

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 750 175**

51 Int. Cl.:

H04N 19/70

(2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **09.04.2013 PCT/US2013/035809**

87 Fecha y número de publicación internacional: **24.10.2013 WO13158415**

96 Fecha de presentación y número de la solicitud europea: **09.04.2013 E 13717688 (9)**

97 Fecha y número de publicación de la concesión europea: **24.07.2019 EP 2839642**

54 Título: **Codificación de vídeo con soporte mejorado para adaptación y empalme de flujos**

30 Prioridad:

20.04.2012 US 201261636566 P
04.05.2012 US 201261643100 P
02.07.2012 US 201261667371 P
12.03.2013 US 201313797458

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
25.03.2020

73 Titular/es:

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

72 Inventor/es:

WANG, YE-KUI

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 750 175 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

Codificación de vídeo con soporte mejorado para adaptación y empalme de flujos

5 **CAMPO TÉCNICO**

[0001] Esta divulgación se refiere a la codificación de vídeo y, más en particular, a técnicas relacionadas con el acceso aleatorio a datos de vídeo codificados.

10 **ANTECEDENTES**

15 **[0002]** Las capacidades del vídeo digital pueden incorporarse en una amplia gama de dispositivos, incluyendo televisores digitales, sistemas de radiodifusión directa digital, sistemas de radiodifusión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, ordenadores de tableta, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos celulares o de radio por satélite, los denominados "teléfonos inteligentes", dispositivos de videoconferencia, dispositivos de transmisión continua de vídeo, transcodificadores, encaminadores y similares. Los dispositivos de vídeo digital implementan técnicas de compresión de vídeo, tales como las descritas en las normas definidas por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, parte 10, codificación de vídeo avanzada (AVC), la norma de codificación de vídeo de alta eficacia (HEVC) actualmente en desarrollo y las ampliaciones de dichas normas. Los dispositivos de vídeo pueden transmitir, recibir, codificar, descodificar y/o almacenar información de vídeo digital con más eficacia implementando dichas técnicas de compresión de vídeo.

25 **[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 (es decir, una trama de vídeo o una parte de una trama de vídeo) puede dividirse en bloques de vídeo, que también pueden denominarse bloques de árbol, unidades de árbol de codificación (CTU), unidades de codificación (CU) y/o nodos de codificación. Los bloques de vídeo en un fragmento intracodificado (I) de una imagen se codifican usando predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen. Los bloques de vídeo de un fragmento intercodificado (P o B) de una imagen pueden usar la predicción espacial con respecto a unas muestras de referencia de unos bloques vecinos de la misma imagen, o la predicción temporal con respecto a unas muestras de referencia de otras imágenes. Las imágenes pueden denominarse tramas y las imágenes de referencia pueden denominarse tramas de referencia. Una secuencia de vídeo puede denominarse también flujo de bits.

35 **[0004]** La predicción espacial o temporal da como resultado un bloque predictivo para un bloque a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original a codificar y el bloque predictivo. Un bloque intercodificado se codifica de acuerdo con un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo, y los datos residuales que indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intracodificado se codifica de acuerdo con un modo de intracodificación y de acuerdo con datos residuales. Para una mayor compresión, los datos residuales pueden transformarse desde el dominio del píxel a un dominio de transformada, dando como resultado coeficientes de transformada residuales, que a continuación se pueden cuantificar. Los coeficientes de transformada cuantificados, dispuestos inicialmente en una formación bidimensional, pueden explorarse con el fin de generar un vector unidimensional de coeficientes de transformada, y puede aplicarse codificación de entropía para lograr aún más compresión.

45 **[0005]** El documento JCTVC-I0404 de SULLIVAN (MICROSOFT) G J: "CRA pictures with broken links", divulga el añadir un indicador de enlace roto que indicaría la posible presencia de imágenes principales no descodificables, y que una imagen CRA con una indicación de enlace roto podrá tener imágenes principales asociadas que no deben descodificarse o proporcionarse.

50 **[0006]** El documento JCTVC-I0275, DESHPANDE S ET AL: "AHG11: On Leading Pictures", divulga la señalización explícita de imágenes principales posteriores a una imagen CRA en orden de descodificación y anteriores a la misma en el orden de salida, mediante un tipo de unidad NAL (LCF) para indicar estas imágenes principales.

55 **[0007]** El documento m21712, HENDRY ET AL: "Random Access Detection and Notification", divulga la detección de imágenes de enlaces rotos a partir de información en la cabecera de la unidad NAL, y divulga además la detección de imágenes de enlaces rotos mediante un sistema que se ejecuta en un nivel superior al descodificador.

60 **SUMARIO**

65 **[0008]** Esta divulgación describe varias técnicas para mejorar el soporte de la adaptación y empalme de flujos en función de imágenes de acceso aleatorio limpio (CRA). Esta divulgación propone, en lugar de usar un indicador de la cabecera de fragmento para indicar que una imagen de enlace roto está presente, puede usarse un tipo de unidad de capa de abstracción de red (NAL) distinto para indicar la presencia de una imagen de enlace roto. En algunas implementaciones de las técnicas de esta divulgación, se puede usar un primer tipo de unidad NAL distinto para indicar

la presencia de una imagen de enlace roto con imágenes principales, mientras que un segundo tipo de unidad NAL distinto indica la presencia de una imagen de enlace roto sin imágenes principales. Además, en algunas implementaciones, se puede usar un tercer tipo de unidad NAL distinto para indicar la presencia de una imagen de enlace roto con imágenes principales descodificables.

5 **[0009]** En un ejemplo, un procedimiento para procesar datos de vídeo incluye recibir una primera unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo y basada en un tipo de unidad NAL de la primera unidad NAL, que detecta una imagen de enlace roto.

10 **[0010]** En otro ejemplo, un procedimiento para procesar datos de vídeo incluye recibir una unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo, determinar que la unidad NAL comprende una imagen de enlace roto y establecer un tipo de unidad NAL para la unidad NAL como un tipo de unidad NAL que indica que la unidad NAL comprende la imagen de enlace roto.

15 **[0011]** En otro ejemplo, un dispositivo para procesar datos de vídeo incluye uno o más procesadores configurados para recibir una primera unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo; y en función de un tipo de unidad NAL de la primera unidad NAL, detectar una imagen de enlace roto.

20 **[0012]** En otro ejemplo, un dispositivo para procesar datos de vídeo incluye uno o más procesadores configurados para recibir una unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo; determinar que la unidad NAL comprende una imagen de enlace roto; y establecer un tipo de unidad NAL para la unidad NAL como un tipo de unidad NAL que indica que la unidad NAL comprende la imagen de enlace roto.

25 **[0013]** En otro ejemplo, un dispositivo para procesar datos de vídeo incluye medios para recibir una primera unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo; y medios para detectar una imagen de enlace roto en función de un tipo de unidad NAL de la primera unidad NAL.

30 **[0014]** En otro ejemplo, un dispositivo para procesar datos de vídeo incluye medios para recibir una unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo; medios para determinar que la unidad NAL comprende una imagen de enlace roto; y medios para establecer un tipo de unidad NAL para la unidad NAL como un tipo de unidad NAL que indica que la unidad NAL comprende la imagen de enlace roto.

35 **[0015]** En otro ejemplo, un medio de almacenamiento legible por ordenador almacena instrucciones que cuando son ejecutadas por uno o más procesadores hacen que el uno o más procesadores reciban una primera unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo; y en función de un tipo de unidad NAL de la primera unidad NAL, detectar una imagen de enlace roto.

40 **[0016]** En otro ejemplo, un medio de almacenamiento legible por ordenador almacena instrucciones que cuando son ejecutadas por uno o más procesadores hacen que el uno o más procesadores reciban una unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo; determinar que la unidad NAL comprende una imagen de enlace roto; y establecer un tipo de unidad NAL para la unidad NAL como un tipo de unidad NAL que indica que la unidad NAL comprende la imagen de enlace roto.

45 **[0017]** Los detalles de uno o más ejemplos se exponen en los dibujos adjuntos y en la siguiente descripción. La invención se expone en las reivindicaciones adjuntas.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

50 **[0018]**
La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación y descodificación de vídeo de ejemplo que puede utilizar las técnicas descritas en esta divulgación.

55 Las FIG. 2A y 2B son diagramas conceptuales que ilustran el concepto de imagen de acceso aleatorio limpio.

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

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

La FIG. 5 es un diagrama de bloques que ilustra un conjunto de ejemplo de dispositivos que forman parte de una red para comunicación de datos de vídeo.

65 La FIG. 6 es un diagrama que ilustra una secuencia de vídeo de ejemplo que incluye una imagen RAP de acuerdo con las técnicas descritas en esta divulgación.

La FIG. 7 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar imágenes RAP de acuerdo con uno o más ejemplos descritos en esta divulgación.

5 La FIG. 8 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar imágenes RAP de acuerdo con uno o más ejemplos descritos en esta divulgación.

La FIG. 9 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar imágenes RAP de acuerdo con uno o más ejemplos descritos en esta divulgación.

10 La FIG. 10 es un diagrama de flujo que ilustra un procedimiento de ejemplo para descodificar un fragmento de acuerdo con uno o más ejemplos descritos en esta divulgación.

15 La FIG. 11 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar un fragmento de acuerdo con uno o más ejemplos descritos en esta divulgación.

La FIG. 12 es un diagrama de flujo que ilustra una técnica de procesamiento de vídeo de ejemplo de esta divulgación.

20 DESCRIPCIÓN DETALLADA

[0019] Esta divulgación describe varias técnicas para mejorar el soporte de la adaptación y concatenación de flujos en función de imágenes de acceso aleatorio limpio (CRA). El acceso aleatorio se refiere en general a la descodificación de un flujo de bits de vídeo empezando por una imagen codificada que no es la primera imagen codificada del flujo de bits. El acceso aleatorio a un flujo de bits puede ser deseable en muchas aplicaciones de vídeo, tales como radiodifusión y transmisión continua. Por ejemplo, el acceso aleatorio puede permitir que un usuario cambie entre diferentes canales, salte a partes específicas de un vídeo, o cambie a un flujo de bits diferente para la adaptación del flujo (por ejemplo, adaptación de la velocidad de transmisión de bits, la velocidad de tramas, la resolución espacial, etc.). El acceso aleatorio se puede habilitar insertando imágenes de acceso aleatorio, también conocidas como imágenes de punto de acceso aleatorio (RAP), en intervalos regulares o irregulares (pero típicamente regulares) en el flujo de bits de vídeo.

[0020] Se contempla que las técnicas de esta divulgación pueden implementarse junto con una variedad de diferentes normas de codificación. Con fines explicativos, las técnicas de esta divulgación se describirán principalmente con referencia a la norma emergente de codificación de vídeo de alta eficiencia (HEVC). Un borrador intermedio de la norma HEVC, denominado "HEVC Working Draft 6" o "WD6", se describe en el documento JCTVC-H1003, de Bross et al., titulado "High efficiency video coding (HEVC) text specification draft 6", Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, octava conferencia: San José, California, EE. UU., febrero de 2012, que, a partir del 11 de marzo de 2013, puede descargarse desde: http://phenix.int-evry.fr/jct/doc_end_user/documents/8_San%20Jose/wg11/JCTVC-H1003-v22.zip. La norma HEVC continúa evolucionando, y un borrador más reciente de la norma, denominado "HEVC Working Draft 10" o "WD10", se describe en el documento JCTVC-L1003_v18, de Bross et al., titulado "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, duodécima conferencia: Ginebra, Suiza, del 14 al 23 de enero de 2013, que, a partir del 11 de marzo de 2013, puede descargarse desde http://phenix.it-sudparis.eu/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v18.zip.

[0021] Se puede utilizar una imagen de actualización instantánea de descodificación (IDR) para el acceso aleatorio. Sin embargo, dado que una imagen IDR inicia una secuencia de vídeo codificada y siempre limpia la memoria intermedia de imágenes descodificadas (DPB), las imágenes que siguen a la IDR en orden de descodificación no pueden usar imágenes descodificadas antes que la imagen IDR como imágenes de referencia. Por consiguiente, los flujos de bits que dependen de imágenes IDR para el acceso aleatorio pueden tener una eficiencia de codificación significativamente menor. Para mejorar la eficiencia de codificación, el concepto de imágenes CRA se introdujo para permitir a imágenes que sigan a la imagen CRA en orden de descodificación, pero precedan a la imagen CRA en orden de salida, usar imágenes descodificadas antes de la imagen CRA como imágenes de referencia (por ejemplo, imágenes de referencia para la interpolación).

[0022] Las imágenes que siguen a una imagen CRA en orden de descodificación, pero que preceden a la imagen CRA en orden de salida, se denominan imágenes principales de la imagen CRA. Las imágenes principales de una imagen CRA se pueden descodificar correctamente si la descodificación comienza desde una imagen IDR o CRA anterior a la imagen CRA actual. No obstante, las imágenes principales de una imagen CRA no se pueden descodificar correctamente cuando se produce un acceso aleatorio desde la imagen CRA. En consecuencia, estas imágenes principales se descartan típicamente (es decir, se eliminan del flujo de bits) durante la descodificación de acceso aleatorio. Para evitar la propagación de errores a partir de imágenes de referencia que pueden no estar disponibles dependiendo de dónde comienza la descodificación, todas las imágenes que siguen a una imagen CRA tanto en orden

de descodificación como en orden de salida no usan ninguna imagen que preceda a la imagen CRA en orden de descodificación u orden de salida (lo cual incluye las imágenes principales) como imágenes de referencia.

5 **[0023]** Las funcionalidades de acceso aleatorio similares a las descritas anteriormente se admiten en la norma H.264/AVC con el uso de puntos de recuperación en mensajes de información de mejora suplementaria (SEI). Una implementación de descodificador de vídeo H.264/AVC puede o no admitir dichas funciones. En HEVC, un flujo de bits que comienza con una imagen CRA se considera un flujo de bits conforme a la norma. Cuando un flujo de bits comienza con una imagen CRA, las imágenes principales de la imagen CRA pueden referirse a imágenes de referencia no disponibles y, por lo tanto, pueden no descodificarse correctamente. Sin embargo, la norma HEVC
10 especifica que las imágenes principales de la imagen CRA inicial no se facilitan, de ahí el nombre "acceso aleatorio limpio". Para establecer la conformidad del flujo de bits, HEVC especifica un proceso de descodificación para generar imágenes de referencia no disponibles para la descodificación de las imágenes principales de no salida, pero las implementaciones de descodificador válidas no tienen que seguir ese proceso de descodificación, siempre que el descodificador pueda generar una salida idéntica en comparación con cuando el proceso de descodificación se realiza desde el comienzo de la secuencia de vídeo codificada.
15

[0024] En algunas versiones de la norma HEVC emergente, un flujo de bits válido puede no contener ninguna imagen IDR y, en consecuencia, puede contener un subconjunto de una secuencia de vídeo codificada o una secuencia de vídeo codificada incompleta. En HEVC, una secuencia de vídeo codificada se define como una secuencia de unidades de acceso que consiste, en orden de descodificación, en una unidad de acceso IDR seguida de cero o más unidades de acceso no IDR, incluidas todas las unidades de acceso posteriores hasta, pero sin incluir, cualquier unidad de acceso IDR posterior.
20

[0025] El concepto de "imágenes CRA con enlaces rotos" se ha introducido en varias propuestas de HEVC. Comparado con el concepto de CRA, se propone que imágenes CRA con enlaces rotos permitan adicionalmente a las imágenes CRA que no están al principio del flujo de bits tener imágenes principales no descodificables, como puede tenerlas una imagen CRA que comienza el flujo de bits. En HEVC WD6, se permite que una imagen CRA que comienza un flujo de bits esté seguida en orden de descodificación (también denominado como orden de flujo de bits) por imágenes principales que no se pueden descodificar debido a la ausencia de imágenes de referencia anteriores.
25 Sin embargo, no se permite que una imagen CRA que está situada en medio de un flujo de bits tenga dichas imágenes principales no descodificables. Se ha propuesto que esta restricción se elimine añadiendo un indicador de "enlace roto" que indique la posible presencia de tales imágenes principales no descodificables.
30

[0026] El indicador de "enlace roto" se ha propuesto como información de nivel de imagen de imágenes CRA en la cabecera de fragmento o en algún otro lugar para información de nivel de imagen, tal como el conjunto de parámetros de adaptación (APS). Cuando el indicador de enlace roto es igual a 1, se permitirá que el flujo de bits contenga imágenes principales de la imagen CRA que no son descodificables debido a que faltan imágenes de referencia precedentes, aunque el flujo de bits comience con una imagen IDR o una imagen CRA anterior en el orden de flujo de bits.
35

[0027] Para una imagen CRA con el indicador de "enlace roto" igual a 1, las imágenes CRA también pueden tener imágenes principales no descodificables como se mencionó anteriormente. Los enfoques existentes de "imágenes CRA con enlaces rotos", y otros enfoques existentes de acceso aleatorio de vídeo, están asociados a diversos problemas y deficiencias posibles. Por ejemplo, la señalización del indicador de "enlace roto" en la cabecera de segmento o el APS requiere una entidad (por ejemplo, un servidor, un elemento de red sensible a los medios (MANE) o un editor/empalmador de vídeo) que cambie una imagen CRA normal a una imagen CRA de enlace roto (BLC) cuando sea necesario, para poder codificar por entropía y analizar sintácticamente la cabecera de segmento y/o el APS para codificar el indicador. Esto también requiere una entidad (por ejemplo, un servidor, un MANE o un editor de vídeo) que identifique una imagen BLC, cuando sea necesario, que sea capaz de descodificar por entropía y analizar sintácticamente la cabecera de fragmento y/o el APS para encontrar el indicador. Las imágenes BLC también se denominan en ocasiones imágenes de acceso de enlace roto (BLA), por ejemplo, en versiones más recientes de la norma HEVC. En esta divulgación, estos términos pueden considerarse intercambiables.
40
45
50

[0028] Esta divulgación describe técnicas que pueden abordar las deficiencias descritas anteriormente. Esta divulgación propone, en lugar de usar un indicador de la cabecera de fragmento para indicar que una imagen BLC está presente, puede usarse un tipo de unidad de capa de abstracción de red (NAL) distinto para indicar la presencia de una imagen BLC. En algunas implementaciones de las técnicas de esta divulgación, se puede usar un primer tipo de unidad NAL distinto para indicar la presencia de una imagen BLC (también conocida como BLA) con imágenes principales, mientras que un segundo tipo de unidad NAL distinto indica la presencia de una imagen BLC sin imágenes principales. Además, en algunas implementaciones, se puede usar un tercer tipo de unidad NAL distinto para indicar la presencia de una imagen de enlace roto con imágenes principales descodificables.
55
60

[0029] Los datos de vídeo codificados se organizan en unidades NAL, cada una de las cuales es efectivamente un paquete que contiene un número entero de octetos. El primer octeto de cada unidad NAL es un octeto de cabecera que contiene una indicación del tipo de datos en la unidad NAL, y los octetos restantes contienen datos de carga útil del tipo indicado por la cabecera. Las unidades NAL se pueden dividir, en general, en dos categorías principales,
65

unidades NAL que contienen datos de capa de codificación de vídeo y unidades NAL que no incluyen datos de capa de codificación de vídeo. Estas dos categorías de unidades NAL se denominan, en general, unidades NAL VCL y unidades NAL no VCL, respectivamente. Las unidades NAL no VCL incluyen, por ejemplo, tipos de unidades NAL para información de mejora suplementaria y tipos de unidades NAL para información de conjunto de parámetros.

[0030] De acuerdo con las técnicas de esta divulgación, en un ejemplo, un tipo de unidad NAL igual a 2 (que está reservado en HEVC WD6) se puede usar para indicar la presencia de una imagen de enlace roto, por ejemplo, una imagen BLC, también conocida como imagen BLA. Al usarse un tipo de unidad NAL dedicada, como se describe en esta divulgación, una entidad (por ejemplo, un servidor, un MANE o editor/empalmador de vídeo) puede cambiar una imagen CRA normal, para lo cual, si dicha imagen no inicia el flujo de bits, todas las imágenes principales asociadas deben ser descodificables (es decir, pueden descodificarse correctamente), para ser una imagen BLC cuando sea necesario sin codificación o descodificación de entropía de sintaxis de cabecera de fragmento, sintaxis APS u otra sintaxis de flujo de bits. Además, al implementarse las técnicas de esta divulgación, una entidad (por ejemplo, un servidor, un MANE o un editor de vídeo) puede identificar una imagen BLC cuando sea necesario sin descodificación de entropía de sintaxis de cabecera de fragmento, sintaxis APS u otra sintaxis de flujo de bits.

[0031] El uso de uno o más tipos distintos de unidades NAL para indicar la presencia de una imagen de enlace roto es una técnica introducida en esta divulgación. También se introducirán a continuación técnicas adicionales. Estas técnicas adicionales se pueden utilizar de forma alternativa a o junto con las unidades NAL dedicadas ya introducidas.

[0032] La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación y descodificación de vídeo 10 de ejemplo que puede utilizar las técnicas descritas en esta divulgación. Como se muestra en la FIG. 1, el sistema 10 incluye un dispositivo de origen 12 que genera datos de vídeo codificados que un dispositivo de destino 14 puede descodificar más tarde. El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender cualquiera entre una amplia gama de dispositivos, incluidos ordenadores de escritorio, ordenadores plegables (es decir, portátiles), ordenadores de tableta, descodificadores, equipos telefónicos de mano tales como los denominados teléfonos "inteligentes", los denominados paneles "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, dispositivos de transmisión continua de vídeo o similares. En algunos casos, el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar equipados para la comunicación inalámbrica.

[0033] En algunos casos, el codificador de vídeo 20 puede codificar una imagen o unas imágenes que se hallan después de un punto de acceso aleatorio o punto de adaptación de flujo, tal como un punto de conmutación de capa temporal. Por ejemplo, este podría ser un punto de conmutación para la adaptación de la velocidad de bits, la velocidad de tramas o la resolución espacial. Una o más de estas imágenes pueden ser imágenes principales de una imagen CRA en una secuencia de vídeo. Un descodificador puede descodificar correctamente las imágenes principales de una imagen CRA si la descodificación de la secuencia de vídeo comienza por una imagen RAP antes que la imagen CRA actual en la secuencia de vídeo. No obstante, las imágenes principales de una imagen CRA no se pueden descodificar correctamente cuando se produce un acceso aleatorio desde la imagen CRA. Por ejemplo, las imágenes principales pueden apuntar a bloques para referencia de predicción que no están disponibles. En consecuencia, la imagen principal puede no ser descodificable en el descodificador de vídeo 30. En consecuencia, un dispositivo de destino 14 puede típicamente descartar estas imágenes principales durante la descodificación de acceso aleatorio.

[0034] En otro ejemplo, para una imagen BLA, es decir, BLC, el codificador 20 puede codificar un indicador, por ejemplo, `no_output_of_prior_pics_flag` (que de forma más correcta puede denominarse elemento sintáctico) de modo que no se proporciona ninguna de las imágenes anteriores en la DPB. En algunos ejemplos, este indicador, o elemento sintáctico, puede estar al principio en la cabecera de fragmento antes de la descodificación de entropía de modo que pueda descodificarse más fácilmente y la información esté disponible más pronto en el proceso de codificación. Por ejemplo, esto se hace a fin de que dispositivos menos avanzados, tal como un MANE, puedan tener acceso a la información sin necesidad de un descodificador de entropía, ya que, por ejemplo, no se necesita realizar la descodificación de entropía de `no_output_of_prior_pics_flag`. Como ilustración, el indicador `no_output_of_prior_pics` se puede presentar como un elemento sintáctico no sometido a codificación de entropía tal como, por ejemplo, un elemento de longitud fija $u(1)$, en lugar de un elemento sintáctico sometido a codificación de entropía tal como, por ejemplo, un elemento de longitud variable $ue(v)$. El indicador `no_output_of_prior_pics` se puede presentar, por ejemplo, justo después de un indicador `first_slice_segment_in_pic` y antes que cualquier elemento sintáctico sometido a codificación de entropía.

[0035] En un ejemplo, el descodificador de vídeo 30 puede marcar imágenes almacenadas en una memoria intermedia de imágenes de referencia como no usadas para referencia cuando, por ejemplo, el uso de estas imágenes puede dar como resultado una imagen descodificada incorrectamente. Por ejemplo, una imagen de referencia que precede a una imagen BLA o BLC en orden de descodificación u orden de salida puede ser inutilizable como referencia para una imagen principal que sigue a la imagen BLA o BLC en orden de descodificación. En consecuencia, la imagen de referencia puede marcarse como no usada para referencia por el descodificador de vídeo 30 en respuesta a la recepción de una imagen de referencia que precede a una imagen BLA o BLC en orden de descodificación u orden de salida y puede ser inutilizable como referencia para una imagen principal que sigue a la imagen BLA o BLC en orden de descodificación.

[0036] En un ejemplo, un codificador de vídeo 20 puede estar configurado para incluir tipos de unidad NAL asignados para indicar cuándo las imágenes BLA o BLC tienen y no tienen imágenes principales. Por ejemplo, en una norma se incluyen los tipos de unidad NAL 16, BLA_W_LP (BLA con imagen principal); 17, BLA_W_DLP (BLA con imagen principal descodificable); y 18, BLA_N_LP (BLA sin imagen principal). Una imagen BLA que tenga un tipo de unidad NAL igual a BLA_W_LP puede tener imágenes principales descodificables o no descodificables asociadas presentes en el flujo de bits. Una imagen BLA que tiene un tipo de unidad NAL igual a BLA_W_DLP no tiene imágenes principales no descodificables asociadas presentes en el flujo de bits, pero puede tener imágenes principales descodificables asociadas en el flujo de bits. Una imagen BLA que tiene un tipo de unidad NAL igual a BLA_N_LP puede no tener imágenes principales asociadas (descodificables o no descodificables) presentes en el flujo de bits.

[0037] En un ejemplo, el codificador de vídeo 20 puede procesar tipos de unidad NAL asignados para indicar cuándo las imágenes BLA tienen y no tienen imágenes principales. Por ejemplo, el codificador de vídeo 20 puede codificar imágenes de acuerdo con uno de una pluralidad de tipos diferentes de unidades de capa de abstracción de red (NAL). La pluralidad de tipos de unidades NAL incluye uno o más de (1) un fragmento codificado de una imagen de acceso de enlace roto (BLA), donde la imagen BLA es una imagen BLA con imágenes principales asociadas en un flujo de bits, (2) un fragmento codificado de una imagen BLA, donde la imagen BLA es una imagen BLA con imágenes principales descodificables asociadas en el flujo de bits, y (3) un fragmento codificado de una imagen BLA, donde la imagen BLA es una imagen BLA sin imágenes principales asociadas en el flujo de bits.

[0038] El dispositivo de destino 14 puede recibir los datos de vídeo codificados. El dispositivo de destino puede descodificar los datos recibidos a través de un enlace 16. El enlace 16 puede comprender cualquier tipo de medio o dispositivo capaz de transferir los datos de vídeo codificados desde el dispositivo de origen 12 hasta el dispositivo de destino 14. En un ejemplo, el enlace 16 puede comprender un medio de comunicación para permitir que el dispositivo de origen 12 transmita datos de vídeo codificados directamente a un dispositivo de destino 14 en tiempo real. Los datos de vídeo codificados se pueden modular de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y transmitir al dispositivo de destino 14. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrico o cableado, tal como un espectro de radiofrecuencia (RF) o una o más líneas físicas de transmisión. El medio de comunicación puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área extensa o una red global tal como Internet. El medio de comunicación puede incluir encaminadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 hasta el dispositivo de destino 14.

[0039] De forma alternativa, los datos codificados pueden emitirse desde la interfaz de salida 22 hasta un dispositivo de almacenamiento 34. De forma similar, una interfaz de entrada puede acceder a los datos codificados del dispositivo de almacenamiento 34. El dispositivo de almacenamiento 34 puede incluir cualquiera de una variedad de medios de almacenamiento de datos de acceso distribuido o local, tales como una unidad de disco duro, discos Blu-ray, DVD, CD-ROM, una memoria flash, una memoria volátil o no volátil o cualquier otro medio de almacenamiento digital adecuado para almacenar datos de vídeo codificados. En otro ejemplo, el dispositivo de almacenamiento 34 puede corresponder a un servidor de archivos o a otro dispositivo de almacenamiento intermedio que puede contener el vídeo codificado generado por el dispositivo de origen 12. El dispositivo de destino 14 puede acceder a datos de vídeo almacenados en el dispositivo de almacenamiento 34 a través de transmisión continua o descarga. El servidor de archivos puede ser cualquier tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir esos datos de vídeo codificados al dispositivo de destino 14. Servidores de archivos de ejemplo incluyen un servidor web (por ejemplo, para un sitio web), un servidor FTP, dispositivos de almacenamiento conectados en red (NAS) o una unidad de disco local. El dispositivo de destino 14 puede acceder a los datos de vídeo codificados a través de cualquier conexión de datos estándar, incluida una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión Wi-Fi), una conexión cableada (por ejemplo, DSL, módem 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 34 puede ser una transmisión continua, una transmisión de descarga o una combinación de ambas cosas.

[0040] Las técnicas de esta divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas se pueden aplicar a la codificación de vídeo como apoyo a cualquiera de una variedad de aplicaciones multimedia, tales como radiodifusiones de televisión por aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones continuas de vídeo, por ejemplo, a través de Internet, codificación de vídeo digital para su almacenamiento en un medio de almacenamiento de datos, descodificación de vídeo digital almacenado en un medio de almacenamiento de datos, u otras aplicaciones. En algunos ejemplos, el sistema 10 puede estar configurado para prestar soporte a una transmisión de vídeo unidireccional o bidireccional, a fin de prestar soporte a aplicaciones tales como la transmisión continua de vídeo, la reproducción de vídeo, la difusión de vídeo y/o la videotelefonía.

[0041] En el ejemplo de la FIG. 1, el dispositivo de origen 12 incluye una fuente de vídeo 18, un codificador de vídeo 20 y una interfaz de salida 22. En algunos casos, la interfaz de salida 22 puede incluir un modulador/desmodulador (módem) y/o un transmisor. En el dispositivo de origen 12, la fuente de vídeo 18 puede incluir una fuente tal como un dispositivo de captación de vídeo, por ejemplo, una videocámara, un archivo de vídeo que contiene vídeo captado

previamente, una interfaz de alimentación de vídeo para recibir vídeo desde un proveedor de contenido de vídeo y/o un sistema de gráficos por ordenador para generar datos de gráficos por ordenador como el vídeo de origen, o una combinación de dichas fuentes. En un ejemplo, si la fuente de vídeo 18 es una videocámara, el dispositivo de origen 12 y el dispositivo de destino 14 pueden formar los denominados teléfonos con cámara o videoteléfonos. Sin embargo, las técnicas descritas en esta divulgación pueden ser aplicables a la codificación de vídeo en general, y se pueden aplicar a aplicaciones inalámbricas y/o cableadas.

[0042] El codificador de vídeo 20 puede codificar el vídeo capturado, capturado previamente o generado por ordenador. Los datos de vídeo codificados se pueden transmitir directamente al dispositivo de destino 14 por medio de la interfaz de salida 22 del dispositivo de origen 12. Los datos de vídeo codificados se pueden almacenar de forma adicional (o alternativa) en el dispositivo de almacenamiento 34 para un posterior acceso por el dispositivo de destino 14 u otros dispositivos, para su descodificación y/o reproducción.

[0043] El dispositivo de destino 14 incluye una interfaz de entrada 28, un descodificador de vídeo 30 y un dispositivo de visualización 32. En algunos casos, la interfaz de entrada 28 puede incluir un receptor y/o un módem. La interfaz de entrada 28 del dispositivo de destino 14 recibe los datos de vídeo codificados a través del enlace 16. Los datos de vídeo codificados transmitidos a través del enlace 16, o proporcionados en el dispositivo de almacenamiento 34, pueden incluir una diversidad de elementos sintácticos generados por el codificador de vídeo 20 para su uso por un descodificador de vídeo, tal como el descodificador de vídeo 30, en la descodificación de los datos de vídeo. Dichos elementos sintácticos se pueden incluir con los datos de vídeo codificados transmitidos en un medio de comunicación, almacenar en un medio de almacenamiento o almacenar en un servidor de archivos.

[0044] En algunos ejemplos, el descodificador de vídeo 30 del dispositivo de destino 14 puede descodificar una imagen o varias imágenes que se hallan después de un punto de acceso aleatorio o un punto de adaptación de flujo, tal como un punto de conmutación de capa temporal. Por ejemplo, este podría ser un punto de conmutación para adaptación de velocidad de bits, velocidad de tramas (es decir, un punto de conmutación de capa temporal) o resolución espacial. Una o más de estas imágenes pueden ser imágenes principales. Las imágenes principales no se pueden descodificar correctamente cuando se produce un acceso aleatorio desde una imagen de enlace roto (es decir, BLA o BLC).

[0045] En un ejemplo, para evitar la propagación de errores a partir de imágenes de referencia que pueden no estar disponibles dependiendo de dónde comienza la descodificación, el descodificador de vídeo 30 no puede usar ninguna imagen que preceda a la imagen de enlace roto en orden de descodificación u orden de salida (lo cual incluye las imágenes principales) como imagen de referencia.

[0046] En diversos ejemplos, el descodificador de vídeo 30 puede marcar todas las imágenes de referencia de la DPB como no usadas para referencia antes de descodificar la imagen BLA cuando se descodifica una imagen BLA o BLC. Por ejemplo, el descodificador de vídeo 30 puede marcar imágenes de referencia de la memoria intermedia de imágenes descodificadas (DPB) como no usadas para referencia.

[0047] En otro ejemplo, el codificador 20 puede incluir en el flujo de bits, y el descodificador 30 puede recibir, un indicador u otro elemento sintáctico, por ejemplo, un `no_output_of_prior_pics_flag`, para una imagen BLA que se va a descodificar. Cuando el indicador es igual a 1, indica que no se proporciona ninguna de las imágenes anteriores en la DPB para su visualización. En particular, cuando `no_output_of_prior_pics_flag` es igual a 1, el descodificador 30 vacía todas las memorias intermedias de almacenamiento de imágenes de la memoria intermedia de imágenes descodificadas sin proporcionar las imágenes que contienen. En algunos ejemplos, este indicador o elemento sintáctico puede presentarse muy al principio de la cabecera de fragmento antes de la descodificación de entropía, de tal forma que puede descodificarse más fácilmente, sin necesidad de codificación de entropía, y la información está disponible más pronto en el proceso de codificación. Por ejemplo, esto se hace para que dispositivos menos avanzados, como un MANE, puedan tener acceso a la información sin necesidad de un descodificador ya que, por ejemplo, no es necesario realizar la descodificación de entropía de `no_output_of_prior_pics_flag`.

[0048] En otro ejemplo, el descodificador de vídeo 30 puede procesar tipos de unidad NAL asignados para indicar cuándo las imágenes BLA o BLC tienen y no tienen imágenes principales. De nuevo, como se ha mencionado anteriormente, las imágenes BLA son, conceptualmente, iguales que las imágenes BLC en la medida en que las imágenes BLA y BLC representan imágenes CRA de enlace roto, pero difieren en terminología. En un ejemplo, el descodificador de vídeo 30 puede descodificar imágenes de acuerdo con uno de una pluralidad de tipos diferentes de unidades de capa de abstracción de red (NAL). La pluralidad de tipos de unidades NAL incluye uno o más de (1) un fragmento codificado de una imagen de acceso de enlace roto (BLA), donde la imagen BLA es una imagen BLA con imágenes principales asociadas en un flujo de bits, (2) un fragmento codificado de una imagen BLA, donde la imagen BLA es una imagen BLA con imágenes principales descodificables asociadas en el flujo de bits, y (3) un fragmento codificado de una imagen BLA, donde la imagen BLA es una imagen BLA sin imágenes principales asociadas en el flujo de bits. En respuesta a la detección de una imagen BLA, el descodificador de vídeo 30 puede activar uno o más conjuntos de parámetros nuevos. Los conjuntos de parámetros se explicarán con más detalle más adelante en esta divulgación.

[0049] En algunos ejemplos, el descodificador de vídeo 30 está configurado para marcar todas las imágenes de referencia como no usadas para referencia si detecta un tipo de unidad NAL BLA. Si el descodificador 30 detecta un tipo de unidad NAL BLA, las imágenes de referencia no son válidas para descodificar la imagen BLA o cualquier imagen que siga a la imagen BLA en orden de descodificación o de salida.

[0050] El dispositivo de visualización 32 puede estar integrado en, o ser externo a, el dispositivo de destino 14. En algunos ejemplos, el dispositivo de destino 14 puede incluir un dispositivo de visualización integrado y también estar configurado para interconectarse con un dispositivo de visualización externo. En otros ejemplos, el dispositivo de destino 14 puede ser un dispositivo de visualización. En general, el dispositivo de visualización 32 visualiza los datos de vídeo descodificados ante un usuario, y puede comprender cualquiera de una variedad de dispositivos de visualización, tales como una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos emisores de luz (OLED) u otro tipo de dispositivo de visualización.

[0051] El codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con una norma de compresión de vídeo, tal como la norma de codificación de vídeo de alta eficiencia (HEVC), actualmente en fase de desarrollo, y pueden ajustarse al modelo de prueba de HEVC (HM). De forma alternativa, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con otras normas de propiedad o industriales, tales como la norma ITU-T H.264, denominada de forma alternativa MPEG-4, parte 10, codificación avanzada de vídeo (AVC), o ampliaciones de dichas normas. Sin embargo, las técnicas de esta divulgación no están limitadas a ninguna norma de codificación particular. Otros ejemplos de normas de compresión de vídeo incluyen MPEG-2 e ITU-T H.263. El HM supone varias capacidades adicionales de los dispositivos de codificación de vídeo respecto a los dispositivos existentes, de acuerdo con, por ejemplo, la norma ITU-T H.264/AVC. Por ejemplo, mientras que la norma H.264 proporciona nueve modos de codificación mediante intrapredicción, el HM puede proporcionar hasta treinta y tres modos de codificación mediante intrapredicción.

[0052] Aunque no se muestra en la FIG. 1, en algunos aspectos, tanto el codificador de vídeo 20 como el descodificador de vídeo 30 pueden estar integrados con un codificador y descodificador de audio, y pueden incluir unidades MUX-DEMUX adecuadas, u otro tipo de hardware y software, para gestionar la codificación tanto de audio como de vídeo en un flujo de datos común o en flujos de datos independientes. Si procede, en algunos ejemplos, las unidades MUX-DEMUX pueden ajustarse al protocolo de multiplexor ITU H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

[0053] Tanto el codificador de vídeo 20 como el descodificador de vídeo 30 pueden implementarse como cualquiera de 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 puertas programables in situ (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. Cuando las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio adecuado no transitorio legible por ordenador, y ejecutar las instrucciones en hardware usando uno o más procesadores para realizar las técnicas de esta divulgación. Tanto el codificador de vídeo 20 como el descodificador de vídeo 30 se pueden incluir en uno o más codificadores o descodificadores, cualquiera de los cuales se puede integrar como parte de un codificador/descodificador (CÓDEC) combinado en un dispositivo respectivo.

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

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

[0056] La norma HEVC admite transformaciones de acuerdo con las TU, que pueden ser diferentes para diferentes CU. El tamaño de las TU se basa típicamente en el tamaño de las PU dentro de una CU dada definida para una LCU

dividida, aunque puede que no sea siempre así. Las TU son típicamente del mismo tamaño o de un tamaño más pequeño que las PU. En algunos ejemplos, las muestras residuales correspondientes a una CU pueden subdividirse en unidades más pequeñas utilizando una estructura de árbol cuaternario conocida como "árbol cuaternario residual" (RQT). Los nodos hoja del RQT pueden denominarse unidades de transformada (TU). Los valores de diferencias de píxeles asociados a las TU pueden transformarse para generar coeficientes de transformada, que pueden cuantificarse.

[0057] En general, una PU incluye datos relacionados con el proceso de predicción. Por ejemplo, cuando la PU se codifica en intramodo, la PU puede incluir datos que describen un modo de intrapredicción para la PU. Como otro ejemplo, cuando la PU se codifica en intermodo, la PU puede incluir datos que definen un vector de movimiento para la PU. Los datos que definen el vector de movimiento para una PU pueden describir, por ejemplo, una componente horizontal del vector de movimiento, una componente vertical del vector de movimiento, una resolución para el vector de movimiento (por ejemplo, precisión de un cuarto de píxel o precisión de un octavo de píxel), una imagen de referencia a la que apunta el vector de movimiento y/o una lista de imágenes de referencia (por ejemplo, lista 0, lista 1 o lista C) para el vector de movimiento.

[0058] En general, se usa una TU para los procesos de transformada y cuantificación. Una CU dada que tiene una o más PU también puede incluir una o más unidades de transformada (TU). Tras la predicción, el codificador de vídeo 20 puede calcular valores residuales correspondientes a la PU. Los valores residuales comprenden valores de diferencias de píxeles que pueden ser transformados en coeficientes de transformada, cuantificados y explorados usando las TU, para generar coeficientes de transformada en serie para la codificación de entropía. Esta divulgación usa típicamente el término "bloque de vídeo" para referirse a un nodo de codificación de una CU. En algunos casos específicos, esta divulgación también puede usar el término "bloque de vídeo" para referirse a un bloque de árbol, es decir, una LCU o una CU, que incluye un nodo de codificación y unas PU y TU.

[0059] Una secuencia de vídeo incluye típicamente una serie de tramas o imágenes de vídeo. Un grupo de imágenes (GOP) comprende, en general, una serie de una o más de las imágenes de vídeo. Un GOP puede incluir datos sintácticos en una cabecera del GOP, una cabecera de una o más de las imágenes, o en otras ubicaciones, que describen una serie de imágenes incluidas en el GOP. Cada fragmento de una imagen puede incluir datos sintácticos de fragmento que describen un modo de codificación para el fragmento respectivo. El codificador de vídeo 20 actúa típicamente en bloques de vídeo dentro de fragmentos de vídeo individuales a fin de codificar los datos de vídeo. Un bloque de vídeo puede corresponder a un nodo de codificación dentro de una CU. Los bloques de vídeo pueden tener tamaños fijos o variables y pueden diferir en tamaño de acuerdo con una norma de codificación especificada.

[0060] Como un ejemplo, el HM presta soporte a la predicción en diversos tamaños de PU. Si se supone que el tamaño de una CU particular es $2N \times 2N$, el HM admite la intrapredicción en tamaños de PU de $2N \times 2N$ o $N \times N$, así como la interpredicción en tamaños de PU simétricas de $2N \times 2N$, $2N \times N$, $N \times 2N$ o $N \times N$. El HM también admite la división asimétrica para la interpredicción en tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$. En la división asimétrica, una dirección de una CU no está dividida, mientras que la otra dirección está dividida entre un 25 % y un 75 %. La parte de la CU correspondiente a la partición del 25 % está indicada por una "n" seguida de una indicación de "arriba", "abajo", "izquierda" o "derecha". Así pues, por ejemplo, " $2N \times nU$ " se refiere a una CU de tamaño $2N \times 2N$ que está dividida horizontalmente, con una PU de tamaño $2N \times 0,5N$ encima y una PU de tamaño $2N \times 1,5N$ debajo.

[0061] En esta divulgación, " $N \times N$ " y " N por N " pueden usarse indistintamente para hacer referencia a las dimensiones de píxeles de un bloque de vídeo en términos de dimensiones verticales y horizontales, por ejemplo, 16×16 píxeles o 16 por 16 píxeles. En general, un bloque de 16×16 tendrá 16 píxeles en una dirección vertical ($y = 16$) y 16 píxeles en una dirección horizontal ($x = 16$). Asimismo, un bloque de tamaño $N \times N$ tiene, en general, N píxeles en una dirección vertical y N píxeles en una dirección horizontal, donde N representa un valor entero no negativo. Los píxeles de un bloque se pueden disponer en filas y columnas. Además, no es necesario que los bloques tengan necesariamente el mismo número de píxeles en la dirección horizontal que en la dirección vertical. Por ejemplo, los bloques pueden comprender $N \times M$ píxeles, donde M no es necesariamente igual a N .

[0062] Tras la codificación intrapredictiva o interpredictiva, usando las PU de una CU, el codificador de vídeo 20 puede calcular datos residuales para las TU de la CU. Las PU pueden comprender datos de píxeles en el dominio espacial (también denominado dominio del píxel) y las TU pueden comprender coeficientes en el dominio de la transformada tras la aplicación de una transformada, por ejemplo, una transformada discreta de coseno (DCT), una transformada de enteros, una transformada de ondículas o una transformada conceptualmente similar, a los datos de vídeo residuales. Los datos residuales pueden corresponder a diferencias de píxeles entre píxeles de la imagen no codificada y valores de predicción correspondientes a las PU. El codificador de vídeo 20 puede formar las TU incluyendo los datos residuales para la CU y, a continuación, transformar las TU para generar coeficientes de transformada para la CU.

[0063] Tras cualquier transformada para generar coeficientes de transformada, el codificador de vídeo 20 puede realizar la cuantificación de los coeficientes de transformada. La cuantificación se refiere, en general, a un proceso en el que unos coeficientes de transformada se cuantifican para reducir posiblemente la cantidad de datos usados para representar los coeficientes, proporcionando una compresión adicional. El proceso de cuantificación puede reducir la

profundidad de bits asociada a algunos, o a la totalidad, de los coeficientes. Por ejemplo, un valor de n bits puede redondearse a la baja hasta un valor de m bits durante la cuantificación, donde n es mayor que m .

[0064] En algunos ejemplos, el codificador de vídeo 20 puede utilizar un orden de exploración predefinido para explorar los coeficientes de transformada cuantificados para producir un vector serializado que se pueda codificar por entropía. En otros ejemplos, el codificador de vídeo 20 puede realizar una exploración adaptativa. Después de explorar los coeficientes de transformada cuantificados para formar un vector unidimensional, el codificador de vídeo 20 puede realizar la codificación de entropía del vector unidimensional, por ejemplo, de acuerdo con la codificación de longitud variable adaptativa al contexto (CAVLC), la codificación aritmética binaria adaptativa al contexto (CABAC), la codificación aritmética binaria adaptativa al contexto basada en la sintaxis (SBAC), la codificación de entropía de división de intervalos de probabilidad (PIPE) u otra metodología de codificación de entropía. El codificador de vídeo 20 también puede realizar la codificación de entropía de los elementos sintácticos asociados a los datos de vídeo codificados, para su uso por el descodificador de vídeo 30 en la descodificación de los datos de vídeo.

[0065] Para realizar la codificación CABAC, el codificador de vídeo 20 puede asignar un contexto, dentro de un modelo de contexto, a un símbolo a transmitir. El contexto se puede referir, por ejemplo, a si los valores vecinos del símbolo son distintos de cero o no. Para realizar la codificación CAVLC, el codificador de vídeo 20 puede seleccionar un código de longitud variable para un símbolo a transmitir. Las palabras de código en la VLC se pueden construir de modo que los códigos relativamente más cortos correspondan a símbolos más probables, mientras que los códigos más largos correspondan a símbolos menos probables. De esta forma, el uso de la VLC puede lograr un ahorro en bits con respecto, por ejemplo, al uso de palabras de código de igual longitud para cada símbolo que se va a transmitir. La determinación de la probabilidad se puede basar en un contexto asignado al símbolo.

[0066] El codificador de vídeo 20 se puede configurar para codificar datos de vídeo de manera jerárquica. Por ejemplo, el codificador de vídeo 20 puede categorizar jerárquicamente los datos de vídeo como incluyendo una pluralidad de capas, una secuencia de imágenes dentro de una capa dada, una imagen dentro de una secuencia, fragmentos dentro de una imagen y bloques (por ejemplo, macrobloques o unidades de árbol de codificación) dentro de un fragmento. Un conjunto de parámetros de vídeo (VPS) puede señalar parámetros que cambian con poca frecuencia para una pluralidad de secuencias a través de capas respectivas. Se pueden usar conjuntos de parámetros de secuencia (SPS) para señalar parámetros que cambian con poca frecuencia para una secuencia de imágenes, y se pueden usar conjuntos de parámetros de imagen (PPS) para señalar parámetros que cambian con poca frecuencia para imágenes individuales.

[0067] Como se indicó anteriormente, esta divulgación describe técnicas para mejorar el soporte de adaptación y empalme de flujos en función de imágenes CRA. Una imagen CRA es una imagen codificada usando técnicas de intracodificación, de modo que la descodificación de las imágenes CRA no depende de la información de otras imágenes.

[0068] El acceso aleatorio se refiere, en general, a la capacidad de descodificación de un flujo de bits de vídeo empezando por una imagen codificada que no es la primera imagen codificada del flujo de bits. El acceso aleatorio a un flujo de bits de datos de vídeo codificados puede ser deseable en muchas aplicaciones de vídeo, tales como radiodifusión y transmisión continua. La transmisión continua de datos de vídeo a través de una red se puede realizar, por ejemplo, usando la transmisión continua dinámica adaptativa sobre HTTP (DASH). Por ejemplo, los espectadores de datos de vídeo pueden desear cambiar entre diferentes canales, saltar a partes específicas de un vídeo, realizar el denominado "modo truco", tal como avanzar o retroceder rápidamente, o cambiar a un flujo de bits diferente (por ejemplo, una representación diferente en DASH) para la adaptación del flujo (por ejemplo, adaptación de la velocidad de bits (por ejemplo para acomodar el ancho de banda de red fluctuante), velocidad de tramas, resolución espacial, etc.).

[0069] Una característica de acceso aleatorio puede habilitarse insertando imágenes de acceso aleatorio o puntos de acceso aleatorios, muchas veces en intervalos regulares, en un flujo de bits de datos de vídeo. Una unidad IDR pueden usarse para acceso aleatorio. Sin embargo, dado que una unidad IDR inicia típicamente una secuencia de vídeo codificada y limpia una DPB, las imágenes que siguen a la unidad IDR en orden de descodificación no pueden usar imágenes descodificadas antes que la unidad IDR como referencias. Por consiguiente, los flujos de bits que dependen de unidades IDR para el acceso aleatorio tienen en ocasiones una eficiencia de codificación significativamente menor. Para mejorar la eficiencia de codificación asociada a las unidades IDR, el concepto de imágenes CRA se introdujo en la HEVC para permitir a imágenes que sigan a la imagen CRA en orden de descodificación, pero que la precedan en orden de salida, usar imágenes descodificadas antes de la imagen CRA como referencias.

[0070] Las FIG. 2A y 2B son diagramas conceptuales que ilustran el concepto de imágenes CRA. La FIG. 2A muestra una secuencia de imágenes 202, donde el orden de izquierda a derecha indica el orden en que se descodifican las imágenes. Cada imagen también tiene un valor de cómputo de orden de imágenes (POC) que representa un orden de visualización para las imágenes. El valor POC se muestra en relación con la imagen CRA 204, de modo que los valores negativos indican imágenes que se muestran antes de la imagen CRA 204 y los valores positivos indican imágenes que se muestran después de la imagen CRA 204. La secuencia de imágenes 202 incluye

la imagen CRA 204. La secuencia de imágenes 202 incluye además una serie de imágenes retardadas 206, una serie de imágenes principales 208 y una serie de imágenes normales 210. Las imágenes retardadas 206 son anteriores a la imagen CRA 204 en el orden de descodificación, pero la siguen en el orden de visualización (es decir, el POC de las imágenes retardadas 206 es mayor que el POC de la imagen CRA 204). Las imágenes principales 208 siguen a la imagen CRA 204 en el orden de descodificación, pero preceden a la imagen CRA 204 en el orden de salida. Las imágenes normales 210 siguen a la imagen CRA 204 tanto en el orden de descodificación como en el orden de salida.

[0071] La FIG. 2B muestra una secuencia de imágenes 203. La secuencia de imágenes 203 comprende las mismas imágenes que la secuencia de imágenes 202, pero la secuencia de imágenes 203 muestra las imágenes en el orden de visualización, de modo que el orden de las imágenes de izquierda a derecha corresponde al orden en que se emitirán las imágenes. Las imágenes que siguen a una imagen CRA en orden de descodificación, pero que preceden a la imagen CRA en orden de salida, se denominan imágenes principales de la imagen CRA (por ejemplo, las imágenes principales 208). Las imágenes principales de una imagen CRA se pueden descodificar correctamente si la descodificación comienza desde una unidad IDR o una imagen CRA antes de la imagen CRA actual. Sin embargo, las imágenes principales de una imagen CRA no se pueden descodificar correctamente cuando se produce un acceso aleatorio desde la imagen CRA. Por tanto, estas imágenes principales se descartan típicamente durante la descodificación de acceso aleatorio. Para evitar la propagación de errores a partir de imágenes de referencia que pueden no estar disponibles dependiendo de dónde comienza la descodificación, todas las imágenes que siguen a una imagen CRA tanto en orden de descodificación como en orden de salida no deben usar ninguna imagen que preceda a la imagen CRA en orden de descodificación u orden de salida (lo cual incluye las imágenes principales) como referencia.

[0072] Funcionalidades de acceso aleatorio similares son admitidas en H.264/AVC con el mensaje de información de mejora suplementaria (SEI) de punto de recuperación. Una implementación de descodificador H.264/AVC puede admitir o no la funcionalidad. En HEVC, un flujo de bits que comienza con una imagen CRA se considera un flujo de bits conforme a la norma. Cuando un flujo de bits comienza con una imagen CRA, las imágenes principales de la imagen CRA pueden hacer referencia a imágenes de referencia no disponibles y, por lo tanto, no pueden descodificarse correctamente. Sin embargo, la norma HEVC especifica que las imágenes principales de la imagen CRA inicial no se proporcionan, de ahí el nombre "acceso aleatorio limpio". Para el establecimiento de un requisito de conformidad con el flujo de bits, HEVC especifica un proceso de descodificación para generar imágenes de referencia no disponibles para la descodificación de las imágenes principales de no salida. Sin embargo, las implementaciones de descodificador conformes a la norma no tienen que seguir ese proceso de descodificación, siempre que el descodificador pueda generar un resultado idéntico al del proceso de descodificación que se realiza desde el comienzo de la secuencia de vídeo codificada.

[0073] Cabe señalar que en HEVC, un flujo de bits conforme a la norma puede no contener ninguna unidad IDR y, en consecuencia, puede contener un subconjunto de una secuencia de vídeo codificada o una secuencia de vídeo codificada incompleta. En HEVC, una secuencia de vídeo codificada se define como una secuencia de unidades de acceso que consiste, en orden de descodificación, en una unidad de acceso IDR seguida de cero o más unidades de acceso no IDR, incluidas todas las unidades de acceso subsiguientes hasta, pero sin incluir, cualquier unidad de acceso IDR subsiguiente.

[0074] El concepto de "imágenes CRA con enlaces rotos" se introdujo en la especificación JCTVC-10404 (disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/9_Geneva/wg11/JCTVC-I0404-v1.zip). Comparado con el concepto CRA incluido en HEVC WD6, en JCTVC-I0404 se propone el permitir de forma adicional a las imágenes CRA que no están al principio del flujo de bits tener imágenes principales no descodificables, como puede tenerlas una imagen CRA que comienza el flujo de bits. En HEVC WD6, se permite que una imagen CRA que comienza un flujo de bits esté seguida (en orden de flujo de bits, también denominado orden de descodificación) de imágenes principales que no se pueden descodificar debido a la ausencia de imágenes de referencia anteriores. Sin embargo, no se permite que una imagen CRA que está situada en medio de un flujo de bits, o en algún otro sitio que no sea al principio del flujo de bits, tenga dichas imágenes principales no descodificables. En JCTVC-I0404 se propone la eliminación de esta restricción mediante adición de un indicador de "enlace roto", que indicaría la posible presencia de dichas imágenes principales no descodificables.

[0075] El indicador de "enlace roto" se propone en JCTVC-I0404 como información de nivel de imagen de imágenes CRA en la cabecera de fragmento o en algún otro lugar para información de nivel de imagen, tal como el conjunto de parámetros de adaptación (APS). Cuando el indicador es igual a 1, se permitirá que el flujo de bits contenga imágenes principales de la imagen CRA que no sean descodificables debido a que faltan imágenes de referencia precedentes, aunque el flujo de bits comience con una imagen IDR o una imagen CRA anterior en el orden de flujo de bits.

[0076] Para una imagen CRA con el indicador de "enlace roto" igual a 1, además de poder tener imágenes principales no descodificables como se mencionó anteriormente, un bit más significativo (MSB) de cómputo de orden de imágenes (POC) puede fijarse a 0, contiene un `no_output_of_prior_pics_flag` que actúa de la misma manera que para una imagen IDR, y un `random_access_pic_id` que actúa de la misma manera que el `idr_pic_id` de las imágenes IDR. Además, el `idr_pic_id` actual (como en HEVC WD 6) se renombra como `random_access_pic_id` y sus restricciones deberán realizarse para que se apliquen tanto a las imágenes CRA como a las imágenes IDR, en lugar

de solo a las imágenes IDR. Al igual que una imagen IDR, una imagen CRA con `broken_link_flag` igual a 1 podría activar un conjunto de parámetros de secuencia (SPS) diferente, cambiar el tamaño de imagen, etc.

[0077] El enfoque de "imágenes CRA con enlaces rotos" en JCTVC-I0404 y otros enfoques de acceso aleatorio de vídeo existentes pueden, en algunos casos, estar asociados a varios problemas o deficiencias. Por ejemplo, la señalización del indicador de "enlace roto" en la cabecera de fragmento o el APS requiere una entidad (por ejemplo, un servidor, un MANE o un editor/empalmador de vídeo) que cambie una imagen CRA normal para que sea una imagen BLC, cuando sea necesario, para poder codificar por entropía y analizar sintácticamente la cabecera de fragmento y/o el APS para codificar el indicador. Esto también requiere que la entidad que identifica una imagen BLC cuando sea necesario sea capaz de descodificar por entropía y analizar sintácticamente la cabecera de fragmento y/o el APS para encontrar el indicador.

[0078] Como otro ejemplo, puede que no se permita que una imagen CRA de enlace roto (BLA) active un SPS, un conjunto de parámetros de imagen (PPS) o un APS (cuando la imagen haga referencia a este) si el ID de conjunto de parámetros es el mismo que el SPS, el PPS o el APS activo. Sin embargo, debido a que una imagen BLC se origina, en general, a partir de un flujo de bits diferente al de la imagen anterior en orden de descodificación, la imagen BLC usa, en general, diferentes cargas útiles de secuencia de bits sin procesar SPS (RBSP), RBSP PBS y RBSP APS, y es probable que tanto la imagen BLC como la imagen anterior en orden de descodificación se refieran (directa o indirectamente) al mismo valor de ID de SPS o PPS. También es posible que hagan referencia al mismo valor de ID de APS. El uso del SPS, PPS o APS activo para la imagen anterior en la descodificación puede hacer, en algunos casos, que la descodificación de la imagen BLC y de las siguientes imágenes (no solo las imágenes principales) sea incorrecta.

[0079] Como otro ejemplo, sin incluir el `random_access_pic_id` y `no_output_of_prior_pics_flag` en imágenes CRA normales, se requiere una entidad que cambie una imagen CRA normal para que sea una imagen BLC, cuando sea necesario, para poder codificar por entropía y analizar sintácticamente la cabecera de fragmento y/o el APS para codificar los elementos sintácticos. Como otro ejemplo más, dado que una imagen BLC se origina, en general, a partir de un flujo de bits diferente al de la imagen anterior en orden de descodificación, si `no_output_of_prior_pics_flag` es igual a 0, la memoria intermedia de imágenes descodificadas puede desbordarse; por tanto, todas las imágenes siguientes en orden de descodificación pueden descodificarse incorrectamente o incluso el descodificador puede fallar.

[0080] Como otro ejemplo, cuando una parte de un primer flujo de bits y una parte de un segundo flujo de bits, donde la parte del segundo flujo de bits comienza desde una imagen CRA `picA`, se empalman o concatenan, es posible no cambiar la imagen CRA a una imagen BLC, sino más bien mantener `picA` como imagen CRA en caso de que, cuando la descodificación comience desde la imagen CRA o la imagen IDR o la imagen BLC anterior, la calidad de descodificación de las imágenes principales de la imagen CRA `picA` sea aceptable pero no perfecta (por ejemplo, cuando las sumas de verificación de una o más de las imágenes principales no coinciden con las sumas de verificación indicadas en los mensajes SEI hash de imagen descodificada como en HVEC WD6). Sin embargo, no se dispone de un mecanismo para indicar la información anterior en el flujo de bits.

[0081] Esta divulgación describe técnicas que, en algunos casos, pueden mejorar algunas de las deficiencias analizadas anteriormente. Las diversas técnicas descritas en el presente documento pueden implementarse individualmente o en combinación con otras técnicas descritas. Las siguientes técnicas se basan en el concepto de imágenes CRA con enlaces rotos como se establece en JCTVC-I0404.

[0082] Según una técnica, en lugar de usar un indicador en la cabecera de fragmento para indicar una imagen CRA con un enlace roto (es decir, una imagen BLC), un tipo de unidad de capa de abstracción de red (NAL) distinto (por ejemplo, tipo de unidad NAL igual a 2 que está reservado en HEVC WD6) puede usarse para indicar que la unidad NAL pertenece a una imagen de enlace roto, que puede denominarse imagen BLC o BLA. Una imagen BLC puede comprender más de una unidad NAL. Dicha técnica puede ser menos compleja para una entidad de red ya que la técnica puede no requerir codificación o descodificación de entropía para que la entidad cambie una imagen CRA normal (por ejemplo, cambiando un tipo de unidad NAL) para que sea una imagen BLC, cuando sea necesario. Además, puede no requerir codificación de entropía para que la entidad identifique una imagen BLC cuando sea necesario. Para una imagen CRA normal, que no inicia un flujo de bits, todas las imágenes principales asociadas pueden necesitar ser descodificables (es decir, pueden descodificarse correctamente).

[0083] Según otra técnica, en lugar de permitir que una imagen BLC active un SPS diferente, se requiere que una imagen BLC active un SPS, incluso si el ID de SPS al que hace referencia la imagen BLC (indirectamente a través de un PPS como en HEVC WD6 o cualquier otro medio, por ejemplo, indirectamente a través de un conjunto de parámetros de grupo como se describe en JCTVC-I0338, o directamente, por ejemplo, cuando el ID de SPS se incluye directamente en la cabecera de fragmento, o indirectamente a través de un mensaje SEI de período de almacenamiento en memoria intermedia) es el mismo que el ID de SPS del SPS activo para la imagen anterior en orden de descodificación. Esto es así porque una imagen BLC era típicamente de un flujo de bits diferente al de la imagen anterior en orden de descodificación, y típicamente se aplican RBSP PPS diferentes, con ID de PPS idénticos o diferentes.

[0084] También se requiere que una imagen BLC active un PPS, incluso si el ID de PPS al que hace referencia la imagen BLC (indirectamente a través de un conjunto de parámetros de grupo como el descrito en JCTVC-10338, o directamente, por ejemplo, cuando el ID de PSS se incluye directamente en la cabecera de fragmento como en HEVC WD6) es igual al ID de PSS del PPS activo para la imagen anterior en orden de descodificación. De nuevo, esto es así porque una imagen BLC era típicamente de un flujo de bits diferente al de la imagen anterior en orden de descodificación, y típicamente se aplican RBSP PPS diferentes, con ID de PPS idénticos o diferentes.

[0085] También se requiere que una imagen BLC active un APS si hace referencia a un APS, incluso si el ID de APS al que hace referencia la imagen BLC es el mismo que el ID de APS del PPS activo para la imagen anterior en orden de descodificación. De nuevo, esto es así porque una imagen BLC era típicamente de un flujo de bits diferente al de la imagen anterior en orden de descodificación, y típicamente se aplican RBSP APS diferentes, con ID de APS idénticos o diferentes.

[0086] Tal como se usa en esta divulgación, una imagen BLC se puede definir como una imagen codificada para la cual el tipo de unidad NAL es el de la imagen BLC (por ejemplo, tipo de unidad NAL igual a 2 que está reservado en HEVC WD6). Una imagen BLC puede considerarse un tipo especial de imagen CRA. El mismo proceso de descodificación para una imagen CRA no BLC cuando comienza el flujo de bits y las imágenes principales asociadas se aplica para descodificar una imagen BLC y las imágenes principales asociadas, incluso si la imagen BLC no es la primera imagen del flujo de bits. De forma alternativa, las imágenes BLC pueden excluirse de las imágenes CRA, es decir, puede considerarse que una imagen BLC no es una imagen CRA. En ese caso, el mismo proceso de descodificación para una imagen CRA cuando comienza el flujo de bits y las imágenes principales asociadas se aplica para descodificar una imagen BLC y las imágenes principales asociadas, incluso si la imagen BLC no es la primera imagen del flujo de bits. En la siguiente descripción de este documento, se supone que se aplica esta alternativa. Una unidad de acceso BLC puede definirse como una unidad de acceso en la que la imagen codificada es una imagen BLC. La definición de secuencia de vídeo codificada puede cambiarse de la siguiente manera: una secuencia de unidades de acceso que incluye, en orden de descodificación, una unidad de acceso IDR o una unidad de acceso BLC seguida de cero o más unidades de acceso no IDR y no BLC, incluidas todas las unidades de acceso subsiguientes hasta, pero sin incluir, cualquier unidad de acceso IDR o BLC subsiguiente.

[0087] Según otra técnica, en lugar de tener uno o ambos de `random_access_pic_id` y `no_output_of_prior_pics_flag` solo para imágenes IDR e imágenes BLC como en JCTVC-I0404, cualquiera de los dos campos puede estar siempre presente para todas las imágenes IDR, imágenes BLC y todas las imágenes CRA. Para cada imagen CRA, puede requerirse que `no_output_of_prior_pics_flag` sea igual a 0. Dicha técnica puede, en algunos casos, facilitar que una entidad de red cambie una imagen CRA a una imagen BLC, cuando sea necesario. De forma alternativa, para una imagen BLC, puede requerirse que `no_output_of_prior_pics_flag` sea igual a 1. De forma alternativa, cada imagen BLC puede no tener un `no_output_of_prior_pics_flag` señalado, pero el comportamiento de salida de imagen puede ser el mismo que si tuviera `no_output_of_prior_pics_flag` igual a 1. De forma alternativa, cada imagen BLC puede tener señalado `no_output_of_pior_pics_flag`, pero el comportamiento de salida de imagen puede ser el mismo que si tuviera `no_output_of_prior_pics_flag` igual a 1, independientemente del valor del `no_output_of_prior_pics_flag` señalado.

[0088] Según otra técnica, el valor MSB POC para una imagen BLC puede señalarse, por ejemplo, en la cabecera de fragmento. Si se señala, el valor se seguiría considerando igual a 0 en el proceso de descodificación, independientemente del valor. De forma alternativa, el valor MSB POC señalado se usa en el proceso de descodificación, pero a continuación el empalmador necesita verificar y posiblemente cambiar el valor para que sea coherente con los valores POC de imágenes anteriores en orden de descodificación.

[0089] Según otra técnica, cuando una parte de un primer flujo de bits y una parte de un segundo flujo de bits, donde la parte del segundo flujo de bits comienza desde una imagen CRA `picA`, se empalman o concatenan, el empalmador puede mantener `picA` como una imagen CRA en caso de que, cuando la descodificación comienza desde la imagen CRA o la imagen IDR o la imagen BLC anterior, la calidad de descodificación de las imágenes principales de la imagen CRA `picA` es aceptable pero no perfecta (por ejemplo, cuando las sumas de verificación de una o más de las imágenes principales no coinciden con las sumas de verificación señaladas en los mensajes SEI hash de imágenes descodificadas como en HVEC WD6). Una indicación de la información anterior se puede señalar en el flujo de bits. La información puede señalarse a través de una indicación asociada a `picA`, por ejemplo, como un indicador en la cabecera de unidad NAL o la cabecera de fragmento o el APS al que se hace referencia, o un mensaje SEI asociado a `picA`. El indicador se puede denominar indicador de coincidencia exacta, el valor 1 indica que la suma de verificación de cada imagen principal asociada a `picA` coincide con la suma de verificación señalada en el mensaje SEI hash de imágenes descodificadas, si está presente, y el valor 0 indica que la suma de verificación de cada imagen principal asociada a `picA` puede o no coincidir con la suma de verificación señalada en el mensaje SEI hash de imágenes descodificadas, si está presente.

[0090] La FIG. 3 es un diagrama de bloques que ilustra un ejemplo de codificador de vídeo 20 que puede implementar las técnicas descritas en esta divulgación. El codificador de vídeo 20 puede estar configurado para proporcionar vídeo a una entidad de posprocesamiento 27. La entidad de posprocesamiento 27 pretende representar

un ejemplo de una entidad de vídeo, tal como un dispositivo MANE o de empalme/edición, que puede procesar datos de vídeo codificados del codificador de vídeo 20. En algunos casos, la entidad de posprocesamiento puede ser un ejemplo de una entidad de red. En algunos sistemas de codificación de vídeo, la entidad de posprocesamiento 27 y el codificador de vídeo 20 pueden formar parte de dispositivos separados, mientras que, en otros casos, la funcionalidad descrita con respecto a la entidad de posprocesamiento 27 se puede realizar por el mismo dispositivo que comprende el codificador de vídeo 20.

[0091] El codificador de vídeo 20 puede realizar la intracodificación y la intercodificación de bloques de vídeo dentro de fragmentos de vídeo. La intracodificación se basa en la predicción espacial para reducir o eliminar la redundancia espacial de un vídeo dentro de una trama o imagen de vídeo dada. La intercodificación se basa en la predicción temporal para reducir o eliminar la redundancia temporal de un vídeo dentro de tramas o imágenes adyacentes de una secuencia de vídeo. El intramodo (modo I) puede referirse a cualquiera de varios modos de compresión espacial. Los intermodos, tales como la predicción unidireccional (modo P) o la predicción bidireccional (modo B), pueden referirse a cualquiera de varios modos de compresión temporal.

[0092] En el ejemplo de la FIG. 3, el codificador de vídeo 20 incluye una unidad de división 35, una unidad de procesamiento de predicción 41, una unidad de filtro 63, una memoria de imágenes 64, un sumador 50, una unidad de procesamiento de transformada 52, una unidad de cuantificación 54 y una unidad de codificación de entropía 56. La unidad de procesamiento de predicción 41 incluye una unidad de estimación de movimiento 42, una unidad de compensación de movimiento 44 y una unidad de procesamiento de intrapredicción 46. Para la reconstrucción de bloques de vídeo, el codificador de vídeo 20 incluye también una unidad de cuantificación inversa 58, una unidad de procesamiento de transformada inversa 60 y un sumador 62. La unidad de filtro 63 está destinada a representar uno o más filtros de bucle tales como un filtro de eliminación de bloques, un filtro de bucle adaptativo (ALF) y un filtro de desplazamiento adaptativo de muestras (SAO). Aunque la unidad de filtro 63 que se ilustra en la FIG. 3 es un filtro de bucle, en otras configuraciones, la unidad de filtro 63 puede implementarse como un filtro posbucle.

[0093] Como se ilustra en la FIG. 3, el codificador de vídeo 20 recibe datos de vídeo, y la unidad de división 35 divide los datos en bloques de vídeo. Esta división también puede incluir la división en fragmentos, mosaicos u otras unidades mayores, así como la división en bloques de vídeo, por ejemplo, de acuerdo con una estructura de árbol cuaternario de LCU y CU. El codificador de vídeo 20 ilustra, en general, los componentes que codifican bloques de vídeo dentro de un fragmento de vídeo que se va a codificar. El fragmento se puede dividir en múltiples bloques de vídeo (y, posiblemente, en conjuntos de bloques de vídeo denominados mosaicos). La unidad de procesamiento de predicción 41 puede seleccionar uno entre una pluralidad de posibles modos de codificación, tal como uno entre una pluralidad de modos de intracodificación, o uno entre una pluralidad de modos de intercodificación, para el bloque de vídeo actual basándose en resultados de error (por ejemplo, la velocidad de codificación y el nivel de distorsión). La unidad de procesamiento de predicción 41 puede proporcionar el bloque intracodificado o intercodificado resultante al sumador 50 para generar datos de bloque residuales y al sumador 62 para reconstruir el bloque codificado para su uso como imagen de referencia.

[0094] Como se ha analizado anteriormente, en algunos casos, el codificador de vídeo 20 puede codificar un punto de acceso aleatorio o un punto de adaptación de flujo, tal como un punto de conmutación de capa temporal, tal como una imagen BLA o BLC. Por ejemplo, la codificación puede tener lugar dentro de la unidad de codificación de entropía 56, que puede realizar codificación de entropía y de no entropía. Una o más de estas imágenes pueden ser imágenes principales de la imagen CRA. Las imágenes principales de una imagen CRA se pueden descodificar correctamente si la descodificación comienza por una imagen RAP anterior a la imagen CRA actual. Sin embargo, las imágenes principales de una imagen CRA no se pueden descodificar correctamente cuando se produce un acceso aleatorio desde la imagen CRA. Por ejemplo, las imágenes principales pueden apuntar a bloques para referencia de predicción que no están disponibles. En consecuencia, la imagen principal puede no ser descodificable en el descodificador de vídeo 30. En consecuencia, estas imágenes principales se descartan típicamente durante la descodificación de acceso aleatorio.

[0095] En un ejemplo, el codificador de vídeo 20 puede proporcionar un indicador en la cabecera de fragmento, por ejemplo, `no_output_of_prior_pics_flag` o un elemento sintáctico, de modo que no se proporciona ninguna de las imágenes anteriores, es decir, imágenes anteriores a la imagen BLA o BLC, en la DPB. En algunos ejemplos, este indicador (o elemento sintáctico) puede estar al principio en la cabecera de fragmento antes de la codificación de entropía, de modo que puede descodificarse más fácilmente en el descodificador 30, por ejemplo, y la información puede estar disponible antes en el proceso de codificación. El elemento sintáctico o indicador puede codificarse, por ejemplo, en una cabecera de fragmento para una imagen BLA o BLC, mediante la unidad de codificación de entropía 56 (que puede realizar codificación de no entropía). Esto puede ser útil, por ejemplo, para dispositivos intermedios tales como un MANE, de modo que la información proporcionada por el elemento sintáctico o indicador puede estar disponible para el dispositivo intermedio sin descodificación de entropía, aunque también puede ser útil para que un descodificador tenga acceso a dicha información antes de la descodificación de entropía.

[0096] Por ejemplo, el codificador de vídeo 20 (por ejemplo, la unidad de codificación de entropía 56) puede incluir un elemento sintáctico que indica que una memoria intermedia de almacenamiento de imágenes se vacía sin proporcionar ninguna imagen de la memoria intermedia de almacenamiento de imágenes. El elemento sintáctico, una

vez establecido, puede hacer que las imágenes, anteriores a una imagen actual en orden de descodificación y que residen en la memoria intermedia de almacenamiento de imágenes en el momento de la descodificación de la imagen actual, se extraigan de la memoria intermedia de almacenamiento de imágenes sin que se proporcionen. En algunos ejemplos, el elemento sintáctico puede ser uno de una pluralidad de elementos sintácticos. Adicionalmente, la pluralidad de elementos sintácticos puede incluir elementos sintácticos de cabecera de fragmento sometidos a codificación de entropía y elementos sintácticos de cabecera de fragmento no sometidos a codificación de entropía. En un ejemplo, el elemento sintáctico, que indica que una memoria intermedia de almacenamiento de imágenes se vacía sin proporcionar ninguna imagen de la memoria intermedia de almacenamiento de imágenes, se incluye en la cabecera de fragmento antes de cualquier elemento sintáctico de cabecera de fragmento sometido a codificación de entropía de modo que no se realiza la codificación de entropía del propio elemento sintáctico.

[0097] En algunos ejemplos, el elemento sintáctico puede ser `no_output_of_prior_pics_flag` y el `no_output_of_prior_pics_flag` puede estar incluido en la cabecera de fragmento justo después de un `first_slice_in_pic_flag`. El `first_slice_in_pic_flag` puede ser un indicador que indica si un fragmento es el primer fragmento, en orden de descodificación, de una imagen.

[0098] En un ejemplo, un codificador de vídeo 20 puede estar configurado para incluir tipos de unidad NAL asignados para indicar cuándo las imágenes BLA o BLC tienen y no tienen imágenes principales. Por ejemplo, en una norma, se incluyen los tipos de unidad NAL 16, `BLA_W_LP` (BLA con imagen principal); 17, `BLA_W_DLP` (BLA con imagen principal descodificable); y 18, `BLA_N_LP` (BLA sin imagen principal). La unidad de codificación de entropía 56 (que puede realizar una codificación de no entropía) puede codificar estos tipos de unidad NAL. En consecuencia, basándose en el tipo de unidad NAL, un descodificador puede saber cuándo una imagen BLA tiene una imagen principal y cuándo la imagen principal no es descodificable, por ejemplo, cuando la descodificación comienza por una imagen RAP anterior. En consecuencia, esta información podría usarse para determinar cuándo las imágenes principales pueden marcarse como no usadas para referencia, lo que puede hacer que el descodificador marque imágenes de referencia de la memoria intermedia de imágenes descodificadas como no usadas para referencia.

[0099] La unidad de procesamiento de intrapredicción 46, situada en la unidad de procesamiento de predicción 41, puede realizar la codificación intrapredictiva del bloque de vídeo actual con respecto a uno o más bloques vecinos en la misma trama o fragmento que el bloque actual que se va a codificar, para proporcionar compresión espacial. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 de la unidad de procesamiento de predicción 41 realizan la codificación interpredictiva del bloque de vídeo actual con respecto a uno o más bloques para referencia de predicción de una o más imágenes de referencia para proporcionar una compresión temporal.

[0100] La unidad de estimación de movimiento 42 puede estar configurada para determinar el modo de interpredicción para un fragmento de vídeo de acuerdo con un patrón predeterminado para una secuencia de vídeo. El patrón predeterminado puede designar fragmentos de vídeo en la secuencia como fragmentos P, fragmentos B o fragmentos GPB. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar sumamente integradas, pero se ilustran por separado con fines conceptuales. La estimación del movimiento, realizada por la unidad de estimación de movimiento 42, es el proceso de generación de vectores de movimiento, que estiman el movimiento para los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una PU de un bloque de vídeo de una trama o imagen de vídeo actual en relación con un bloque predictivo de una imagen de referencia.

[0101] Un bloque predictivo es un bloque que se corresponde estrechamente con la PU del bloque de vídeo que se va a codificar en términos de diferencia de píxel, que puede determinarse mediante una suma de diferencia absoluta (SAD), una suma de diferencia al cuadrado (SSD) u otras métricas de diferencia. En algunos ejemplos, el codificador de vídeo 20 puede calcular los valores para las posiciones fraccionarias de píxel de imágenes de referencia almacenadas en la memoria de imágenes 64. Por ejemplo, el codificador de vídeo 20 puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones fraccionarias de píxel de la imagen de referencia. Por lo tanto, la unidad de estimación de movimiento 42 puede realizar una búsqueda de movimiento relativa a las posiciones de píxel completo y las posiciones fraccionarias de píxel, y proporcionar un vector de movimiento con una precisión de píxel fraccionaria.

[0102] La unidad de estimación de movimiento 42 calcula un vector de movimiento para una PU de un bloque de vídeo en un fragmento intercodificado, comparando la posición de la PU con la posición de un bloque predictivo de una imagen de referencia. La imagen de referencia puede seleccionarse entre una primera lista de imágenes de referencia (Lista 0) o una segunda lista de imágenes de referencia (Lista 1), cada una de las cuales identifica una o más imágenes de referencia almacenadas en la memoria de imágenes 64. La unidad de estimación de movimiento 42 envía el vector de movimiento calculado a la unidad de codificación de entropía 56 y a la unidad de compensación de movimiento 44.

[0103] La compensación de movimiento, realizada por la unidad de compensación de movimiento 44, puede implicar extraer o generar el bloque predictivo en función del vector de movimiento determinado mediante estimación de movimiento, realizando posiblemente interpolaciones hasta una precisión de subpíxel. Tras recibir el vector de movimiento para la PU del bloque de vídeo actual, la unidad de compensación de movimiento 44 puede localizar el

bloque predictivo al que apunta el vector de movimiento en una de las listas de imágenes de referencia. El codificador de vídeo 20 forma un bloque de vídeo residual restando los valores de píxel del bloque predictivo a los valores de píxel del bloque de vídeo actual que se está codificando, formando valores de diferencia de píxel. Los valores de diferencia de píxel forman datos residuales para el bloque, y pueden incluir componentes de diferencia tanto de luminancia como de crominancia. El sumador 50 representa el componente o los componentes que realizan esta operación de resta. La unidad de compensación de movimiento 44 también puede generar elementos sintácticos asociados a los bloques de vídeo y al fragmento de vídeo para su uso por el descodificador de vídeo 30 en la descodificación de los bloques de vídeo del fragmento de vídeo.

[0104] La unidad de intrapredicción 46 puede intrapredecir un bloque actual, como alternativa a la interpredicción llevada a cabo por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como se describe anteriormente. En particular, la unidad de intrapredicción 46 puede determinar un modo de intrapredicción a usar para codificar un bloque actual. En algunos ejemplos, la unidad de intrapredicción 46 puede codificar un bloque actual usando diversos modos de intrapredicción, por ejemplo, durante pasadas de codificación independientes, y la unidad de intrapredicción 46 (o unidad de selección de modo 40, en algunos ejemplos) puede seleccionar un modo de intrapredicción apropiado para usar, entre los modos probados. Por ejemplo, la unidad de intrapredicción 46 puede calcular valores de velocidad-distorsión usando un análisis de velocidad-distorsión para los diversos modos de intrapredicción probados, y seleccionar el modo de intrapredicción que tenga las mejores características de velocidad-distorsión entre los modos probados. El análisis de velocidad-distorsión determina, en general, una cantidad de distorsión (o error) entre un bloque codificado y un bloque original no codificado que fue codificado para generar el bloque codificado, así como una velocidad de bits (es decir, un número de bits) usada para generar el bloque codificado. La unidad de intrapredicción 46 puede calcular razones a partir de las distorsiones y velocidades para los diversos bloques codificados, para determinar qué modo de intrapredicción presenta el mejor valor de velocidad-distorsión para el bloque.

[0105] En cualquier caso, tras seleccionar un modo de intrapredicción para un bloque, la unidad de intrapredicción 46 puede proporcionar a la unidad de codificación de entropía 56 información que indica el modo de intrapredicción seleccionado para el bloque. La unidad de codificación de entropía 56 puede codificar la información que indica el modo de intrapredicción seleccionado de acuerdo con las técnicas de esta divulgación. El codificador de vídeo 20 puede incluir datos de configuración en el flujo de bits transmitido. El flujo de bits puede incluir una pluralidad de tablas de índices de modos de intrapredicción y una pluralidad de tablas de índices de modos de intrapredicción modificadas (también denominadas tablas de correlación de palabras de código), definiciones de contextos de codificación para diversos bloques e indicaciones del modo de intrapredicción más probable, una tabla de índices de modos de intrapredicción y una tabla de índices de modos de intrapredicción modificada que se va a usar para cada uno de los contextos.

[0106] Después de que la unidad de procesamiento de predicción 41 genere el bloque predictivo para el bloque de vídeo actual, ya sea mediante interpredicción o intrapredicción, el codificador de vídeo 20 forma un bloque de vídeo residual restando el bloque predictivo al bloque de vídeo actual. Los datos de vídeo residuales del bloque residual pueden estar incluidos en una o más TU y aplicarse a la unidad de procesamiento de transformada 52. La unidad de procesamiento de transformada 52 transforma los datos de vídeo residuales en coeficientes de transformada residuales usando una transformada, tal como una transformada de coseno discreta (DCT) o una transformada conceptualmente similar. La unidad de procesamiento de transformada 52 puede convertir los datos de vídeo residuales de un dominio de píxel a un dominio de transformada, tal como un dominio de frecuencia.

[0107] La unidad de procesamiento de transformada 52 puede enviar los coeficientes de transformada resultantes a la unidad de cuantificación 54. La unidad de cuantificación 54 cuantifica los coeficientes de transformada para reducir más la velocidad de bits. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos, o a la totalidad, de los coeficientes. El grado de cuantificación se puede modificar ajustando un parámetro de cuantificación. En algunos ejemplos, la unidad de cuantificación 54 puede realizar, a continuación, una exploración de la matriz que incluye los coeficientes de transformada cuantificados. De forma alternativa, la unidad de codificación de entropía 56 puede realizar la exploración.

[0108] Tras la cuantificación, la unidad de codificación de entropía 56 realiza la codificación de entropía de los coeficientes de transformada cuantificados. Por ejemplo, la unidad de codificación de entropía 56 puede realizar una codificación de longitud variable adaptativa al contexto (CAVLC), una codificación aritmética binaria adaptativa al contexto (CABAC), una codificación aritmética binaria adaptativa al contexto basada en la sintaxis (SBAC), una codificación de entropía por división de intervalos de probabilidad (PIPE) u otra metodología o técnicas de codificación de entropía. Tras la codificación de entropía mediante la unidad de codificación de entropía 56, el flujo de bits codificado se puede transmitir al descodificador de vídeo 30, o archivarse para su posterior transmisión o recuperación mediante el descodificador de vídeo 30. La unidad de codificación de entropía 56 también puede realizar una codificación de entropía de los vectores de movimiento y los otros elementos sintácticos para el fragmento de vídeo actual que se está codificando.

[0109] La unidad de cuantificación inversa 58 y la unidad de procesamiento de transformada inversa 60 aplican una cuantificación inversa y una transformada inversa, respectivamente, para reconstruir el bloque residual en el dominio

del píxel, para su posterior uso como bloque de referencia de una imagen de referencia. La unidad de compensación de movimiento 44 puede calcular un bloque de referencia añadiendo el bloque residual a un bloque predictivo de una de las imágenes de referencia de una de las listas de imágenes de referencia. La unidad de compensación de movimiento 44 también puede aplicar uno o más filtros de interpolación al bloque residual reconstruido para calcular valores fraccionarios de píxeles, para su uso en la estimación de movimiento. El sumador 62 añade el bloque residual reconstruido al bloque predictivo con compensación de movimiento generado por la unidad de compensación de movimiento 44 para generar un bloque de referencia para su almacenamiento en la memoria de imágenes 64. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden usar el bloque de referencia como bloque de referencia para realizar una interpredicción de un bloque en una trama o imagen de vídeo subsiguiente.

[0110] La FIG. 4 es un diagrama de bloques que ilustra un descodificador de vídeo 30 de ejemplo que puede implementar las técnicas descritas para un apoyo mejorado de adaptación y empalme de flujos en función de imágenes RAP de enlace roto descritas anteriormente. En el ejemplo de la FIG. 4, el descodificador de vídeo 30 incluye una unidad de descodificación de entropía 80, una unidad de procesamiento de predicción 81, una unidad de cuantificación inversa 86, una unidad de transformada inversa 88, un sumador 90, una unidad de filtro 91 y una memoria de imágenes 92. La unidad de procesamiento de predicción 81 incluye una unidad de compensación de movimiento 82 y una unidad de procesamiento de intrapredicción 84. En algunos ejemplos, el descodificador de vídeo 30 puede realizar una pasada de descodificación que, en general, es recíproca a la pasada de codificación descrita con respecto al codificador de vídeo 20 de la FIG. 3.

[0111] Durante el proceso de descodificación, el descodificador de vídeo 30 recibe un flujo de bits de vídeo codificado, que representa bloques de vídeo de un fragmento de vídeo codificado y elementos sintácticos asociados, desde el codificador de vídeo 20. El descodificador de vídeo 30 puede recibir el flujo de bits de vídeo codificado desde una entidad de red 29. La entidad de red 29 puede ser, por ejemplo, un servidor, un MANE, un editor/empalmador de vídeo u otro dispositivo similar configurado para implementar una o más de las técnicas descritas anteriormente. La entidad de red 29 puede incluir o no un codificador de vídeo 20. Como se ha descrito anteriormente, algunas de las técnicas descritas en esta divulgación pueden ser implementadas por la entidad de red 29 antes de que la entidad de red 29 transmita el flujo de bits de vídeo codificado al descodificador de vídeo 30. En algunos sistemas de descodificación de vídeo, la entidad de red 29 y el descodificador de vídeo 30 pueden formar parte de dispositivos independientes, mientras que, en otros casos, el mismo dispositivo que comprende el descodificador de vídeo 30 puede realizar la funcionalidad descrita con respecto a la entidad de red 29.

[0112] La entidad de red 29 representa un ejemplo de un dispositivo de procesamiento de vídeo configurado para recibir una primera unidad de capa de abstracción de red (NAL) que comprende una parte de datos de vídeo y, en función de un tipo de unidad NAL de la primera unidad NAL, detectar una imagen de enlace roto. La entidad de red 29 también puede representar un ejemplo de un dispositivo de procesamiento de vídeo configurado para recibir una unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo; determinar que la unidad NAL comprende una imagen de enlace roto; y establecer un tipo de unidad NAL para la unidad NAL como un tipo de unidad NAL que indica que la unidad NAL comprende la imagen de enlace roto.

[0113] Como se analiza anteriormente, en algunos ejemplos, el descodificador de vídeo 30 puede descodificar una imagen RAP de enlace roto, tal como una imagen BLA, o imágenes que se hallan después de un punto de acceso aleatorio o un punto de adaptación de flujo, tal como un punto de conmutación de capa temporal, tales como imágenes principales. Las imágenes principales no se pueden descodificar correctamente cuando se produce un acceso aleatorio desde la imagen RAP de enlace roto.

[0114] En un ejemplo, para evitar la propagación de errores a partir de imágenes de referencia que pueden no estar disponibles dependiendo de dónde comienza la descodificación, el descodificador de vídeo 30 puede no usar ninguna imagen que preceda a la imagen RAP de enlace roto en orden de descodificación u orden de salida (lo cual incluye las imágenes principales) como imágenes de referencia. Por ejemplo, la unidad de procesamiento de predicción 81 puede no usar ninguna imagen almacenada en la memoria de imágenes 92 que preceda a la imagen RAP de enlace roto en orden de descodificación u orden de salida (lo cual incluye las imágenes principales) como imágenes de referencia.

[0115] En diversos ejemplos, el descodificador de vídeo 30 puede marcar todas las imágenes de referencia de la DPB como no usadas para referencia antes de descodificar la imagen BLA como no usada para referencia. Por ejemplo, la unidad de descodificación de entropía 80, que puede realizar descodificación de entropía y de no entropía, puede marcar imágenes de referencia de la memoria de imágenes 92, a veces denominada memoria intermedia de imágenes descodificadas (DPB), como no usadas para referencia. El descodificador de vídeo 30 (por ejemplo, la unidad de descodificación de entropía 80) puede determinar que una imagen actual es una imagen BLA o BLC y marcar las imágenes de referencia de una memoria intermedia de almacenamiento de imágenes como no usadas para referencia antes de descodificar la imagen BLA o BLC. Determinar que una imagen actual es una imagen BLA o BLC puede incluir determinar que la imagen actual es una imagen CRA y determinar que la imagen actual es una imagen RAP. La imagen CRA actual es una imagen BLA cuando la imagen actual es tanto una imagen CRA como

una imagen RAP. En algunos ejemplos, esta determinación por el descodificador 30 podría realizarse en función de que la imagen tiene un tipo de unidad NAL BLA.

5 **[0116]** En otro ejemplo, cuando se descodifica una imagen BLA, el descodificador 30 puede recibir un indicador o elemento sintáctico en el flujo de bits codificado, por ejemplo, `no_output_of_prior_pics_flag`, de modo que no se proporciona ninguna de las imágenes anteriores de la DPB. En algunos ejemplos, este indicador puede presentarse al principio de una cabecera de fragmento, de un fragmento de una imagen BLA, antes de la descodificación de entropía de modo que puede descodificarse más fácilmente y la información está disponible más pronto en el proceso de codificación. La unidad de descodificación de entropía 80, que puede realizar descodificación de entropía y de no entropía, puede descodificar el indicador o elemento sintáctico. Colocar el indicador u otro elemento sintáctico al principio de la cabecera de fragmento antes de la codificación de entropía puede permitir que dispositivos menos avanzados, tales como un MANE, tengan acceso a la información sin necesidad de un descodificador de entropía ya que, por ejemplo, no es necesario realizar la descodificación de entropía de `no_output_of_prior_pics_flag`.

15 **[0117]** En un ejemplo, el descodificador de vídeo 30 puede colocar un elemento sintáctico, por ejemplo, `no_output_of_prior_pics_flag`, en el flujo de bits para que lo reciba un descodificador. El elemento sintáctico puede indicar que una memoria intermedia de almacenamiento de imágenes se vacía sin proporcionar ninguna imagen de la memoria intermedia de almacenamiento de imágenes. El elemento sintáctico, una vez establecido, puede hacer que las imágenes, anteriores a una imagen actual en orden de descodificación y que residen en la memoria intermedia de almacenamiento de imágenes en el momento de la descodificación de la imagen actual, se extraigan de la memoria intermedia de almacenamiento de imágenes sin que se proporcionen. En algunos ejemplos, el elemento sintáctico puede ser uno de una pluralidad de elementos sintácticos. De forma adicional, la pluralidad de elementos sintácticos puede incluir uno o más elementos sintácticos de cabecera de fragmento sometidos a codificación de entropía y uno o más elementos sintácticos de cabecera de fragmento sometidos a codificación de no entropía. En un ejemplo, el elemento sintáctico que indica que una memoria intermedia de almacenamiento de imágenes se vacía sin proporcionar ninguna imagen de la memoria intermedia de almacenamiento de imágenes se incluye en la cabecera de fragmento, por ejemplo, como elemento `u(1)`, antes de cualquier elemento sintáctico de cabecera de fragmento sometido a codificación de entropía, por ejemplo, antes de cualquier elemento `ue(v)`. En algunos ejemplos, el elemento sintáctico puede ser un `no_output_of_prior_pics_flag` y el `no_output_of_prior_pics_flag` se puede incluir en la cabecera de fragmento justo después de un `first_slice_in_pic_flag` y antes de cualquier elemento sometido a codificación de entropía. El `first_slice_in_pic_flag` puede ser un indicador que indica si un fragmento es el primer fragmento, en orden de descodificación, de una imagen.

35 **[0118]** En otro ejemplo, el descodificador de vídeo 30 puede procesar tipos de unidad NAL asignados para indicar cuándo las imágenes BLA tienen y no tienen imágenes principales. El descodificador de vídeo 30 puede estar configurado para incluir tipos de unidad NAL asignados para indicar cuándo las imágenes BLA tienen y no tienen imágenes principales. Por ejemplo, la unidad de descodificación de entropía 80, que puede realizar descodificación de entropía y de no entropía, puede procesar los tipos de unidad NAL.

40 **[0119]** En un ejemplo, el descodificador de vídeo 30 (por ejemplo, la unidad de descodificación de entropía 80) puede descodificar imágenes de acuerdo con uno de una pluralidad de diferentes tipos de unidades de capa de abstracción de red (NAL). La pluralidad de tipos de unidades NAL incluye uno o más de (1) un fragmento codificado de una imagen de enlace roto (por ejemplo, denominada BLA o BLC), donde la imagen de enlace roto es una imagen de enlace roto con imágenes principales asociadas (ya sean descodificables o no descodificables) en un flujo de bits, 45 (2) un fragmento codificado de una imagen de enlace roto, donde la imagen de enlace roto es una imagen BLA con imágenes principales descodificables asociadas en el flujo de bits, y (3) un fragmento codificado de una imagen de enlace roto, donde la imagen de enlace roto es una imagen de enlace roto sin imágenes principales asociadas en el flujo de bits. En un ejemplo, la imagen principal comprende una imagen que precede a una imagen de acceso aleatorio (RAP) en un orden de visualización, pero sigue a la imagen de acceso aleatorio en orden de descodificación.

50 **[0120]** La unidad de descodificación de entropía 80 del descodificador de vídeo 30 descodifica por entropía el flujo de bits para generar coeficientes cuantificados, vectores de movimiento y otros elementos sintácticos. La unidad de descodificación de entropía 80 reenvía los vectores de movimiento y otros elementos sintácticos a la unidad de procesamiento de predicción 81. El descodificador de vídeo 30 puede recibir los elementos sintácticos en el nivel de fragmento de vídeo y/o el nivel de bloque de vídeo.

60 **[0121]** Cuando el fragmento de vídeo se codifica como un fragmento intracodificado (I), la unidad de procesamiento de intrapredicción 84 de la unidad de procesamiento de predicción 81 puede generar datos de predicción para un bloque de vídeo del fragmento de vídeo actual basándose en un modo de intrapredicción señalado, y datos de bloques previamente descodificados de la trama o imagen actual. Cuando la trama de vídeo se codifica como un fragmento intercodificado (es decir, B, P o GPB), la unidad de compensación de movimiento 82 de la unidad de procesamiento de predicción 81 genera bloques de referencia de predicción para un bloque de vídeo del fragmento de vídeo actual basándose en los vectores de movimiento y en otros elementos sintácticos recibidos desde la unidad de descodificación de entropía 80. Los bloques de referencia de predicción pueden generarse a partir de una de las 65 imágenes de referencia de una de las listas de imágenes de referencia. El descodificador de vídeo 30 puede construir

las listas de tramas de referencia, la Lista 0 y la Lista 1, usando técnicas de construcción predeterminadas, basándose en las imágenes de referencia almacenadas en la memoria de imágenes 92.

5 **[0122]** La unidad de compensación de movimiento 82 determina información de predicción para un bloque de vídeo del fragmento de vídeo actual analizando sintácticamente los vectores de movimiento y otros elementos sintácticos, y usa la información de predicción para generar los bloques de referencia de predicción para el bloque de vídeo actual que se está descodificando. Por ejemplo, la unidad de compensación de movimiento 82 usa algunos de los elementos sintácticos recibidos para determinar un modo de predicción (por ejemplo, intrapredicción o interpredicción) usado para codificar los bloques de vídeo del fragmento de vídeo, un tipo de fragmento de interpredicción (por ejemplo, un fragmento B, un fragmento P o un fragmento GPB), información de construcción para una o más de las listas de imágenes de referencia para el fragmento, vectores de movimiento para cada bloque de vídeo intercodificado del fragmento, el estado de interpredicción para cada bloque de vídeo intercodificado del fragmento y otra información para descodificar los bloques de vídeo en el fragmento de vídeo actual. Cuando las imágenes de la DPB están marcadas como no usadas para referencia, no hay imágenes de referencia disponibles. En consecuencia, el descodificador de vídeo 30 no podrá descodificar una imagen principal que hace referencia a una imagen de referencia anterior para la interpredicción.

20 **[0123]** La unidad de compensación de movimiento 82 también puede realizar una interpolación basándose en filtros de interpolación. La unidad de compensación de movimiento 82 puede usar filtros de interpolación como los usados por el codificador de vídeo 20 durante la codificación de los bloques de vídeo para calcular valores interpolados para píxeles fraccionarios de los bloques de referencia. En este caso, la unidad de compensación de movimiento 82 puede determinar los filtros de interpolación usados por el codificador de vídeo 20 a partir de los elementos sintácticos recibidos y usar los filtros de interpolación para generar bloques de referencia de predicción.

25 **[0124]** La unidad de cuantificación inversa 86 cuantifica inversamente, es decir, descuantifica, los coeficientes de transformada cuantificados proporcionados en el flujo de bits y descodificados por la unidad de descodificación de entropía 80. El proceso de cuantificación inversa puede incluir el uso de un parámetro de cuantificación calculado por el codificador de vídeo 20 para cada bloque de vídeo del fragmento de vídeo para determinar un grado de cuantificación y, asimismo, un grado de cuantificación inversa que se debería aplicar. La unidad de procesamiento de transformada inversa 88 aplica una transformada inversa, por ejemplo una DCT inversa, una transformada entera inversa o un proceso de transformada inversa conceptualmente similar, a los coeficientes de transformada a fin de generar bloques residuales en el dominio del píxel.

35 **[0125]** Una vez que la unidad de compensación de movimiento 82 ha generado el bloque predictivo para el bloque de vídeo actual basándose en los vectores de movimiento y en otros elementos sintácticos, el descodificador de vídeo 30 forma un bloque de vídeo descodificado sumando los bloques residuales procedentes de la unidad de procesamiento de transformada inversa 88 a los correspondientes bloques de referencia de predicción generados por la unidad de compensación de movimiento 82. El sumador 90 representa el componente o los componentes que realizan esta operación de suma. Si se desea, también pueden usarse filtros de bucle (ya sea en el bucle de codificación o después del bucle de codificación) para suavizar las transiciones de píxeles o mejorar de otro modo la calidad de vídeo. La unidad de filtro 91 está destinada a representar uno o más filtros de bucle tales como un filtro de eliminación de bloques, un filtro de bucle adaptativo (ALF) y un filtro de desplazamiento adaptativo de muestras (SAO). Aunque la unidad de filtro 91 se ilustra en la FIG. 4 como un filtro de bucle, en otras configuraciones, la unidad de filtro 91 puede implementarse como un filtro posbucle. Los bloques de vídeo descodificados en una trama o imagen dada se almacenan a continuación en la memoria de imágenes 92, que almacena imágenes de referencia usadas para la posterior compensación de movimiento. La memoria de imágenes 92 también almacena vídeo descodificado para su presentación posterior en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la FIG. 1.

50 **[0126]** De esta manera, el descodificador de vídeo 30 de la FIG. 4 representa un ejemplo de un descodificador de vídeo configurado para recibir una primera unidad de capa de abstracción de red (NAL) que comprende una parte de datos de vídeo y, en función de un tipo de unidad NAL de la primera unidad NAL, detectar una imagen de enlace roto.

55 **[0127]** La FIG. 5 es un diagrama de bloques que ilustra un conjunto de ejemplo de dispositivos que forman parte de la red 100. En este ejemplo, la red 100 incluye los dispositivos de encaminamiento 104A, 104B (dispositivos de encaminamiento 104) y el dispositivo de transcodificación 106. Los dispositivos de encaminamiento 104 y el dispositivo de transcodificación 106 están concebidos para representar un pequeño número de dispositivos que pueden formar parte de la red 100. Otros dispositivos de red, tales como conmutadores, concentradores, pasarelas, cortafuegos, puentes y otros dispositivos de ese tipo también pueden estar incluidos dentro de la red 100. Además, pueden proporcionarse dispositivos de red adicionales a lo largo de un trayecto de red entre un dispositivo servidor 102 y un dispositivo cliente 108. El dispositivo servidor 102 puede corresponder a un dispositivo de origen 12 (FIG. 1), mientras que el dispositivo cliente 108 puede corresponder a un dispositivo de destino 14 (FIG. 1), en algunos ejemplos.

65 **[0128]** En general, los dispositivos de encaminamiento 104 implementan uno o más protocolos de encaminamiento para intercambiar datos de red a través de la red 100. En algunos ejemplos, los dispositivos de encaminamiento 104 pueden estar configurados para realizar operaciones de apoderado (*proxy*) o de memoria caché. Por lo tanto, en algunos ejemplos, los dispositivos de encaminamiento 104 pueden denominarse dispositivos de apoderado. En

general, los dispositivos de encaminamiento 104 ejecutan protocolos de encaminamiento para descubrir rutas a través de la red 100. Al ejecutar dichos protocolos de encaminamiento, un dispositivo de encaminamiento 104B puede descubrir una ruta de red desde sí mismo hasta el dispositivo servidor 102 por medio del dispositivo de encaminamiento 104A.

5 **[0129]** Dispositivos de red, tales como dichos dispositivos de encaminamiento 104 y dispositivo de transcodificación 106, pueden implementar las técnicas de esta divulgación, aunque un dispositivo cliente 108 también las puede implementar. De esta manera, los dispositivos de encaminamiento 104, el dispositivo de transcodificación 106 y el dispositivo cliente 108 representan ejemplos de dispositivos configurados para realizar las técnicas de esta divulgación. Por otro lado, los dispositivos de la FIG. 1, el codificador ilustrado en la FIG. 3 y el descodificador ilustrado en la FIG. 4 son también dispositivos a modo de ejemplo que pueden estar configurados para realizar las técnicas de esta divulgación.

15 **[0130]** Por ejemplo, el dispositivo servidor 102 puede incluir un codificador para codificar una imagen o unas imágenes que se hallan después de un punto de acceso aleatorio o un punto de adaptación de flujo, tal como un punto de conmutación de capa temporal u otro punto de adaptación de flujo. Por ejemplo, este punto podría ser un punto de conmutación para la adaptación de la velocidad de bits, la velocidad de tramas (es decir, el punto de conmutación de capa temporal) o la resolución espacial. Asimismo, el dispositivo cliente 108 puede descodificar una imagen o imágenes que están después de un punto de acceso aleatorio o punto de adaptación de flujo, tal como un punto de conmutación de capa temporal. De nuevo, este podría ser un punto de conmutación para adaptación de velocidad de bits, velocidad de tramas (es decir, un punto de conmutación de capa temporal) o resolución espacial. Una o más de estas imágenes pueden ser imágenes principales. Las imágenes principales no pueden descodificarse correctamente en el dispositivo cliente 108 cuando se produce un acceso aleatorio desde la imagen BLA.

25 **[0131]** En un ejemplo, para evitar la propagación de errores a partir de imágenes de referencia que pueden no estar disponibles dependiendo de dónde comienza la descodificación, el dispositivo cliente 108 puede no aplicar imágenes almacenadas en la memoria intermedia de imágenes descodificadas (DPB) que preceden a la imagen BLA en orden de descodificación o en orden de salida (lo cual incluye las imágenes principales) como imágenes de referencia.

30 **[0132]** En diversos ejemplos, el dispositivo cliente 108 puede marcar todas las imágenes de referencia de la DPB como no usadas para referencia antes de descodificar la imagen BLA. Por ejemplo, el dispositivo cliente 108 puede marcar imágenes de referencia de la memoria intermedia de imágenes descodificadas (DPB) como no usadas para referencia.

35 **[0133]** En otro ejemplo, el dispositivo servidor 102, el dispositivo cliente 108 o ambos pueden incluir un elemento sintáctico o indicador en una cabecera de fragmento y codificar el indicador dentro de un flujo de bits, por ejemplo, un `no_output_of_prior_pics`, de modo que ninguna de las imágenes anteriores de la DPB procesadas por el descodificador de vídeo 30 para descodificar imágenes principales se proporcionan desde la DPB, por ejemplo, para su presentación en un monitor de visualización. En algunos ejemplos, este indicador puede aparecer al principio de la cabecera de fragmento antes de la descodificación de entropía de modo que pueda descodificarse más fácilmente y la información esté disponible más pronto en el proceso de codificación. En un ejemplo, uno de estos dispositivos de elementos de red puede convertir una CRA en una imagen de enlace roto cuando, por ejemplo, se necesita un empalme, una conmutación de canales o una adaptación de flujo, tal como una conmutación de capa temporal. Tener el indicador accesible sin codificación de entropía permite que los elementos de red sin la capacidad de descodificación de entropía tengan acceso al indicador.

50 **[0134]** La pluralidad de tipos de unidades NAL incluye uno o más de (1) un fragmento codificado de una imagen de enlace roto, donde la imagen de enlace roto es una imagen de enlace roto con imágenes principales asociadas (ya sean descodificables o no descodificables) en un flujo de bits, (2) un fragmento codificado de una imagen de enlace roto, donde la imagen de enlace roto es una imagen de enlace roto con imágenes principales descodificables asociadas en el flujo de bits, y (3) un fragmento codificado de una imagen de enlace roto, donde la imagen de enlace roto es una imagen de enlace roto sin imágenes principales asociadas en el flujo de bits. En un ejemplo, la imagen principal comprende una imagen que precede a una imagen de acceso aleatorio (RAP) en un orden de visualización pero sigue a la imagen de acceso aleatorio en orden de descodificación.

55 **[0135]** En diversos ejemplos, los dispositivos de encaminamiento 104A, 104B y el dispositivo de transcodificación 106 que componen la red 100 también pueden realizar algún procesamiento en una imagen o unas imágenes que se hallan después de un punto de acceso aleatorio o punto de adaptación de flujo, tal como un punto de conmutación de capa temporal. Por ejemplo, este podría ser un punto de conmutación para adaptación de velocidad de bits, velocidad de tramas (es decir, un punto de conmutación de capa temporal) o resolución espacial. Como se analiza anteriormente, una o más de estas imágenes pueden ser imágenes principales que no pueden descodificarse correctamente.

60 **[0136]** En un ejemplo, uno o más de los dispositivos de encaminamiento 104A, 104B, y el dispositivo de transcodificación 106 pueden no usar ninguna imagen que precede a la imagen CRA en orden de descodificación u orden de salida (que incluye las imágenes principales) como imágenes de referencia. En otro ejemplo, uno o más de los dispositivos de encaminamiento 104A, 104B y el dispositivo de transcodificación 106 pueden marcar todas las

imágenes de referencia de la DPB como no usadas para referencia antes de descodificar la imagen BLA. Por ejemplo, uno o más de los dispositivos de encaminamiento 104A, 104B y el dispositivo de transcodificación 106 pueden marcar imágenes de referencia de la memoria intermedia de imágenes descodificadas (DPB) como no usadas para referencia. En otro ejemplo, uno o más de los dispositivos de encaminamiento 104A, 104B, y el dispositivo de transcodificación 106 o un servidor de transmisión continua pueden usar un indicador, por ejemplo, un `no_output_of_prior_pics_flag`, de modo que no se proporcionen ninguna de las imágenes anteriores de la DPB. Adicionalmente, los dispositivos de encaminamiento 104A, 104B y el dispositivo de transcodificación 106 pueden procesar los tipos de unidades NAL asignados para indicar cuándo las imágenes BLA tienen y no tienen imágenes principales.

[0137] La FIG. 6 es un diagrama que ilustra un ejemplo de acuerdo con las técnicas descritas en esta divulgación. La FIG. 6 ilustra ejemplos de casos en los que las imágenes principales son descodificables y no descodificables. La capacidad de descodificación de las imágenes principales puede basarse en la ubicación del bloque predictivo. Adicionalmente, la capacidad de descodificación de las imágenes principales puede basarse en si la imagen CRA actual es una imagen CRA que no es una imagen BLA o si la imagen CRA actual es una imagen CRA que también es una imagen BLA. (Las imágenes BLA son un subconjunto de las imágenes CRA).

[0138] La parte 200 de la FIG. 6 ilustra una serie de imágenes en orden de descodificación. Inicialmente, el descodificador de vídeo 30 (FIG. 1 y 4) o el dispositivo cliente 108 (FIG. 5) por ejemplo, pueden descodificar una imagen RAP en la ubicación 202 como primera imagen de una secuencia de vídeo. El descodificador de vídeo 30 o el dispositivo cliente 108 pueden descodificar a continuación una imagen o parte de una imagen que puede actuar como imagen de referencia en la ubicación 204. Como se ilustra en la FIG. 6, la ubicación 204 es una ubicación posible de la imagen de referencia en orden de descodificación. Si la imagen de referencia está situada en la ubicación 204 y la imagen de la ubicación 206 es una imagen CRA que no es una imagen BLA, entonces la imagen principal de la ubicación 208 será descodificable. Por el contrario, si la imagen de referencia está situada en la ubicación 204 y la imagen de la ubicación 206 es una imagen CRA que también es una imagen de enlace roto, entonces la imagen principal de la ubicación 208 no será descodificable. (las imágenes de enlace roto, denominadas imágenes BLA o BLC, son un subconjunto de las imágenes CRA).

[0139] Si la imagen RAP de la ubicación 202 es la imagen RAP, donde comienza la descodificación, y la imagen CRA actual en la ubicación 206 no es una imagen de enlace roto, entonces, como se describe anteriormente, la imagen principal de la ubicación 208 es descodificable. Por el contrario, si la imagen CRA actual en la ubicación 206 es la RAP, entonces la imagen CRA actual de la ubicación 206 también es una imagen de enlace roto y la imagen principal de la ubicación 208 no es descodificable. Esto se debe a que el bloque predictivo de la ubicación 204 no está disponible para una imagen de enlace roto que es una imagen CRA actual en la ubicación 206. En consecuencia, el bloque predictivo de la ubicación 204 puede (1) marcarse como no usado para referencia y (2) el `no_output_of_prior_pics_flag` puede indicar que las imágenes anteriores, hasta e incluyendo la imagen principal de la ubicación 208, no deberían proporcionarse.

[0140] Como se ilustra en la FIG. 6, la ubicación 210 es otra ubicación posible de la imagen de referencia en orden de descodificación. Si la imagen de referencia está situada en la ubicación 210, entonces la imagen principal de la ubicación 208 será descodificable.

[0141] La parte 212 de la FIG. 6 ilustra una serie de imágenes en orden de salida. Inicialmente, el descodificador de vídeo 30 (FIG. 1 y 4) o el dispositivo cliente 108 (FIG. 5) es una imagen RAP 202. El descodificador de vídeo 30 o el dispositivo cliente 108 pueden descodificar, a continuación, una imagen o parte de una imagen que puede actuar como bloque predictivo en la ubicación 204. Como se ilustra en la FIG. 6, la ubicación 204 es una ubicación posible del bloque predictivo en orden de salida.

[0142] En el orden de salida, la imagen principal 208 puede proporcionarse antes que la imagen CRA actual de la ubicación 206, como se ilustra en la FIG. 6. Como se ilustra en la FIG. 6, la ubicación 210 es otra ubicación posible del bloque predictivo en orden de salida.

[0143] La FIG. 7 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar imágenes RAP de acuerdo con uno o más ejemplos descritos en esta divulgación. Como se ilustra en la FIG. 7, en algunos ejemplos, un codificador de vídeo, por ejemplo, el codificador de vídeo 20 o el descodificador de vídeo 30, puede codificar una imagen BLA para un punto de acceso aleatorio o punto de adaptación de flujo, tal como un punto de conmutación de capa temporal en una secuencia de vídeo. Por ejemplo, este podría ser un punto de conmutación para la adaptación de la velocidad de bits, la velocidad de tramas o la resolución espacial. La imagen BLA puede incluir una o más imágenes principales. Las imágenes principales no pueden descodificarse correctamente (por ejemplo, mediante el descodificador 30, un MANE u otro dispositivo de descodificación) cuando se produce un acceso aleatorio desde la imagen BLA.

[0144] En un ejemplo, para evitar la propagación de errores a partir de imágenes de referencia que pueden no estar disponibles dependiendo de dónde comienza la descodificación, el codificador de vídeo puede no usar ninguna imagen que preceda a la imagen BLA en orden de descodificación o en orden de salida (lo cual incluye las imágenes principales) como imagen de referencia.

[0145] Por ejemplo, el descodificador de vídeo 30 puede recibir y descodificar un elemento sintáctico que indica que una memoria intermedia de almacenamiento de imágenes se vacía sin proporcionar ninguna imagen de la memoria intermedia de almacenamiento de imágenes (700). El codificador 20 o un elemento de red intermedio puede incluir el elemento sintáctico en el flujo de bits de vídeo codificado. El descodificador de vídeo 30 puede determinar si el elemento sintáctico indica que no debe producirse una salida de imágenes anteriores; por ejemplo, el descodificador de vídeo 30 puede verificar el elemento sintáctico (702) para determinar si está establecido, es decir, si es igual a 1. Cuando el descodificador de vídeo 30 recibe un elemento sintáctico que está establecido, el descodificador de vídeo 30 puede hacer que las imágenes anteriores a una imagen actual en orden de descodificación y que residen en la memoria intermedia de almacenamiento de imágenes en el momento de la descodificación de la imagen actual se extraigan de la memoria intermedia de almacenamiento de imágenes sin que se proporcionen (704).

[0146] En algunos ejemplos, el elemento sintáctico puede ser uno de una pluralidad de elementos sintácticos. Adicionalmente, la pluralidad de elementos sintácticos puede incluir elementos sintácticos de cabecera de fragmento sometidos a codificación de entropía y elementos sintácticos de cabecera de fragmento no sometidos a codificación de entropía. En un ejemplo, el elemento sintáctico que puede indicar que una memoria intermedia de almacenamiento de imágenes se vacía sin proporcionar ninguna imagen de la memoria intermedia de imágenes descodificadas, está incluido en la cabecera de fragmento antes de cualquier elemento sintáctico de cabecera de fragmento sometido a codificación de entropía. En otro ejemplo, el elemento sintáctico que puede indicar que los datos de la memoria intermedia de almacenamiento de imágenes deben ignorarse y/o sobrescribirse sin proporcionar ninguna imagen de la memoria intermedia de almacenamiento de imágenes está incluido en la cabecera de fragmento antes de cualquier elemento sintáctico de cabecera de fragmento sometido a codificación de entropía. En algunos ejemplos, el elemento sintáctico puede ser un `no_output_of_prior_pics_flag` y el `no_output_of_prior_pics_flag` puede estar incluido en la cabecera de fragmento justo después de un `first_slice_in_pic_flag`. El `first_slice_in_pic_flag` puede ser un indicador que indica si un fragmento es el primer fragmento, en orden de descodificación, de una imagen.

[0147] La FIG. 8 es un diagrama de flujo que ilustra un procedimiento de ejemplo de acuerdo con uno o más ejemplos descritos en esta divulgación. En el ejemplo ilustrado, un codificador de vídeo puede determinar que la imagen actual es una imagen CRA (800). El codificador de vídeo también puede determinar que la imagen actual es una imagen RAP (802). El codificador de vídeo puede determinar que la imagen actual es una imagen BLA cuando la imagen actual es tanto una imagen CRA como una imagen RAP (804). En algunos ejemplos, un codificador de vídeo puede usar un tipo de unidad NAL para determinar si la imagen actual es una imagen BLA. El codificador de vídeo puede marcar imágenes de referencia como no usadas para referencia (806).

[0148] En diversos ejemplos, el codificador de vídeo puede marcar todas las imágenes de referencia de la DPB como no usadas para referencia antes de descodificar la imagen BLA. Por ejemplo, el descodificador de vídeo 30 puede marcar imágenes de referencia de la memoria de imágenes 92, algunas veces denominada memoria intermedia de imágenes descodificadas (DPB), como no usadas para referencia. En consecuencia, las imágenes no se usarán para la intercodificación, lo que evitará posibles errores, y en algunos ejemplos permitirán abordar problemas con la adaptación espacial. Adicionalmente, en general, estas imágenes no se proporcionarán, por ejemplo, a un monitor o pantalla donde puedan visualizarse.

[0149] El descodificador de vídeo 30 puede determinar que una imagen actual es una imagen de enlace roto y marcar una imagen de referencia de una memoria intermedia de almacenamiento de imágenes como no usada para referencia antes de descodificar la imagen BLA. En un ejemplo, el marcar la imagen de referencia en la memoria intermedia de almacenamiento de imágenes puede tener lugar cuando la imagen de enlace roto comprende una imagen de enlace roto no descodificable, lo que se puede determinar en algunos ejemplos en función del tipo de unidad NAL. En algunos ejemplos, uno o más de (1) marcar imágenes en una DPB como no usadas para referencia, (2) usar un elemento sintáctico como un `no_output_of_prior_pics_flag` y (3) usar tipos de unidades NAL que indican tipos de imágenes de enlace roto se pueden usar de manera independiente o en cualquier combinación.

[0150] La FIG. 9 es un diagrama de flujo que ilustra un procedimiento de ejemplo de acuerdo con uno o más ejemplos descritos en esta divulgación. Las técnicas de la FIG. 9 se pueden realizar por un dispositivo de procesamiento de vídeo, tal como el descodificador de vídeo 30, la entidad de red 29, el dispositivo de encaminamiento 104, el dispositivo de transcodificación 106 o algún otro dispositivo de procesamiento de vídeo. Un dispositivo de procesamiento de vídeo puede procesar unidades NAL, donde tipos de unidades NAL para las unidades NAL se asignan para indicar cuándo las imágenes BLA tienen y no tienen imágenes principales. En algunos casos, el dispositivo de procesamiento de vídeo puede, por ejemplo, detectar que una imagen es una imagen CRA y determinar si la imagen CRA es una imagen BLA sin imágenes principales, una imagen BLA con imágenes principales descodificables o una imagen BLA con imágenes principales no descodificables. Según la determinación, el dispositivo de procesamiento de vídeo puede establecer un tipo de unidad NAL para los datos de vídeo para indicar que la unidad NAL incluye una imagen BLA sin imágenes principales, una imagen BLA con imágenes principales descodificables o una imagen BLA con imágenes principales no descodificables.

[0151] En un ejemplo, el dispositivo de procesamiento de vídeo puede codificar imágenes de acuerdo con uno de una pluralidad de tipos diferentes de unidades de capa de abstracción de red (NAL) que incluyen uno o más de lo

siguiente. El dispositivo de procesamiento de vídeo puede determinar que una imagen BLA no tiene imágenes principales asociadas (900) y establecer una unidad NAL como un tipo de unidad NAL que indica que la unidad NAL comprende la imagen BLA sin imágenes principales asociadas (902). El dispositivo de procesamiento de vídeo puede, por ejemplo, codificar un fragmento codificado de la imagen BLA o toda la imagen BLA usando un tipo de unidad NAL que indica que la imagen BLA es una imagen BLA con ninguna imagen principal asociada en un flujo de bits. El dispositivo de procesamiento de vídeo puede determinar que una imagen BLA tiene imágenes principales descodificables asociadas (904) y establecer una unidad NAL como un tipo de unidad NAL que indica que la unidad NAL comprende la imagen BLA con imágenes principales descodificables asociadas (906). El dispositivo de procesamiento de vídeo puede, por ejemplo, codificar un fragmento codificado o la imagen BLA completa usando un tipo de unidad NAL que indica que la imagen BLA es una imagen BLA con imágenes principales descodificables asociadas en el flujo de bits. El dispositivo de procesamiento de vídeo puede determinar que una imagen BLA tiene imágenes principales asociadas (908) y establecer una unidad NAL como un tipo de unidad NAL que indica que la unidad NAL comprende la imagen BLA con imágenes principales asociadas (910). El dispositivo de procesamiento de vídeo puede, por ejemplo, codificar un fragmento codificado de la imagen BLA o toda la imagen BLA usando un tipo de unidad NAL que indica que la imagen BLA es una imagen BLA con imágenes principales asociadas en un flujo de bits. En un ejemplo, si el descodificador de vídeo 30 detecta un tipo de unidad NAL BLA, entonces el descodificador de vídeo 30 puede, por ejemplo, marcar imágenes de la DPB como no usadas para referencia.

[0152] En algunos ejemplos, uno o más de (1) marcar imágenes de una DPB como no usadas para referencia, (2) usar un elemento sintáctico tal como un `no_output_of_prior_pics_flag` y (3) usar tipos de unidad NAL que indican imágenes BLA, se pueden usar de manera independiente o en cualquier combinación. Por ejemplo, en algunos casos se pueden usar los tres puntos anteriores. En otros ejemplos, los tipos de unidad NAL de ejemplo podrían usarse junto con el marcado de imágenes de la DPB como no usadas para referencia. En otro ejemplo, podría usarse el elemento sintáctico que indica que no se proporcionan imágenes anteriores y el marcado de las imágenes como no usadas para referencia. En otro ejemplo, podrían usarse los elementos sintácticos que indican que no se proporcionan imágenes anteriores y los tipos de unidades NAL.

[0153] Como se usa en el presente documento, BLC se refiere al acceso aleatorio limpio de enlace roto, BLCL se refiere a una imagen BLC con imágenes principales asociadas en el flujo de bits y BLCNL se refiere a una imagen BLC sin imágenes principales asociadas en el flujo de bits. Como se analiza en el presente documento, las imágenes BLC son, en general, las mismas que las imágenes BLA. CRA se refiere a un acceso aleatorio limpio, CRAL se refiere a una imagen CRA con imágenes principales asociadas en el flujo de bits, y CRANL se refiere a una imagen CRA sin imágenes principales asociadas en el flujo de bits. IDR se refiere a una actualización de descodificación instantánea, LPR se refiere a una imagen principal asociada a una imagen de punto de acceso aleatorio, NSP se refiere a "normal, sin nada especial" (*Nothing Special Plain*), RAP se refiere a un punto de acceso aleatorio y RPS se refiere a un conjunto de imágenes de referencia. Como se usa en el presente documento, TLA se refiere a acceso a capa temporal, TLAL se refiere a una imagen TLA que también es una imagen LRP, TLANL se refiere a una imagen TLA que no es una imagen LRP.

[0154] Una unidad de acceso BLA se refiere a una unidad de acceso en la que la imagen codificada es una imagen BLA. Una imagen BLC es una imagen RAP en la que la cabecera de fragmento de los fragmentos codificados incluye sintaxis RPS mientras que el RPS se deriva como vacío sin usar la sintaxis RPS. Una unidad de acceso BLCL es una unidad de acceso en la que la imagen codificada es una imagen BLCL. Las imágenes BLCL son imágenes BLA para las cuales las imágenes LPR asociadas están presentes en el flujo de bits. En algunos ejemplos, una unidad de acceso BLCL puede ser equivalente a una combinación de `BLA_W_DLP` y `BLA_W_LP`. Una unidad de acceso BLCNL es una unidad de acceso en la que la imagen codificada es una imagen BLCNL. En algunos ejemplos, una unidad de acceso BLCNL puede ser equivalente a `BLA_N_LP`. Las imágenes BLCNL son imágenes BLA para las cuales las imágenes LPR asociadas no están presentes en el flujo de bits.

[0155] En un ejemplo, una unidad de acceso CRA es una unidad de acceso en la que la imagen codificada es una imagen CRA. Las imágenes CRA son imágenes RAP en las que la cabecera de fragmento de los fragmentos codificados incluye la sintaxis RPS, y la sintaxis RPS se usa para la obtención del RPS. Una unidad de acceso CRAL es una unidad de acceso en la que la imagen codificada es una imagen CRAL. Las imágenes CRAL son imágenes CRA para las cuales las imágenes LPR asociadas están presentes en el flujo de bits. Una unidad de acceso CRANL es una unidad de acceso en la que la imagen codificada es una imagen CRANL. Las imágenes CRANL son imágenes CRA para las cuales las imágenes LPR asociadas no están presentes en el flujo de bits.

[0156] En un ejemplo, una unidad de acceso IDR es una unidad de acceso en la que la imagen codificada es una imagen IDR. Las imágenes IDR son imágenes RAP en las que la cabecera de fragmento de los fragmentos codificados no incluye la sintaxis RPS, y el RPS se deriva como vacío.

[0157] En un ejemplo, si todas las unidades de acceso precedentes en orden de descodificación no están presentes, siempre que cada conjunto de parámetros al que hace referencia la imagen codificada y todas las imágenes codificadas subsiguientes en orden de descodificación estén presentes antes de su activación, una imagen IDR y todas las imágenes codificadas subsiguientes en orden de descodificación pueden descodificarse correctamente. De forma alternativa, en otro ejemplo, las imágenes IDR pueden ser como se define en HEVC con la adición de lo anterior.

[0158] En otros ejemplos, una imagen IDR puede ser una imagen codificada que solo contiene fragmentos I. Adicionalmente, para una imagen IDR de ejemplo, todas las imágenes codificadas que siguen a la imagen IDR en orden de descodificación no usan interpredicción a partir de ninguna imagen que precede a la imagen IDR en orden de descodificación. En un ejemplo, cualquier imagen que precede a la imagen IDR en orden de descodificación también precede a la imagen IDR en orden de salida.

[0159] Una imagen principal es una imagen codificada que no es una imagen RAP y que sigue a otra imagen particular en orden de descodificación y precede a la imagen particular en orden de salida. Una imagen LPR es una imagen principal que está asociada a una imagen RAP o una imagen principal de una imagen RAP.

[0160] El cómputo de orden de imágenes puede ser una variable asociada a cada imagen codificada y tiene un valor que aumenta al aumentar la posición de la imagen en orden de salida con relación a la imagen RAP anterior en orden de descodificación.

[0161] En un ejemplo, una unidad de acceso RAP es una unidad de acceso en la que la imagen codificada es una imagen RAP. Una imagen RAP puede ser una imagen codificada que solo contiene fragmentos I. Para una imagen RAP, todas las imágenes codificadas que siguen a la imagen RAP tanto en orden de descodificación como en orden de salida no usan interpredicción a partir de ninguna imagen que preceda a la imagen RAP en orden de descodificación u orden de salida. La salida de cualquier imagen que preceda a la imagen RAP en orden de descodificación debe preceder a la salida de la imagen RAP. Si todas las unidades de acceso precedentes en orden de descodificación no están presentes, siempre que cada conjunto de parámetros al que hace referencia la imagen codificada y todas las imágenes codificadas subsiguientes en orden de descodificación estén presentes antes de su activación, una imagen RAP y todas las imágenes codificadas subsiguientes en orden de descodificación y orden de salida se pueden descodificar correctamente.

[0162] De forma alternativa, la imagen RAP se puede definir de conformidad con el análisis anterior y tal como se indica a continuación. Una imagen RAP puede ser una imagen codificada que solo contiene fragmentos I y para la cual todas las imágenes codificadas que siguen a la imagen RAP tanto en orden de descodificación como en orden de salida no usan interpredicción a partir de ninguna imagen que preceda a la imagen RAP en orden de descodificación u orden de salida. Cualquier imagen que preceda a la imagen RAP en orden de descodificación también precede a la imagen RAP en orden de salida.

[0163] Una unidad de acceso TLA es una unidad de acceso en la que la imagen codificada es una imagen TLA. Una imagen TLA es una imagen codificada para la cual la imagen TLA y todas las imágenes codificadas tienen un temporal_id mayor que o igual al temporal_id de la imagen TLA. Una imagen TLA que sigue a la imagen TLA en orden de descodificación no usará interpredicción a partir de ninguna imagen con un temporal_id mayor que o igual al temporal_id de una imagen TLA que preceda a la imagen TLA en orden de descodificación. Una unidad de acceso TLAL es una unidad de acceso en la que la imagen codificada es una imagen TLA.

[0164] En algunos ejemplos, se pueden definir los siguientes tipos de unidades NAL VCL diferenciados. Como primer ejemplo, se puede proporcionar un tipo de unidad NAL para un fragmento codificado de una imagen IDR (por ejemplo, nal_unit_type = 5). Para este tipo de unidad NAL, se aplica el concepto de imagen IDR de HEVC WD6. Una característica única de este tipo de unidad NAL VCL en comparación con otros tipos de unidades NAL VCL es que no hay ninguna sintaxis de conjunto de imágenes de referencia (RPS) incluida en la cabecera de fragmento.

[0165] Algunos ejemplos incluyen un fragmento codificado de una imagen BLCNL (una imagen BLC sin imágenes principales asociadas en el flujo de bits, por ejemplo, nal_unit_type = 2). Comparado con un fragmento codificado de una imagen IDR, un fragmento codificado de una imagen BLCNL incluye sintaxis RPS en la cabecera de fragmento, pero la sintaxis RPS no se usa para la obtención de RPS, sino que todos los subconjuntos RPS se derivan como vacíos.

[0166] Algunos ejemplos incluyen un fragmento codificado de una imagen BLCL (una imagen BLC con imágenes principales asociadas en el flujo de bits, por ejemplo, nal_unit_type = 3). En comparación con una imagen BLCNL, hay imágenes principales asociadas a una imagen BLCL en el flujo de bits.

[0167] Algunos ejemplos incluyen un fragmento codificado de una imagen CRANL (una CRA sin imágenes principales asociadas en el flujo de bits, por ejemplo, nal_unit_type = 15). En comparación con un fragmento codificado de una imagen BLCNL, un fragmento codificado de una imagen CRANL incluye sintaxis RPS en la cabecera de fragmento, y la sintaxis RPS se usa para la obtención de RPS.

[0168] Algunos ejemplos incluyen un fragmento codificado de una imagen CRAL (una imagen CRA con imágenes principales asociadas en el flujo de bits, por ejemplo, nal_unit_type = 4). En comparación con una imagen CRANL, hay imágenes principales asociadas a una imagen CRAL en el flujo de bits.

[0169] Algunos ejemplos incluyen un fragmento codificado de una imagen TLANL (una imagen TLA que no es una imagen LFR, por ejemplo, `nal_unit_type = 16`). Algunos ejemplos incluyen un fragmento codificado de una imagen TLAL (una imagen TLA que también es una imagen LPR, por ejemplo, `nal_unit_type = 17`). Algunos ejemplos incluyen un fragmento codificado de una imagen NSP (una imagen normal, sin nada especial - ninguna de las anteriores, `nal_unit_type = 1`).

[0170] Antes de descodificar los datos de fragmento de cada imagen BLC (BLCL o BLCNL), todas las imágenes de referencia de la memoria intermedia de imágenes descodificadas (DPB) deben marcarse como "no usadas para referencia" por el descodificador 30, como se describe anteriormente. Solo con esto, el descodificador 30 puede aplicar directamente el proceso de descodificación actual, especificado actualmente en HEVC WD 6 para las imágenes principales de una imagen CRA que inicia el flujo de bits, para las imágenes principales de una imagen BLA, independientemente de si la imagen BLA cambia la resolución espacial.

[0171] Sin lo anterior, si una imagen BLA no cambia la resolución espacial, el descodificador 30 puede aplicar directamente el proceso de descodificación actual, especificado actualmente en HEVC WD 6 para imágenes principales de una imagen CRA que inicia el flujo de bits, para imágenes principales de la imagen BLA. Sin embargo, si una imagen BLA cambia la resolución espacial, el proceso de descodificación actual especificado actualmente en HEVC WD 6 para imágenes principales de una imagen CRA que inicia el flujo de bits no se puede aplicar directamente para imágenes principales de la imagen BLA, ya que puede darse la situación en la que la resolución espacial sea diferente para una imagen actual y una imagen de referencia para la imagen actual.

[0172] Una forma de garantizar que todas las imágenes de referencia de la DPB se marquen como "no usadas para referencia" antes de descodificar los datos de fragmento de cada imagen BLA es derivar el RPS de cada imagen BLA como vacío, independientemente de si la señalización RPS de la cabecera de fragmento indica un RPS no vacío. Por ejemplo, incluso si hay un RPS, el descodificador de vídeo 30 puede anularlo y entonces derivar o tratar el RPS como vacío si la imagen es una imagen BLA.

[0173] En realidad, si la señalización RPS de la cabecera de fragmento indica un RPS vacío para una imagen BLA o una imagen CRA (CRAL o CRANL), entonces la imagen debería haberse codificado como una imagen IDR.

[0174] En algunos ejemplos, una imagen RAP puede definirse como una imagen codificada que solo contiene fragmentos I. Para una imagen RAP, todas las imágenes codificadas que siguen a la imagen RAP tanto en orden de descodificación como en orden de salida no usan interpredicción a partir de ninguna imagen que preceda a la imagen RAP en orden de descodificación u orden de salida. Adicionalmente, la salida de cualquier imagen que preceda a la imagen RAP en orden de descodificación puede preceder a la salida de la imagen RAP.

[0175] Una manera de garantizar que la salida de cualquier imagen que precede a la imagen RAP en orden de descodificación precederá a la salida de la imagen RAP es que un codificador de vídeo 20 establezca `no_output_of_prior_pics_flag` igual a 1 en un flujo de bits codificado transmitido a, por ejemplo, un descodificador de vídeo 30, como se analiza anteriormente. En otro ejemplo, un descodificador de vídeo 30 puede inferir que `no_output_of_prior_pics_flag` es igual a 1 (independientemente de su valor) para una imagen BLA. De esta manera se permiten operaciones de empalme en una imagen BLA, donde en el flujo de bits empalmado el valor POC de una imagen anterior a la imagen BLA es mayor que el valor POC de la imagen BLA. En particular, si el valor POC de la imagen BLA se deriva como igual a su LSB POC (suponiendo que MSB POC sea igual a 0), lo anterior es fácil que suceda. Otra forma de garantizar esto es asegurarse de que los tiempos de salida de las imágenes que preceden a la imagen BLA en orden de descodificación sean anteriores a los de la imagen BLA.

[0176] Algunos ejemplos permiten que el codificador de vídeo 20 y/o un empalmador de flujo de bits determinen cuál de las formas descritas en el presente documento se va a usar. En consecuencia, el codificador de vídeo 20, por ejemplo, puede incluir un `no_output_of_prior_pics_flag` en la cabecera de fragmento de imágenes BLA en algunos ejemplos. En otros ejemplos, el codificador de vídeo 20 puede incluir tipos de unidad NAL que indican si las imágenes almacenadas en una memoria intermedia en el descodificador de vídeo 30 se pueden usar para referencia. En consecuencia, el descodificador de vídeo 30 puede marcar imágenes como no usadas para referencia, por ejemplo, cuando el uso de dichas imágenes en la descodificación dará como resultado la descodificación incorrecta de una imagen.

[0177] Para permitir la reescritura simple de un CRA en una imagen BLA por un elemento de red, además de poner el indicador `no_output_of_prior_pics_flag` en la cabecera de fragmento de una imagen CRA, el `no_output_of_prior_pics_flag` también puede incluirse lo más al principio posible en la cabecera de fragmento. Esto puede ser antes de cualquier parámetro de cabecera de fragmento sometido a codificación de entropía, por ejemplo, justo después de `first_slice_in_pic_flag` en algunos ejemplos, como se analiza anteriormente.

[0178] En un ejemplo, dos imágenes BLA consecutivas tienen el mismo LSB POC, y solo es posible diferenciarlas mediante `random_access_pic_id` (también denominado `rap_pic_id`). Por lo tanto, podría ser preferible usar una codificación de longitud fija para `rap_pic_id`, y colocar `rap_pic_id` relativamente al principio en la cabecera de fragmento, preferentemente no después de cualquier parámetro de cabecera de fragmento sometido a codificación

de entropía, por ejemplo, justo después de `first_slice_in_pic_flag` y `no_output_of_prior_pics_flag`, por ejemplo, para imágenes CRA e imágenes BLA. Otros elementos sintácticos de cabecera de fragmento que pueden usarse para la detección de límites de imagen, por ejemplo, `pic_parameter_set_id` y LSB POC (es decir, `pic_order_cnt_lsb`) pueden ser similares.

5 **[0179]** Todos los conjuntos de parámetros de un tipo particular (por ejemplo, SPS) o todos los tipos pueden estar presentes al principio del flujo de bits, es decir, incluidos en la primera unidad de acceso del flujo de bits. Si es así, es conveniente buscar todos los conjuntos de parámetros de un tipo particular y enviarlos fuera de banda. Por ejemplo, un codificador puede incluir un parámetro de protocolo de descripción de sesión (SDP) que se usa durante la negociación de la sesión. Por lo tanto, puede ser beneficioso para un codificador incluir una indicación en el flujo de bits de que todos los conjuntos de parámetros de un tipo particular o todos los tipos están presentes al principio del flujo de bits. La indicación puede incluirse en un mensaje SEI, un delimitador de unidad de acceso o un conjunto de parámetros. Por ejemplo, un tipo distinto de unidad NAL puede usarse para el SPS, el conjunto de parámetros de imagen (PPS) o el conjunto de parámetros de adaptación (APS) para indicar que todos los SPS (o PPS o APS) están presentes al principio del flujo de bits.

10 **[0180]** El elemento sintáctico `slice_type` puede estar presente condicionalmente en la cabecera de fragmento, en función de si el tipo de unidad NAL indica que la imagen que contiene el fragmento es una imagen IDR, una imagen CRA o una imagen BLA. Por ejemplo, si el tipo de unidad NAL indica que la imagen que contiene el fragmento es una imagen IDR, una imagen CRA o una imagen BLA, `slice_type` no está presente en la cabecera de fragmento. De lo contrario, el codificador inserta `slice_type` en la cabecera de fragmento. Cuando no está presente, el valor de `slice_type` indica que el fragmento es un fragmento I.

20 **[0181]** La FIG. 10 es un diagrama de flujo que ilustra un procedimiento de ejemplo para descodificar un fragmento de acuerdo con uno o más ejemplos descritos en esta divulgación. En un ejemplo, el descodificador de vídeo 30 puede descodificar datos de vídeo, incluyendo la recepción de un fragmento de una imagen actual que se va a descodificar para una secuencia de datos de vídeo (1000). El descodificador de vídeo 30 puede recibir, en una cabecera de fragmento del fragmento, al menos un elemento sintáctico sometido a codificación de entropía y al menos un elemento sintáctico no sometido a codificación de entropía. El elemento sintáctico no sometido a codificación de entropía puede hallarse antes del elemento sintáctico sometido a codificación de entropía en la cabecera de fragmento. Adicionalmente, el elemento sintáctico no sometido a codificación de entropía puede indicar si las imágenes anteriores a la imagen actual en orden de descodificación deben extraerse de una memoria intermedia de almacenamiento de imágenes sin proporcionarse (1002). En un ejemplo, el elemento sintáctico puede ser un `no_output_of_prior_pics_flag`. El `no_output_of_prior_pics_flag` puede establecerse en "1", por ejemplo, para indicar cuándo las imágenes anteriores a la imagen actual en orden de descodificación deben extraerse de una memoria intermedia de almacenamiento de imágenes sin proporcionarse. El descodificador de vídeo 30 puede descodificar el fragmento basándose en el elemento sintáctico no sometido a codificación de entropía (1004).

25 **[0182]** La FIG. 11 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar un fragmento de acuerdo con uno o más ejemplos descritos en esta divulgación. El codificador 20 puede codificar datos de vídeo. El codificador de vídeo 20 puede codificar un fragmento de una imagen actual para una secuencia de datos de vídeo (1100).

30 **[0183]** El codificador de vídeo 20 codifica, en una cabecera de fragmento del fragmento, al menos un elemento sintáctico sometido a codificación de entropía y al menos un elemento sintáctico no sometido a codificación de entropía, donde el elemento sintáctico no sometido a codificación de entropía se halla antes del elemento sintáctico sometido a codificación de entropía en la cabecera de fragmento e indica si las imágenes anteriores a la imagen actual en orden de descodificación deben extraerse de una memoria intermedia de almacenamiento de imágenes sin proporcionarse (1102). En un ejemplo, el elemento sintáctico puede ser un `no_output_of_prior_pics_flag`. El `no_output_of_prior_pics_flag` puede establecerse en "1", por ejemplo, para indicar cuándo las imágenes anteriores a la imagen actual en orden de descodificación deben extraerse de una memoria intermedia de almacenamiento de imágenes sin proporcionarse.

35 **[0184]** La FIG. 12 es un diagrama de flujo que ilustra una técnica de ejemplo para procesar datos de vídeo, como se describe esta divulgación. Las técnicas de la FIG. 12 se pueden realizar por un dispositivo de procesamiento de vídeo, tal como el descodificador de vídeo 30, la entidad de red 29, el dispositivo de encaminamiento 104, el dispositivo de transcodificación 106 o algún otro dispositivo de procesamiento de vídeo. El dispositivo de procesamiento de vídeo puede recibir en los datos de vídeo una unidad NAL (1210) y, basándose en un tipo de unidad NAL de la primera unidad NAL, detectar una imagen de enlace roto (1220). El tipo de unidad NAL puede ser un tipo de unidad NAL que indica que la imagen de enlace roto comprende una imagen de enlace roto con imágenes principales, o el tipo de unidad NAL puede ser un tipo de unidad NAL diferente que indica que la imagen de enlace roto comprende una imagen de enlace roto sin imágenes principales. La entidad de procesamiento de vídeo puede detectar la imagen de enlace roto sin realizar la descodificación de entropía de elementos sintácticos de datos de flujo de bits de vídeo en la unidad NAL.

65

[0185] Según el tipo de unidad NAL detectada, el dispositivo de procesamiento de vídeo puede realizar una o más de una variedad de operaciones. Un MANE, por ejemplo, puede tomar una o más decisiones de encaminamiento para la unidad NAL basándose en el tipo de unidad NAL. Como ejemplo, si el dispositivo de procesamiento de vídeo es un MANE, el dispositivo de procesamiento de vídeo puede, en respuesta a una unidad NAL que indica una imagen de enlace roto con imágenes principales, descartar las imágenes principales y transmitir la unidad NAL sin las imágenes principales. En los casos en que la unidad NAL indica una imagen de enlace roto sin imágenes principales, el dispositivo de procesamiento de vídeo puede transmitir la unidad NAL sin tener que descartar ninguna imagen principal. El MANE puede, por ejemplo, transmitir la unidad NAL a un descodificador de vídeo, potencialmente a través de uno o más dispositivos de encaminamiento.

[0186] Si, por ejemplo, el dispositivo de procesamiento de vídeo es un descodificador de vídeo en lugar de un MANE, entonces el descodificador de vídeo puede realizar una o más de una variedad de operaciones diferentes. Por ejemplo, en respuesta a la detección de una imagen de enlace roto, el descodificador de vídeo puede activar uno o más conjuntos de parámetros nuevos para datos de vídeo.

[0187] En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o cualquier combinación de los mismos. Si se implementan en software, las funciones se pueden almacenar en o transmitir por un medio legible por ordenador como una o más instrucciones o código, y ejecutarse mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que corresponden a un medio tangible tal como medios de almacenamiento de datos, o medios de comunicación que incluyan cualquier medio que facilite la transferencia de un programa informático desde un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder, en general, a (1) medios de almacenamiento tangibles y legibles por ordenador que sean no transitorios o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que se pueda acceder mediante uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

[0188] En otros ejemplos adicionales, esta divulgación contempla un medio legible por ordenador que comprende una estructura de datos almacenada en el mismo, donde la estructura de datos incluye un flujo de bits codificado de conformidad con esta divulgación. En particular, las estructuras de datos pueden incluir los diseños de unidades NAL descritas en el presente documento.

[0189] A modo de ejemplo, y no de limitación, tales medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda usarse para almacenar código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. Además, cualquier conexión recibe adecuadamente la denominación de medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde un sitio web, un servidor u otro origen remoto usando un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, la DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas se incluyen en la definición de medio. Sin embargo, debería entenderse que los medios de almacenamiento legibles por ordenador y los medios de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales ni otros medios transitorios, sino que, en cambio, se orientan a medios de almacenamiento tangibles no transitorios. El término disco, como se usa en el presente documento, incluye el disco compacto (CD), el disco láser, el disco óptico, el disco versátil digital (DVD), el disco flexible y el disco Blu-ray, donde algunos discos normalmente reproducen datos de manera magnética, mientras que otros discos reproducen datos ópticamente con láseres. Las combinaciones de lo anterior se deberían incluir también dentro del alcance de los medios legibles por ordenador.

[0190] Las instrucciones pueden ser ejecutadas por uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados específicos de la aplicación (ASIC), formaciones lógicas programables in situ (FPGA) u otros circuitos lógicos equivalentes, integrados o discretos. En consecuencia, el término "procesador", como se usa en el presente documento, se puede referir a cualquiera de las estructuras anteriores o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento se puede proporcionar dentro de módulos de hardware y/o software dedicados, configurados para la codificación y la descodificación, o incorporarse en un códec combinado. Además, las técnicas se pueden implementar totalmente en uno o más circuitos o elementos lógicos.

[0191] Las técnicas de esta divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluidos un equipo manual inalámbrico, un circuito integrado (IC) o un conjunto de IC (por ejemplo, un conjunto de chips). Diversos componentes, módulos o unidades se describen en esta divulgación para enfatizar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no requieren necesariamente su realización mediante diferentes unidades de hardware. En cambio, como se describe anteriormente, diversas unidades

se pueden combinar en una unidad de hardware de códec o proporcionarse mediante un grupo de unidades de hardware interoperativas, incluidos uno o más procesadores, como se describe anteriormente, junto con software y/o firmware adecuados.

- 5 **[0192]** Se han descrito diversos ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un procedimiento de procesamiento de datos de vídeo, comprendiendo el procedimiento:

5 recibir una primera unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo para una imagen, donde un tipo de unidad NAL de la primera unidad NAL indica que la imagen es una imagen de enlace roto; y

10 en función del tipo de unidad NAL de la primera unidad NAL, detectar que la imagen es una imagen de enlace roto, donde:

15 una imagen de enlace roto es una imagen de acceso aleatorio que no está al comienzo de un flujo de bits y puede incluir imágenes principales, y dichas imágenes principales pueden hacer referencia a imágenes de referencia no disponibles que pueden no estar descodificadas correctamente; y

el tipo de unidad NAL es:

20 un primer tipo de unidad NAL distinto que indica que la imagen es una imagen de enlace roto con imágenes principales; o

un segundo tipo de unidad NAL distinto que indica que la imagen es una imagen de enlace roto sin imágenes principales.

25 2. El procedimiento según la reivindicación 1, en el que el tipo de unidad NAL es el primer tipo de unidad NAL distinto, y el procedimiento comprende además:

descartar las imágenes principales; y

30 transmitir la primera unidad NAL a un dispositivo de procesamiento de vídeo.

3. El procedimiento según la reivindicación 1, en el que el tipo de unidad NAL es el segundo tipo de unidad NAL distinto, y el procedimiento comprende además:

35 transmitir la primera unidad NAL a un dispositivo de procesamiento de vídeo; o

40 recibir una segunda unidad NAL, determinar un tipo de unidad NAL para la segunda unidad NAL y, en función del tipo de unidad NAL para la segunda unidad NAL, detectar una imagen de enlace roto con imágenes principales, donde el tipo de unidad NAL para la segunda unidad NAL es diferente al primer tipo de unidad NAL.

4. El procedimiento según la reivindicación 1, donde el procedimiento es realizado por una entidad de procesamiento de vídeo, y donde la entidad de procesamiento de vídeo detecta la imagen de enlace roto sin realizar la descodificación de entropía de elementos sintácticos de datos de flujo de bits de vídeo en la unidad NAL.

45 5. El procedimiento según la reivindicación 1, que comprende además:

en respuesta a la detección de una imagen de enlace roto, activar uno o más conjuntos de parámetros nuevos para codificar los datos de vídeo.

50 6. El procedimiento según la reivindicación 5, en el que el uno o más conjuntos de parámetros nuevos comprenden uno o más de un conjunto de parámetros de adaptación (APS), un conjunto de parámetros de secuencia (SPS) y un conjunto de parámetros de imagen (PPS).

55 7. El procedimiento según la reivindicación 5, en el que una identificación (ID) para uno de los uno o más conjuntos de parámetros es la misma que la de un conjunto de parámetros activo para una imagen previamente descodificada.

60 8. El procedimiento según la reivindicación 1, donde el procedimiento se realiza mediante un descodificador de vídeo o mediante un elemento de red sensible a los medios (MANE).

9. Un procedimiento de procesamiento de datos de vídeo, comprendiendo el procedimiento:

65 recibir una unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo para una imagen;

determinar si la imagen es una imagen de enlace roto, donde:

una imagen de enlace roto es una imagen de acceso aleatorio que no está al comienzo de un flujo de bits y puede incluir imágenes principales, y dichas imágenes principales pueden hacer referencia a imágenes de referencia no disponibles que pueden no estar descodificadas correctamente; y

5 en respuesta a la determinación de que la imagen es una imagen de enlace roto con imágenes principales, establecer el tipo de unidad NAL para la unidad NAL como un primer tipo de unidad NAL distinto que indica una imagen de enlace roto con imágenes principales;

10 en respuesta a la determinación de que la imagen es una imagen de enlace roto sin imágenes principales, establecer el tipo de unidad NAL para la unidad NAL como un segundo tipo de unidad NAL distinto que indica una imagen de enlace roto sin imágenes principales.

10. Un dispositivo para procesar datos de vídeo, comprendiendo el dispositivo:

15 medios para recibir una primera unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo para una imagen, donde un tipo de unidad NAL de la primera unidad NAL indica que la imagen es una imagen de enlace roto; y

20 medios para detectar que la imagen es una imagen de enlace roto en función del tipo de unidad NAL de la primera unidad NAL, donde:

25 una imagen de enlace roto es una imagen de acceso aleatorio que no está al comienzo de un flujo de bits y puede incluir imágenes principales, y dichas imágenes principales pueden hacer referencia a imágenes de referencia no disponibles que pueden no estar descodificadas correctamente; y

en el que el tipo de unidad NAL es:

30 un primer tipo de unidad NAL distinto que indica que la imagen es una imagen de enlace roto con imágenes principales; o

un segundo tipo de unidad NAL distinto que indica que la imagen es una imagen de enlace roto sin imágenes principales.

35 11. Un dispositivo para procesar datos de vídeo, comprendiendo el dispositivo:

medios para recibir una unidad de capa de abstracción de red (NAL) que comprende una parte de los datos de vídeo para una imagen;

40 medios para determinar si la imagen es una imagen de enlace roto; y

medios para establecer un tipo de unidad NAL para la unidad NAL, donde:

45 una imagen de enlace roto es una imagen de acceso aleatorio que no está al comienzo de un flujo de bits y puede incluir imágenes principales, y dichas imágenes principales pueden hacer referencia a imágenes de referencia no disponibles que pueden no estar descodificadas correctamente; y

50 en respuesta a la determinación de que la imagen es una imagen de enlace roto con imágenes principales, los medios para establecer el tipo de unidad NAL para la unidad NAL establece el tipo de unidad NAL en un primer tipo de unidad NAL distinto que indica una imagen de enlace roto con imágenes principales;

55 en respuesta a la determinación de que la imagen es una imagen de enlace roto sin imágenes principales, los medios para establecer el tipo de unidad NAL para la unidad NAL establecen el tipo de unidad NAL como un segundo tipo de unidad NAL distinto que indica una imagen de enlace roto sin imágenes principales.

12. Un medio de almacenamiento legible por ordenador que almacena instrucciones que, cuando son ejecutadas por uno o más procesadores, hacen que el uno o más procesadores lleven a cabo el procedimiento de acuerdo con una cualquiera de las reivindicaciones 1 a 9.

60

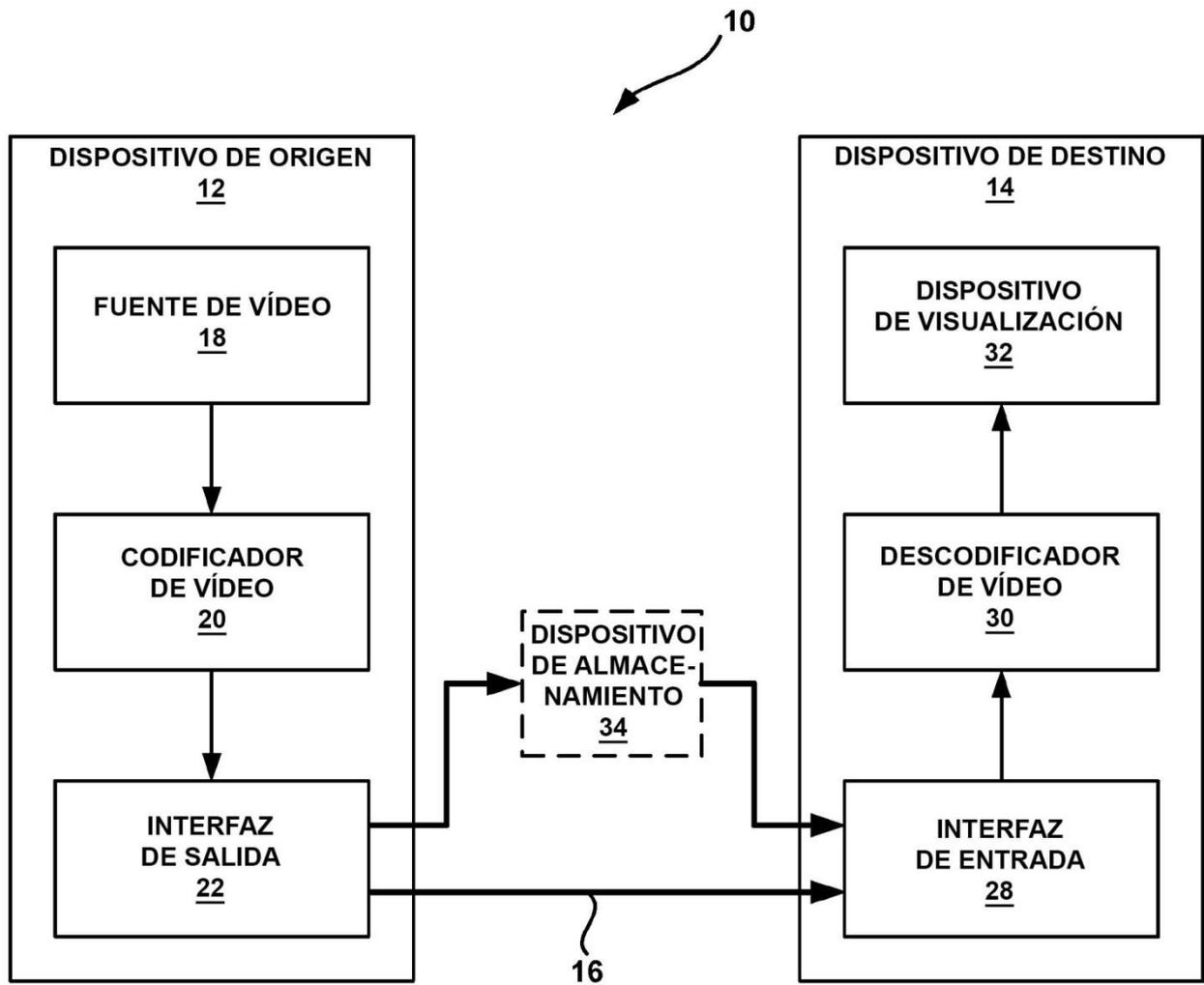


FIG. 1

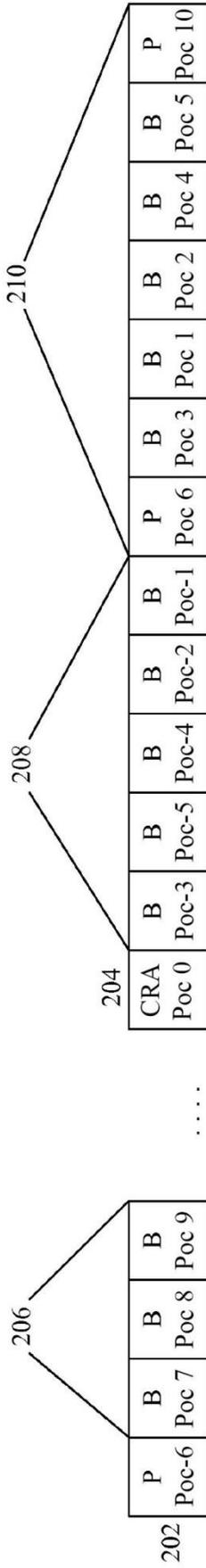


FIG. 2A

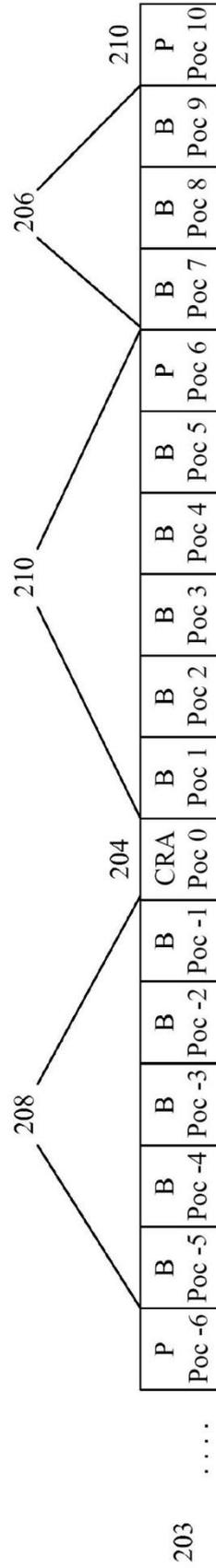


FIG. 2B

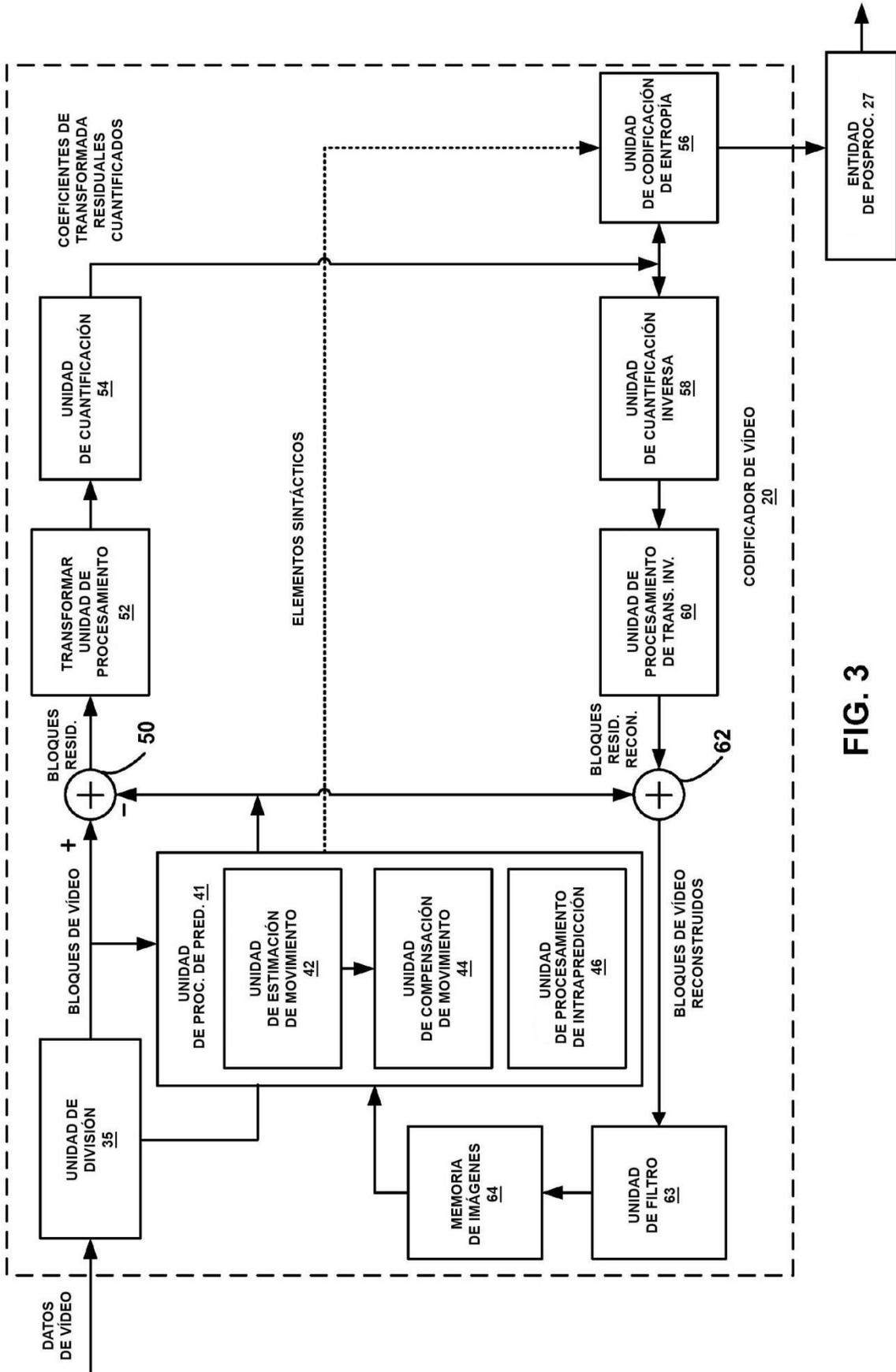


FIG. 3

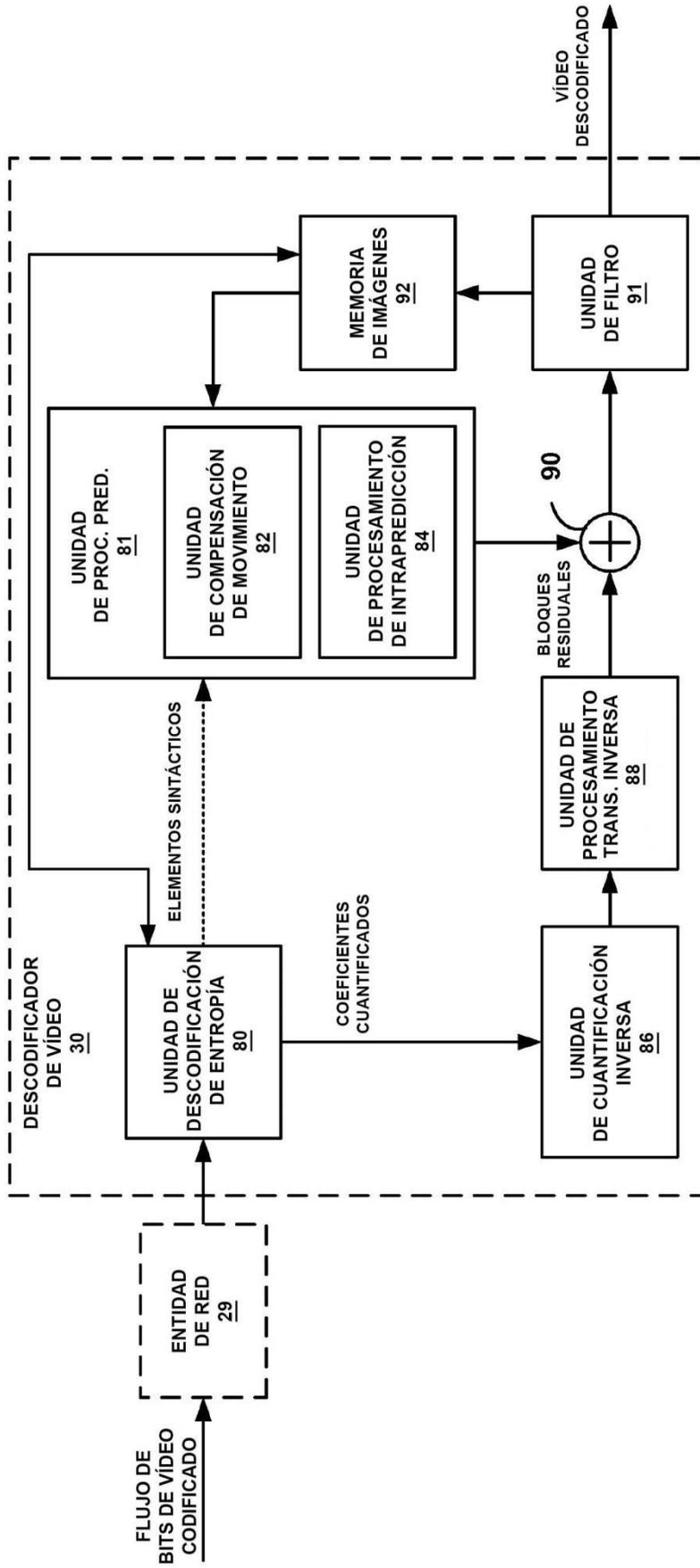


FIG. 4

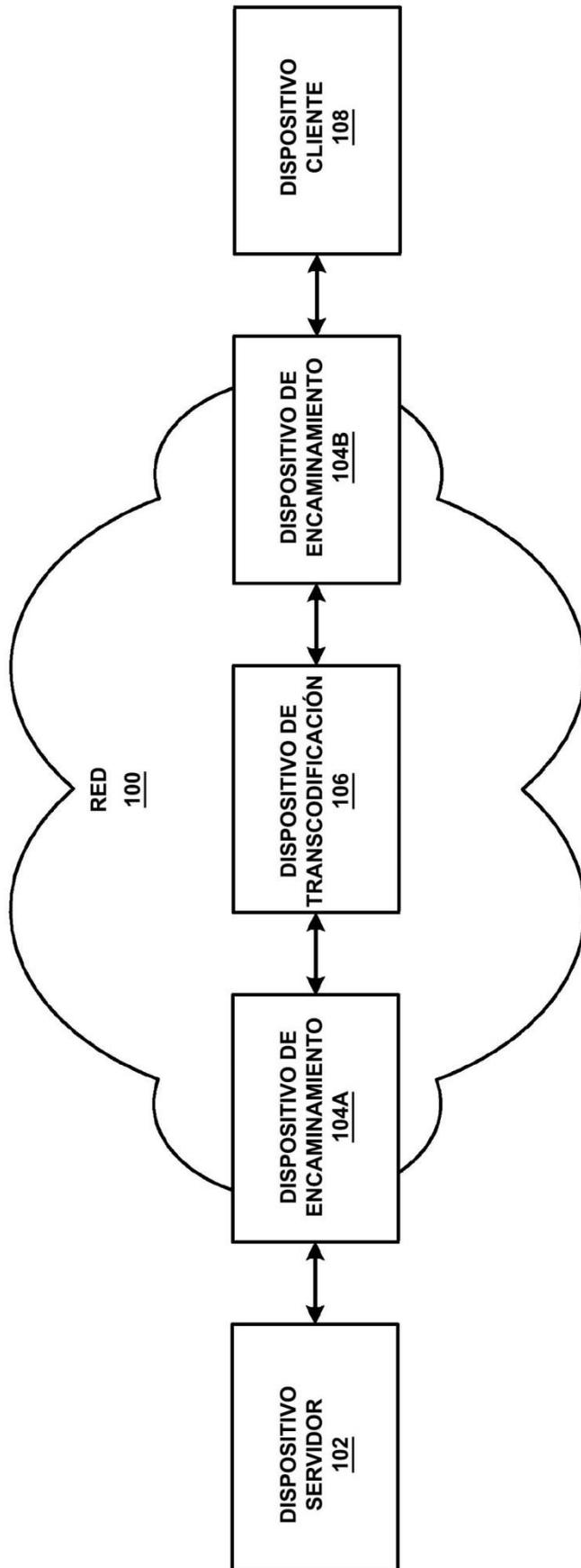


FIG. 5

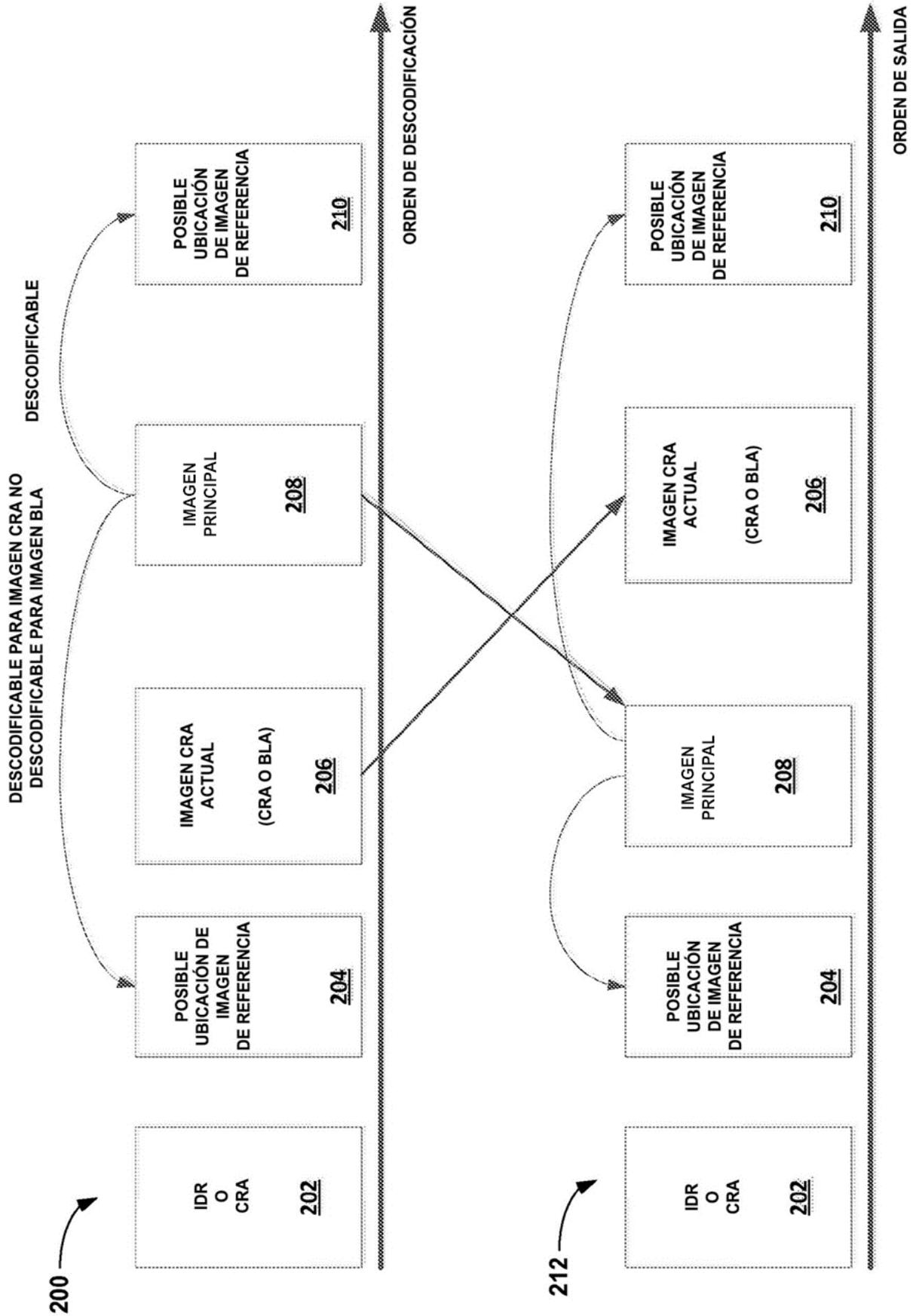


FIG. 6

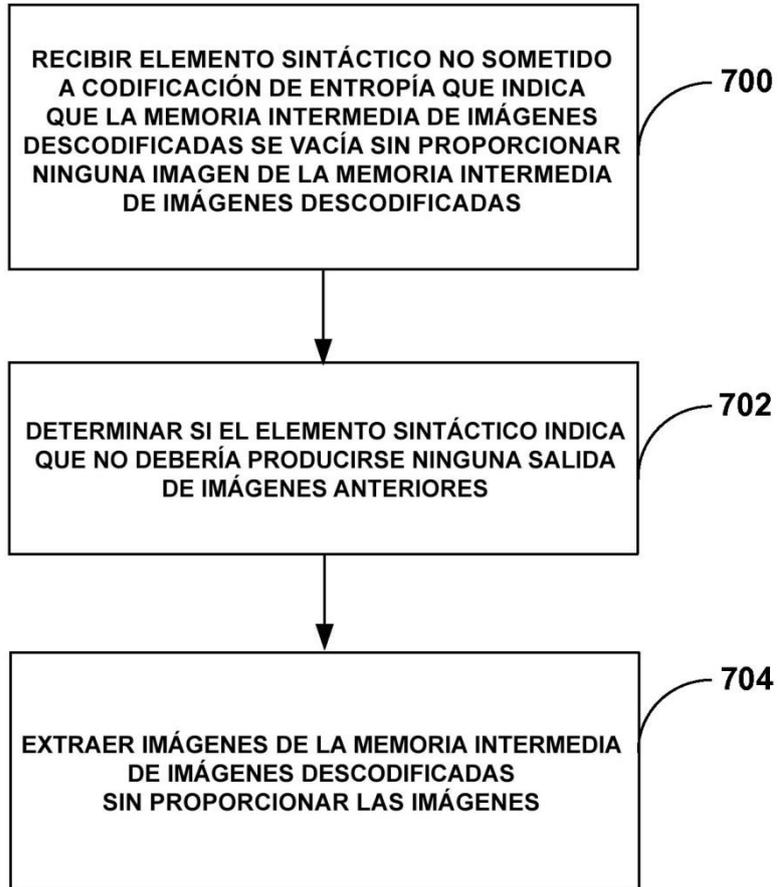


FIG. 7

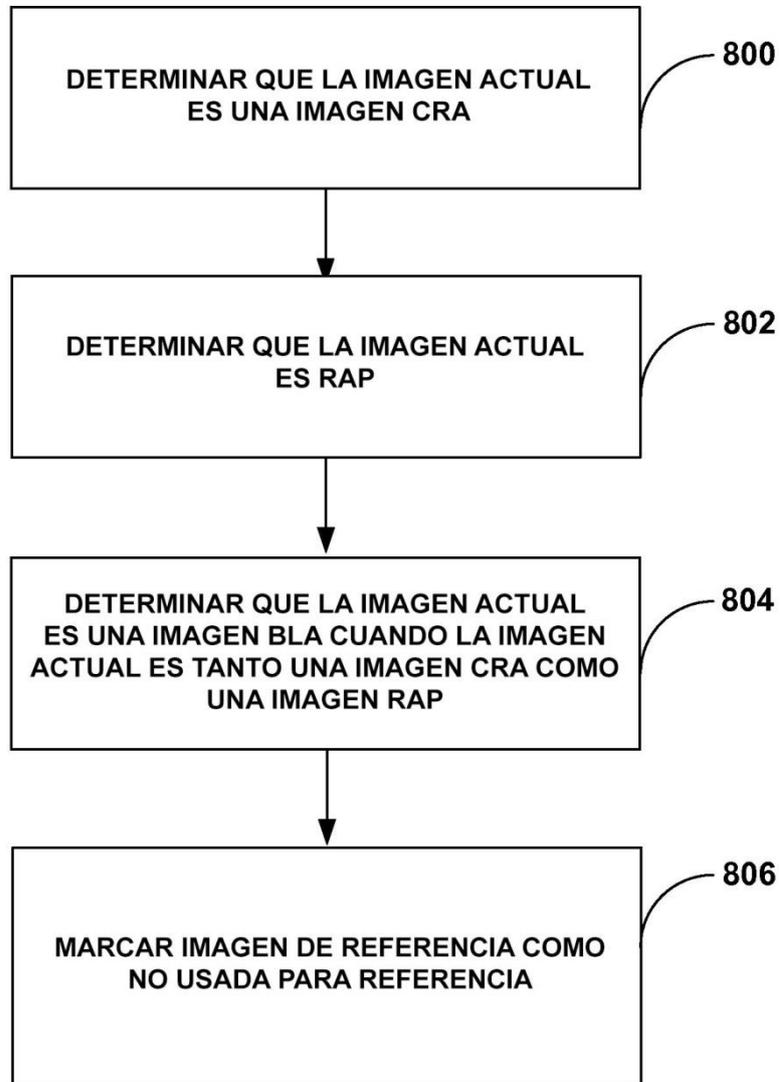


FIG. 8

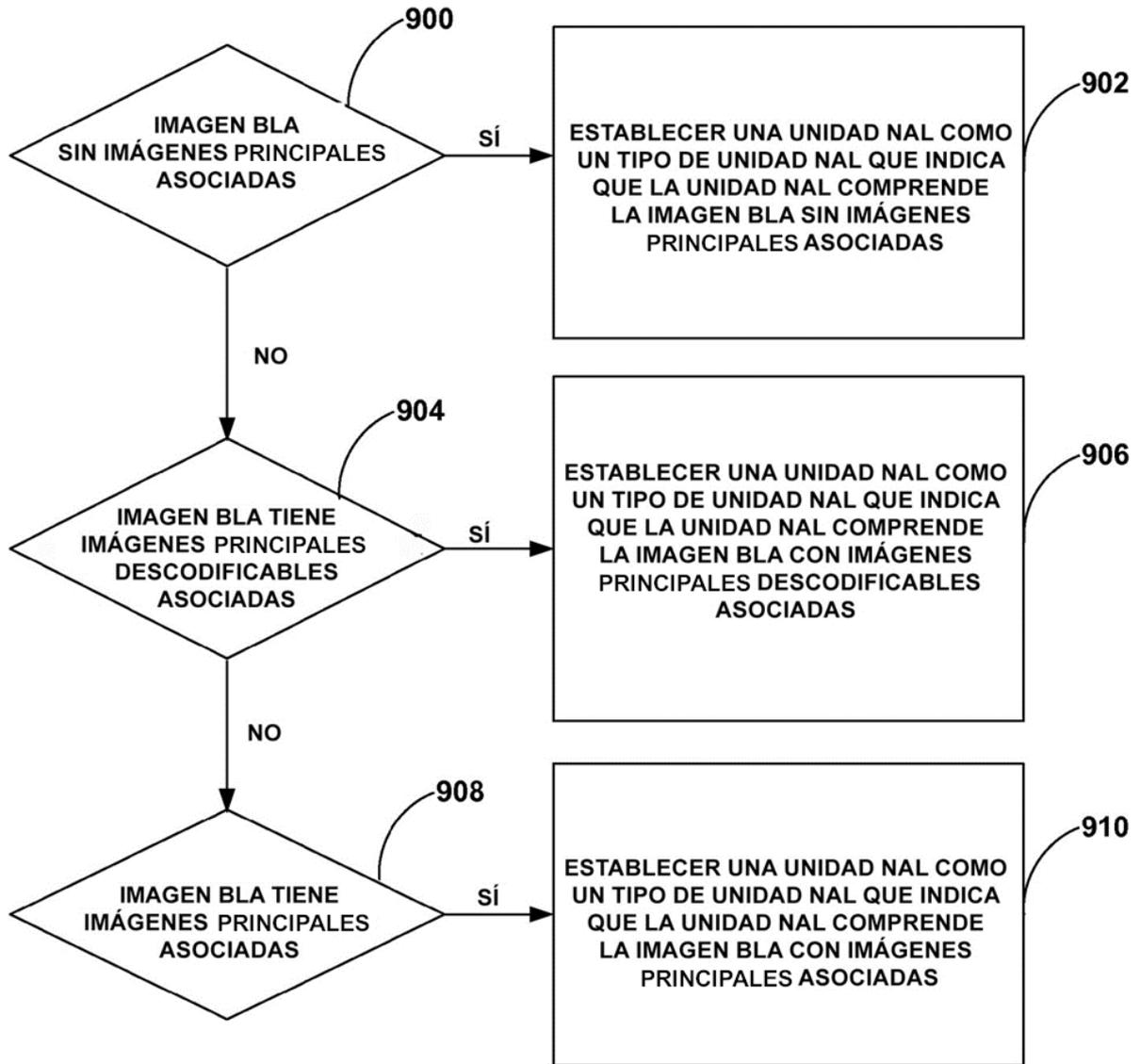


FIG. 9

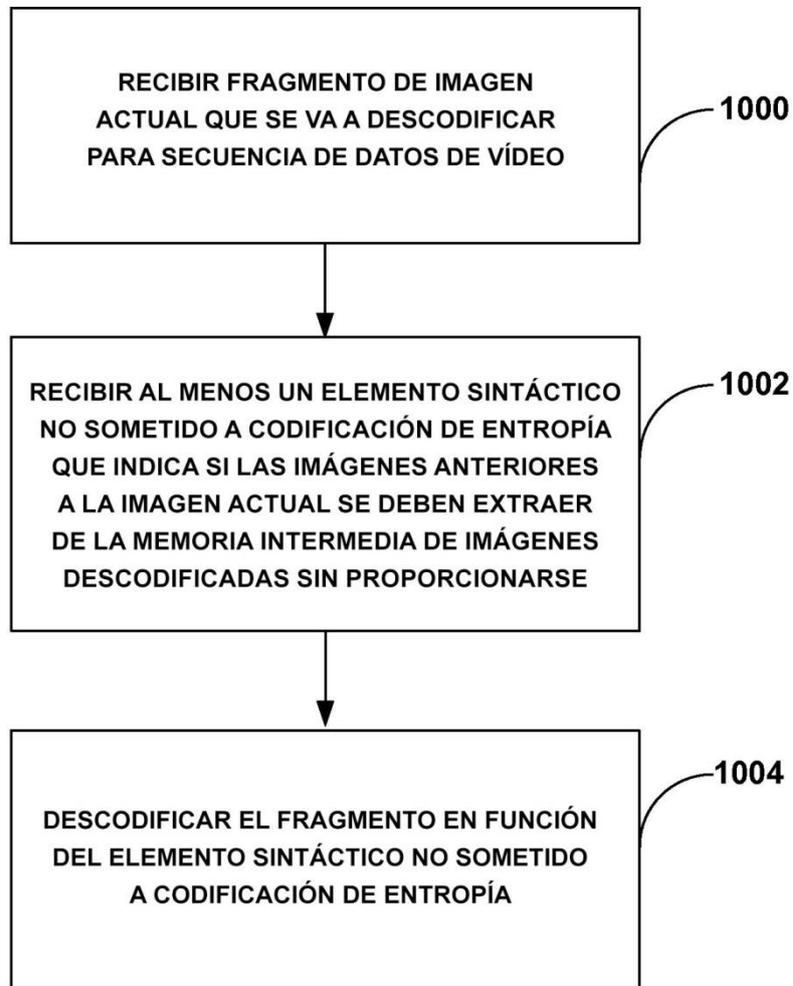


FIG. 10

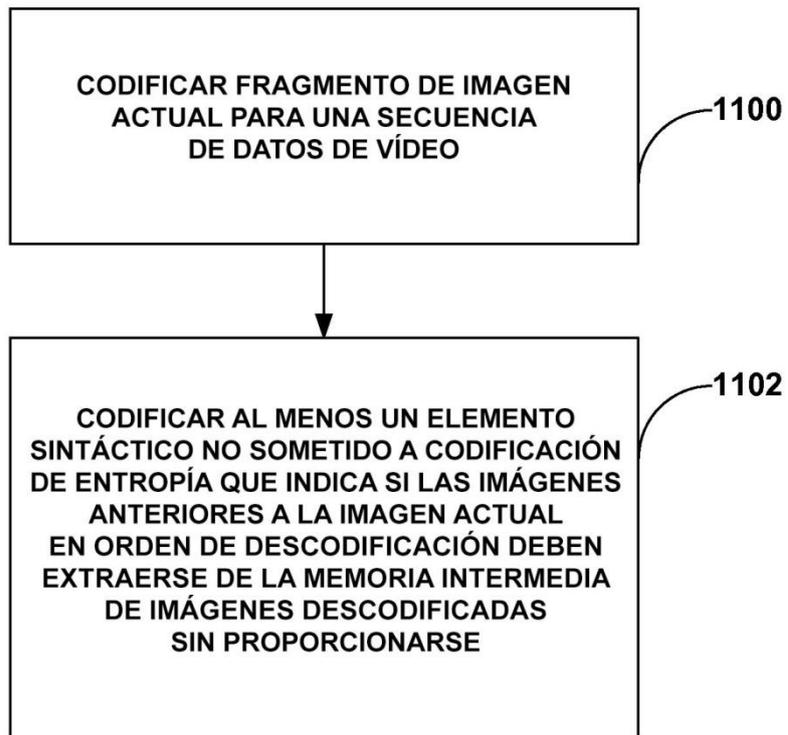


FIG. 11

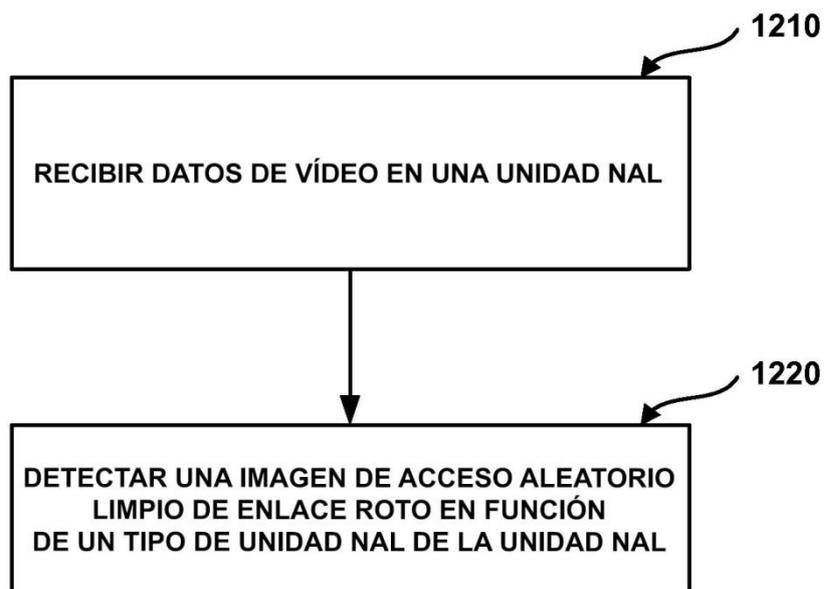


FIG. 12