

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 810 202**

51 Int. Cl.:

H04N 21/234 (2011.01)

H04N 21/6336 (2011.01)

H04N 21/647 (2011.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **17.06.2013 PCT/US2013/046152**

87 Fecha y número de publicación internacional: **03.01.2014 WO14004150**

96 Fecha de presentación y número de la solicitud europea: **17.06.2013 E 13732052 (9)**

97 Fecha y número de publicación de la concesión europea: **06.05.2020 EP 2868105**

54 Título: **Adaptación de transmisión continua basada en imágenes de acceso aleatorio limpio (CRA)**

30 Prioridad:

28.06.2012 US 201261665667 P
21.02.2013 US 201313773060

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
08.03.2021

73 Titular/es:

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

72 Inventor/es:

WANG, YE-KUI y
CHEN, YING

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 810 202 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

Adaptación de transmisión continua basada en imágenes de acceso aleatorio limpio (CRA)

5 **CAMPO TÉCNICO**

[0001] La presente divulgación se refiere, en general, al procesamiento de datos de vídeo, y, más en particular, a técnicas para admitir el acceso aleatorio en transmisiones continuas de vídeo comprimido.

10 **ANTECEDENTES**

[0002] Las capacidades del vídeo digital se pueden incorporar 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 los estándares definidos por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, parte 10, Codificación de Vídeo Avanzada (Advanced Video Coding, AVC), el estándar de Codificación de Vídeo de Alta Eficacia (High Efficiency Video Coding, HEVC) actualmente en desarrollo, estándares propietarios, formatos de compresión de vídeo abiertos tales como VP8, y extensiones de dichos estándares, técnicas o formatos. Los dispositivos de vídeo pueden transmitir, recibir, codificar, descodificar y/o almacenar información de vídeo digital más eficazmente implementando dichas técnicas de compresión de vídeo.

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

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

[0005] En el documento de entrada del JCT-VC nº JCTVC-10404 "CRA pictures with broken links" de Gary Sullivan (Equipo Conjunto de Colaboración en Codificación de Vídeo (JCT-VC) de ITU-T SG 16 WP 3 e ISO/IEC JTC 1/SC 29/WG 11 9º congreso: Ginebra, CH, 27 de abril - mayo de 2012), se divulga una propuesta para permitir que las imágenes CRA que no están al comienzo del flujo de bits tengan "imágenes principales" no descodificables, como puede tener una imagen CRA que comienza el flujo de bits. En el borrador del estándar HEVC en ese punto, una imagen CRA que comienza un flujo de bits puede estar seguida (en orden de flujo de bits) por "imágenes principales" que no se pueden descodificar debido a la falta de imágenes de referencia anteriores. Sin embargo, actualmente no se permite que una imagen CRA que está situada en medio de un flujo de bits tenga dichas "imágenes principales" no descodificables. En JCTVC-10404 se propone eliminar esta restricción y añadir un indicador de "enlace roto", que indicaría la presencia potencial de dichas imágenes principales no descodificables.

60 **BREVE EXPLICACIÓN**

[0006] En un ejemplo, las técnicas de la presente divulgación se refieren a tratar una imagen de acceso aleatorio limpio (CRA) como una imagen de acceso de enlace roto (BLA) en base a una indicación externa. Para una imagen CRA, las imágenes principales asociadas se pueden descodificar correctamente en base al comienzo de la descodificación a partir de una imagen de acceso aleatorio (RAP) situada antes de la imagen CRA en orden de descodificación y, para una imagen BLA, las imágenes principales asociadas son una de no correctamente descodificable o correctamente descodificable en base al comienzo de la descodificación a partir de una imagen

RAP situada antes de la imagen BLA en orden de descodificación. Una imagen que sigue a una imagen CRA o BLA en orden de descodificación pero precede a la imagen CRA o BLA respectiva en orden de salida es una imagen principal asociada con la imagen CRA o BLA respectiva. Por ejemplo, un descodificador de vídeo u otro dispositivo puede recibir una indicación externa. El descodificador de vídeo puede tratar entonces una imagen CRA como una imagen BLA en base a la indicación externa. En algunos ejemplos, se define un indicador para una imagen CRA y la indicación externa indica si se debe establecer el indicador en el descodificador de vídeo. En consecuencia, el descodificador de vídeo puede establecer el indicador en base a la indicación externa. El descodificador o alguna funcionalidad interna, tal como una unidad de procesamiento de indicación externa o un módulo de predicción pueden verificar entonces el indicador. En un ejemplo, el módulo de predicción puede tratar una imagen CRA como una imagen BLA en base a la indicación externa. Por ejemplo, un descodificador puede tratar la imagen CRA como una imagen BLA en base al indicador.

[0007] La invención se define en las reivindicaciones adjuntas. Cualquier ejemplo y modo de realización de la descripción que no esté dentro del alcance de las reivindicaciones no forma parte de la invención reivindicada y solamente se proporciona con propósitos ilustrativos.

[0008] Los detalles de uno o más ejemplos se exponen en los dibujos adjuntos y en la siguiente descripción. Otras características, objetivos y ventajas resultarán evidentes a partir de la descripción y de los dibujos, y a partir de las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0009]

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

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

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

La FIG. 4 es un diagrama de bloques que ilustra un conjunto ejemplar de dispositivos que forman parte de una red.

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

La FIG. 6 es un diagrama de flujo que ilustra un procedimiento de ejemplo de acuerdo con uno o más ejemplos descritos en la presente divulgación.

La FIG. 7 es un diagrama de flujo que ilustra un procedimiento de ejemplo de acuerdo con uno o más ejemplos descritos en la presente divulgación.

La FIG. 8 es un diagrama de flujo que ilustra un funcionamiento ejemplar de un primer dispositivo que envía una indicación externa y acciones de respuesta de un segundo dispositivo que recibe la indicación externa.

DESCRIPCIÓN DETALLADA

[0010] La presente divulgación describe técnicas para la adaptación de transmisión continua basada en imágenes de acceso aleatorio limpio (CRA). Se describen diversos diseños mejorados de codificación de vídeo, que pueden estar relacionados con la adaptación de transmisión continua basada en imágenes CRA, la salida de imágenes antes de imágenes de punto de acceso aleatorio (RAP) y la señalización de información de temporización de imagen.

[0011] En primer lugar se describen unos breves antecedentes de algunos estándares de codificación de vídeo. Los estándares de codificación de vídeo incluyen ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual e ITU-T H.264 (también conocido como ISO/IEC MPEG-4 AVC), incluyendo sus ampliaciones de Codificación de Vídeo Escalable (SVC) y Codificación de Vídeo Multivista (MVC).

[0012] Además, existe un nuevo estándar de codificación de vídeo, a saber la Codificación de Vídeo de Alta Eficiencia (HEVC) que se está desarrollando por el Equipo de Colaboración Conjunta sobre Codificación de Vídeo (JCT-VC) del Grupo de Expertos sobre Codificación de Vídeo (VCEG) de ITU-T y el Grupo de Expertos sobre Imágenes en Movimiento (MPEG) de ISO/IEC. Un borrador de trabajo (WD) de HEVC y denominado HEVC WD7

en adelante en el presente documento, está disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/9_Geneva/wg11/JCTVC-I1003-v5.zip.

5 **[0013]** Un borrador de trabajo (WD) más reciente de HEVC y denominado HEVC WD9 en adelante en el presente documento, está disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/9_Geneva/wg11/JCTVC-I1003-v10.zip.

10 **[0014]** En un ejemplo, las técnicas de la presente divulgación se refieren a tratar una imagen de acceso aleatorio limpio (CRA) como una imagen de acceso de enlace roto (BLA) en base a una indicación externa. Por ejemplo, un descodificador de vídeo u otro dispositivo puede recibir una indicación externa. El descodificador de vídeo puede tratar entonces una imagen CRA como una imagen BLA en base a la indicación externa. En algunos ejemplos, se define un indicador para una imagen CRA y la indicación externa indica si se debe establecer el indicador en el descodificador de vídeo. En consecuencia, el descodificador de vídeo puede establecer el indicador en base a la indicación externa. El descodificador o alguna funcionalidad interna, tal como una unidad de procesamiento de indicación externa o un módulo de predicción pueden verificar entonces el indicador. En un ejemplo, el módulo de predicción puede tratar una imagen CRA como una imagen BLA en base a la indicación externa. Por ejemplo, un descodificador puede tratar la imagen CRA como una imagen BLA en base al indicador.

15 **[0015]** En otro ejemplo, se define un indicador para una imagen CRA, y un descodificador u otro dispositivo puede recibir una indicación externa de que se debe establecer el indicador. El descodificador u otro dispositivo puede establecer entonces el indicador en base a la indicación externa. A continuación, el descodificador puede comprobar el indicador. Cuando se establece el indicador, el descodificador puede tratar la imagen CRA como una imagen BLA.

20 **[0016]** El acceso aleatorio se refiere a la descodificación de un flujo de bits de vídeo comenzando a partir de una imagen codificada que no es la primera imagen codificada en el flujo de bits. Se necesita acceso aleatorio a un flujo de bits en muchas aplicaciones de vídeo, tales como radiodifusión y transmisión continua, por ejemplo, para que los usuarios conmuten entre diferentes canales, para saltar a partes específicas del vídeo o para conmutar a un flujo de bits diferente para la adaptación de transmisión continua (por ejemplo, de la velocidad binaria, la velocidad de tramas, la resolución espacial, y así sucesivamente). Esta característica se puede habilitar insertando imágenes de acceso aleatorio o puntos de acceso aleatorio, muchas veces a intervalos regulares, en el flujo de bits de vídeo.

25 **[0017]** El empalme de flujos de bits se refiere a la concatenación de dos o más flujos de bits o partes de los mismos. Por ejemplo, un segundo flujo de bits se puede anexar a un primer flujo de bits, posiblemente con algunas modificaciones en uno cualquiera o en ambos flujos de bits para generar un flujo de bits empalmado. La primera imagen codificada del segundo flujo de bits también se denomina punto de empalme. Por lo tanto, las imágenes que siguen al punto de empalme en el flujo de bits empalmado se originan a partir del segundo flujo de bits, mientras que las imágenes que preceden al punto de empalme en el flujo de bits empalmado se originan a partir del primer flujo de bits.

30 **[0018]** Los empalmadores de flujos de bits pueden realizar el empalme de flujos de bits. Los empalmadores de flujos de bits son a menudo menos complicados, menos sofisticados y/o menos inteligentes que los codificadores. Por ejemplo, pueden no estar equipados con capacidades de codificación y descodificación por entropía. Los empalmadores de flujos de bits se pueden incorporar en cualquiera de los dispositivos descritos en el presente documento, incluyendo dispositivos de codificación o dispositivos de red.

35 **[0019]** La conmutación de flujo de bits se puede usar en entornos de transmisión continua adaptativa. Una operación de conmutación de flujo de bits en una imagen determinada en el flujo de bits al que se va a conmutar es efectivamente una operación de empalme de flujos de bits en la que el punto de empalme es el punto de conmutación del flujo de bits, es decir, la primera imagen del flujo de bits al que se va a conmutar.

40 **[0020]** Las imágenes de actualización de descodificación instantánea (IDR) como se especifica en AVC o HEVC se pueden usar para el acceso aleatorio. Sin embargo, dado que las imágenes que siguen a una imagen IDR en orden de descodificación no pueden usar imágenes descodificadas anteriores a la imagen IDR como referencia, los flujos de bits que dependen de imágenes IDR para el acceso aleatorio pueden tener una eficiencia de codificación significativamente menor.

45 **[0021]** Para mejorar la eficiencia de codificación, el concepto de imágenes de acceso aleatorio limpio (CRA) se introdujo en la HEVC para permitir que las imágenes que siguen a una imagen CRA en orden de descodificación, pero la preceden en orden de salida usen imágenes descodificadas antes de la imagen CRA como referencia. Las imágenes que siguen a una imagen CRA en orden de descodificación, pero preceden a la imagen CRA en orden de salida, se denominan imágenes principales asociadas con la imagen CRA (o imágenes principales de la imagen CRA). Las imágenes principales de una imagen CRA son correctamente descodificables si la descodificación comienza a partir de una imagen IDR o CRA anterior a la imagen CRA actual. Sin embargo, las imágenes principales de una imagen CRA pueden ser no correctamente descodificables cuando se produce un acceso

aleatorio a partir de la imagen CRA. Por lo tanto, los descodificadores en general descartan las imágenes principales durante la descodificación de acceso aleatorio. Para evitar la propagación de errores por 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 que incluye a las imágenes principales) como referencia.

[0022] El concepto de una imagen de acceso de enlace roto (BLA) se introdujo aún más en la HEVC después de la introducción de las imágenes CRA, y se basa en el concepto de las imágenes CRA. Una imagen BLA se origina típicamente a partir de un empalme de flujos de bits en la posición de una imagen CRA y, en el flujo de bits empalmado, la imagen CRA del punto de empalme se puede cambiar por una imagen BLA. La imagen IDR, la imagen CRA y la imagen BLA se denominan conjuntamente imágenes de punto de acceso aleatorio (RAP).

[0023] Una diferencia entre las imágenes BLA y las imágenes CRA es la siguiente. Para una imagen CRA, las imágenes principales asociadas son correctamente descodificables si la descodificación comienza a partir de una imagen RAP anterior a la imagen CRA en orden de descodificación. La imagen CRA puede ser no correctamente descodificable cuando se produce un acceso aleatorio a partir de la imagen CRA. Por ejemplo, cuando la descodificación comienza a partir de la imagen CRA, o en otras palabras, cuando la imagen CRA es la primera imagen en el flujo de bits. Para una imagen BLA, las imágenes principales asociadas pueden ser no correctamente descodificables en todos los casos, incluso cuando la descodificación comienza a partir de una imagen RAP anterior a la imagen BLA en orden de descodificación.

[0024] Para una imagen CRA o BLA particular, algunas de las imágenes principales asociadas son correctamente descodificables incluso cuando la imagen CRA o BLA es la primera imagen en el flujo de bits. Estas imágenes principales se denominan imágenes principales descodificables (DLP), y otras imágenes principales se denominan imágenes principales no descodificables (NLP) o imágenes principales descodificables de acceso aleatorio (RADL). Las NLP también se denominan imágenes etiquetadas para descarte (TFD) o imágenes principales de omisión de acceso aleatorio (RASL).

[0025] En algunos casos, los siguientes problemas pueden estar asociados con algunos procedimientos existentes (1) en la adaptación de transmisión continua basada en imágenes CRA, el cambio de una imagen CRA a una imagen BLA en general se debe realizar mediante un servidor de medios o un elemento de red intermedio, por ejemplo, un elemento de red con conocimiento de medios (MANE) o incluso un elemento de red sin conocimiento de medios, tal como una caché HTTP o un proxy web, MANE, que típicamente es preferente que sea menos complicado, menos sofisticado y/o menos inteligente y puede no ser capaz de cambiar el flujo de bits en modo alguno, (2) la salida de imágenes antes de una imagen IDR o BLA en orden de descodificación se puede controlar de alguna manera usando el `no_output_of_prior_pics_flag`. Cuando `no_output_of_prior_pics_flag` se establece a "1" o se infiere que es igual a 1, todas las imágenes descodificadas antes en orden de descodificación que la imagen IDR o BLA se descartan después de descodificar la imagen IDR o BLA sin salida/visualización. Sin embargo, a veces visualizar más de esas imágenes puede proporcionar una mejor experiencia de usuario. Actualmente no hay una manera de habilitar la salida/visualización de más imágenes en dichas situaciones, (3) se permite la salida de imágenes DLP. Dado que su orden de salida o sus tiempos de salida son anteriores a la imagen RAP asociada, el tiempo de presentación más temprano cuando se accede aleatoriamente a partir de la imagen RAP no se puede conocer simplemente comprobando la unidad de acceso que contiene la imagen RAP. Sin embargo, cuando se accede aleatoriamente a partir de una imagen RAP, el sistema debe intentar averiguar el inicio de reproducción más temprano para ver si esa imagen RAP se ajusta a la solicitud de acceso aleatorio del usuario.

[0026] En la presente divulgación se exponen varias técnicas que en general pueden abordar o mejorar uno o más de los problemas identificados anteriormente. Son posibles varias formas diferentes de gestionar dicho mensaje, recibido o inferido. A continuación se analizan varios ejemplos; estos incluyen (1) gestionar una imagen CRA como una imagen BLA, (2) cambiar una imagen CRA a una imagen BLA, y (3) gestionar una imagen CRA como una imagen CRA que inicia un flujo de bits.

[0027] En un ejemplo, un descodificador puede gestionar una imagen CRA como una imagen BLA. Un descodificador se puede configurar de modo que una imagen CRA se gestione como una imagen BLA cuando así se indica mediante medios externos. Dicha indicación externa puede ser un mensaje como se describe anteriormente (que determinada imagen CRA se debe gestionar como una imagen BLA) que se pasa al descodificador, mediante una función del lado del descodificador, a través de la inferencia o recepción de un servidor o un elemento de red intermedio.

[0028] Más específicamente, el proceso de descodificación se puede cambiar para que sea como sigue. Se puede usar una variable independiente que puede estar asociada con cada imagen CRA. Por ejemplo, la variable `HandleCraAsBlaFlag` está asociada con cada imagen CRA. En otras palabras, cada imagen CRA puede tener una variable `HandleCraAsBlaFlag` (también denominada indicador) asociada con la misma. El valor de `HandleCraAsBlaFlag` para algunas imágenes CRA se puede especificar mediante medios externos. Cuando el

valor de HandleCraAsBlaFlag para una imagen CRA particular no se especifica mediante medios externos, se puede establecer a "0" (por ejemplo, HandleCraAsBlaFlag de una imagen CRA por defecto es "0", indicando el valor "0" que una imagen CRA no se trata como una imagen BLA). En dicho ejemplo, un valor de "1" puede indicar que una imagen CRA se trata como una imagen BLA. En otros ejemplos, lo contrario puede ser cierto, un valor de "1" puede indicar que una imagen CRA no se trata como una imagen BLA y un valor de "0" puede indicar que una imagen CRA se trata como una imagen BLA.

[0029] El siguiente ejemplo supone el caso en el que HandleCraAsBlaFlag tiene un valor predeterminado de "0", que indica que una imagen CRA no se trata como una imagen BLA, y un valor de "1" indica que una imagen CRA se trata como una imagen BLA. Cuando se descodifica (incluyendo el análisis) cada unidad NAL de fragmento codificado, si HandleCraAsBlaFlag es igual a "1", por ejemplo, se gestiona una imagen CRA como una imagen BLA, y nal_unit_type indica una imagen CRA (por ejemplo, el valor es igual a "4" o "5" de acuerdo con HEVC WD7), se aplica lo siguiente, (1) el valor de nal_unit_type se cambia para indicar una imagen BLA (por ejemplo, el valor se incrementa en 2 de acuerdo con HEVC WD7), (2) el valor de no_output_of_prior_pics_flag se establece a 1, (3) si la imagen anterior en orden de descodificación es una imagen RAP y el rap_pic_id del fragmento actual es igual al rap_pic_id de la imagen anterior, se aplica lo siguiente. En primer lugar, si la siguiente imagen en orden de descodificación no es una imagen RAP, el valor de rap_pic_id del fragmento actual se cambia para que sea diferente al rap_pic_id de la imagen anterior en orden de descodificación, pero todavía en el intervalo de valores permitidos del elemento de sintaxis. En segundo lugar, de otro modo (la siguiente imagen en orden de descodificación es una imagen RAP), el valor de rap_pic_id de la imagen actual se cambia para que sea un valor diferente al rap_pic_id de la imagen anterior y la siguiente imagen en orden de descodificación, pero todavía en el intervalo de valores permitidos del elemento de sintaxis.

[0030] De forma alternativa, cuando se cambia una imagen CRA a una imagen BLA, un descodificador puede realizar lo siguiente, si los mensajes SEI de temporización de imagen están presentes y los tiempos de salida de la DPB para todas las imágenes en la DPB son menores que el tiempo de salida de la DPB de la imagen actual, el valor de no_output_of_prior_pics_flag se establece a 1; de otro modo, si el valor de no_output_of_prior_pics_flag se establece a "0".

[0031] En algunos ejemplos, HandleCraAsBlaFlag puede ser un primer indicador y no_output_of_prior_pictures_flag puede ser un segundo indicador. En algunos ejemplos, el indicador no_output_of_prior_pictures_flag puede ser una variable de contexto.

[0032] Con los cambios anteriores en HEVC WD7, puede ser posible eliminar aún más los procesos especiales de descodificación para una imagen CRA que es la primera imagen en el flujo de bits y las imágenes TFD asociadas. En este caso, cuando un flujo de bits comienza con una imagen CRA, la primera imagen CRA en el flujo de bits se debe gestionar como una imagen BLA, estableciendo el valor de HandleCraAsBlaFlag a "1" para la imagen CRA que inicia el flujo de bits, independientemente de si el valor se especifica mediante medios externos, si los hay, y aplicando el proceso de descodificación modificado anterior.

[0033] De forma alternativa, cuando se descodifica (incluyendo el análisis) cada unidad NAL de fragmento codificado, si la imagen actual es la primera imagen en el flujo de bits y nal_unit_type indica una imagen CRA (por ejemplo, el valor es igual a "4" o "5" conforme a HEVC WD7), se puede aplicar lo siguiente, el valor de nal_unit_type se cambia para indicar una imagen BLA (por ejemplo, el valor se incrementa en 2 conforme a HEVC WD5). En este ejemplo, no es necesario cambiar los valores de no_output_of_prior_pics_flag y rap_pic_id. De forma alternativa, el valor de HandleCraAsBlaFlag se puede indicar mediante un elemento de sintaxis en el flujo de bits, por ejemplo, un nuevo elemento de sintaxis que se puede incluir en la cabecera del fragmento o un nuevo mensaje SEI.

[0034] Un ejemplo se refiere a una adaptación de transmisión continua basada en imágenes CRA. En dicho ejemplo, en lugar de depender de un servidor o un elemento de red intermedio para cambiar una imagen BLA a una imagen CRA, un servidor o un elemento de red intermedio puede generar un mensaje que se enviará al lado del descodificador (es decir, el cliente). El mensaje puede notificar a un descodificador, por ejemplo, que se ha producido una operación de conmutación de flujo de bits en determinada imagen CRA y que la imagen CRA se debe gestionar como una imagen BLA. En el contexto de la transmisión continua adaptativa dinámica a través de HTTP (DASH), el lado del descodificador también puede inferir dicho mensaje por sí mismo a través del cambio del localizador de recursos uniforme (URL) que usó para solicitar los datos de transmisión continua y la recepción de los datos de medios asociados con el URL modificado.

[0035] En otro ejemplo, una imagen CRA se puede cambiar de modo que si los mensajes SEI de temporización de imagen están presentes y los tiempos de salida de la DPB para todas las imágenes en la DPB son menores que el tiempo de salida de la DPB de la imagen actual. El valor de no_output_of_prior_pics_flag se puede establecer a 1. De otro modo, el valor de no_output_of_prior_pics_flag se puede establecer a "0".

[0036] La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación y descodificación de vídeo 10 de ejemplo que puede usar las técnicas descritas en la presente divulgación. Como se muestra en la FIG. 1, el

sistema 10 incluye un dispositivo de origen 12 que genera los datos de vídeo codificados que se van a descodificar en un momento posterior por un dispositivo de destino 14. Las técnicas descritas en el presente documento se refieren, en general, a tratar una imagen CRA como una imagen BLA en base a una indicación externa. En consecuencia, estas técnicas se pueden aplicar, en general, al dispositivo de destino 14, que en general puede recibir la indicación externa y, en respuesta a dicha indicación externa, puede tratar una imagen CRA recibida en el dispositivo de destino como una imagen BLA cuando se procesa dentro del dispositivo de destino. Sin embargo, en algunos ejemplos, el dispositivo de origen 12 u otro dispositivo de red, tal como un MANE, puede proporcionar una indicación externa al dispositivo de destino 14 que hace que el dispositivo de destino 14 trate una imagen CRA recibida en el dispositivo de destino como una imagen BLA.

[0037] El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender cualquiera de una amplia gama de dispositivos, incluyendo ordenadores de escritorio, ordenadores plegables (es decir, portátiles), ordenadores de tableta, descodificadores, equipos telefónicos tales como los denominados teléfonos "inteligentes", los denominados paneles "inteligentes", televisiones, 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.

[0038] El dispositivo de destino 14 puede recibir los datos de vídeo codificados que se vayan a descodificar por medio de un enlace 16. El enlace 16 puede comprender cualquier tipo de medio o dispositivo que pueda mover los datos de vídeo codificados desde el dispositivo de origen 12 al 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 los datos de vídeo codificados directamente a un dispositivo de destino 14 en tiempo real. Un modulador puede modular los datos de vídeo codificados de acuerdo con un estándar de comunicación, tal como un protocolo de comunicación inalámbrica, y transmitirlos al dispositivo de destino 14. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrica o por cable, tal como un espectro de radiofrecuencia (RF) o una o más líneas físicas de transmisión. El medio de comunicación puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área amplia o una red global, tal como Internet. El medio de comunicación puede incluir encaminadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 al dispositivo de destino 14.

[0039] De forma alternativa, los datos codificados se pueden emitir desde la interfaz de salida 22 a un dispositivo de almacenamiento 32. De forma similar, la interfaz de entrada puede acceder a los datos codificados del dispositivo de almacenamiento 32. El dispositivo de almacenamiento 36 puede incluir cualquiera de una variedad de medios de almacenamiento de datos de acceso local o distribuidos, 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 36 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 36 por medio de transmisión continua o descarga. El servidor de archivos puede ser cualquier tipo de servidor que pueda 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 una página 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, incluyendo una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión Wi-Fi), una conexión por cable (por ejemplo, DSL, módem de cable, etc.) o una combinación de ambas que sea adecuada para acceder a datos de vídeo codificados almacenados en un servidor de archivos. La transmisión de datos de vídeo codificados desde el dispositivo de almacenamiento 36 puede ser una transmisión continua, una transmisión de descarga o una combinación de ambas.

[0040] Las técnicas de la presente divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas se pueden aplicar a la codificación de vídeo en apoyo de cualquiera de una variedad de aplicaciones multimedia, tales como radiodifusiones de televisión por el 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 se puede configurar para admitir una transmisión continua de vídeo unidireccional o bidireccional para admitir aplicaciones tales como transmisión continua de vídeo, reproducción de vídeo, radiodifusión de vídeo y/o 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/demodulador (módem) y/o un transmisor. En el dispositivo de origen 12, la fuente de vídeo 18 puede incluir una fuente tal como un dispositivo de captura de vídeo. Por ejemplo, una videocámara, un archivo de vídeo que contenga vídeo previamente capturado, una interfaz de alimentación de vídeo para recibir vídeo desde un

proveedor de contenido de vídeo y/o un sistema de gráficos de ordenador para generar datos de gráficos de ordenador como el vídeo de origen, o una combinación de dichas fuentes. Como 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 por cable.

[0042] El codificador de vídeo 20 puede codificar el vídeo capturado, precapturado 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. De forma alternativa, los datos de vídeo codificados se pueden almacenar en el dispositivo de almacenamiento 36 para un acceso posterior por el dispositivo de destino 14 u otros dispositivos, para su descodificación y/o reproducción. En otros ejemplos, se pueden realizar ambas opciones.

[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 comunicados a través del enlace 16, o proporcionados en el dispositivo de almacenamiento 36, pueden incluir una variedad de elementos de sintaxis 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 de sintaxis se pueden incluir con los datos de vídeo codificados transmitidos en un medio de comunicación, almacenados en un medio de almacenamiento o almacenados en un servidor de archivos.

[0044] En un ejemplo, el descodificador de vídeo 30 u otro dispositivo puede recibir una indicación externa. El descodificador de vídeo 30 puede tratar entonces una imagen de acceso aleatorio limpio (CRA) como una imagen de acceso de enlace roto (BLA) en base a la indicación externa. En algunos ejemplos, la indicación externa indica si se debe establecer un indicador en el descodificador de vídeo. En consecuencia, el descodificador de vídeo 30 puede establecer el indicador en base a la indicación externa. El descodificador de vídeo 30 o alguna funcionalidad interna, tal como una unidad de procesamiento de indicación externa 72 o un módulo de predicción 81 pueden comprobar entonces el indicador. En un ejemplo, el módulo de predicción 81 puede tratar una imagen CRA como una imagen BLA en base a la indicación externa que indica que la imagen CRA se debe tratar como una imagen BLA en base al indicador.

[0045] En otro ejemplo, el descodificador de vídeo 30 u otro dispositivo puede recibir una indicación externa de que se debe establecer un indicador. El descodificador de vídeo 30 u otro dispositivo puede establecer entonces el indicador en base a la indicación externa. El descodificador 30 puede comprobar entonces el indicador. Cuando se establece el indicador, el descodificador de vídeo 30 trata la imagen CRA como una imagen BLA.

[0046] El dispositivo de visualización 32 se puede integrar con, o ser externo a, el dispositivo de destino 14. En algunos ejemplos, el dispositivo de destino 14 puede incluir un dispositivo de visualización integrado y también estar configurado para interconectarse con un dispositivo de visualización externo. En otros ejemplos, el dispositivo de destino 14 puede ser un dispositivo de visualización. En general, el dispositivo de visualización 32 visualiza los datos de vídeo descodificados ante un usuario, y puede comprender cualquiera de una variedad de dispositivos de visualización, tales como una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos emisores de luz (OLED) u otro tipo de dispositivo de visualización.

[0047] El codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con un estándar de compresión de vídeo, tal como el estándar de codificación de vídeo de alta eficacia (HEVC), actualmente en desarrollo, y pueden cumplir con el modelo de prueba de HEVC (HM). Un borrador reciente de la HEVC está disponible, con fecha del 27 de junio de 2012, en http://we11.sc29.org/jct/doc_end_user/current_document.php?id=5885/JCTVC-I1003-v5. De forma alternativa, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con otros estándares privados o industriales, tales como el estándar ITU-T H.264, denominado de forma alternativa MPEG-4, Parte 10, Codificación avanzada de vídeo (AVC), o ampliaciones de dichos estándares. Sin embargo, las técnicas de esta divulgación no están limitadas a ningún estándar de codificación particular. Otros ejemplos de estándares de compresión de vídeo incluyen MPEG-2 e ITU-T H.263, así como formatos abiertos tales como VP8.

[0048] 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 cada uno con un codificador y un 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 separados. Si procede, en algunos ejemplos, las unidades MUX-DEMUX se pueden ajustar al protocolo de multiplexación ITU H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

[0049] Tanto el codificador de vídeo 20 como el descodificador de vídeo 30 se pueden implementar 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), matrices de

puertas programables por campo (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, de los que cualquiera se puede integrar como parte de un codificador/descodificador (CÓDEC) combinado en un dispositivo respectivo.

[0050] El JCT-VC está trabajando en el desarrollo del estándar HEVC. Los esfuerzos de estandarización de la HEVC se basan en un modelo en evolución de un dispositivo de codificación de vídeo, denominado modelo de prueba de la HEVC (HM). El HM supone varias capacidades adicionales de los dispositivos de codificación de vídeo relativos a los dispositivos existentes de acuerdo con, por ejemplo, ITU-T H.264/AVC. Por ejemplo, mientras que el estándar H.264 proporciona nueve modos de codificación mediante intrapredicción, el HM puede proporcionar hasta treinta y tres modos de codificación mediante intrapredicción.

[0051] En general, el modelo de trabajo del HM describe que una trama o imagen de vídeo se puede dividir en una secuencia de bloques del árbol de codificación, o bloques arbolados, o unidades de codificación más grandes (LCU), que incluyen tanto muestras de luminancia como de crominancia. Un bloque arbolado tiene un propósito similar al de un macrobloque del estándar H.264. Un fragmento incluye varios bloques arbolados consecutivos en orden de codificación. Una trama o imagen de vídeo se puede dividir en uno o más fragmentos. Cada bloque arbolado se puede dividir en unidades de codificación (CU) de acuerdo con un árbol cuaternario. Por ejemplo, un bloque arbolado, como un nodo raíz del árbol cuaternario, se puede separar en cuatro nodos hijo, y cada nodo hijo puede, a su vez, ser un nodo padre y separarse en otros cuatro nodos hijo. Un nodo hijo final, no separado, 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 se puede separar un bloque arbolado, y también pueden definir un tamaño mínimo de los nodos de codificación.

[0052] Una CU incluye un nodo de codificación y unidades de predicción (PU) y unidades de transformada (TU) asociadas al nodo de codificación. Un tamaño de la CU corresponde a un tamaño del nodo de codificación y debe tener conformación cuadrada. El tamaño de la CU puede variar desde 8x8 píxeles hasta el tamaño del bloque arbolado, 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 se pueden dividir para tener una conformación 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 conformación cuadrada o no cuadrada.

[0053] El estándar HEVC admite transformaciones de acuerdo con las TU, que pueden ser diferentes para CU diferentes. 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 más pequeñas que las PU. En algunos ejemplos, las muestras residuales correspondientes a una CU se pueden subdividir en unidades más pequeñas usando una estructura de árbol cuaternario conocida como "árbol cuaternario residual" (RQT). Los nodos hoja del RQT se pueden denominar unidades de transformada (TU). Los valores de diferencias de píxeles asociados a las TU se pueden transformar para producir coeficientes de transformada, que se pueden cuantificar.

[0054] En general, una PU incluye datos relacionados con el procedimiento 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.

[0055] En general, se usa una TU para los procedimientos de transformada y cuantificación. Una CU dada que tiene una o más PU también puede incluir una o más unidades de transformada (TU). Seguidamente a 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 se pueden transformar en coeficientes de transformada, cuantificar y explorar usando las TU, para producir coeficientes de transformada en serie para la codificación por entropía. La presente divulgación usa típicamente el término "bloque de vídeo" para referirse a un nodo de codificación de una CU. En algunos casos específicos, la presente divulgación también puede usar el término "bloque de vídeo" para referirse a un bloque arbolado, es decir, una LCU o una CU, que incluye un nodo de codificación y PU y TU.

- 5 **[0056]** 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 funciona típicamente en bloques de vídeo dentro de fragmentos de vídeo individuales para 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 un estándar de codificación especificado.
- 10 **[0057]** En un ejemplo, el HM admite 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.
- 15 **[0058]** En la presente divulgación, " $N \times N$ " y "N por N" se pueden usar de manera intercambiable 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.
- 20 **[0059]** Después de la codificación de intrapredicción o interpredicción mediante 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 de píxel) y las TU pueden comprender coeficientes en el dominio de transformada seguidamente a 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 producir coeficientes de transformada para la CU.
- 25 **[0060]** Después de cualquier transformada para producir 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 procedimiento en el que coeficientes de transformada se cuantifican para reducir posiblemente la cantidad de datos usados para representar los coeficientes, proporcionando compresión adicional. El proceso de cuantificación puede reducir la profundidad de bits asociada con algunos, o con la totalidad, de los coeficientes. Por ejemplo, un valor de n bits se puede redondear a la baja hasta un valor de m bits durante la cuantificación, donde n es mayor que m.
- 30 **[0061]** En algunos ejemplos, el codificador de vídeo 20 puede usar un orden de exploración predefinido para explorar los coeficientes de transformada cuantificados, para producir un vector en serie 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 por 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 por entropía por división de intervalos de probabilidad (PIPE) u otra metodología de codificación por entropía. El codificador de vídeo 20 también puede codificar por entropía los elementos de sintaxis asociados a los datos de vídeo codificados, para su uso por el descodificador de vídeo 30 en la descodificación de los datos de vídeo.
- 35 **[0062]** Para realizar la CABAC, el codificador de vídeo 20 puede asignar un contexto, dentro de un modelo de contexto, a un símbolo que se va a transmitir. El contexto se puede referir, por ejemplo, a si los valores vecinos del símbolo son distintos de cero o no. Para realizar la codificación CAVLC, el codificador de vídeo 20 puede seleccionar un código de longitud variable para un símbolo a transmitir. Las palabras de código en la VLC se pueden construir de modo que los códigos relativamente más cortos correspondan a símbolos más probables, mientras que los códigos más largos correspondan a símbolos menos probables. De esta forma, el uso de la VLC puede lograr un ahorro en bits con respecto, por ejemplo, al uso de palabras de código de igual longitud para cada
- 40
- 45
- 50
- 55
- 60
- 65

símbolo que se va a transmitir. La determinación de la probabilidad se puede basar en un contexto asignado al símbolo.

5 **[0063]** De acuerdo con la presente divulgación, el dispositivo de origen 12 (o posiblemente otro dispositivo intermedio no mostrado en la FIG. 1) puede proporcionar una indicación externa 34 al dispositivo de destino 14 que hace que el dispositivo de destino 14 trate una imagen CRA recibida en el dispositivo de destino como una imagen BLA. Por ejemplo, el dispositivo de origen 12 (o posiblemente otro dispositivo intermedio no mostrado en la FIG. 1) puede determinar que un usuario ha realizado un cambio, tal como solicitar un vídeo de diferente resolución o calidad, o se puede producir un enlace roto. Cuando se produce un cambio en la resolución o se produce un enlace roto, por ejemplo, esto puede significar que una imagen CRA se debe tratar como una imagen BLA porque cualquier información de imágenes anteriores almacenada en el dispositivo de recepción puede no ser válida para descodificar el flujo de bits entrante.

15 **[0064]** La FIG. 2 es un diagrama de bloques que ilustra un codificador de vídeo 20 de ejemplo que puede implementar las técnicas descritas en la presente divulgación. Como se analiza anteriormente, las técnicas descritas en el presente documento se refieren, en general, a tratar una imagen CRA como una imagen BLA en base a una indicación externa recibida en el dispositivo de destino 14. Sin embargo, en algunos ejemplos, el dispositivo de origen 12 u otro dispositivo de red, tal como un MANE, puede proporcionar una indicación externa al dispositivo de destino 14 que hace que el dispositivo de destino 14 trate una imagen CRA recibida en el dispositivo de destino como una imagen BLA.

25 **[0065]** 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) se puede referir a cualquiera de varios modos de compresión de base espacial. Los intermodos, tales como la predicción unidireccional (modo P) o la bipredicción (modo B), se pueden referir a cualquiera de varios modos de compresión de base temporal.

30 **[0066]** En el ejemplo de la FIG. 2, el codificador de vídeo 20 incluye un módulo de división 35, un módulo de predicción 41, un módulo de filtro 63, una memoria de imágenes de referencia 64, un sumador 50, un módulo de transformada 52, un módulo de cuantificación 54 y un módulo de codificación por entropía 56. El módulo de predicción 41 incluye un módulo de estimación de movimiento 42, un módulo de compensación de movimiento 44 y un módulo de intrapredicción 46. Para la reconstrucción de bloques de vídeo, el codificador de vídeo 20 incluye además un módulo de cuantificación inversa 58, un módulo de transformada inversa 60 y un sumador 62. El módulo de filtro 63 está destinado 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 según muestras (SAO). Aunque el módulo de filtro 63 que se muestra en la FIG. 2 es un filtro en bucle, en otras configuraciones, el módulo de filtro 63 se puede implementar como un filtro posbucle.

40 **[0067]** El dispositivo de origen 12 u otro dispositivo de red, tal como un MANE, puede proporcionar una indicación externa 34 al dispositivo de destino 14 que hace que el dispositivo de destino 14 trate una imagen CRA recibida en el dispositivo de destino como una imagen BLA. Por ejemplo, la indicación externa 34, que en general es externa al dispositivo de destino 14, y en general no se transmite como parte del flujo de bits, se puede generar mediante el módulo de predicción 41, que puede tener acceso a indicaciones relacionadas con el estado del flujo de bits. Este es solo un ejemplo, sin embargo, otras unidades o módulos en el dispositivo de origen 12, o en otros dispositivos externos al dispositivo de origen 12, también pueden generar una indicación externa.

50 **[0068]** Como se representa en la FIG. 2, el codificador de vídeo 20 recibe datos de vídeo, y el módulo 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 en árbol cuaternario de LCU y CU. El codificador de vídeo 20 ilustra, en general, los componentes que codifican los 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). El módulo de predicción 41 puede seleccionar uno de entre una pluralidad de posibles modos de codificación, tal como uno de entre una pluralidad de modos de intracodificación, o uno de entre una pluralidad de modos de intercodificación, para el bloque de vídeo actual, basándose en resultados de errores (por ejemplo, la velocidad de codificación y el nivel de distorsión). El módulo de predicción 41 puede proporcionar el bloque intracodificado o intercodificado resultante al sumador 50 para generar datos de bloque residual, y al sumador 62 para reconstruir el bloque codificado para su uso como imagen de referencia.

60 **[0069]** El módulo de intrapredicción 46, dentro del módulo de predicción 41, puede realizar la codificación intrapredictiva del bloque de vídeo actual con respecto a uno o más bloques contiguos de la misma trama o fragmento que el bloque actual que se vaya a codificar para proporcionar compresión espacial. El módulo de estimación de movimiento 42 y el módulo de compensación de movimiento 44, dentro del módulo de predicción

41, realizan la codificación interpredictiva del bloque de vídeo actual con respecto a uno o más bloques predictivos en una o más imágenes de referencia, para proporcionar compresión temporal.

5 **[0070]** El módulo de estimación de movimiento 42 puede estar configurado 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. El módulo de estimación de movimiento 42 y el módulo de compensación de movimiento 44 pueden estar altamente integrados, pero se ilustran por separado con fines conceptuales. La estimación de movimiento, realizada por el módulo de estimación de movimiento 42, es el proceso de generar vectores de movimiento, que estiman el movimiento para los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar la desviación de una PU de un bloque de vídeo dentro de una trama o imagen de vídeo actual relativa a un bloque predictivo dentro de una imagen de referencia.

15 **[0071]** Un bloque predictivo es un bloque que se encuentra que coincide estrechamente con la PU del bloque de vídeo que se va a codificar en términos de diferencia de píxel, que se puede determinar 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 valores para posiciones fraccionarias de píxeles de imágenes de referencia almacenadas en la memoria de imágenes de referencia 64. Por ejemplo, el codificador de vídeo 20 puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones fraccionarias de píxel de la imagen de referencia. Por lo tanto, el módulo de estimación de movimiento 42 puede realizar una búsqueda de movimiento con respecto a las posiciones de píxel completo y a las posiciones de píxel fraccionario, y emitir un vector de movimiento con una precisión de píxel fraccionario.

25 **[0072]** El módulo de estimación de movimiento 42 calcula un vector de movimiento para una PU de un bloque de vídeo en un segmento intercodificado comparando la posición de la PU con la posición de un bloque predictivo de una imagen de referencia. La imagen de referencia se puede seleccionar de una primera lista (lista 0) de imágenes de referencia o de una segunda lista (lista 1) de imágenes de referencia, de las que cada una identifica una o más imágenes de referencia almacenadas en una memoria de imágenes de referencia 64. El módulo de estimación de movimiento 42 envía el vector de movimiento calculado al módulo de codificación por entropía 56 y al módulo de compensación de movimiento 44.

35 **[0073]** La compensación de movimiento, realizada por el módulo de compensación de movimiento 44, puede implicar buscar o generar el bloque predictivo basándose en el vector de movimiento determinado por la estimación de movimiento, realizando posiblemente interpolaciones a la precisión de subpíxel. Tras recibir el vector de movimiento para la PU del bloque de vídeo actual, el módulo 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 esté codificando, formando valores de diferencia de píxel. Los valores de diferencia de píxel forman datos residuales para el bloque, y pueden incluir componentes de diferencia tanto de luma como de croma. El sumador 50 representa el componente o los componentes que realizan esta operación de resta. El módulo de compensación de movimiento 44 también puede generar elementos de sintaxis asociados a los bloques de vídeo y al segmento de vídeo para su uso por el descodificador de vídeo 30 en la descodificación de los bloques de vídeo del segmento de vídeo.

45 **[0074]** El módulo de intrapredicción 46 puede realizar la intrapredicción de un bloque actual, como alternativa a la interpredicción realizada por el módulo de estimación de movimiento 42 y el módulo de compensación de movimiento 44, descrita anteriormente. En particular, el módulo de intrapredicción 46 puede determinar un modo de intrapredicción que se vaya a usar para codificar un bloque actual. En algunos ejemplos, el módulo de intrapredicción 46 puede codificar un bloque actual usando diversos modos de intrapredicción, por ejemplo, durante pasadas de codificación independientes, y el módulo de intrapredicción 46 (o el módulo de selección de modo 40, en algunos ejemplos) puede seleccionar un modo de intrapredicción adecuado que se va a usar de entre los modos probados. Por ejemplo, el módulo de intrapredicción 46 puede calcular valores de velocidad-distorsión mediante un análisis de velocidad-distorsión para los diversos modos de intrapredicción probados, y seleccionar el modo de intrapredicción que tenga las mejores características de velocidad-distorsión entre los modos probados. El análisis de velocidad-distorsión determina, en general, una cantidad de distorsión (o error) entre un bloque codificado y un bloque original no codificado que se codificó para producir el bloque codificado, así como una tasa de bits (es decir, un número de bits) usada para producir el bloque codificado. El módulo de intrapredicción 46 puede calcular relaciones 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.

60 **[0075]** En cualquier caso, tras seleccionar un modo de intrapredicción para un bloque, el módulo de intrapredicción 46 puede proporcionar información que indique el modo de intrapredicción seleccionado para el bloque al módulo de codificación por entropía 56. El módulo de codificación por entropía 56 puede codificar la información que indica el modo de intrapredicción seleccionado de acuerdo con las técnicas de la presente divulgación. El codificador de vídeo 20 puede incluir datos de configuración en el flujo de bits transmitido. Los datos de configuración pueden incluir una pluralidad de tablas de índices de modos de intrapredicción y una pluralidad

de tablas de índices de modos de intrapredicción modificadas (también denominadas tablas de correlación de palabras de código), definiciones de contextos de codificación para varios bloques e indicaciones del modo de intrapredicción más probable, una tabla de índices de modalidades de intrapredicción y una tabla de índices de modos de intrapredicción modificada, a usar para cada uno de los contextos.

5

[0076] Después de que el módulo de predicción 41 genere el bloque predictivo para el bloque de vídeo actual, ya sea mediante interpredicción o intrapredicción, el codificador 20 de vídeo forma un bloque de vídeo residual restando el bloque predictivo del bloque de vídeo actual. Los datos de vídeo residuales del bloque residual pueden incluirse en una o más TU y aplicarse al módulo de transformada 52. El módulo de transformada 52 transforma los datos de vídeo residuales en coeficientes de transformada residuales mediante una transformada, tal como una transformada de coseno discreta (DCT) o una transformada conceptualmente similar. El módulo de transformada 52 puede convertir los datos de vídeo residuales de un dominio de píxel en un dominio de transformada, tal como un dominio de frecuencia.

10

15

[0077] El módulo de transformada 52 puede enviar los coeficientes de transformada resultantes al módulo de cuantificación 54. El módulo de cuantificación 54 cuantifica los coeficientes de transformada para reducir más la velocidad binaria. El proceso de cuantificación puede reducir la profundidad de bits asociada con algunos, o con la totalidad, de los coeficientes. El grado de cuantificación se puede modificar ajustando un parámetro de cuantificación. En algunos ejemplos, el módulo de cuantificación 54 puede realizar, a continuación, una exploración de la matriz que incluya los coeficientes de transformada cuantificados. De forma alternativa, el módulo de codificación por entropía 56 puede realizar la exploración.

20

25

[0078] Tras la cuantificación, el módulo de codificación por entropía 56 realiza la codificación por entropía de los coeficientes de transformada cuantificados. Por ejemplo, el módulo de codificación por entropía 56 puede realizar una codificación de longitud variable adaptativa según el contexto (CAVLC), una codificación aritmética binaria adaptativa según el contexto (CABAC), una codificación aritmética binaria adaptativa según el contexto basada en la sintaxis (SBAC), una codificación por entropía de partición de intervalo de probabilidad (PIPE) u otra metodología o técnica de codificación por entropía. Tras la codificación por entropía realizada por el módulo de codificación por entropía 56, el flujo de bits codificado puede transmitirse al descodificador de vídeo 30, o archivar para su posterior transmisión o recuperación por el descodificador de vídeo 30. El módulo de codificación por entropía 56 también puede realizar la codificación por entropía de los vectores de movimiento y de los otros elementos de sintaxis para el segmento de vídeo actual que se esté codificando.

30

35

[0079] El módulo de cuantificación inversa 58 y el módulo 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 uso posterior como bloque de referencia de una imagen de referencia. El módulo de compensación de movimiento 44 puede calcular un bloque de referencia sumando el bloque residual a un bloque predictivo de una de las imágenes de referencia de una de las listas de imágenes de referencia. El módulo de compensación de movimiento 44 también puede aplicar uno o más filtros de interpolación al bloque residual reconstruido para calcular valores de píxel de subentero para su uso en la estimación de movimiento. El sumador 62 añade el bloque residual reconstruido al bloque de predicción con compensación de movimiento producido por el módulo de compensación de movimiento 44 para generar un bloque de referencia para su almacenamiento en la memoria de imágenes de referencia 64. El módulo de estimación de movimiento 42 y el módulo de compensación de movimiento 44 pueden usar el bloque de referencia como bloque de referencia para realizar la interpredicción de un bloque en una trama o imagen de vídeo subsiguiente.

40

45

[0080] El codificador de vídeo 20 de la FIG. 2 representa un ejemplo de un codificador de vídeo que se puede configurar para señalar que una imagen de acceso aleatorio limpio (CRA) se debe tratar como una imagen de acceso de enlace roto (BRA), como se describe en el presente documento.

50

[0081] La FIG. 3 es un diagrama de bloques que ilustra un descodificador de vídeo de ejemplo 30 que puede implementar las técnicas de la presente divulgación, que, en general, se refieren a tratar una imagen CRA como una imagen BLA en base a una indicación externa 70, que se puede generar mediante una entidad de red 29, tal como un MANE o algún otro dispositivo externo (no mostrado). En un ejemplo, un descodificador de vídeo 30 recibe una indicación externa 70 de que se debe establecer un indicador 74. El descodificador de vídeo 30 recibe la indicación externa 70. En otros ejemplos, la indicación externa 70 puede recibirse y procesarse externamente al descodificador de vídeo 30. La unidad de procesamiento de indicación externa 72 establece el indicador 74 en base a la indicación externa. A continuación, el indicador se pasa al módulo de predicción 81. En el ejemplo ilustrado, la unidad de procesamiento de indicación externa 72 está dentro del descodificador de vídeo 30. En otros ejemplos, la unidad de procesamiento de indicación externa 72 puede ser externa a y estar separada del descodificador de vídeo 30. En un descodificador de vídeo 30, el módulo de predicción (81) comprueba el indicador, y cuando el indicador se establece trata una imagen de acceso aleatorio limpio (CRA) como una imagen de acceso de enlace roto (BLA).

55

60

[0082] En algunos ejemplos, un valor predeterminado del indicador es "0" y un valor establecido del indicador es "1". En otros ejemplos, lo contrario puede ser cierto, el valor predeterminado del indicador es "1" y un valor establecido del indicador es "0". En otras palabras, el indicador puede ser activo alto ("1") o activo bajo ("0").

5 **[0083]** En algunos ejemplos, cuando se descodifica una unidad de capa de abstracción de red (NAL) de fragmento codificado, si se establece el primer indicador, el módulo de predicción 81 puede cambiar un tipo de unidad NAL de la unidad NAL. Cuando se descodifica la unidad de capa de abstracción de red (NAL) de fragmento codificado, si se establece el primer indicador, el módulo de predicción 81 puede cambiar el valor de un segundo indicador. El segundo indicador puede ser el indicador no_output_of_prior_pics. Además, cuando se descodifica la unidad de capa de abstracción de red (NAL) de fragmento codificado, si se establece el indicador, el módulo de predicción puede establecer el valor de un segundo indicador a "1".

15 **[0084]** En un ejemplo, cuando una imagen actual es una imagen CRA, y cuando se dispone de alguna indicación externa para establecer una variable que indica que una imagen CRA se debe gestionar como una imagen BLA (por ejemplo, HandleCraAsBlaFlag), entonces la variable (por ejemplo, HandleCraAsBlaFlag) se puede establecer al valor proporcionado por los medios externos. De lo contrario, el valor de la variable (por ejemplo, HandleCraAsBlaFlag) se puede establecer para indicar que la imagen CRA no se debe gestionar como una imagen BRA. Por ejemplo, el HandleCraAsBlaFlag se puede establecer a "1" para indicar que la imagen CRA se debe gestionar como una imagen BRA y establecer a "0" para indicar que la imagen CRA no se debe gestionar como una imagen BRA.

[0085] Cabe destacar que, si bien algunos ejemplos de indicación externa se pueden describir en el presente documento, estos no pretenden ser una lista exhaustiva. Se podrían utilizar muchas posibles indicaciones externas.

25 **[0086]** En algunos ejemplos, cuando la imagen actual es una imagen CRA y una variable que indica que una imagen CRA se debe gestionar como una imagen BLA (por ejemplo, HandleCraAsBlaFlag) es igual a "1", donde "1" indica que la imagen CRA se debe gestionar como una imagen BLA, el valor de no_output_of_prior_pics_flag se puede establecer a "1", y aplica los siguiente durante los procesos de análisis y descodificación para cada unidad NAL de segmento de fragmento codificado.

30 **[0087]** En un ejemplo, el no_output_of_prior_pics_flag especifica cómo se tratan las imágenes previamente descodificadas en la memoria intermedia de imágenes descodificadas después de descodificar una imagen IDR o BLA. En un ejemplo, cuando la imagen IDR o BLA es la primera imagen en el flujo de bits, el valor de no_output_of_prior_pics_flag no tiene efecto en el proceso de descodificación. Cuando la imagen IDR o BLA no es la primera imagen en el flujo de bits y el valor de pic_width_in_luma_samples o pic_height_in_luma_samples o sps_max_dec_pic_buffering[sps_max_temporal_layers_minus1] derivado del conjunto de parámetros de secuencia activo es diferente del valor de pic_width_in_luma_samples o pic_height_in_luma_samples o sps_max_dec_pic_buffering[sps_max_temporal_layers_minus1] derivado del conjunto de parámetros de secuencia activo para la imagen anterior, el descodificador puede inferir (pero no debería) que no_output_of_prior_pics_flag es igual a 1, independientemente del valor real de no_output_of_prior_pics_flag.

40 **[0088]** En el ejemplo de la FIG. 3, el descodificador 30 de vídeo incluye un módulo de descodificación por entropía 80, un módulo de predicción 81, un módulo de cuantificación inversa 86, un módulo de transformada inversa 88, un sumador 90, un módulo de filtro 91 y una memoria de imágenes de referencia 92. El módulo de predicción 81 incluye un módulo de compensación de movimiento 82 y un módulo de intrapredicción 84. En algunos ejemplos, el descodificador de vídeo 30 puede realizar un pase de descodificación, en general, recíproco al pase de codificación descrito con respecto al codificador de vídeo 20 de la FIG. 2.

50 **[0089]** Durante el procedimiento 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 de sintaxis 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 de este tipo configurado para implementar una o más de las técnicas descritas anteriormente. Como se ha descrito anteriormente, algunas de las técnicas descritas en la presente divulgación se pueden implementar mediante 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 ser partes de dispositivos separados, mientras que, en otros casos, la funcionalidad descrita con respecto a la entidad de red 29 se puede realizar por el mismo dispositivo que comprende el descodificador de vídeo 30.

60 **[0090]** Como se analiza anteriormente, un dispositivo de red, tal como la entidad de red 29, que puede ser un MANE, puede proporcionar una indicación externa 34 al dispositivo de destino 14 que hace que el dispositivo de destino 14 trate una imagen CRA recibida en el dispositivo de destino como una imagen BLA. Por ejemplo, la indicación externa 34, que en general es externa al dispositivo de destino 14, y en general no se transmite como parte del flujo de bits, se puede generar mediante el módulo de predicción 41, que puede tener acceso a indicaciones relacionadas con el estado del flujo de bits. Este es solo un ejemplo, sin embargo, otras unidades o

módulos en el dispositivo de origen 12, o en otros dispositivos externos al dispositivo de origen 12, también pueden generar una indicación externa.

5 **[0091]** El módulo de descodificación por entropía 80 del descodificador 30 de vídeo realiza la descodificación por entropía del flujo de bits para generar coeficientes cuantificados, vectores de movimiento y otros elementos de sintaxis. El módulo de descodificación por entropía 80 reenvía los vectores de movimiento y otros elementos de sintaxis al módulo de predicción 81. El descodificador de vídeo 30 puede recibir los elementos de sintaxis a nivel de fragmento de vídeo y/o a nivel de bloque de vídeo.

10 **[0092]** Cuando el fragmento de vídeo se codifica como un fragmento intracodificado (I), el módulo de intrapredicción 84 del módulo 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 descodificados previamente de la trama o imagen actual. Cuando la trama de vídeo se codifica como un fragmento intercodificado (es decir, B, P o GPB), el módulo de compensación de movimiento 82 del módulo de predicción 81 genera bloques predictivos para un bloque de vídeo del fragmento de vídeo actual, basándose en los vectores de movimiento y otros elementos de sintaxis recibidos desde el módulo de descodificación por entropía 80. Los bloques predictivos se pueden producir a partir de una de las imágenes de referencia dentro de una de las listas de imágenes de referencia. El descodificador de vídeo 30 puede construir las listas de tramas de referencia, lista 0 y lista 1, usando técnicas de construcción por defecto en base a imágenes de referencia almacenadas en la memoria de imágenes de referencia 92.

20 **[0093]** El módulo de compensación de movimiento 82 determina la información de predicción para un bloque de vídeo del fragmento de vídeo actual analizando los vectores de movimiento y otros elementos de sintaxis, y usa la información de predicción para generar los bloques predictivos para bloque de vídeo actual que se está descodificando. Por ejemplo, el módulo de compensación de movimiento 82 puede usar algunos de los elementos de sintaxis recibidos para determinar un modo de predicción (por ejemplo, intrapredicción o interpredicción) usado para codificar los bloques de vídeo del fragmento de vídeo, un tipo de fragmento de interpredicción (por ejemplo, fragmento B, fragmento P o fragmento GPB). Adicionalmente, el módulo de compensación de movimiento 82 puede usar 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, 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.

25 **[0094]** El módulo de compensación de movimiento 82 también puede realizar la interpolación basándose en filtros de interpolación. El módulo de compensación de movimiento 82 puede usar filtros de interpolación como los usados por el codificador de vídeo 20 durante la codificación de los bloques de vídeo para calcular valores interpolados para píxeles de subentero de bloques de referencia. En este caso, el módulo de compensación de movimiento 82 puede determinar los filtros de interpolación usados por el codificador de vídeo 20 a partir de los elementos de sintaxis recibidos y usar los filtros de interpolación para generar bloques predictivos.

30 **[0095]** El módulo de cuantificación inversa 86 realiza la cuantificación inversa, es decir, la descuantificación, de los coeficientes de transformada cuantificados proporcionados en el flujo de bits y descodificados por el módulo de descodificación por entropía 80. El procedimiento 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, del mismo modo, un grado de cuantificación inversa que se debería aplicar. El módulo 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, con el fin de generar bloques residuales en el dominio del píxel.

35 **[0096]** Después de que el módulo de compensación de movimiento 82 genera el bloque predictivo para el bloque de vídeo actual basándose en los vectores de movimiento y otros elementos de sintaxis, el descodificador de vídeo 30 forma un bloque de vídeo descodificado sumando los bloques residuales del módulo de transformación inversa 88 y los correspondientes bloques predictivos generados por el módulo de compensación de movimiento 82. El sumador 90 representa el componente o los componentes que realizan esta operación de suma. Si se desea, también se pueden usar filtros de bucle (en el bucle de codificación o bien después del bucle de codificación) para suavizar las transiciones de píxeles o mejorar de otro modo la calidad de vídeo. El módulo de filtro 91 está destinado 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 según muestras (SAO). Aunque el módulo de filtro 91 que se muestra en la FIG. 3 es un filtro en bucle, en otras configuraciones, el módulo de filtro 91 se puede implementar como un filtro posbucle. Los bloques de vídeo descodificados en una trama o imagen dada se almacenan a continuación en la memoria de imágenes de referencia 92, que almacena imágenes de referencia usadas para una compensación de movimiento subsiguiente. La memoria de imágenes de referencia 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.

65

[0097] El descodificador de vídeo 30 de la FIG. 3 representa un ejemplo de un descodificador de vídeo configurado para tratar una imagen de acceso aleatorio limpio (CRA) como una imagen de acceso de enlace roto (BRA), tal como se describe en el presente documento.

5 **[0098]** La FIG. 4 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. En consecuencia, el dispositivo servidor 102 en general no recibe la indicación externa para tratar una imagen CRA como una imagen BLA. Sin embargo, el servidor 102 puede proporcionar una indicación externa 34 al dispositivo cliente 108 que hace que el dispositivo cliente 108 trate una imagen CRA recibida en el dispositivo de destino como una imagen BLA. Sin embargo, de forma similar, los dispositivos de encaminamiento 104A, 104B (dispositivos de encaminamiento 104) y el dispositivo de transcodificación 106 en general no reciben la indicación externa para tratar una imagen CRA como una imagen BLA, pero pueden proporcionar una indicación externa 34 al dispositivo cliente 108 para que el dispositivo cliente 108 trate una imagen CRA recibida en el dispositivo de destino como una imagen BLA. Algunos ejemplos descritos en el presente documento pueden incluir uno o más de los siguientes: dispositivos de red, servidores MANE, cachés de protocolo de transferencia de hipertexto (HTTP) o proxies web.

25 **[0099]** En algunos ejemplos, el dispositivo cliente 108 puede establecer el indicador después de recibir un mensaje de que se ha producido un cambio en la velocidad binaria de un flujo de bits. En consecuencia, el dispositivo cliente puede establecer el indicador en base al cambio de la velocidad binaria. En algunos ejemplos, un descodificador en el dispositivo cliente 108 puede descodificar una unidad NAL de fragmento codificado. Un módulo de predicción en un descodificador en el dispositivo cliente 108 puede analizar cada unidad NAL de fragmento codificado para identificar el tipo de unidad NAL. Adicionalmente, el módulo de predicción puede descodificar la unidad NAL de fragmento codificado en base al tipo de unidad NAL.

35 **[0100]** 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 *proxy* o de caché. Por lo tanto, en algunos ejemplos, los dispositivos de encaminamiento 104 pueden denominarse dispositivos proxy. 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.

45 **[0101]** Dispositivos de red, tales como dichos dispositivos de encaminamiento 104 y dispositivo de transcodificación 106, pueden implementar las técnicas de la presente 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, incluyendo las técnicas enumeradas en la porción REIVINDICACIONES de la presente divulgación. Además, los dispositivos de la FIG. 1 y el codificador mostrado en la FIG. 2 y el descodificador mostrado en la FIG. 3 son también dispositivos ejemplares que pueden estar configurados para realizar las técnicas de la presente divulgación, incluyendo las técnicas citadas en la porción REIVINDICACIONES de la presente divulgación.

55 **[0102]** La FIG. 5 es un diagrama de flujo que ilustra un procedimiento de ejemplo de acuerdo con uno o más ejemplos descritos en la presente divulgación. En un ejemplo, el descodificador de vídeo 30 u otro dispositivo recibe una indicación externa (500). El descodificador de vídeo 30 trata entonces una imagen de acceso aleatorio limpio (CRA) como una imagen de acceso de enlace roto (BLA) en base a la indicación externa (502). En algunos ejemplos, la indicación externa indica si se debe establecer un indicador en el descodificador de vídeo. En consecuencia, el descodificador de vídeo puede establecer el indicador en base a la indicación externa, como se analiza con mayor detalle con respecto a la FIG. 6. El descodificador o alguna funcionalidad interna, tal como una unidad de procesamiento de indicación externa o un módulo de predicción pueden comprobar entonces el indicador. En un ejemplo, el módulo de predicción puede tratar una imagen CRA como una imagen BLA en base a la indicación externa que indica que la imagen CRA se debe tratar como una imagen BLA en base al indicador.

65 **[0103]** La FIG. 6 es un diagrama de flujo que ilustra otro procedimiento de ejemplo de acuerdo con uno o más ejemplos descritos en la presente divulgación. En el ejemplo ilustrado, un descodificador de vídeo 30 recibe una indicación externa 70 de que se debe establecer un indicador 74 (600). En el ejemplo ilustrado, la indicación externa

70 se recibe mediante el descodificador de vídeo 30. En otros ejemplos, la indicación externa 70 puede recibirse y procesarse externamente al descodificador de vídeo 30.

5 **[0104]** La unidad de procesamiento de indicación externa 72 establece el primer indicador 74 en base a la indicación externa (602). A continuación, el primer indicador se pasa al módulo de predicción 81. En el ejemplo ilustrado, la unidad de procesamiento de indicación externa 72 está dentro del descodificador de vídeo 30. En otros ejemplos, la unidad de procesamiento de indicación externa 72 puede ser externa a y estar separada del descodificador de vídeo 30.

10 **[0105]** En el descodificador 30, el módulo de predicción 81 comprueba el indicador, y cuando el indicador se establece trata una imagen de acceso aleatorio limpio (CRA) como una imagen de acceso de enlace roto (BLA) (604). En un ejemplo, cuando se descodifica una unidad de capa de abstracción de red (NAL) de fragmento codificado, si el primer indicador es igual a "1" y un tipo de unidad NAL de una unidad NAL de fragmento codificado indica una imagen CRA (por ejemplo, el valor es igual a "4" o "5" conforme a HEVC WD7) la unidad de procesamiento de indicación externa 72 u otra unidad dentro del descodificador de vídeo 30 cambia el valor del tipo de unidad NAL para indicar una imagen BLA (por ejemplo, aumenta el valor del tipo de unidad NAL en 2 conforme a HEVC WD7). Adicionalmente, el módulo de predicción 81 establece el valor de un segundo indicador a 1. Si la imagen anterior en orden de descodificación es una imagen RAP y el rap_pic_id del fragmento actual es igual al rap_pic_id de la imagen anterior, aplica lo siguiente. Si la siguiente imagen en orden de descodificación no es una imagen RAP, se cambia el valor de rap_pic_id de un fragmento actual para que sea diferente de un rap_pic_id de la imagen anterior en orden de descodificación. De otro modo, el valor del rap_pic_id de la imagen actual se cambia para que sea un valor diferente del rap_pic_id de la imagen anterior y la siguiente imagen en orden de descodificación.

25 **[0106]** En otro ejemplo, en el descodificador de vídeo 30, el módulo de predicción 81 u otra unidad dentro del descodificador de vídeo 30 descodifica una unidad NAL de fragmento codificado. Si el primer indicador es igual a "1" y un tipo de unidad NAL de una unidad NAL de fragmento codificado indica una imagen CRA (por ejemplo, el valor es igual a "4" o "5" conforme a HEVC WD7) el módulo de predicción (u otra unidad) cambia el valor del tipo de unidad NAL para indicar una imagen BLA (por ejemplo, aumenta el valor del tipo de unidad NAL en 2 conforme a HEVC WD7). Posiblemente adicionalmente, si los mensajes SEI de temporización de imagen están presentes y los tiempos de salida de la DPB para todas las imágenes en una DPB son menores que el tiempo de salida de la DPB de una imagen actual, el módulo de predicción 81 u otra unidad establece un valor de un segundo indicador a 1. De otro modo, si el valor del segundo indicador se establece a "0" y si una imagen anterior en orden de descodificación es una imagen RAP y un rap_pic_id de un fragmento actual es igual al rap_pic_id de la imagen anterior, aplica lo siguiente. Si la siguiente imagen en orden de descodificación no es la imagen RAP, el módulo de predicción 81 u otra unidad cambia un valor de rap_pic_id de un fragmento actual para que sea diferente de un rap_pic_id de la imagen anterior en orden de descodificación. De otro modo, el valor del rap_pic_id del módulo de predicción u otra unidad cambia la imagen actual para que sea un valor diferente del rap_pic_id de la imagen anterior y de la siguiente imagen en orden de descodificación.

40 **[0107]** La FIG. 7 es un diagrama de flujo que ilustra un procedimiento de ejemplo de acuerdo con uno o más ejemplos descritos en la presente divulgación. Un dispositivo, tal como un dispositivo de red, por ejemplo, un MANE, recibe un flujo de bits que incluye una imagen CRA (700). El dispositivo de red determina que la imagen CRA se debe tratar como una imagen BLA (702). Por ejemplo, el dispositivo de red puede determinar que una imagen CRA se debe tratar como una imagen BLA para permitir la salida y/o visualización de más imágenes cuando la salida de todas las imágenes descodificadas antes en orden de descodificación que una imagen IDR o BLA se descarta después de descodificar la imagen IDR o BLA sin salida y/o visualización. A veces visualizar más de esas imágenes puede proporcionar una mejor experiencia de usuario. En consecuencia, el dispositivo de red transmite la imagen CRA y una indicación externa de que la imagen CRA se debe convertir en una imagen BLA (704).

55 **[0108]** La FIG. 8 es un diagrama de flujo que ilustra un funcionamiento ejemplar de un primer dispositivo que envía una indicación externa y acciones de respuesta de un segundo dispositivo que recibe la indicación externa. Un dispositivo de origen, tal como un dispositivo de red, por ejemplo, un MANE, recibe un flujo de bits que incluye una imagen CRA (800). El dispositivo de red determina que la imagen CRA se debe tratar como una imagen BLA (802). En consecuencia, el dispositivo de red transmite la imagen CRA y una indicación externa de que la imagen CRA se debe convertir en una imagen BLA (804). Un descodificador de vídeo 30 recibe la imagen CRA y la indicación externa 70 de que la imagen CRA se debe convertir en una imagen BLA (806).

60 **[0109]** La unidad de procesamiento de indicación externa 72 establece un indicador 74 en base a la indicación externa (808). A continuación, el indicador se puede pasar al módulo de predicción 81. En un ejemplo, la unidad de procesamiento de indicación externa 72 está dentro del descodificador de vídeo 30. En otros ejemplos, la unidad de procesamiento de indicación externa 72 puede ser externa a y estar separada del descodificador de vídeo 30. En el descodificador 30, el módulo de predicción 81 comprueba el indicador, y cuando el indicador se establece trata las imágenes CRA como una imagen BLA (810).

65

[0110] En el ejemplo de la FIG. 8, se usa un indicador para indicar que se ha recibido una indicación externa. En otros ejemplos, similares a la FIG. 5, el descodificador de vídeo 30 u otro dispositivo recibe una indicación externa y, a continuación, trata la imagen CRA como una imagen BLA en base a la indicación externa.

5 **[0111]** En un ejemplo, un descodificador cambia una imagen CRA a una imagen BLA como una función en el lado del descodificador. En la recepción o inferencia de dicho mensaje, una función del lado del descodificador puede realizar el cambio de la imagen CRA identificada a una imagen BLA del flujo de bits, antes de que la imagen codificada se envíe al descodificador para su descodificación.

10 **[0112]** Una imagen CRA se puede cambiar a una imagen BLA. Para cada unidad NAL de fragmento codificado, si nal_unit_type indica una imagen CRA, por ejemplo, el valor es igual a "4" o "5" conforme a HEVC WD7, aplica lo siguiente, (1) el valor de nal_unit_type se cambia para indicar una imagen BLA, por ejemplo, el valor se incrementa en 2, (2) el valor de no_output_of_prior_pics_flag se establece a 1, (3) si la imagen anterior en orden de descodificación es una imagen RAP y el rap_pic_id del fragmento actual es igual al rap_pic_id de la imagen anterior, aplica lo siguiente: (a) si la siguiente imagen en orden de descodificación no es una imagen RAP, el valor de rap_pic_id del fragmento actual se cambia para que sea diferente al rap_pic_id de la imagen anterior en orden de descodificación, pero todavía en el intervalo de valores permitidos del elemento de sintaxis, o (b) de otro modo (la siguiente imagen en orden de descodificación es una imagen RAP), el valor de rap_pic_id de la imagen actual se cambia para que sea un valor diferente al rap_pic_id de la imagen anterior y la siguiente imagen en orden de descodificación, pero todavía en el intervalo de valores permitidos del elemento de sintaxis.

20 **[0113]** A continuación se describirá la gestión de una imagen CRA como una imagen CRA que inicia un flujo de bits. Una indicación de que una imagen CRA particular se debe gestionar como una imagen BLA, como se describe anteriormente, también se puede cambiar a o interpretar como una indicación de que una imagen CRA particular se debe gestionar como una imagen CRA que es la primera imagen en un flujo de bits, siempre que los cambios descritos a continuación se realicen en la especificación del borrador de HEVC.

25 **[0114]** En un ejemplo, la variable CralsFirstPicFlag está asociada con cada imagen CRA. El valor de CralsFirstPicFlag para algunas imágenes CRA se puede especificar mediante medios externos. Si una imagen CRA es la primera imagen en el flujo de bits, el valor de CralsFirstPicFlag para la imagen CRA se establece a 1, independientemente del valor indicado por la indicación externa (si está presente). De otro modo, cuando el valor de CralsFirstPicFlag para la imagen CRA no se especifica mediante medios externos, se establece a "0".

30 **[0115]** Cuando se descodifica (incluyendo el análisis) cada unidad NAL de fragmento codificado, si CralsFirstPicFlag es igual a "1" y nal_unit_type es igual a "4" o "5", el valor de no_output_of_prior_pics_flag se puede establecer a 1. Si la imagen anterior en orden de descodificación es una imagen RAP y el rap_pic_id del fragmento actual es igual al rap_pic_id de la imagen anterior, entonces, si la siguiente imagen en orden de descodificación no es una imagen RAP, el valor de rap_pic_id del fragmento actual se cambia para que sea diferente del rap_pic_id de la imagen anterior en orden de descodificación, pero todavía en el intervalo de valores permitidos del elemento de sintaxis. De otro modo (la siguiente imagen en orden de descodificación es una imagen RAP), el valor de rap_pic_id de la imagen actual se cambia para que sea un valor diferente al rap_pic_id de la imagen anterior y la siguiente imagen en orden de descodificación, pero todavía en el intervalo de valores permitidos del elemento de sintaxis.

35 **[0116]** De forma alternativa, en lugar de establecer el valor de no_output_of_prior_pics_flag a 1, el módulo de predicción 81 puede, si los mensajes SEI de temporización de imagen están presentes y los tiempos de salida de la DPB para todas las imágenes en la DPB son menores que el tiempo de salida de la DPB de la imagen actual, establecer el valor de no_output_of_prior_pics_flag a 1, de otro modo, el valor de no_output_of_prior_pics_flag se establece a "0".

40 **[0117]** En otros ejemplos, diversas definiciones de recuento de orden de imágenes, imágenes etiquetadas para descarte (TFD), se pueden cambiar en el HEVC WD9 u otros borradores de trabajo del estándar. En consecuencia, las definiciones proporcionadas a continuación pueden ser diferentes del estándar. Estas definiciones pueden no aplicarse a algunos o todos los ejemplos descritos en el presente documento.

45 **[0118]** En algunos ejemplos, una secuencia de vídeo codificado es una secuencia de unidades de acceso que puede incluir, en orden de descodificación, una unidad de acceso CRA que puede tener un CralsFirstPicFlag igual a 1, una unidad de acceso IDR o una unidad de acceso BLA, seguida de cero o más unidades de acceso no IDR y no BLA, incluyendo todas las unidades de acceso posteriores hasta pero sin incluir alguna unidad de acceso IDR o BLA posterior.

50 **[0119]** En algunos ejemplos, un recuento de orden de imágenes es una variable que se puede asociar con cada imagen codificada y tiene un valor que aumenta con el aumento de la posición de la imagen en orden de salida con respecto a una de las siguientes imágenes codificadas: (1) la imagen IDR anterior en orden de descodificación, si la hay (2) la imagen BLA anterior en orden de descodificación, si la hay, y (3) la imagen CRA anterior en orden de descodificación, si la hay, y, en algunos ejemplos, si la imagen CRA anterior tiene CralsFirstPicFlag igual a 1.

- 5 **[0120]** En algunos ejemplos, si está presente más de una de las imágenes codificadas anteriores, el recuento de orden de imágenes es relativo a la última de dichas imágenes codificadas en orden de descodificación. Una imagen etiquetada para descarte (TFD): Una imagen codificada para la cual cada fragmento tiene `nal_unit_type` igual a 2; una imagen TFD está asociada con la imagen CRA o la imagen BLA anterior en orden de descodificación y precede a la imagen asociada en orden de salida; cuando la imagen asociada es una imagen BLA, o cuando la imagen asociada es una imagen CRA que puede tener un `CraIsFirstPicFlag` igual a 1, la imagen TFD puede ser no correctamente descodificable y no se envía.
- 10 **[0121]** En algunos ejemplos, la semántica de `no_output_of_prior_pics_flag` se puede cambiar de modo que `no_output_of_prior_pics_flag` especifique cómo se tratan las imágenes descodificadas previamente en la memoria intermedia de imágenes descodificadas después de descodificar una imagen CRA con `CraIsFirstPicFlag` igual a "1" o una imagen IDR o BLA.
- 15 **[0122]** En algunos ejemplos, cuando la imagen CRA con `CraIsFirstPicFlag` igual a "1" o la imagen IDR o BLA es la primera imagen en el flujo de bits, el valor de `no_output_of_prior_pics_flag` no tiene efecto en el proceso de descodificación. Cuando la imagen CRA con `CraIsFirstPicFlag` igual a "1" o la imagen IDR o BLA no es la primera imagen en el flujo de bits y el valor de `pic_width_in_luma_samples` o `pic_height_in_luma_samples` o `sps_max_dec_pic_buffering[sps_max_temporal_layers_minus1]` derivado del conjunto de parámetros de secuencia activo es diferente del valor de `pic_width_in_luma_samples` o `pic_height_in_luma_samples` o `sps_max_dec_pic_buffering[sps_max_temporal_layers_minus1]` derivado del conjunto de parámetros de secuencia activo para la imagen anterior, el descodificador puede (pero no debe) inferir `no_output_of_prior_pics_flag` igual a "1", independientemente del valor real de `no_output_of_prior_pics_flag`.
- 20 **[0123]** En algunos ejemplos, se puede hacer un cambio a lo siguiente en la subcláusula 8.1 de HEVC WD7, por ejemplo, cambiar: si la primera imagen codificada en el flujo de bits es una imagen CRA, y la imagen actual es una imagen TFD asociada con la imagen CRA, o si la imagen RAP anterior que precede a la imagen actual en orden de descodificación es una imagen BLA y la imagen actual es una imagen TFD asociada con la imagen BLA, `PicOutputFlag` se establece igual a "0" y se invoca el proceso de descodificación para generar imágenes de referencia no disponibles especificado en la subcláusula 8.3.3 (solo es necesario invocarlo para un fragmento de una imagen) para: si una imagen CRA tiene `CraIsFirstPicFlag` igual a 1, y la imagen actual es una imagen TFD asociada con la imagen CRA, o si la imagen RAP anterior que precede a la imagen actual en orden de descodificación es una imagen BLA y la imagen actual es una imagen TFD asociada con la imagen BLA, `PicOutputFlag` se establece igual a "0" y se invoca el proceso de descodificación para generar imágenes de referencia no disponibles especificado en la subcláusula 8.3.3 (solo es necesario invocarlo para un fragmento de una imagen).
- 25 **[0124]** En algunos ejemplos, se puede hacer un cambio de lo siguiente en la subcláusula 8.3.1 de HEVC WD7, por ejemplo, cambiar: la imagen actual es una imagen CRA y es la primera imagen codificada en el flujo de bits a la imagen actual es una imagen CRA con `CraIsFirstPicFlag` igual a 1.
- 30 **[0125]** En algunos ejemplos, se puede hacer un cambio a lo siguiente en la subcláusula 8.3.1 de HEVC WD7, por ejemplo, cambiar: si la imagen actual es una imagen IDR o BLA, o si la primera imagen codificada en el flujo de bits es una imagen CRA y la imagen actual es la primera imagen codificada en el flujo de bits, `PicOrderCntMsb` se establece igual a "0". De lo contrario, `PicOrderCntMsb` se deriva según lo especificado por el siguiente pseudocódigo a si la imagen actual es una imagen IDR o BLA, o una imagen CRA con `CraIsFirstPicFlag` igual a 1, `PicOrderCntMsb` se establece igual a "0". De lo contrario, `PicOrderCntMsb` se deriva como se especifica en el siguiente pseudocódigo.
- 35 **[0126]** En algunos ejemplos, se puede hacer un cambio de lo siguiente en la subcláusula 8.3.2 de HEVC WD7, por ejemplo, cambiar NOTA 4 - Puede haber una o más imágenes de referencia que se incluyen en el conjunto de imágenes de referencia pero que no están presentes en la memoria intermedia de imágenes descodificadas. Las entradas en `RefPicSetStFoll` o `RefPicSetLtFoll` que son iguales a "sin imagen de referencia" deberían ignorarse. A menos que cualquiera de las dos condiciones siguientes sea verdadera, se debe inferir una pérdida de imagen involuntaria para cada entrada en `RefPicSetStCurrBefore`, `RefPicSetStCurrAfter` y `RefPicSetLtCurr` que sea igual a "sin imagen de referencia": a) la primera imagen codificada en el flujo de bits es una imagen CRA y la imagen codificada actual es una imagen TFD asociada con la primera imagen codificada en el flujo de bits; b) la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación es una imagen BLA y la imagen codificada actual es una imagen TFD asociada con la imagen BLA a NOTA 4 - Puede haber una o más imágenes de referencia que se incluyen en el conjunto de imágenes de referencia pero no están presentes en la memoria intermedia de imágenes descodificadas. Las entradas en `RefPicSetStFoll` o `RefPicSetLtFoll` que son iguales a "sin imagen de referencia" deberían ignorarse. A menos que la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación sea una imagen CRA con `CraIsFirstPicFlag` igual a "1" o una imagen BLA, y la imagen codificada actual sea una imagen TFD asociada con la imagen RAP anterior, se debe inferir una pérdida de imagen involuntaria para cada entrada en `RefPicSetStCurrBefore`, `RefPicSetStCurrAfter` y `RefPicSetLtCurr` que es igual a "sin imagen de referencia".
- 40 **[0124]** En algunos ejemplos, se puede hacer un cambio de lo siguiente en la subcláusula 8.3.1 de HEVC WD7, por ejemplo, cambiar: la imagen actual es una imagen CRA y es la primera imagen codificada en el flujo de bits a la imagen actual es una imagen CRA con `CraIsFirstPicFlag` igual a 1.
- 45 **[0125]** En algunos ejemplos, se puede hacer un cambio a lo siguiente en la subcláusula 8.3.1 de HEVC WD7, por ejemplo, cambiar: si la imagen actual es una imagen IDR o BLA, o si la primera imagen codificada en el flujo de bits es una imagen CRA y la imagen actual es la primera imagen codificada en el flujo de bits, `PicOrderCntMsb` se establece igual a "0". De lo contrario, `PicOrderCntMsb` se deriva según lo especificado por el siguiente pseudocódigo a si la imagen actual es una imagen IDR o BLA, o una imagen CRA con `CraIsFirstPicFlag` igual a 1, `PicOrderCntMsb` se establece igual a "0". De lo contrario, `PicOrderCntMsb` se deriva como se especifica en el siguiente pseudocódigo.
- 50 **[0126]** En algunos ejemplos, se puede hacer un cambio de lo siguiente en la subcláusula 8.3.2 de HEVC WD7, por ejemplo, cambiar NOTA 4 - Puede haber una o más imágenes de referencia que se incluyen en el conjunto de imágenes de referencia pero que no están presentes en la memoria intermedia de imágenes descodificadas. Las entradas en `RefPicSetStFoll` o `RefPicSetLtFoll` que son iguales a "sin imagen de referencia" deberían ignorarse. A menos que cualquiera de las dos condiciones siguientes sea verdadera, se debe inferir una pérdida de imagen involuntaria para cada entrada en `RefPicSetStCurrBefore`, `RefPicSetStCurrAfter` y `RefPicSetLtCurr` que sea igual a "sin imagen de referencia": a) la primera imagen codificada en el flujo de bits es una imagen CRA y la imagen codificada actual es una imagen TFD asociada con la primera imagen codificada en el flujo de bits; b) la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación es una imagen BLA y la imagen codificada actual es una imagen TFD asociada con la imagen BLA a NOTA 4 - Puede haber una o más imágenes de referencia que se incluyen en el conjunto de imágenes de referencia pero no están presentes en la memoria intermedia de imágenes descodificadas. Las entradas en `RefPicSetStFoll` o `RefPicSetLtFoll` que son iguales a "sin imagen de referencia" deberían ignorarse. A menos que la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación sea una imagen CRA con `CraIsFirstPicFlag` igual a "1" o una imagen BLA, y la imagen codificada actual sea una imagen TFD asociada con la imagen RAP anterior, se debe inferir una pérdida de imagen involuntaria para cada entrada en `RefPicSetStCurrBefore`, `RefPicSetStCurrAfter` y `RefPicSetLtCurr` que es igual a "sin imagen de referencia".
- 55 **[0126]** En algunos ejemplos, se puede hacer un cambio de lo siguiente en la subcláusula 8.3.2 de HEVC WD7, por ejemplo, cambiar NOTA 4 - Puede haber una o más imágenes de referencia que se incluyen en el conjunto de imágenes de referencia pero que no están presentes en la memoria intermedia de imágenes descodificadas. Las entradas en `RefPicSetStFoll` o `RefPicSetLtFoll` que son iguales a "sin imagen de referencia" deberían ignorarse. A menos que cualquiera de las dos condiciones siguientes sea verdadera, se debe inferir una pérdida de imagen involuntaria para cada entrada en `RefPicSetStCurrBefore`, `RefPicSetStCurrAfter` y `RefPicSetLtCurr` que sea igual a "sin imagen de referencia": a) la primera imagen codificada en el flujo de bits es una imagen CRA y la imagen codificada actual es una imagen TFD asociada con la primera imagen codificada en el flujo de bits; b) la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación es una imagen BLA y la imagen codificada actual es una imagen TFD asociada con la imagen BLA a NOTA 4 - Puede haber una o más imágenes de referencia que se incluyen en el conjunto de imágenes de referencia pero no están presentes en la memoria intermedia de imágenes descodificadas. Las entradas en `RefPicSetStFoll` o `RefPicSetLtFoll` que son iguales a "sin imagen de referencia" deberían ignorarse. A menos que la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación sea una imagen CRA con `CraIsFirstPicFlag` igual a "1" o una imagen BLA, y la imagen codificada actual sea una imagen TFD asociada con la imagen RAP anterior, se debe inferir una pérdida de imagen involuntaria para cada entrada en `RefPicSetStCurrBefore`, `RefPicSetStCurrAfter` y `RefPicSetLtCurr` que es igual a "sin imagen de referencia".
- 60 **[0126]** En algunos ejemplos, se puede hacer un cambio de lo siguiente en la subcláusula 8.3.2 de HEVC WD7, por ejemplo, cambiar NOTA 4 - Puede haber una o más imágenes de referencia que se incluyen en el conjunto de imágenes de referencia pero que no están presentes en la memoria intermedia de imágenes descodificadas. Las entradas en `RefPicSetStFoll` o `RefPicSetLtFoll` que son iguales a "sin imagen de referencia" deberían ignorarse. A menos que cualquiera de las dos condiciones siguientes sea verdadera, se debe inferir una pérdida de imagen involuntaria para cada entrada en `RefPicSetStCurrBefore`, `RefPicSetStCurrAfter` y `RefPicSetLtCurr` que sea igual a "sin imagen de referencia": a) la primera imagen codificada en el flujo de bits es una imagen CRA y la imagen codificada actual es una imagen TFD asociada con la primera imagen codificada en el flujo de bits; b) la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación es una imagen BLA y la imagen codificada actual es una imagen TFD asociada con la imagen BLA a NOTA 4 - Puede haber una o más imágenes de referencia que se incluyen en el conjunto de imágenes de referencia pero no están presentes en la memoria intermedia de imágenes descodificadas. Las entradas en `RefPicSetStFoll` o `RefPicSetLtFoll` que son iguales a "sin imagen de referencia" deberían ignorarse. A menos que la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación sea una imagen CRA con `CraIsFirstPicFlag` igual a "1" o una imagen BLA, y la imagen codificada actual sea una imagen TFD asociada con la imagen RAP anterior, se debe inferir una pérdida de imagen involuntaria para cada entrada en `RefPicSetStCurrBefore`, `RefPicSetStCurrAfter` y `RefPicSetLtCurr` que es igual a "sin imagen de referencia".
- 65 **[0126]** En algunos ejemplos, se puede hacer un cambio de lo siguiente en la subcláusula 8.3.2 de HEVC WD7, por ejemplo, cambiar NOTA 4 - Puede haber una o más imágenes de referencia que se incluyen en el conjunto de imágenes de referencia pero que no están presentes en la memoria intermedia de imágenes descodificadas. Las entradas en `RefPicSetStFoll` o `RefPicSetLtFoll` que son iguales a "sin imagen de referencia" deberían ignorarse. A menos que cualquiera de las dos condiciones siguientes sea verdadera, se debe inferir una pérdida de imagen involuntaria para cada entrada en `RefPicSetStCurrBefore`, `RefPicSetStCurrAfter` y `RefPicSetLtCurr` que sea igual a "sin imagen de referencia": a) la primera imagen codificada en el flujo de bits es una imagen CRA y la imagen codificada actual es una imagen TFD asociada con la primera imagen codificada en el flujo de bits; b) la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación es una imagen BLA y la imagen codificada actual es una imagen TFD asociada con la imagen BLA a NOTA 4 - Puede haber una o más imágenes de referencia que se incluyen en el conjunto de imágenes de referencia pero no están presentes en la memoria intermedia de imágenes descodificadas. Las entradas en `RefPicSetStFoll` o `RefPicSetLtFoll` que son iguales a "sin imagen de referencia" deberían ignorarse. A menos que la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación sea una imagen CRA con `CraIsFirstPicFlag` igual a "1" o una imagen BLA, y la imagen codificada actual sea una imagen TFD asociada con la imagen RAP anterior, se debe inferir una pérdida de imagen involuntaria para cada entrada en `RefPicSetStCurrBefore`, `RefPicSetStCurrAfter` y `RefPicSetLtCurr` que es igual a "sin imagen de referencia".

[0127] En algunos ejemplos, se puede hacer un cambio en la subcláusula 8.3.2 de HEVC WD7, por ejemplo, cambiar: Salvo que alguna de las siguientes condiciones sea verdadera, no habrá ninguna entrada en RefPicSetStCurrBefore, RefPicSetStCurrAfter o RefPicSetLtCurr que sea igual a "sin imagen de referencia": a) la primera imagen codificada en el flujo de bits es una imagen CRA y la imagen codificada actual es una imagen TFD asociada con la primera imagen codificada en el flujo de bits; b) la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación es una imagen BLA y la imagen codificada actual es una imagen TFD asociada con la imagen BLA a Salvo que la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación sea una imagen CRA con CralsFirstPicFlag igual a "1" o una imagen BLA, y la imagen codificada actual sea una imagen TFD asociada con la imagen RAP anterior, no habrá ninguna entrada en RefPicSetStCurrBefore, RefPicSetStCurrAfter o RefPicSetLtCurr que sea igual a "ninguna imagen de referencia".

[0128] En algunos ejemplos, un cambio de los primeros tres párrafos en la subcláusula 8.3.3.1 de HEVC WD7 se puede realizar como sigue: este proceso se invoca una vez por imagen codificada, después de la invocación del proceso de descodificación para el conjunto de imágenes de referencia como se especifica en la subcláusula 8.3.2, cuando la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación es una imagen CRA con CralsFirstPicFlag igual a "1" o una imagen BLA, y la imagen codificada actual es una imagen TFD asociada con la imagen RAP anterior. NOTA 1 - La especificación completa en el presente documento del proceso de descodificación para imágenes TFD asociadas con una imagen CRA al comienzo del flujo de bits o para imágenes TFD asociadas con una imagen BLA solo se incluye con el propósito de especificar restricciones en el contenido de la sintaxis permitida de dichas imágenes. En los descodificadores reales, cualquier imagen TFD asociada con una imagen CRA en el comienzo del flujo de bits o cualquier imagen TFD asociada con una imagen BLA sencillamente puede ignorarse (eliminarse del flujo de bits y descartarse), ya que no están especificadas para la salida y no tienen ningún efecto en el proceso de descodificación de cualquier otra imagen que esté especificada para la salida. Cuando la imagen RAP anterior que precede a la imagen codificada actual en orden de descodificación es una imagen CRA con CralsFirstPicFlag igual a "1" o una imagen BLA, y la imagen codificada actual es una imagen TFD asociada con la imagen RAP anterior, se aplica lo siguiente.

[0129] En algunos ejemplos, se puede hacer un cambio en la subcláusula C.4 de HEVC WD7, por ejemplo, cambiar: NOTA 1 - Esta restricción garantiza la capacidad de descodificación de una imagen TFD si su imagen RAP asociada es una imagen CRA y si esa imagen CRA no es la primera imagen codificada en el flujo de bits a NOTA 1 - Esta restricción garantiza la capacidad de descodificación de una imagen TFD si su imagen RAP asociada es una imagen CRA y si esa imagen CRA tiene CralsFirstPicFlag igual a "0".

[0130] En algunos ejemplos, un cambio en el tercer párrafo de la subcláusula C.3.1 de HEVC WD7 se puede hacer de la siguiente manera: Si la imagen actual es una imagen CRA con CralsFirstPicFlag igual a "1" o una imagen IDR o BLA, aplica lo siguiente cuando la imagen CRA con CralsFirstPicFlag igual a "1" o la imagen IDR o BLA no es la imagen descodificada y el valor de pic_width_in_luma_samples o pic_height_in_luma_samples o sps_max_dec_pic_buffering[i] para cualquier valor posible de i derivado del conjunto de parámetros de secuencia activo es diferente del valor de pic_width_in_luma_samples o pic_height_in_luma_samples o sps_max_dec_pic_buffering[i] derivado del conjunto de parámetros de secuencia que estaba activo para la imagen precedente, respectivamente, el HRD infiere que no_output_of_prior_pics_flag es igual a "1", independientemente del valor real de no_output_of_prior_pics_flag. NOTA 1: las implementaciones del descodificador deben intentar gestionar los cambios en la imagen o el tamaño de la DPB más fácilmente que el HRD con respecto a cambios en pic_width_in_luma_samples, pic_height_in_luma_samples o sps_max_dec_pic_buffering[i]. Cuando no_output_of_prior_pics_flag es igual a "1" o se infiere que es igual a 1, todas las memorias intermedias de almacenamiento de imágenes en la DPB se vacían sin proporcionar las imágenes que contienen, y la completitud de la DPB se establece a "0".

[0131] En algunos ejemplos, se puede hacer un cambio de la subcláusula completa C.5.2 de HEVC WD7 como sigue: la eliminación de imágenes de la DPB antes de descodificar la imagen actual (pero después de analizar la cabecera de fragmento del primer fragmento de la imagen actual) se produce instantáneamente cuando la primera unidad de descodificación de la unidad de acceso que contiene la imagen actual se elimina de la CPB y procede como sigue. Se invoca el proceso de descodificación para el conjunto de imágenes de referencia como se especifica en la subcláusula 8.3.2. Si la imagen actual es una imagen CRA con CralsFirstPicFlag igual a "1" o una imagen IDR o BLA, aplica lo siguiente. Cuando la imagen CRA con CralsFirstPicFlag igual a "1" o la imagen IDR o BLA no es la primera imagen descodificada y el valor de pic_width_in_luma_samples o pic_height_in_luma_samples o sps_max_dec_pic_buffering[i] para cualquier valor posible de i derivado del conjunto de parámetros de secuencia activo es diferente del valor de pic_width_in_luma_samples o pic_height_in_luma_samples o sps_max_dec_pic_buffering[i] derivado del conjunto de parámetros de secuencia que estuvo activo para la imagen anterior, respectivamente, el HRD infiere que no_output_of_prior_pics_flag es igual a 1, independientemente del valor real de no_output_of_prior_pics_flag. NOTA: las implementaciones del descodificador deben intentar gestionar los cambios en la imagen o el tamaño de la DPB más fácilmente que el HRD con respecto a cambios en pic_width_in_luma_samples pic_height_in_luma_samples o sps_max_dec_pic_buffering[i]. Cuando no_output_of_prior_pics_flag es igual a "1" o se infiere que es igual a 1, todas las memorias intermedias de almacenamiento de imágenes en la DPB se vacían sin proporcionar las

imágenes que contienen. De otro modo, las memorias intermedias de almacenamiento de imágenes que contienen una imagen que se marca como "no necesaria para la salida" y "no usada para referencia" pueden estar vacías (sin salida). Cuando una o más de las siguientes condiciones son verdaderas, el proceso de "choque" especificado en la subcláusula C.5.2.1 se invoca repetidamente hasta que haya una memoria intermedia de almacenamiento de imágenes vacía para almacenar la imagen descodificada actual. El número de imágenes en la DPB que están marcadas como "necesarias para la salida" es mayor que `sps_num_reorder_pics[temporal_id]`. El número de imágenes en la DPB con `temporal_id` menor o igual que el `temporal_id` de la imagen actual es igual a `sps_max_dec_pic_buffering[temporal_id]`.

[0132] En algunos ejemplos, cuando la imagen actual es una imagen CRA con `CraIsFirstPicFlag` igual a "1" o una imagen IDR o BLA para la cual `no_output_of_prior_pics_flag` no es igual a "1" y no se infiere que es igual a 1, se realizan los dos pasos siguientes. Las memorias intermedias de almacenamiento de imágenes que contienen una imagen que se marca como "no necesaria para la salida" y "no usada para referencia" están vacías (sin salida). Todas las memorias intermedias de almacenamiento de imágenes no vacías en la DPB se vacían invocando repetidamente el proceso de "choque" especificado en la subcláusula C.5.2.1.

[0133] Algunos ejemplos pueden incluir un proceso de "choque". El proceso de "choque" se puede invocar en los siguientes casos: (1) la imagen actual es una imagen CRA con `CraIsFirstPicFlag` igual a "1" o una imagen IDR o BLA y `no_output_of_prior_pics_flag` no es igual a "1" y no se infiere que sea igual a 1, como se especifica en la subcláusula C.5.2, (2) el número de imágenes en la DPB que están marcadas como "necesarias para la salida" es mayor que `sps_num_reorder_pics[temporal_id]`, como se especifica en la subcláusula C.5.2, y (3) el número de imágenes en la DPB con `temporal_id` menor o igual que el `temporal_id` de la imagen actual es igual a `sps_max_dec_pic_buffering[temporal_id]`, como se especifica en la subcláusula C.5.2.

[0134] El proceso de "choque" puede incluir las siguientes etapas ordenadas: (1) la imagen que es la primera para la salida se selecciona como la que tiene el valor más pequeño de `PicOrderCntVal` de todas las imágenes en la DPB marcadas como "necesarias para la salida", (2) la imagen se recorta, usando el rectángulo de recorte especificado en el conjunto de parámetros de secuencia activo para la imagen, la imagen recortada se envía y la imagen se marca como "no necesaria para la salida", (3) si la memoria intermedia de almacenamiento de imágenes que incluye la imagen recortada y la salida contiene una imagen marcada como "no usada para referencia", la memoria intermedia de almacenamiento de imágenes se vacía.

[0135] En algunos ejemplos, con los cambios anteriores de la especificación del borrador de HEVC, es posible eliminar aún más todos los textos para admitir imágenes BLA.

[0136] A continuación se describirá una salida mejorada de imágenes. En un ejemplo, se propone que `no_output_of_prior_pics_flag` se cambie a `output_all_prior_pics_flag`, este indicador igual a "1" tiene el significado equivalente cuando `no_output_of_prior_pics_flag` es igual a "0". Además, cuando este indicador es igual a "0", el número de imágenes anteriores que se pueden usar para la salida/visualización se señala como `num_output_pics`. `num_output_pics` se puede señalar como `u(v)`, este elemento de sintaxis está en el intervalo de 0 a `MaxDpbSize`, exclusivo. Las `num_output_pics` imágenes que se enviarán/visualizarán son las imágenes que tienen órdenes de visualización más próximas a la imagen BLA o IDR y en el primer flujo de bits. `num_output_pics` puede estar relacionado con el número de imágenes principales que no es necesario enviar.

[0137] De forma alternativa, `num_output_pics` se puede señalar como `ue(v)`. De forma alternativa, `no_output_of_prior_pics_flag`, `output_all_prior_pics_flag` o `num_output_pics` no se señalizan y `num_prior_discard_pics` se señala directamente como `u(v)` o `ue(v)`, `num_prior_discard_pics` está en el intervalo de 0 a `MaxDpbSize`, exclusivo. Indica el número de imágenes anteriores que se descartarán. Las `num_prior_discard_pics` imágenes que se descartarán (por tanto, no se visualizarán) son las imágenes que tienen órdenes de visualización más lejanas a la imagen BLA o IDR y en el primer flujo de bits.

[0138] De forma alternativa, se puede añadir un mensaje SEI durante el empalme para indicar la memoria adicional, en términos de número de tramas en el primer flujo de bits, necesaria para visualizar todas las imágenes en el primer flujo de bits que no se han visualizado.

[0139] A continuación se describirá la señalización de la temporización de imagen. La indicación de una o más de información de temporización diferente, por ejemplo, el tiempo de presentación más temprano (es decir, el tiempo de salida de la DPB más temprano) y el valor del recuento de orden de imagen más pequeño de todas las imágenes TFD asociadas con una imagen BLA o CRA, se puede incluir en el flujo de bits. La información se puede incluir en uno o más de la cabecera del fragmento y un mensaje SEI (por ejemplo, el mensaje SEI de punto de recuperación o el mensaje SEI de período de almacenamiento intermedio o el mensaje SEI de temporización de imagen). Se pueden incluir uno o más de los siguientes elementos de sintaxis en la cabecera de fragmento de una imagen RAP o un mensaje SEI asociados con una imagen RAP para señalar la información: (1) `delta_earliest_presentation_time`, que indica la diferencia entre el tiempo de salida de la DPB de la imagen RAP y el tiempo de salida de la DPB más temprano de cualquier imagen cuando la imagen RAP es la primera imagen en el flujo de bits (es decir, el tiempo de salida de la DPB más temprano de todos los DLP asociados con la imagen

RAP), en unidades de tics de reloj como se especifica en el Anexo C de HEVC WD7. El elemento de sintaxis puede estar codificado en $u(v)$, y el número de bits usados para representar el elemento de sintaxis es $cpb_removal_delay_length_minus1 + 1$ bits. El valor "0" indica que la imagen RAP no tiene DLP asociados, (2) $\delta_{earliest_poc}$, que indica la diferencia entre el valor de $PicOrderCntVal$ de la imagen RAP y el valor de $PicOrderCntVal$ más pequeño de cualquier imagen cuando la imagen RAP es la primera imagen en el flujo de bits (es decir, el valor de $PicOrderCntVal$ más temprano y más pequeño de todos los DLP asociados con la imagen RAP). El elemento de sintaxis puede estar codificado en $ue(v)$, y el intervalo de valores puede ser de 0 a $MaxPicOrderCntLsb/2 - 1$, inclusive.

5
10
15
20
[0140] En uno o más ejemplos, las funciones descritas se pueden implementar en hardware, software, firmware o en cualquier combinación de los mismos. Si se implementan en software, las funciones se pueden almacenar en o transmitir a través de un medio legible por ordenador como una o más instrucciones o código, y ejecutarse por una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que correspondan a un medio tangible tal como medios de almacenamiento de datos, o medios de comunicación incluyendo cualquier medio que facilite la transferencia de un programa informático de un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder, en general, a (1) medios de almacenamiento tangibles legibles por ordenador que sean no transitorios o a (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que se pueda acceder desde uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en la presente divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

25
[0141] En algunos ejemplos, el mensaje o el elemento de sintaxis incluyen uno de los siguientes: (1) un $\delta_{earliest_presentation_time}$, que indica una diferencia de tiempo asociada con una o más imágenes CRA tratadas como imágenes BRA; o (2) un $\delta_{earliest_poc}$, que indica una diferencia en el valor del orden de imagen asociada con la una o más imágenes CRA tratadas como imágenes BRA.

30
[0142] En otros ejemplos adicionales, la presente 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.

35
40
[0143] En un ejemplo, una imagen CRA se puede tratar como imágenes BRA. El descodificador de vídeo 30 puede cambiar un valor de un tipo de capa de abstracción de red (NAL), establecer un valor que controle la salida de imágenes anteriores y cambiar un valor de identificación (ID) de imagen asociado con una imagen siguiente. El descodificador de vídeo 30 puede recibir un elemento de sintaxis que indica una operación de conmutación. El elemento de sintaxis o mensaje que forma parte de un flujo de bits comprimido y la operación de conmutación indica a un descodificador que trate una o más imágenes CRA como imágenes BRA. A continuación, el descodificador puede descodificar el flujo de bits basándose, en parte, en el elemento de sintaxis.

45
[0144] En un ejemplo, un codificador de vídeo 20 puede generar un elemento de sintaxis o un mensaje para indicar una operación de conmutación. La operación de conmutación indica a un descodificador que trate una o más imágenes CRA como imágenes BRA. El codificador de vídeo 20 puede enviar el elemento de sintaxis a un dispositivo de descodificación como parte de un flujo de bits comprimido.

50
55
60
[0145] A modo de ejemplo, y no de limitación, dichos medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento en disco óptico, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que se pueda usar para almacenar el código de programa deseado en forma de instrucciones o estructuras de datos y al que se pueda acceder mediante un ordenador. También, cualquier conexión recibe apropiadamente la denominación de medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde una página web, un servidor u otra fuente remota usando un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, la DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas están incluidos en la definición de medio. Sin embargo, se debe entender 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 refieren a medios de almacenamiento tangibles no transitorios. El término disco, como se usa en el presente documento, incluye disco compacto (CD), disco láser, disco óptico, disco versátil digital (DVD), disco flexible y disco Blu-ray, donde unos discos reproducen normalmente los datos magnéticamente, mientras que otros discos reproducen datos ópticamente con láseres. Las combinaciones de los anteriores también se deben incluir dentro del alcance de los medios legibles por ordenador.

65
[0146] Uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados específicos de la aplicación (ASIC), matrices lógicas programables *in situ* (FPGA) u otros circuitos lógicos integrados o discretos equivalentes pueden ejecutar

instrucciones. Por consiguiente, el término "procesador", como se usa en el presente documento, se puede referir a cualquiera de las estructuras anteriores o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento se puede proporcionar dentro de módulos de hardware y/o de software dedicados configurados para codificar y descodificar, o incorporar en un códec combinado. Además, las técnicas se podrían implementar por completo en uno o más circuitos o elementos lógicos.

5

[0147] Las técnicas de esta divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluyendo un teléfono inalámbrico, un circuito integrado (IC) o un conjunto de IC (por ejemplo, un conjunto de chips). En la presente divulgación se describen diversos componentes, módulos o unidades para destacar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no se requiere necesariamente su realización mediante diferentes unidades de hardware. En su lugar, como se describe anteriormente, diversas unidades se pueden combinar en una unidad de hardware de códec o proporcionar mediante un grupo de unidades de hardware interoperativas, que incluya uno o más procesadores como se describe anteriormente, junto con software y/o firmware adecuados.

10

15

[0148] 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 de codificación de vídeo de alta eficiencia, HEVC, comprendiendo el procedimiento:

5

recibir (500, 600, 806) una indicación externa (70) en un descodificador de vídeo (30) para indicar una operación de conmutación, en el que la indicación externa indica si el descodificador de vídeo debe establecer un valor de un indicador a un valor predeterminado o a un valor establecido;

10

recibir un flujo de bits de vídeo en el descodificador de vídeo, en el que el flujo de bits de vídeo comprende datos de vídeo comprimidos y se recibe independientemente de la indicación externa;

15

establecer (602, 808), mediante el descodificador de vídeo, en base a la indicación externa, el valor del indicador al valor establecido;

20

comprobar (604, 810), mediante el descodificador de vídeo, el valor del indicador; y

25

tratar (502, 604, 810), mediante el descodificador de vídeo, una imagen de acceso aleatorio limpio, CRA, como una imagen de acceso de enlace roto, BLA, en base al establecimiento del indicador al valor establecido,

30

en el que una imagen que sigue a una imagen CRA o BLA en orden de descodificación pero precede a la imagen CRA o BLA respectiva en orden de salida es una imagen principal asociada con la imagen CRA o BLA respectiva,

35

en el que, para una imagen CRA, las imágenes principales asociadas son correctamente descodificables en base al comienzo de la descodificación a partir de una imagen de acceso aleatorio, RAP, situada antes de la imagen CRA en orden de descodificación, y

40

en el que, para una imagen BLA, las imágenes principales asociadas son una de no correctamente descodificables o correctamente descodificables en base al comienzo de la descodificación a partir de una imagen RAP situada antes de la imagen BLA en orden de descodificación.

35

2. El procedimiento de la reivindicación 1, en el que:

el indicador está asociado con la imagen CRA, y/o

40

el valor predeterminado del indicador indica que la imagen CRA no se debe tratar como una imagen BLA.

45

3. El procedimiento de la reivindicación 1, en el que:

la indicación externa comprende un mensaje que indica que una imagen CRA se debe tratar como una imagen BLA, y/o

50

4. Un descodificador de vídeo (30) para procesar datos de vídeo de codificación de vídeo de alta eficiencia, HEVC, comprendiendo el descodificador de vídeo:

55

medios para recibir (500, 600, 806) una indicación externa (70) en el descodificador de vídeo para indicar una operación de conmutación, en el que la indicación externa indica si el descodificador de vídeo debe establecer un valor de un indicador a un valor predeterminado o a un valor establecido;

60

medios para recibir un flujo de bits de vídeo en el descodificador de vídeo, en el que el flujo de bits de vídeo comprende datos de vídeo comprimidos y se recibe independientemente de la indicación externa;

medios para establecer, en base a la indicación externa, el valor del indicador al valor establecido;

65

medios para comprobar el valor del indicador; y

medios para tratar (502, 604, 810) una imagen de acceso aleatorio limpio, CRA, como una imagen de acceso de enlace roto, BLA, en base al establecimiento del indicador al valor establecido,

en el que una imagen que sigue a una imagen CRA o BLA en orden de descodificación pero precede a la imagen CRA o BLA respectiva en orden de salida es una imagen principal asociada con la imagen CRA o BLA respectiva,

5 en el que, para una imagen CRA, las imágenes principales asociadas son correctamente descodificables en base al comienzo de la descodificación a partir de una imagen de acceso aleatorio, RAP, situada antes de la imagen CRA en orden de descodificación, y

10 en el que, para una imagen BLA, las imágenes principales asociadas son una de no correctamente descodificables o correctamente descodificables en base al comienzo de la descodificación a partir de una imagen RAP situada antes de la imagen BLA en orden de descodificación.

5. El descodificador de vídeo de la reivindicación 4, en el que:

15 el indicador está asociado con la imagen CRA, y/o

el valor predeterminado del indicador indica que la imagen CRA no se debe tratar como una imagen BLA.

20 6. El descodificador de vídeo de la reivindicación 4, en el que:

la indicación externa comprende un mensaje que indica que una imagen CRA se debe tratar como una imagen BLA, y/o

25 el indicador comprende un HandleCraAsBlaFlag, indicando el HandleCraAsBlaFlag si una imagen CRA se debe tratar como una imagen BLA.

7. Un medio de almacenamiento legible por ordenador que tiene instrucciones almacenadas en el mismo que, cuando se ejecutan mediante uno o más procesadores de un dispositivo, provocan que dicho uno o más procesadores lleven a cabo un procedimiento de acuerdo con cualquiera de las reivindicaciones 1 a 3.

30

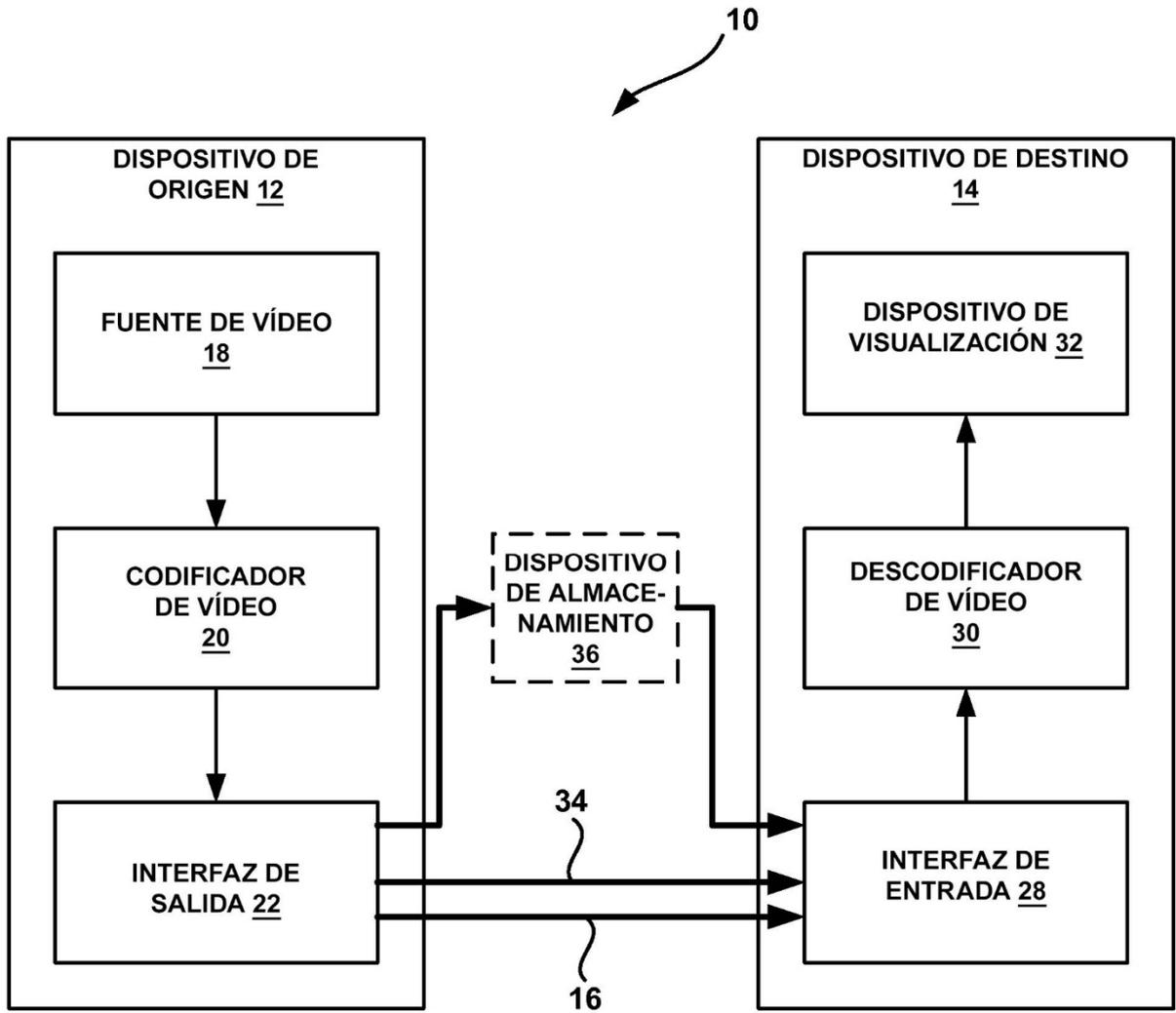


FIG. 1

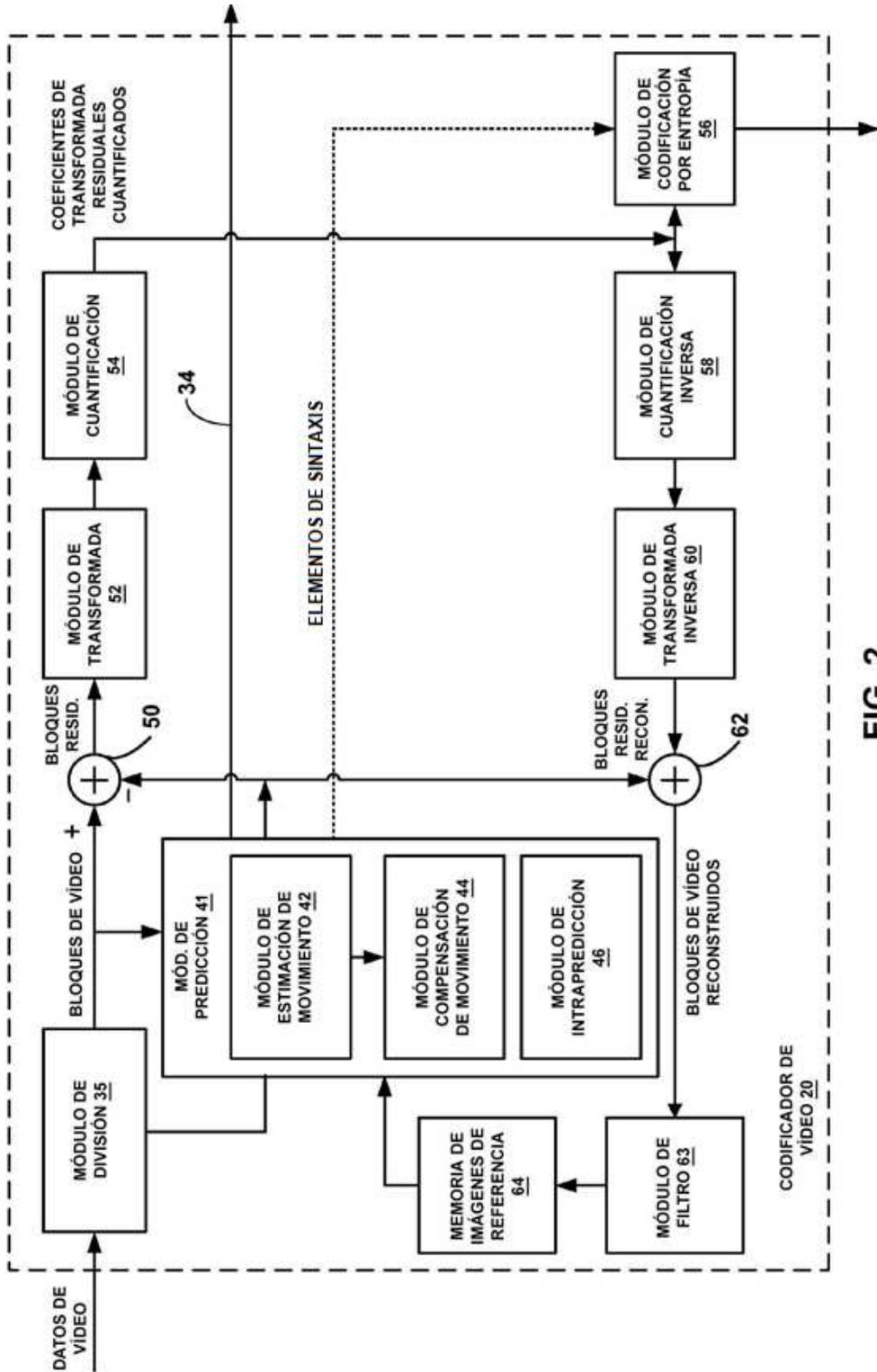


FIG. 2

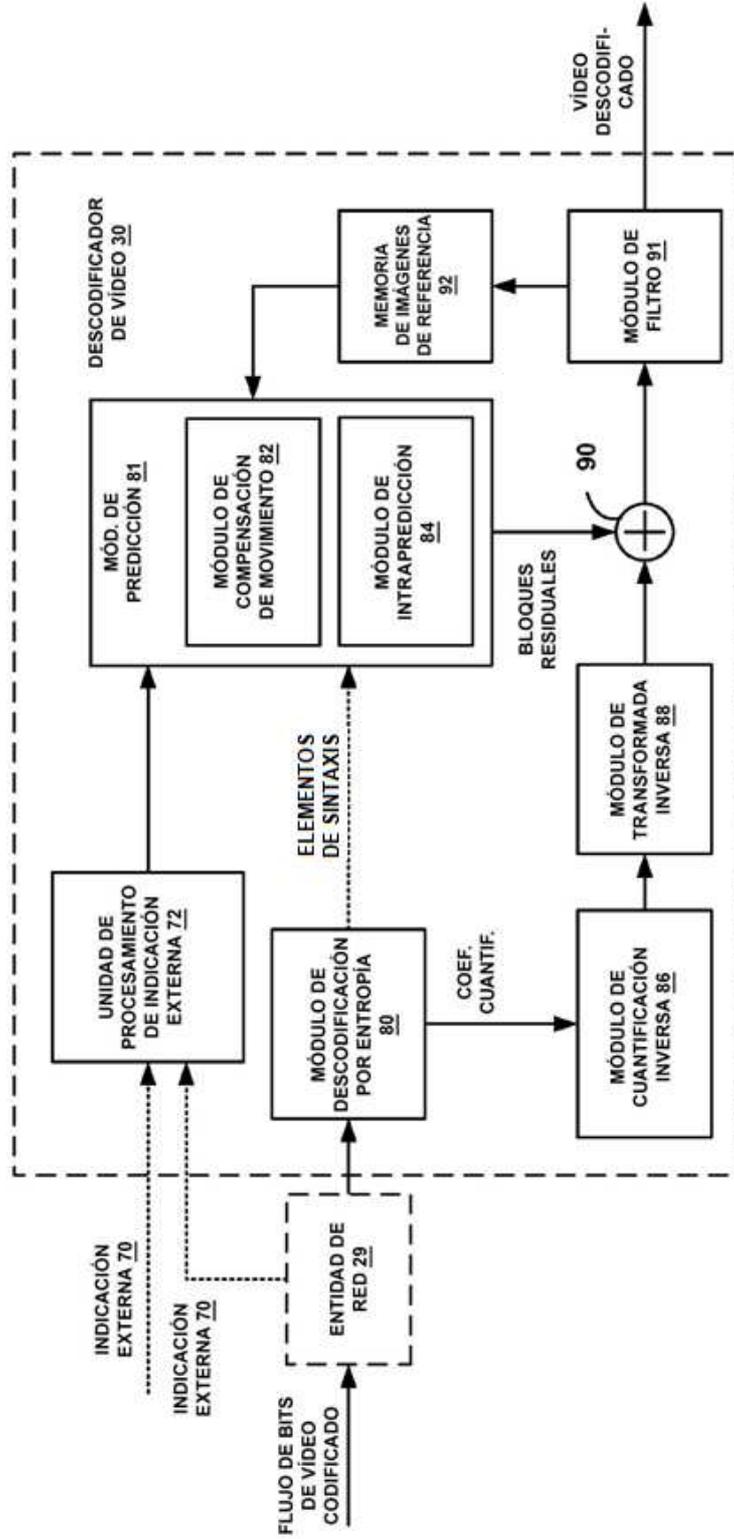


FIG. 3

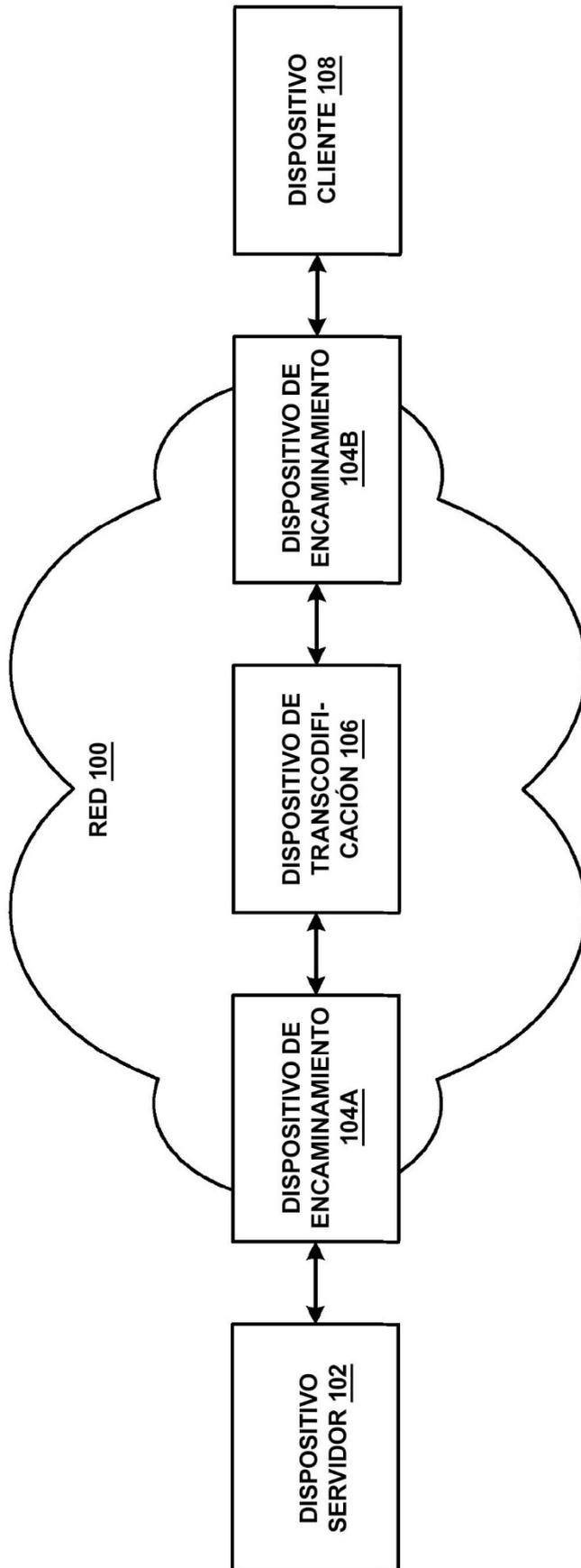


FIG. 4



FIG. 5

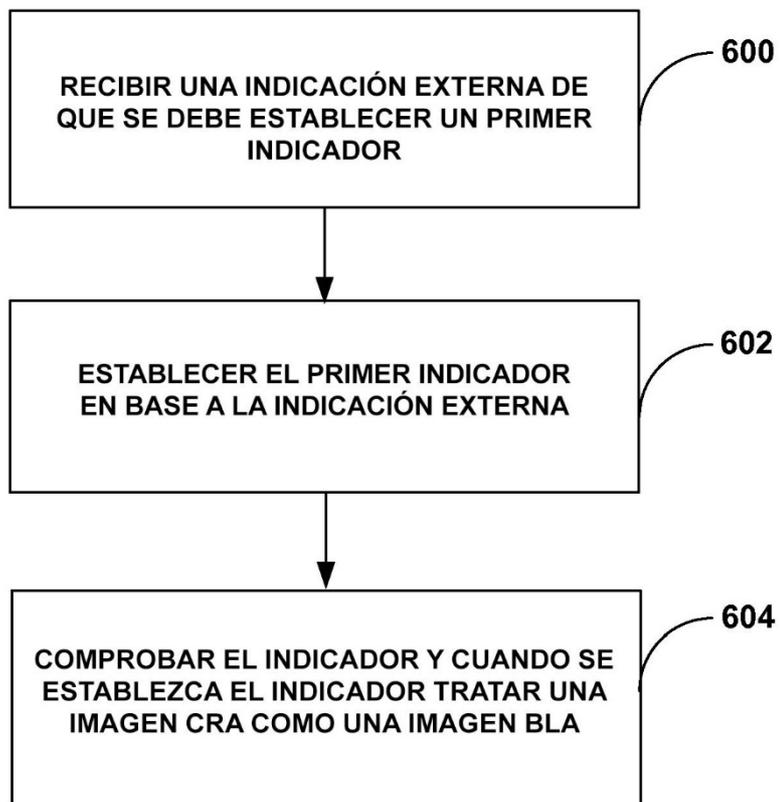


FIG. 6

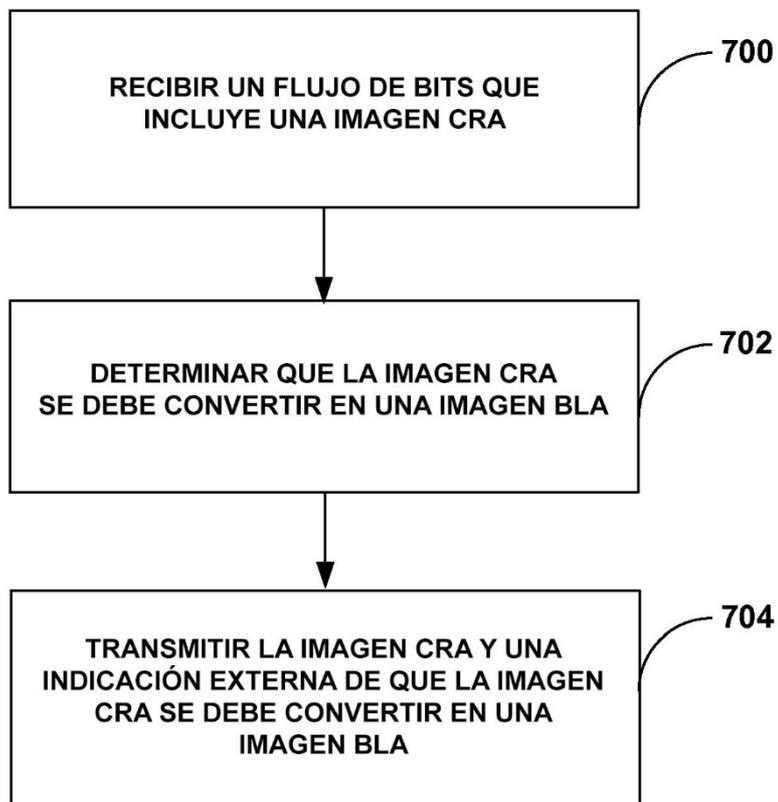


FIG. 7

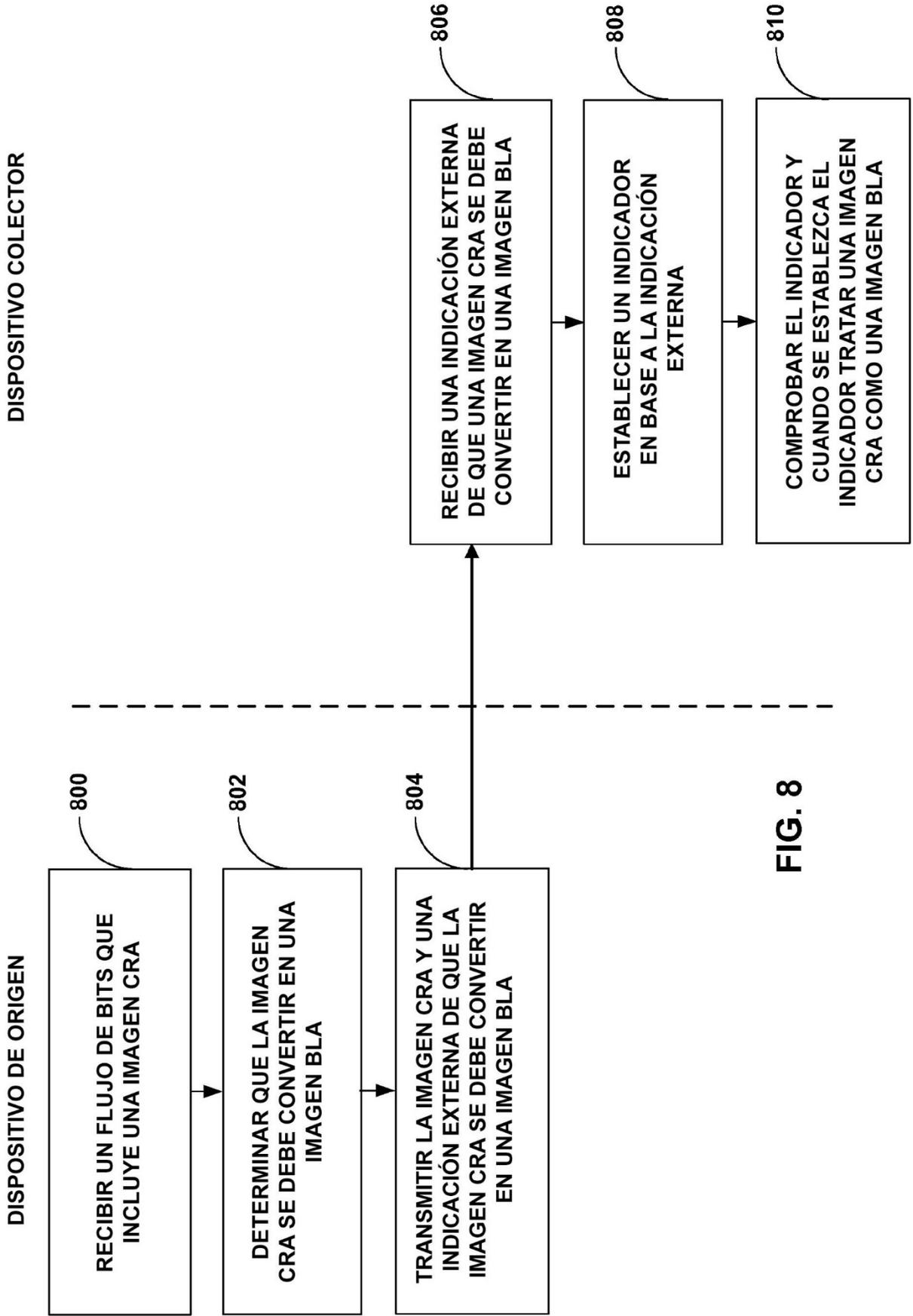


FIG. 8