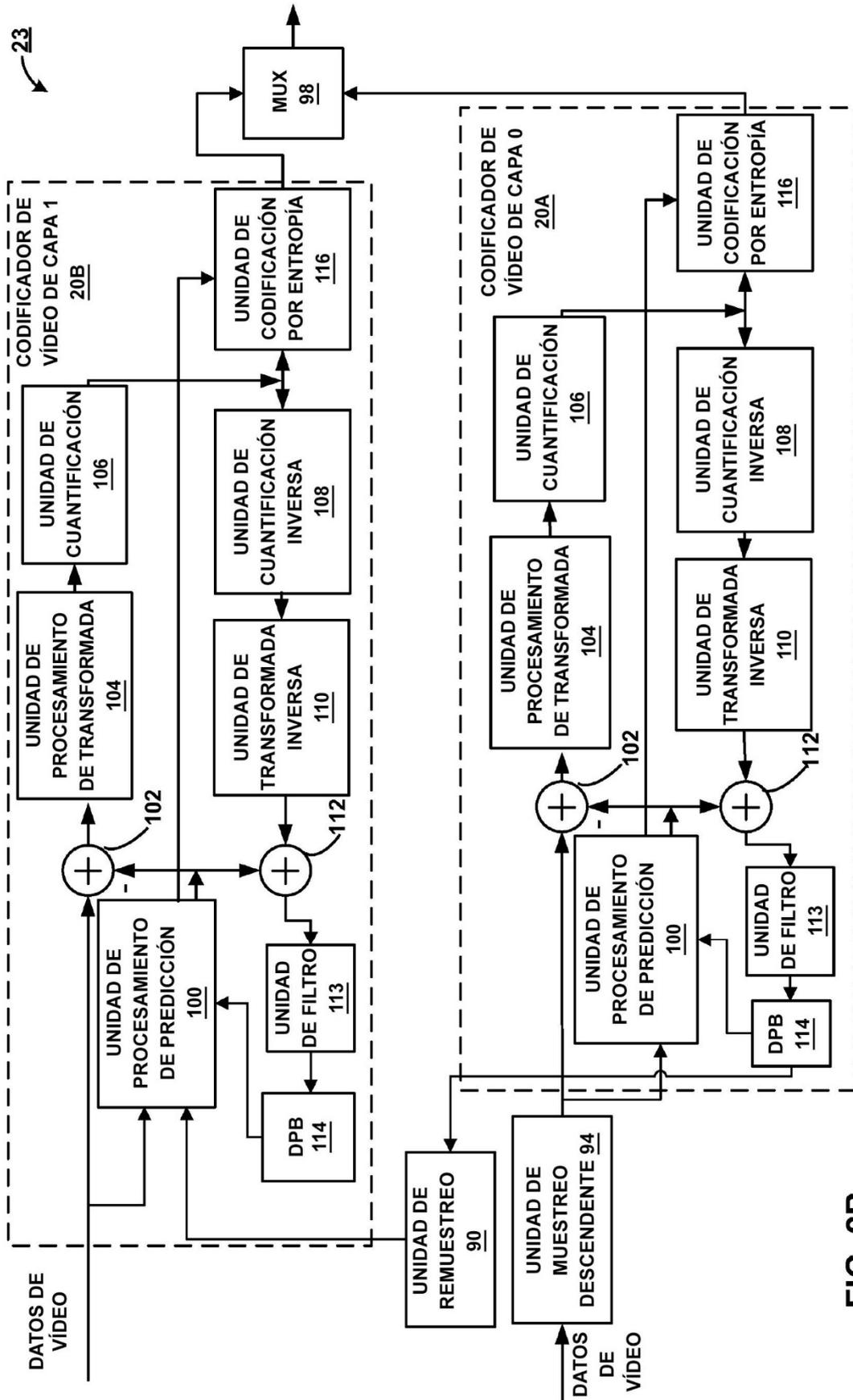


FIG. 2B

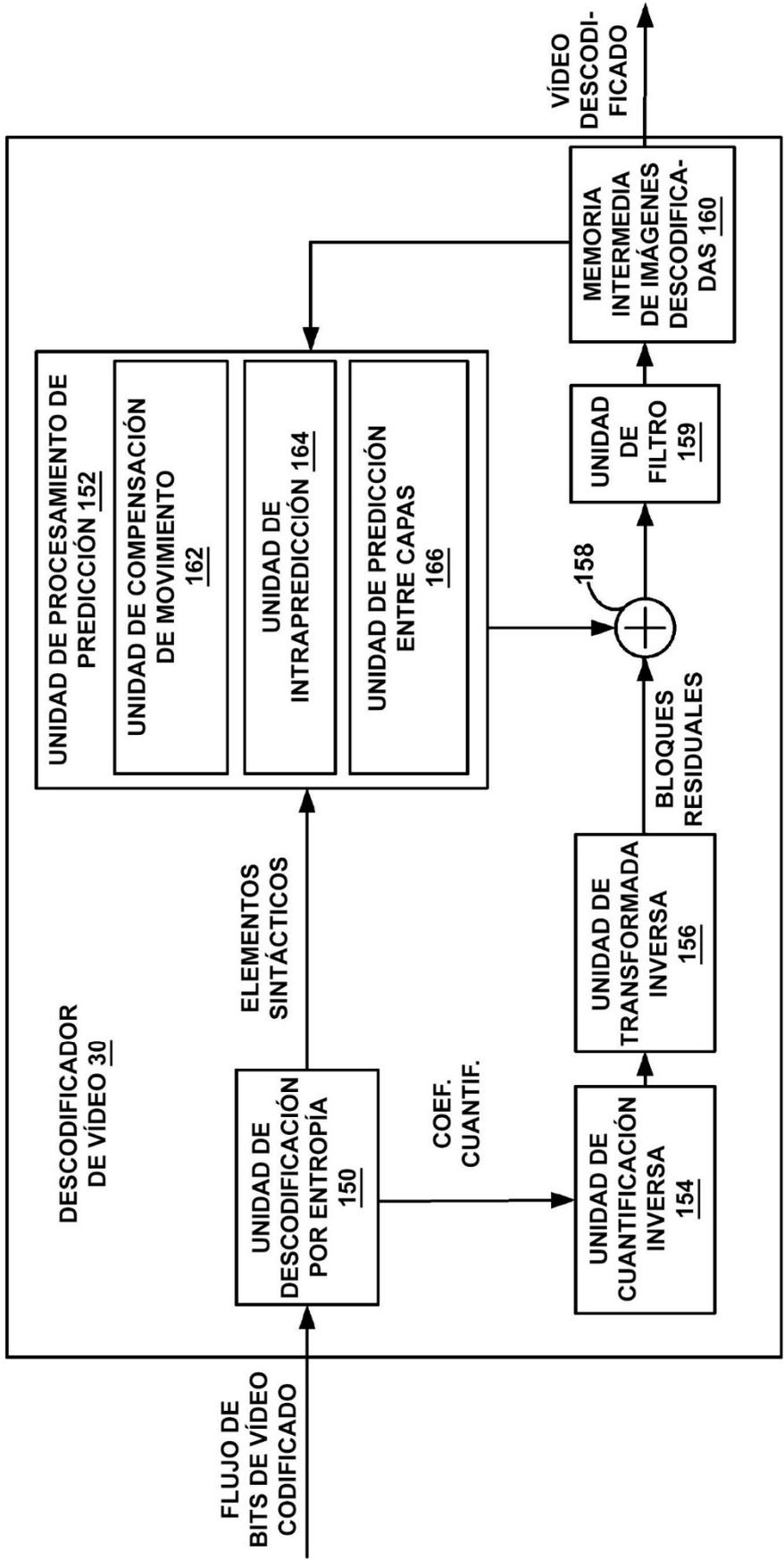


FIG. 3A

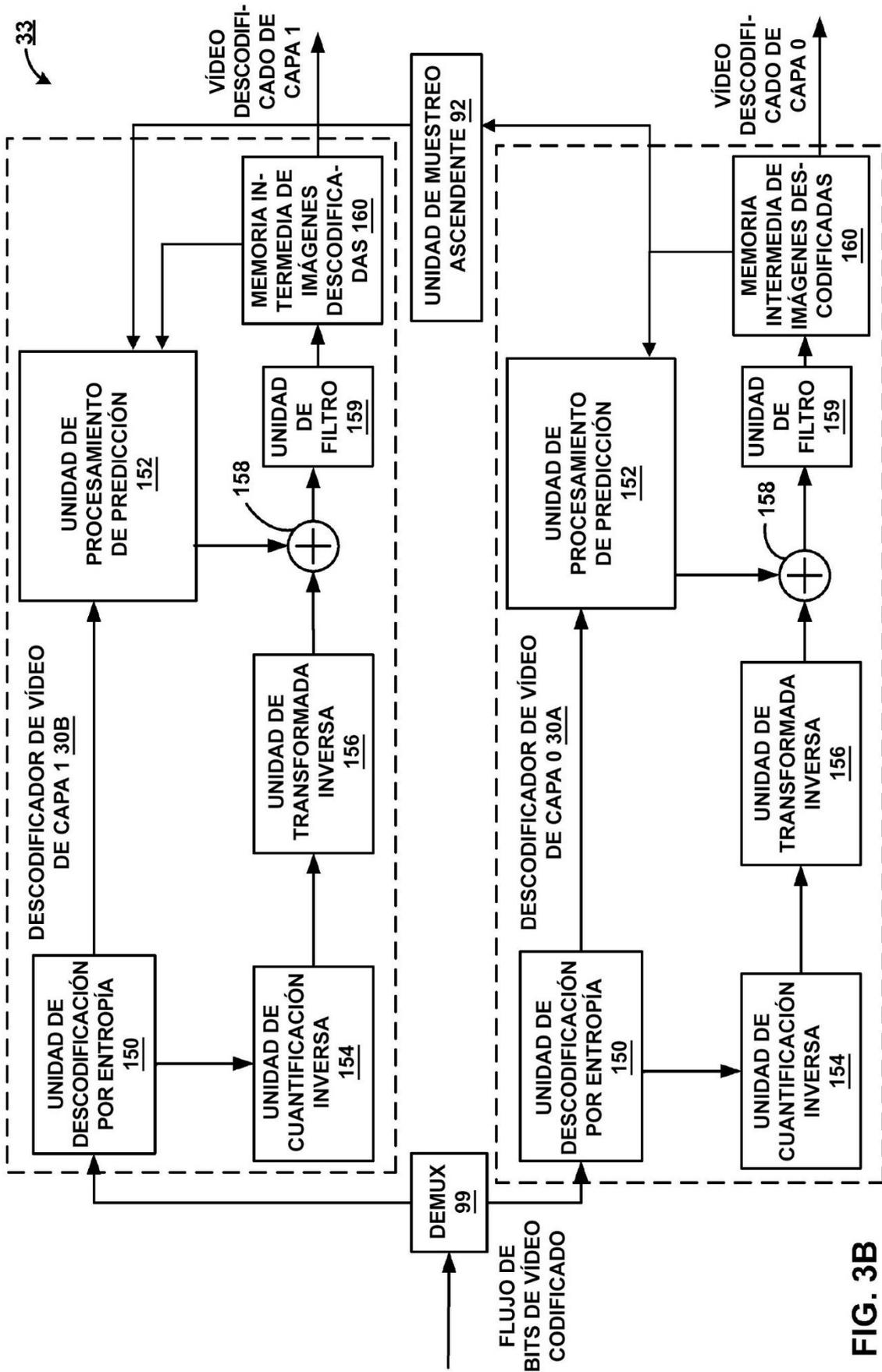


FIG. 3B

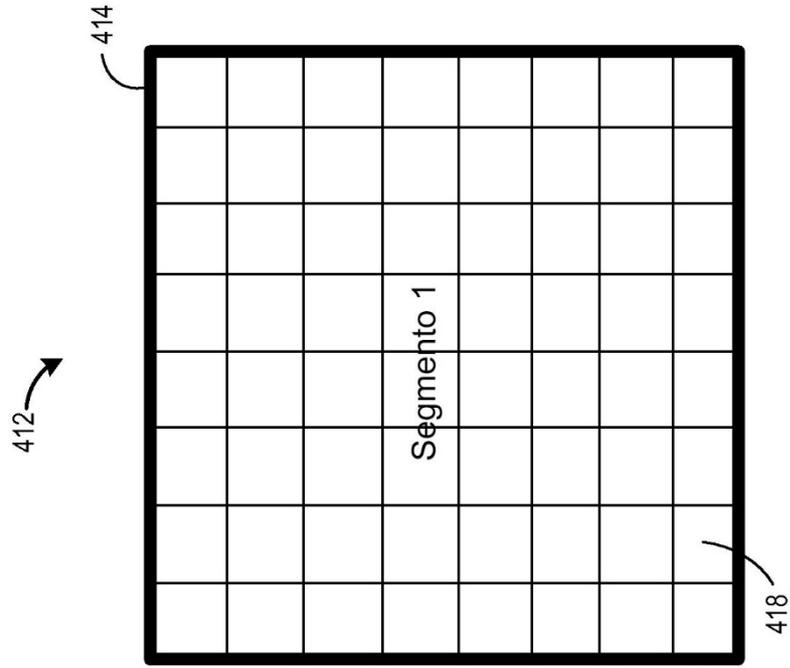


FIG. 4B

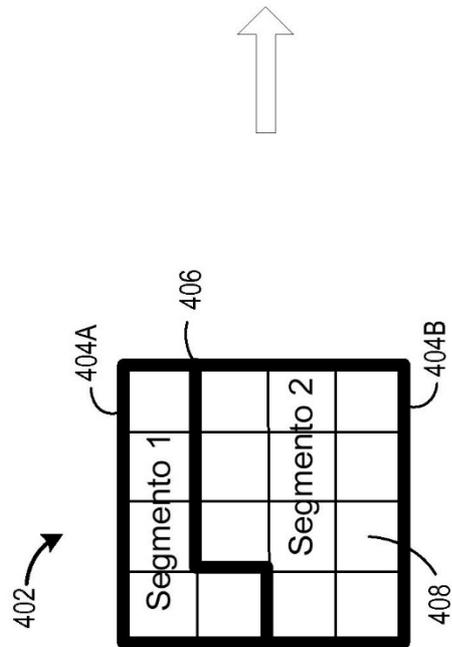


FIG. 4A

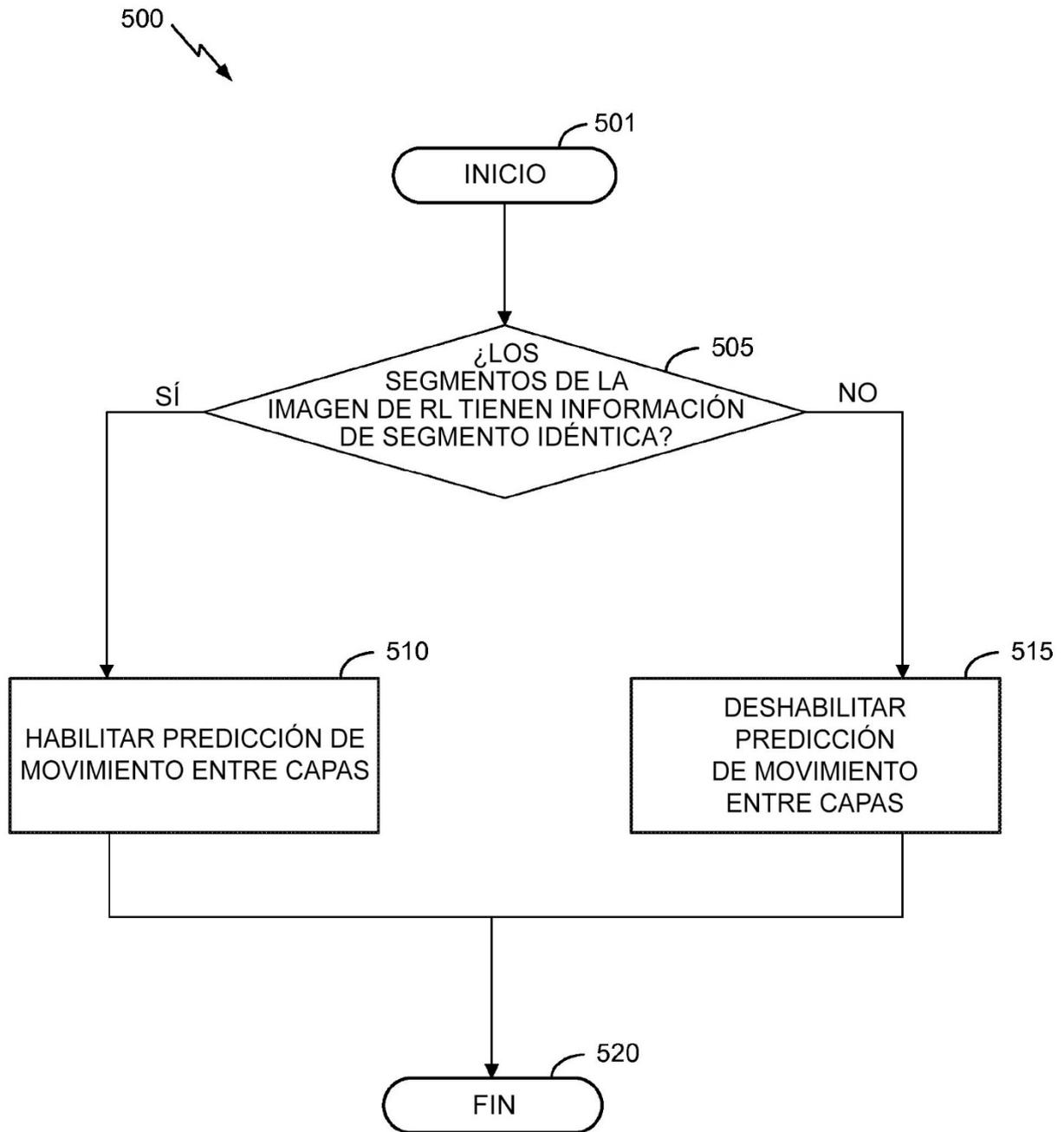


FIG. 5

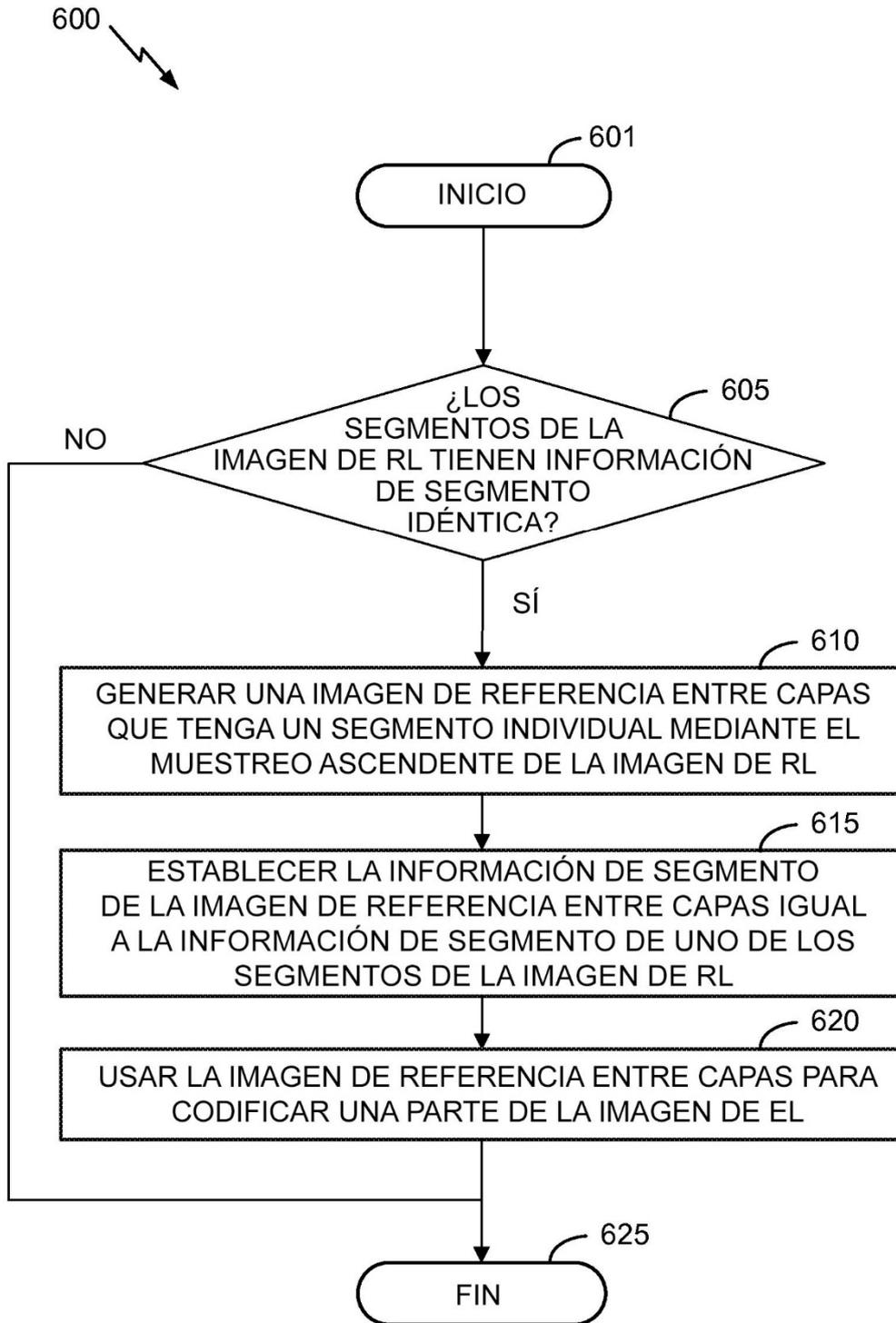


FIG. 6

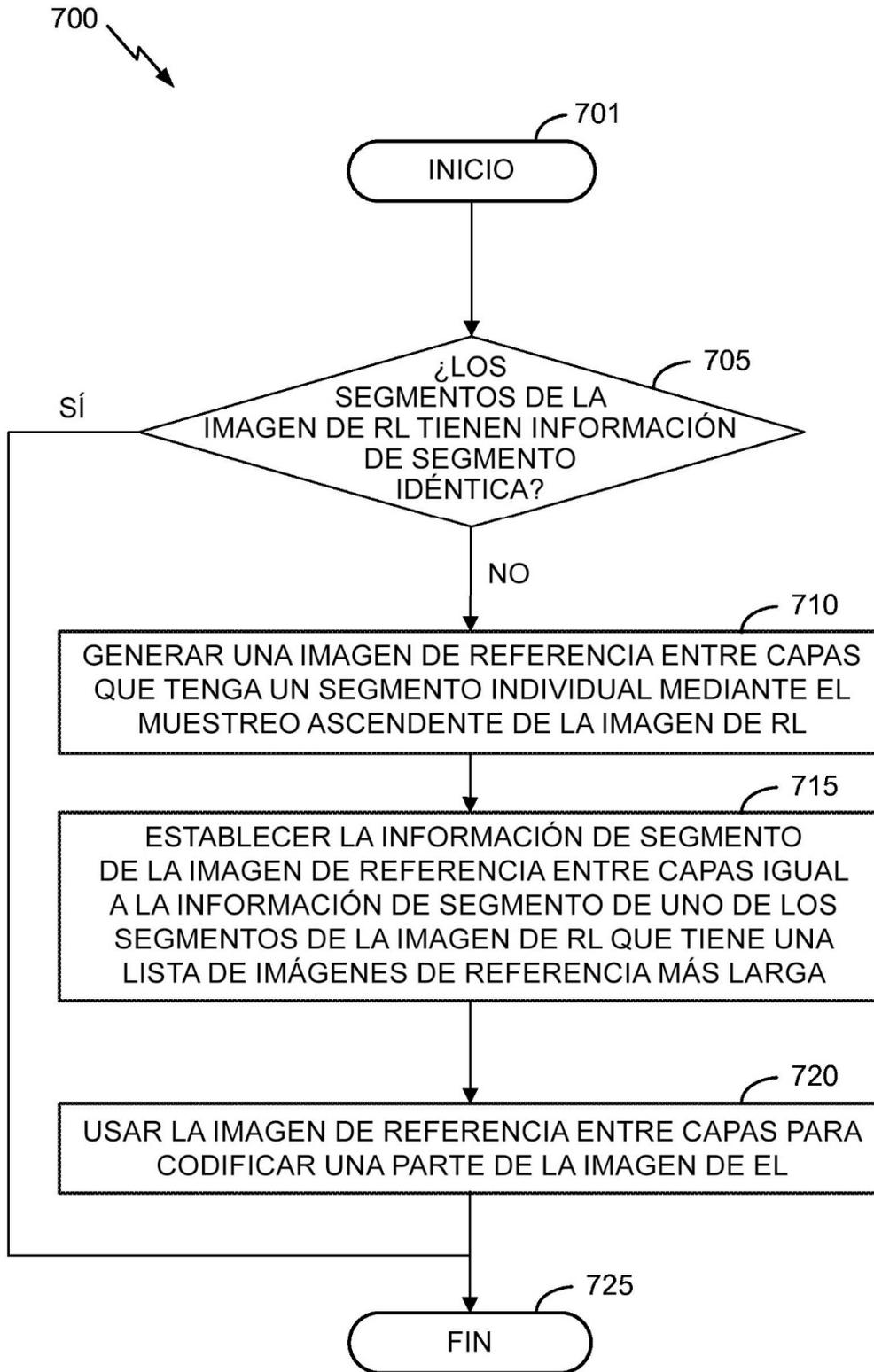


FIG. 7

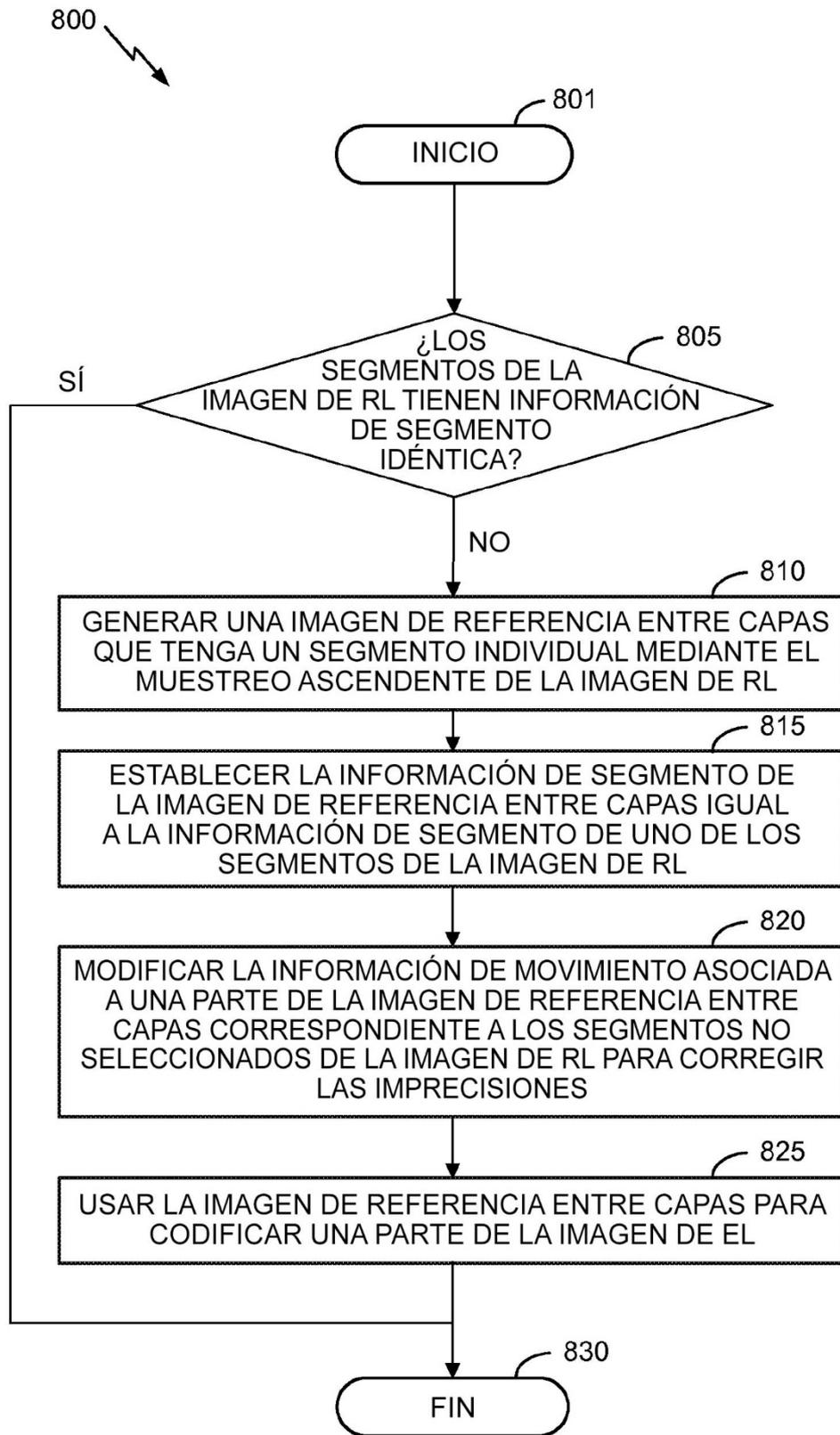


FIG. 8