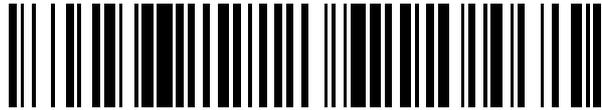


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 701 786**

51 Int. Cl.:

H04N 19/70 (2014.01)

H04N 19/46 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **11.04.2013 PCT/US2013/036224**

87 Fecha y número de publicación internacional: **24.10.2013 WO13158461**

96 Fecha de presentación y número de la solicitud europea: **11.04.2013 E 13718453 (7)**

97 Fecha y número de publicación de la concesión europea: **12.09.2018 EP 2839643**

54 Título: **Procesamiento de memoria intermedia de imágenes descodificadas para imágenes de punto de acceso aleatorio en secuencias de vídeo**

30 Prioridad:

20.04.2012 US 201261636566 P

04.05.2012 US 201261643100 P

02.07.2012 US 201261667371 P

12.03.2013 US 201313796737

12.03.2013 US 201313797458

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

25.02.2019

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)

Attn: International IP Administration, 5775

Morehouse Drive

San Diego, CA 92121-1714, US

72 Inventor/es:

WANG, YE-KUI

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 701 786 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

Procesamiento de memoria intermedia de imágenes descodificadas para imágenes de punto de acceso aleatorio en secuencias de vídeo

5 CAMPO TÉCNICO

[0001] Esta divulgación se refiere en general al procesamiento de datos de vídeo y, más particularmente, a la codificación de imágenes de punto de acceso aleatorio (RAP) en una secuencia de vídeo.

10 ANTECEDENTES

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

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

[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 pueden transformarse desde el dominio del píxel hasta un dominio de transformada, dando como resultado unos coeficientes de transformada residuales, que a continuación se pueden cuantificar. Los coeficientes de transformada cuantificados, dispuestos inicialmente en una matriz bidimensional, pueden examinarse con el fin de generar un vector unidimensional de coeficientes de transformada, y puede aplicarse codificación de entropía para lograr aún más compresión.

[0005] En el documento JCTVC-H1003_dA de Bross et al.: "WD6: High Efficiency Video Coding (HEVC) text specification draft 6 version JCTVC-H1003_dA [WD6: codificación de vídeo de alta eficiencia (HEVC), borrador 6 de especificación de texto, versión JCTVC-H1003_dA]" (Equipo colaborativo conjunto sobre codificación de vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 7.ª reunión, Ginebra, CH, 21-30 de noviembre de 2011, 13 de febrero de 2012), se divulga un proyecto de norma de codificación de vídeo.

[0006] En el documento JCTVC-G159 de Hendry *et al.*: "Random Access Detection and Notification [Detección y notificación de acceso aleatorio]" (Equipo colaborativo conjunto sobre codificación de vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 7.ª reunión, Ginebra, CH, 21-30 de noviembre de 2011) se divulgan dos mecanismos propuestos para la detección de acceso aleatorio y un mecanismo para permitir que cualquier aplicación (por ejemplo, una aplicación de reproductor de vídeo) notifique a un descodificador HEVC cuándo se produce un acceso aleatorio.

[0007] En el documento de Sjöberg *et al.*: "Overview of HEVC high-level syntax and reference picture management [Visión general de la sintaxis de alto nivel de HEVC y la gestión de imágenes de referencia]" (IEEE Transactions on Circuits and Systems for Video Technology [Transacciones IEEE sobre Circuitos y Sistemas para

Tecnología de Video], 13 de abril de 2012) se divulga una visión general de la sintaxis de alto nivel de HEVC, incluidas las cabeceras de unidades NAL, los conjuntos de parámetros, los esquemas de división de imágenes, la gestión de imágenes de referencia y los mensajes SEI.

5 **[0008]** En el documento JCTVC-E279 de Boyce *et al.*: "Extensible High Layer Syntax for Scalability [Sintaxis de alto nivel extensible para escalabilidad]" (Equipo colaborativo conjunto sobre codificación de vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 5.ª reunión, Ginebra, CH, 16-23 de marzo de 2011) se divulga una propuesta de nueva sintaxis de un Conjunto de Parámetros de Dependencia y dos nuevos mensajes SET, así como cambios en la cabecera de unidad NAL, el Conjunto de Parámetros de Secuencia, el Conjunto de Parámetros de Imagen y la cabecera de segmento, para un mejor respaldo de escalabilidad y futuras extensiones de HEVC.

10 **[0009]** En el documento JCTVC-I0113 de Suehring: "High level syntax parsing issues [Problemas en el análisis sintáctico de alto nivel]", (Equipo colaborativo conjunto sobre codificación de vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 9.ª reunión, Ginebra, CH, 27 de abril - 7 de mayo de 2012) se divulgan dos problemas de análisis sintáctico de alto nivel, así como posibles soluciones a los problemas de análisis.

RESUMEN

20 **[0010]** Esta divulgación se refiere a unas técnicas para codificar imágenes de puntos de acceso aleatorio (RAP) en una secuencia de vídeo codificada. En algunos ejemplos, cuando una imagen RAP actual que se va a descodificar es una imagen RAP de enlace roto, como una imagen de acceso aleatorio limpio (CRA) de enlace roto o una imagen de acceso de enlace roto (imagen BLA), un elemento sintáctico indica que no debe facilitarse ninguna de las imágenes anteriores a la imagen BLA en orden de descodificación de una memoria intermedia de imágenes descodificadas. El elemento sintáctico puede proporcionarse en una posición al principio de una cabecera de fragmento de la imagen BLA y anterior a cualquier parámetro de cabecera de fragmento sometido a codificación de entropía de cada fragmento de la imagen BLA. En otros ejemplos, cuando una imagen RAP actual que se va a descodificar es una imagen BLA, todas las imágenes de referencia de una memoria intermedia de imágenes descodificadas se marcan como no usadas para referencia.

30 **[0011]** En un ejemplo, la divulgación describe un procedimiento de descodificación de datos de vídeo, incluyendo el procedimiento recibir, en una unidad de capa de abstracción de red (NAL), un segmento de una imagen actual que se va a descodificar para una secuencia de datos de vídeo, en el que la imagen actual comprende una imagen de acceso de enlace roto (BLA) y la unidad NAL tiene un tipo de unidad NAL que es uno de una pluralidad de tipos de unidades NAL que incluyen: un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales descodificables asociadas en el flujo de bits, y un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits; recibir, en una cabecera de segmento del segmento, uno o más elementos sintácticos sometidos a codificación de entropía y elementos sintácticos no sometidos a codificación de entropía; y descodificar el segmento basándose en los elementos sintácticos no sometidos a codificación de entropía, en el que los elementos sintácticos no sometidos a codificación de entropía de la cabecera de segmento del segmento comprenden un elemento sintáctico `no_output_of_prior_pics_flag` que está incluido en la cabecera de segmento inmediatamente después de un elemento sintáctico que indica si el segmento es un primer segmento, por orden de descodificación, de la imagen actual, en el que el elemento sintáctico `no_output_of_prior_pics_flag` es anterior a todos los elementos sintácticos sometidos a codificación de entropía de la cabecera de segmento e indica si unas imágenes anteriores a la imagen actual por orden de descodificación se deben extraer de una memoria intermedia de imágenes descodificadas sin facilitarlas.

45 **[0012]** En otro ejemplo, la divulgación describe un procedimiento de codificación de datos de vídeo, comprendiendo el procedimiento: codificar un segmento de una imagen actual para una secuencia de datos de vídeo en una unidad de capa de abstracción de red (NAL), en el que la imagen actual comprende una imagen de acceso de enlace roto (BLA) y la unidad NAL tienen un tipo de unidad NAL que es uno de una pluralidad de tipos de unidades NAL que incluye: un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales descodificables asociadas en el flujo de bits, y un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits y codificar, en una cabecera de segmento del segmento, uno o más elementos sintácticos sometidos a codificación de entropía y elementos sintácticos no sometidos a codificación de entropía, en el que los elementos sintácticos no sometidos a codificación de entropía de la cabecera de segmento del segmento comprenden un elemento sintáctico `no_output_of_prior_pics_flag` que está incluido en la cabecera de segmento inmediatamente después de un elemento sintáctico que indica si el segmento es un primer segmento, por orden de descodificación, de la imagen actual, en el que el elemento sintáctico `no_output_of_prior_pics_flag` es anterior a todos los elementos sintácticos sometidos a codificación de entropía en la cabecera de segmento e indica si unas imágenes anteriores

a la imagen actual por orden de decodificación se deben extraer de una memoria intermedia de imágenes decodificadas sin facilitarlas.

5 **[0013]** En un ejemplo, la divulgación describe un medio de almacenamiento legible por ordenador que tiene almacenadas instrucciones que, cuando se ejecutan, hacen que uno o más procesadores de un dispositivo lleven a cabo un procedimiento de codificación o decodificación de datos de vídeo de acuerdo con los ejemplos anteriores.

10 **[0014]** En un ejemplo, la divulgación describe un dispositivo para decodificar datos de vídeo, comprendiendo el dispositivo: medios para recibir, en una unidad de capa de abstracción de red (NAL), un segmento de una imagen actual que se va a decodificar para una secuencia de datos de vídeo, en el que la imagen actual comprende una imagen de acceso de enlace roto (BLA) y la unidad NAL tiene un tipo de unidad NAL que es uno de una pluralidad de tipos de unidades NAL que incluyen: un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales descodificables asociadas en el flujo de bits, y un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits; medios para recibir, en una cabecera de segmento del segmento, uno o más elementos sintácticos sometidos a codificación de entropía y elementos sintácticos no sometidos a codificación de entropía; y medios para decodificar el segmento basándose en los elementos sintácticos no sometidos a codificación de entropía, en el que los elementos sintácticos no sometidos a codificación de entropía de la cabecera de segmento del segmento comprenden un elemento sintáctico `no_output_of_prior_pics_flag` que está incluido en la cabecera de segmento inmediatamente después de un elemento sintáctico que indica si el segmento es un primer segmento, por orden de decodificación, de la imagen actual, en el que el elemento sintáctico `no_output_of_prior_pics_flag` es anterior a todos los elementos sintácticos sometidos a codificación de entropía en la cabecera de segmento e indica si imágenes anteriores a la imagen actual por orden de decodificación se deben extraer de una memoria intermedia de imágenes decodificadas sin facilitarlas.

25 **[0015]** En otro ejemplo, la divulgación describe un dispositivo para decodificar datos de vídeo, incluyendo el dispositivo un dispositivo para decodificar datos de vídeo, comprendiendo el dispositivo: medios para recibir, en una unidad de capa de abstracción de red (NAL), un segmento de una imagen actual que se va a decodificar para una secuencia de datos de vídeo, en el que la imagen actual comprende una imagen de acceso de enlace roto (BLA) y la unidad NAL tiene un tipo de unidad NAL que es uno de una pluralidad de tipos de unidades NAL que incluyen: un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales descodificables asociadas en el flujo de bits, y un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits; medios para recibir, en una cabecera de segmento del segmento, uno o más elementos sintácticos sometidos a codificación de entropía y elementos sintácticos no sometidos a codificación de entropía; y medios para decodificar el segmento basándose en los elementos sintácticos no sometidos a codificación de entropía, en el que los elementos sintácticos no sometidos a codificación de entropía de la cabecera de segmento del segmento comprenden un elemento sintáctico `no_output_of_prior_pics_flag` que está incluido en la cabecera de segmento inmediatamente después de un elemento sintáctico que indica si el segmento es un primer segmento, por orden de decodificación, de la imagen actual, en el que el elemento sintáctico `no_output_of_prior_pics_flag` es anterior a todos los elementos sintácticos sometidos a codificación de entropía de la cabecera de segmento e indica si imágenes anteriores a la imagen actual por orden de decodificación se deben extraer de una memoria intermedia de imágenes decodificadas sin facilitarlas.

45 **[0016]** En otro ejemplo, la divulgación describe un dispositivo para codificar datos de vídeo, comprendiendo el dispositivo: medios para codificar un segmento de una imagen actual para una secuencia de datos de vídeo en una unidad de capa de abstracción de red (NAL), en el que la imagen actual comprende una imagen de acceso de enlace roto (BLA) y la unidad NAL tiene un tipo de unidad NAL que es uno de una pluralidad de tipos de unidades NAL que incluyen: un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales descodificables asociadas en el flujo de bits, y un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits; y medios para codificar, en una cabecera de segmento del segmento, uno o más elementos sintácticos sometidos a codificación de entropía y elementos sintácticos no sometidos a codificación de entropía, en el que los elementos sintácticos no sometidos a codificación de entropía de la cabecera de segmento del segmento comprenden un elemento sintáctico `no_output_of_prior_pics_flag` que está incluido en la cabecera de segmento inmediatamente después de un elemento sintáctico que indica si el segmento es un primer segmento, por orden de decodificación, de la imagen actual, en el que el elemento sintáctico `no_output_of_prior_pics_flag` es anterior a todos los elementos sintácticos sometidos a codificación de entropía en la cabecera de segmento e

indica si unas imágenes anteriores a la imagen actual por orden de descodificación se deben extraer de una memoria intermedia de imágenes descodificadas sin facilitarlas.

5 **[0017]** 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

10 **[0018]**

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

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

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

La figura 4 es un diagrama de bloques que ilustra un conjunto de ejemplo de dispositivos que forman parte de una red para comunicación de datos de vídeo.

25 La figura 5 es un diagrama que ilustra una secuencia de vídeo de ejemplo que incluye una imagen RAP de acuerdo con las técnicas descritas en esta divulgación.

La figura 6 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar imágenes RAP de acuerdo con uno o más ejemplos descritos en esta divulgación.

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

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

La figura 9 es un diagrama de flujo que ilustra un procedimiento de ejemplo para descodificar un fragmento de acuerdo con uno o más ejemplos descritos en esta divulgación.

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

DESCRIPCIÓN DETALLADA

45 **[0019]** Esta divulgación describe diversos diseños de codificación de vídeo, relacionados con imágenes que pueden usarse, por ejemplo, como puntos de acceso aleatorio o puntos de adaptación de flujo, tales como un punto de conmutación de capa temporal, o similares. Por ejemplo, este tipo de imagen podría ser un punto de conmutación para adaptación de velocidad de bits, velocidad de tramas o resolución espacial, cada una de las cuales se denominará en general imágenes RAP en esta divulgación. En algunos casos, un codificador de vídeo o descodificador de vídeo puede codificar una imagen o unas imágenes que están situadas, en una secuencia de vídeo, después de una imagen RAP en orden de descodificación, pero situadas antes del punto de acceso aleatorio o punto de adaptación de flujo, tal como un punto de conmutación de capa temporal en orden de salida, es decir, el orden de visualización. Estas imágenes pueden denominarse imágenes iniciales. Las imágenes iniciales son imágenes que siguen a una imagen RAP en el orden de descodificación, pero preceden a la imagen RAP en el orden de salida. Más específicamente, estas imágenes pueden denominarse imágenes iniciales de la imagen RAP.

55 **[0020]** Una imagen de acceso aleatorio limpio (CRA) es un tipo de imagen RAP. Las imágenes iniciales de una imagen CRA actual pueden descodificarse correctamente si la descodificación comienza por imagen de actualización instantánea de descodificación (IDR) o CRA que está situada en la secuencia de vídeo antes de la imagen CRA actual. No obstante, las imágenes iniciales de una imagen CRA no se pueden descodificar correctamente cuando se produce un acceso aleatorio desde la imagen CRA actual. Esto se debe a que las imágenes iniciales, es decir, las imágenes que están situadas después de la imagen CRA actual en orden de descodificación pero antes de la imagen CRA actual en orden de salida, pueden apuntar a bloques para referencia de predicción de imágenes anteriores que no están disponibles (por ejemplo, imágenes anteriores a la imagen BLA en orden de descodificación).

65

[0021] En particular, las imágenes que residen antes de la imagen CRA actual en orden de descodificación no están disponibles cuando la descodificación de acceso aleatorio comienza por la imagen CRA actual. En consecuencia, las imágenes iniciales no son descodificables en este caso, y típicamente se descartan cuando la descodificación de acceso aleatorio comienza en una imagen CRA. Para evitar la propagación de errores por imágenes que pueden no estar disponibles dependiendo de dónde comienza la descodificación, todas las imágenes que siguen a una imagen CRA tanto en orden de descodificación como en orden de salida no usan ninguna imagen que preceda a la imagen CRA en orden de descodificación u orden de salida (lo cual incluye las imágenes iniciales) como imágenes de referencia.

[0022] Para el caso en el que las imágenes iniciales no son descodificables, por ejemplo, cuando la descodificación comienza por una imagen RAP anterior, una imagen CRA puede incluir un señalizador de enlace roto. Este señalizador indica que las imágenes iniciales de la imagen CRA no son descodificables, por ejemplo, porque una imagen de referencia no es válida para usar en la descodificación de la imagen inicial debido a un cambio del flujo de bits que se está descodificando. Dicha imagen CRA puede denominarse imagen de acceso aleatorio limpio (BLC) de enlace roto o imagen de acceso de enlace roto (BLA).

[0023] Un enlace roto en general se refiere a una ubicación en un flujo de bits en la que se indica que algunas imágenes subsiguientes en orden de descodificación pueden contener artefactos visuales significativos debido a operaciones no especificadas realizadas en la generación del flujo de bits. En lugar de, o además de, usar un señalizador de enlace roto, puede usarse una imagen BLC (o una BLA análoga) para indicar un enlace roto en una secuencia de vídeo. Se puede usar una imagen BLA o BLC, por ejemplo, para empalme de flujos de bits por cualquiera de una variedad de dispositivos, tales como un servidor, un elemento de red adaptable a los medios (MANE) o un editor/empalmador de vídeo. De nuevo, una imagen BLC puede considerarse en general análoga a una imagen BLA, como se describe en HEVC WD9 (véase a continuación). Aunque la terminología es ligeramente diferente, la imagen BLC o BLA en general se puede referir a una imagen CRA o de acceso a capa temporal (TLA) para la cual las imágenes iniciales no son descodificables, estas imágenes pueden no ser descodificables, por ejemplo, cuando la descodificación comienza por una imagen RAP anterior.

[0024] En diversos ejemplos de esta divulgación, para una imagen BLA o BLC, se considera que las imágenes iniciales no son descodificables, dando por sentado que las imágenes de referencia anteriores a la imagen BLA o BLC en orden de descodificación, por ejemplo, antes de un punto de empalme, no están disponibles. De acuerdo con un ejemplo de esta divulgación, para descodificar una imagen BLA o BLC actual, las imágenes de referencia de una memoria intermedia de imágenes descodificadas pueden marcarse como no usadas para referencia por un descodificador. En particular, las imágenes de referencia de la memoria intermedia de imágenes descodificadas (DPB) pueden marcarse como no usadas para referencia por el descodificador cuando la imagen actual que se va a descodificar es una imagen BLA o BLC. En otro ejemplo, para la descodificación de una imagen BLA o BLC, un codificador u otro dispositivo puede generar, y un descodificador puede recibir, un elemento sintáctico tal como un señalizador, por ejemplo, un `no_output_of_prior_pics_flag`, que indica que no se va a facilitar ninguna de las imágenes de referencia anteriores a una imagen CRA o una imagen BLA (o BLC) que puede estar almacenada en la DPB. En algunos ejemplos, un codificador u otro dispositivo pueden colocar este señalizador u otro elemento sintáctico al principio de una cabecera de fragmento de una imagen BLA (o BLC) o CRA, por ejemplo, antes de elementos que están sometidos a descodificación de entropía, de modo que el señalizador puede descodificarse más fácilmente y la información puede estar disponible más pronto en el proceso de descodificación o ser fácilmente accesible para otras entidades aparte de un descodificador de vídeo. Por ejemplo, para que los dispositivos menos capacitados, como un elemento de red adaptable a los medios (MANE), puedan tener acceso a la información sin necesidad de descodificación de entropía, el señalizador `no_output_of_prior_pics_flag` puede colocarse en una posición situada al principio de la cabecera del fragmento y preceder a los parámetros de cabecera de fragmento sometidos a codificación de entropía.

[0025] En otro ejemplo, un codificador puede procesar tipos de unidad NAL asignados para indicar cuándo las imágenes BLA (o BLC) o CRA tienen y no tienen imágenes iniciales. Debe tenerse en cuenta que las imágenes BLA son sustancialmente similares a las imágenes BLC. En general, el cambio de imágenes BLC a imágenes BLA es un cambio de terminología, aunque el procesamiento de imágenes BLA también puede incluir la adición de los tipos de unidades NAL descritos en el presente documento. Por ejemplo, los tipos de unidad NAL pueden incluir una unidad NAL tipo 16, BLA_W_LP (BLA con imagen inicial); una unidad NAL tipo 17, BLA_W_DLP (BLA con imagen inicial descodificable pero sin imagen inicial no descodificable); y una unidad NAL tipo 18, BLA_N_LP (BLA sin imagen inicial). En algunos ejemplos, estos tipos de unidad NAL se pueden usar para indicar si una imagen BLA puede incluir imágenes iniciales descodificables y no descodificables, solo imágenes iniciales descodificables o ninguna imagen inicial. Cuando las imágenes BLA no tienen imágenes iniciales no descodificables, todas las imágenes iniciales asociadas, si están presentes, son descodificables.

[0026] La codificación de vídeo de alta eficiencia (HEVC) que está en proceso de elaboración por el Equipo de colaboración conjunta en codificación de vídeo (JCT-VC) del Grupo de expertos en codificación de vídeo (VCEG) de ITU-T y el Grupo de expertos en imágenes en movimiento (MPEG) de ISO/IEC se describe en diversos documentos de trabajo. Se describe un borrador reciente de la norma HEVC, denominado «HEVC Working Draft 6 [Borrador de trabajo 6 de HEVC]» o «WD6», en el documento JCTVC-H1003, de Bross *et al.*, titulado «High

5 efficiency video coding (HEVC) text specification draft 6 [Borrador 6 de especificación textual de la codificación de vídeo de alta eficiencia (HEVC)]», Equipo de colaboración conjunta en codificación de vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 8.ª reunión: San Jose, California, EE. UU., 1-10 de febrero de 2012, que, a partir del 4 de mayo de 2012, se puede descargar desde http://phenix.int-evry.fr/jct/doc_end_user/documents/8_San%20Jose/wg11/JCTVC-H1003-v22.zip, cuyo contenido completo se incorpora en el presente documento como referencia. Un borrador de trabajo (WD) más reciente de HEVC, en lo sucesivo denominado HEVC WD9, se describe en el documento JCTVC-K1003v13, de Bross *et al.*, «High efficiency video coding (HEVC) text specification draft 9 [Borrador 9 de especificación textual de codificación de vídeo de alta eficiencia (HEVC)]», Equipo de colaboración conjunta en codificación de vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 11.ª reunión: Shanghai, CN, 10-19 de octubre de 2012, que, desde el 27 de diciembre de 2012, puede descargarse desde http://phenix.int-evry.fr/jct/doc_end_user/documents/11_Shanghai/wg11/JCTVC-K1003-v13.zip, cuyo contenido completo se incorpora en el presente documento como referencia. En WD9, la terminología de imágenes BLC usada para referirse a imágenes CRA de enlace roto se ha cambiado por la terminología de imágenes BLA. En consecuencia, 15 la terminología de BLC y BLA se puede utilizar en general de manera intercambiable en esta divulgación para referirse a una imagen CRA de enlace roto.

20 **[0027]** Otras normas de codificación de vídeo de ejemplo incluyen ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual e ITU-T H.264 (también conocida como ISO/IEC MPEG-4 AVC), incluyendo sus ampliaciones de codificación de vídeo escalable (SVC) y de codificación de vídeo multivista (MVC).

25 **[0028]** A continuación se describirán diversos aspectos de las imágenes RAP. El acceso aleatorio se refiere en general a la descodificación de un flujo de bits de vídeo empezando por una imagen codificada que no es la primera imagen codificada del flujo de bits. El acceso aleatorio a un flujo de bits puede necesitarse en muchas aplicaciones de vídeo, como la radiodifusión y la transmisión en continuo, por ejemplo, para que los usuarios alternen entre canales diferentes, para saltar a partes específicas del vídeo, para admitir el empalme de flujos de bits o para alternar a un flujo de bits diferente para adaptación del flujo (de la velocidad de bits, velocidad de tramas, resolución espacial, etc.). Una aplicación de vídeo puede insertar imágenes RAP, muchas veces a intervalos regulares y/o en 30 ubicaciones seleccionadas, en el flujo de bits de vídeo para permitir esta característica.

35 **[0029]** Los codificadores y el descodificador pueden usar la imagen de actualización instantánea de descodificador (IDR) para acceso aleatorio. Sin embargo, dado que una imagen IDR comienza una secuencia de vídeo codificada y siempre limpia la memoria intermedia de imágenes descodificadas (DPB), las imágenes que siguen a la IDR en orden de descodificación no pueden usar imágenes descodificadas antes que la imagen IDR como imágenes de referencia. Por consiguiente, los flujos de bits que dependen de imágenes IDR para el acceso aleatorio pueden tener una eficiencia de codificación significativamente menor. Para mejorar la eficiencia de codificación, se ha introducido el concepto de imágenes de acceso aleatorio limpio (CRA) en la norma HEVC en proceso de elaboración para permitir que las imágenes que siguen a la imagen CRA en orden de descodificación, 40 pero que la preceden en orden de salida, usen imágenes descodificadas antes que la imagen CRA como referencia. Por lo tanto, la imagen CRA puede usarse para acceso aleatorio como la primera imagen descodificada de una secuencia de vídeo, o puede descodificarse como parte de una secuencia de vídeo para la que se han descodificado imágenes RAP anteriores (por ejemplo, IDR o CRA).

45 **[0030]** Las imágenes que siguen a una imagen CRA en orden de descodificación, pero que preceden a la imagen CRA en orden de salida, se denominan imágenes iniciales de la imagen CRA. Las imágenes iniciales de una imagen CRA se pueden descodificar correctamente si la descodificación comienza por una imagen RAP anterior a la imagen CRA actual. No obstante, las imágenes iniciales de una imagen CRA no se pueden descodificar correctamente cuando se produce un acceso aleatorio desde la imagen CRA. En consecuencia, como se ha 50 analizado anteriormente, en el proceso HEVC, estas imágenes iniciales típicamente se descartan durante la descodificación de acceso aleatorio de la imagen CRA. 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 cual incluye las imágenes iniciales) como referencia. 55

60 **[0031]** Las funciones de acceso aleatorio son admitidas en H.264/AVC con el mensaje de punto de recuperación SEI. Una implementación de descodificador H.264/AVC puede o no admitir las funciones. En HEVC, un flujo de bits que comienza con una imagen CRA se considera un flujo de bits conforme a la norma. Cuando un flujo de bits comienza con una imagen CRA, las imágenes iniciales de la imagen CRA pueden referirse a imágenes de referencia no disponibles y, por lo tanto, no pueden descodificarse correctamente. Sin embargo, la norma HEVC especifica que las imágenes iniciales de la imagen CRA inicial no se facilitan, de ahí el nombre «acceso aleatorio limpio» para una imagen CRA. Para el establecimiento de requisitos de conformidad de flujo de bits, la HEVC especifica un proceso de descodificación para generar imágenes de referencia no disponibles para la 65 descodificación de las imágenes iniciales no facilitadas, es decir, las imágenes iniciales que no se visualizan. Sin embargo, no se necesitan implementaciones de descodificador conformes a la norma para seguir ese proceso de

descodificación, siempre que la implementación del descodificador pueda generar un resultado idéntico al del proceso de descodificación que se realiza desde el comienzo de la secuencia de vídeo codificada.

5 **[0032]** En HEVC, un flujo de bits conforme a la norma puede no contener ninguna imagen IDR, y en consecuencia puede contener un subconjunto de una secuencia de vídeo codificada o una secuencia de vídeo codificada incompleta. En HEVC WD6, una secuencia de vídeo codificada se define de la manera siguiente.

10 **[0033]** Una secuencia de vídeo codificada es una secuencia de unidades de acceso que incluye, en orden de descodificación, una unidad de acceso IDR seguida de cero o más unidades de acceso no IDR incluidas todas las unidades de acceso subsiguientes hasta, pero sin incluir, cualquier unidad de acceso IDR subsiguiente.

15 **[0034]** Se ha descrito un concepto de «imágenes CRA con enlaces rotos» en el documento JCTVC-I0404, Sullivan *et al.*, «CRA pictures with broken links» [«imágenes CRA con enlaces rotos»], Equipo de colaboración conjunta en codificación de vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 9.ª reunión: Ginebra, CH, 27 de abril-7 de mayo de 2012, que, a partir del 27 de diciembre de 2012 puede descargarse en: http://phenix.int-evry.fr/ict/doc_end_user/documents/9_Geneva/wg11/JCTVC-I0404-v1.zip, cuyo contenido completo se incorpora en el presente documento como referencia. Comparado con el concepto CRA incluido en HEVC WD6, en un ejemplo, se propone en JCTVC-I0404 permitir de forma adicional a las imágenes CRA que no están al principio del flujo de bits tener imágenes iniciales no descodificables, como puede tenerlas una imagen CRA que comienza el flujo de bits. En HEVC WD6, se permite que una imagen CRA que comienza un flujo de bits (en orden de flujo de bits, también denominado orden de descodificación) esté seguida de imágenes iniciales que no se pueden descodificar debido a la ausencia de imágenes de referencia antes de la imagen BLA en orden de descodificación. Sin embargo, no se permite que una imagen CRA que está situada en medio de un flujo de bits tenga dichas imágenes iniciales no descodificables. En un ejemplo, la eliminación de esta restricción se propone en JCTVC-I0404 mediante adición de un señalizador de «enlace roto» que indicaría la presencia potencial de dichas imágenes iniciales no descodificables.

25 **[0035]** El señalizador de «enlace roto» se propone en JCTVC-I0404 como información de nivel de imagen de imágenes CRA en la cabecera del fragmento o en algún otro lugar para información de nivel de imagen. En un ejemplo que incluye un conjunto de parámetros de adaptación (APS), el señalizador de «enlace roto» puede formar parte del APS. Sin embargo, algunas normas no prevén un APS. Cuando el señalizador es igual a 1, se permitirá que el flujo de bits contenga imágenes iniciales de la imagen CRA que no son descodificables debido a que faltan imágenes de referencia precedentes, aunque el flujo de bits comience con una imagen IDR o una imagen CRA anterior en el orden de flujo de bits.

30 **[0036]** Para una imagen CRA con el señalizador de «enlace roto» igual a 1, aparte de tener permitido tener imágenes iniciales no descodificables como se analiza en el presente documento, sus bits más significativos (MBS) de recuento de orden de imágenes (POC) están establecidos en 0. Además, la imagen CRA con el señalizador de enlace roto igual a 1 puede incluir un señalizador `no_output_of_prior_pics_flag` que hace que la imagen CRA actúe de la misma manera que para una imagen IDR, y un `ID random_access_pic_id` que hace que la imagen CRA actúe de la misma manera que el `ID idr_pic_id` de las imágenes IDR en HEVC WD6. De forma adicional, el `ID idr_pic_id` actual como en HEVC WD6 se renombra como `random_access_pic_id` y sus restricciones deberán realizarse para que se apliquen tanto a las imágenes CRA como a las imágenes IDR, en lugar de solo a las imágenes IDR. Al igual que una imagen IDR, una imagen CRA con `broken_link_flag` igual a 1 podría activar un conjunto de parámetros de secuencia (SPS) diferente, cambiar el tamaño de la imagen, etc.

35 **[0037]** En algunos ejemplos, la señalización del señalizador de «enlace roto» en la cabecera del fragmento o un APS requeriría una entidad (por ejemplo, un servidor, elemento de red adaptable a los medios (MANE) o un editor/empalmador de vídeo) que convierte una imagen CRA normal en una denominada imagen de acceso aleatorio limpio (BLC) de enlace roto, también denominada imagen BLA en esta divulgación. Esta entidad normalmente necesitaría ser capaz de realizar la codificación de entropía y el análisis de la cabecera del fragmento y/o el APS para codificar el señalizador de enlace roto. Del mismo modo, una entidad (por ejemplo, un servidor, un MANE o un editor de vídeo) que identifica una imagen BLA o BLC, cuando sea necesario, necesitaría ser capaz de realizar la descodificación de entropía y el análisis de la cabecera del fragmento y/o el APS para encontrar el señalizador.

40 **[0038]** En algunos ejemplos, puede que no se permita que una imagen CRA (BLA) de enlace roto active un conjunto de parámetros de secuencia (SPS), un conjunto de parámetros de imagen (PPS) o APS (cuando la imagen hace referencia a este) si el ID del conjunto de parámetros es el mismo que el SPS, el PPS o el APS activo. Sin embargo, dado que una imagen BLA normalmente se origina en un flujo de bits diferente al de la imagen anterior en orden de descodificación, por ejemplo, en el caso del empalme de flujos de bits, la imagen BLA o BLC y la imagen anterior pueden usar diferentes cargas útiles de secuencia de bits sin formato (RBSP), PPS RBSP y APS RBSP. En consecuencia, en algunos ejemplos, es probable que tanto la imagen BLA o BLC como la imagen previa en orden de descodificación puedan referirse (directa o indirectamente) al mismo valor de SPS ID o PPS ID. De forma adicional, en algunos ejemplos, también es posible que las imágenes se refieran al mismo valor de identificación de ID de APS. Por lo tanto, el uso del SPS, PPS o APS activo para la imagen previa en la

descodificación es muy probable que haga que la descodificación de la imagen BLA o BLC y las siguientes imágenes (no solo las imágenes iniciales) sea incorrecta.

5 [0039] En ejemplos que no incluyen los elementos sintácticos `random_access_pic_id` y `no_output_of_prior_pics_flag` en imágenes CRA normales, una entidad (por ejemplo, un servidor, un elemento de red adaptable a los medios (MANE) o un editor/empalmador de vídeo) que cambia una imagen CRA normal para que sea una imagen BLA o BLC, cuando sea necesario, necesitaría ser capaz de realizar la codificación o descodificación de entropía y el análisis de la cabecera del fragmento y/o APS para codificar los elementos sintácticos.

10 [0040] Puesto que una imagen BLA o BLC normalmente se originaría a partir de un flujo de bits diferente al de la imagen previa en orden de descodificación, si `no_output_of_prior_pics_flag` es igual a 0, la memoria intermedia de imágenes descodificadas puede desbordarse. Como resultado, todas las siguientes imágenes en orden de descodificación pueden descodificarse incorrectamente o el descodificador puede incluso fallar.

15 [0041] Cuando una parte de un primer flujo de bits y una parte de un segundo flujo de bits, en los que la parte del segundo flujo de bits comienza por una imagen CRA (`picA`), se empalman o concatenan, es posible no convertir la imagen CRA en una imagen BLA o BLC. En su lugar, es posible mantener `picA` como una imagen CRA en un caso en el que la descodificación comienza por la imagen CRA, la imagen IDR, la imagen BLA o la imagen BLC previas y la calidad de descodificación de las imágenes iniciales de la imagen CRA, `picA`, es aceptable aunque no perfecta (por ejemplo, cuando las sumas de comprobación de una o más de las imágenes iniciales no coinciden con las sumas de comprobación señalizadas en los mensajes de información de mejora complementaria (SEI) de *hash* de imágenes descodificadas como en HEVC WD6). Sin embargo, este enfoque carece de un mecanismo para indicar la información anterior en el flujo de bits.

25 [0042] Para abordar algunos de los problemas anteriores, en un ejemplo, en lugar de usar un señalizador para indicar que una imagen CRA tiene un enlace roto, o indicar una imagen CRA (BLA) de enlace roto, puede usarse un tipo de unidad NAL diferenciado (por ejemplo, tipo de unidad NAL igual a 2 que está reservado en HEVC WD6). En dicho ejemplo, la información proporcionada por las unidades NAL puede estar disponible sin codificación o descodificación de entropía. Este enfoque puede permitir que una entidad (por ejemplo, un servidor, un elemento de red adaptable a los medios (MANE) o un editor/empalmador de vídeo) cambie una imagen CRA normal, para la cual, si no comienza el flujo de bits, todas las imágenes iniciales asociadas deben ser descodificables (es decir, pueden descodificarse correctamente), para ser una imagen BLA o BLC, cuando sea necesario. Esto también puede permitir que una entidad (por ejemplo, un servidor, un MANE o un editor de vídeo) identifique una imagen BLA o BLC cuando sea necesario.

35 [0043] En algunos ejemplos, en lugar de permitir que una imagen BLC active un SPS diferente, se requiere que una imagen BLC active un SPS, incluso si el SPS ID al que hace referencia la imagen BLC es el mismo que el SPS ID del SPS activo para la imagen anterior en orden de descodificación. Esto puede realizarse indirectamente a través de un PPS como en HEVC WD6 o mediante otros procedimientos, por ejemplo, indirectamente a través de un conjunto de parámetros de grupo como se describe en JCTVC-I0338, o directamente, por ejemplo, cuando el SPS ID se incluye directamente en la cabecera del fragmento, o indirectamente a través de un mensaje SEI de período de almacenamiento intermedio. Puede requerirse que una imagen BLC active un SPS, porque una imagen BLC era típicamente de un flujo de bits diferente al de la imagen anterior en orden de descodificación. Además, típicamente se aplican diferentes SPS RBSP con SPS ID idénticos o diferentes.

45 [0044] Una imagen BLC también se puede usar para activar un PPS. Esto puede ocurrir incluso si el PPS ID al que se refiere la imagen BLC (indirectamente a través de un conjunto de parámetros de grupo como el descrito en JCTVC-I0338, o directamente, por ejemplo, cuando el PPS ID se incluye directamente en la cabecera del fragmento como en HEVC WD6) es igual al PPS ID del PPS activo para la imagen anterior en orden de descodificación. De nuevo, esto es así porque una imagen BLC era típicamente de un flujo de bits diferente al de la imagen anterior en orden de descodificación, y típicamente se aplican PPS RBSP diferentes, con PPS ID idénticos o diferentes.

55 [0045] En algunos ejemplos, también se requiere que una imagen BLA o BLC active un APS si se refiere a un APS, incluso si el APS ID al que se refiere la imagen BLA o BLC es el mismo que el APS ID del PPS activo para la imagen anterior en orden de descodificación. De nuevo, esto es así porque una imagen BLA o BLC era típicamente de un flujo de bits diferente al de la imagen previa en orden de descodificación. De forma adicional, típicamente se aplican diferentes APS RBSP, con APS ID idénticos o diferentes.

60 [0046] Una imagen BLA o BLC se puede considerar un tipo especial de imagen CRA y se puede definir como una imagen codificada para la cual el tipo de unidad NAL es el de la imagen BLC (por ejemplo, tipo de unidad NAL igual a 2 que está reservado en HEVC WD6). Esta definición puede aplicarse, con diferente terminología pero efecto similar, para imágenes BLA, como se describe en HEVC WD9. Se puede aplicar el mismo proceso de descodificación para una imagen CRA no BLA cuando comienza el flujo de bits y las imágenes iniciales asociadas, para descodificar una imagen BLA o BLC y las imágenes iniciales asociadas, incluso si la imagen BLA o BLC no es la primera imagen del flujo de bits. De forma alternativa, las imágenes BLA o BLC pueden excluirse de las

imágenes CRA, es decir, puede considerarse que una imagen BLA o BLC no es una imagen CRA. En ese caso, se aplica el mismo proceso de decodificación para una imagen CRA cuando comienza el flujo de bits y las imágenes iniciales asociadas, para decodificar una imagen BLA o BLC y las imágenes iniciales asociadas, incluso si la imagen BLA o BLC no es la primera imagen del flujo de bits. En el análisis siguiente, se supone que se aplica esta alternativa.

[0047] En algunos ejemplos, una unidad de acceso BLA o BLC se puede definir como una unidad de acceso en la que la imagen codificada es una imagen BLA o BLC. La definición de secuencia de vídeo codificada puede cambiarse de la siguiente manera: una secuencia de unidades de acceso que incluye, en orden de decodificación, una unidad de acceso IDR o una unidad de acceso BLA o BLC seguida de cero o más unidades de acceso no IDR y no BLA, incluidas todas las unidades de acceso subsiguientes hasta, pero sin incluir, cualquier unidad de acceso IDR o BLA subsiguiente.

[0048] En lugar de tener `random_access_pic_id` y `no_output_of_prior_pics_flag` para todas las imágenes IDR e imágenes BLA solo como en JCTVC-I0404, en otro enfoque, los dos campos están siempre presentes para todas las imágenes IDR, BLA y todas las imágenes CRA. Para cada imagen CRA, se requiere que `no_output_of_prior_pics_flag` sea igual a 0. En algunos ejemplos, puede ser más fácil para una entidad (por ejemplo, un servidor, un elemento de red adaptable a los medios (MANE) o un editor/empalmador de vídeo) cambiar una imagen CRA para que sea una imagen BLA, cuando sea necesario.

[0049] En otro ejemplo alternativo, para una imagen BLA, puede requerirse que `no_output_of_prior_pics_flag` sea igual a 1. De forma alternativa, cada imagen BLA no tiene ningún `no_output_of_prior_pics_flag` señalado, pero el comportamiento de salida de la imagen es el mismo que si tuviera `no_output_of_prior_pics_flag` igual a 1. En otra alternativa, cada imagen BLA tiene señalado `no_output_of_pior_pics_flag`, pero el comportamiento de salida de la imagen es el mismo que si tuviera `no_output_of_prior_pics_flag` igual a 1, independientemente del valor del `no_output_of_prior_pics_flag` señalado.

[0050] El valor de bit más significativo (MSB) del recuento de orden de imágenes (POC) para una imagen BLA puede señalizarse, por ejemplo, en la cabecera del fragmento. Si se señala, el valor se seguiría considerando igual a 0 en el proceso de decodificación, independientemente del valor. De forma alternativa, el valor POC MSB señalado se usa en el proceso de decodificación, pero luego el empalmador necesita verificar y posiblemente cambiar el valor para que sea coherente con los valores POC de imágenes anteriores en orden de decodificación.

[0051] Cuando una parte de un primer flujo de bits y una parte de un segundo flujo de bits, en los que la parte del segundo flujo de bits comienza por una imagen CRA, `picA`, se empalman o concatenan, el empalmador puede mantener una imagen del primer flujo de bits como una imagen CRA. En un ejemplo, cuando la decodificación comienza por la imagen CRA previa, la imagen IDR o la imagen BLA, la calidad de decodificación de las imágenes iniciales de la imagen CRA puede ser aceptable aunque no perfecta. Por ejemplo, la calidad de decodificación puede ser aceptable cuando las sumas de comprobación de una o más de las imágenes iniciales no coinciden con las sumas de comprobación señalizadas en los mensajes SEI de *hash* de imágenes decodificadas como en HVEC WD6.

[0052] Una indicación de la información anterior se puede señalar en el flujo de bits. La información puede señalizarse a través de una indicación asociada con una imagen CRA del primer flujo de bits, por ejemplo, como un señalizador en la cabecera de la unidad NAL o la cabecera del fragmento o el APS al que se refiere, o un mensaje SEI asociado con una imagen CRA del primer flujo de bits. El señalizador puede denominarse señalizador de coincidencia exacta. En un ejemplo, un valor 1 indica que la suma de comprobación de cada imagen inicial asociada con una imagen CRA del primer flujo de bits coincide con la suma de comprobación señalizada en el mensaje SEI de *hash* de imagen decodificada, si está presente. Un valor 0 indica que la suma de comprobación de cada imagen inicial asociada con una imagen CRA del primer flujo de bits puede o no coincidir con la suma de comprobación señalizada en el mensaje SEI de *hash* de imagen decodificada, si está presente.

[0053] Se proporciona un análisis sobre la señalización de imágenes iniciales y la presencia de imágenes iniciales de imágenes CRA en el documento JCTVC-I0275, disponible en: http://phenix.int-evry.fr/jct/doc_end_user/documents/9_Geneva/wg11/JCTVC-I0275-v2.zip; y el documento JCTVC-I0277, disponible en: http://phenix.int-evry.fr/jct/doc_end_user/documents/9_Geneva/wg11/JCTVC-I0277-v3.zip, para un análisis de la señalización de imágenes iniciales y la presencia de imágenes iniciales de imágenes CRA, el contenido completo de los cuales se incorpora en el presente documento como referencia. En algunos ejemplos, se puede aplicar una idea similar para señalar la presencia de imágenes iniciales de imágenes CRA a las imágenes BLA o BLC.

[0054] El uso de imágenes RAP de enlace roto, tales como imágenes BLA o BLC, puede presentar algunos problemas. Se pueden proporcionar diversas mejoras para abordar dichos problemas asociados con imágenes RAP de enlace roto, tales como imágenes BLA o BLC, y otros enfoques de acceso aleatorio de vídeo existentes.

[0055] Un problema es que definir solo un tipo de unidad NAL para las imágenes iniciales asociadas con una imagen CRA puede no ser suficiente, ya que una imagen inicial también puede ser una imagen de acceso a capa temporal (TLA). Por lo tanto, los codificadores y descodificadores que implementan los sistemas y procedimientos descritos en el presente documento pueden necesitar más tipos de unidades NAL para identificar unidades NAL pertenecientes a imágenes iniciales que también son imágenes TLA y unidades NAL pertenecientes a imágenes iniciales que no son imágenes TLA.

[0056] En otro problema, si una imagen BLA o BLC cambia la resolución espacial de las imágenes de la secuencia de vídeo, entonces el proceso de descodificación actual como el especificado en HEVC WD6 para las imágenes iniciales de una imagen CRA que comienza el flujo de bits no se puede aplicar directamente para las imágenes iniciales de la imagen BLA o BLC. En algunos casos, puede parecer que la resolución espacial es diferente para una imagen actual y una imagen de referencia para la imagen actual. En otro problema, en el proceso de descodificación especificado en HEVC WD6, el elemento sintáctico `slice_type` puede estar innecesariamente presente en la cabecera de fragmento de los fragmentos de imágenes IDR, CRA y BLA o BLC.

[0057] Esta divulgación describe diversas técnicas para imágenes RAP en general y para codificar imágenes RAP de enlace roto (por ejemplo, imágenes BLA) en particular en una secuencia de vídeo. Algunas de las técnicas pueden abordar uno o más de los problemas descritos anteriormente, mientras que algunas de las técnicas pueden proporcionar características adicionales que pueden ser útiles en la codificación de imágenes de acceso aleatorio.

[0058] En algunos ejemplos, cuando una imagen RAP actual que se va a descodificar es una imagen RAP de enlace roto, como una imagen (imagen BLA) de acceso aleatorio limpio (CRA) de enlace roto, un elemento sintáctico indica que no se va a facilitar ninguna de las imágenes de referencia anteriores a la imagen BLA en orden de descodificación de una memoria intermedia de imágenes descodificadas. Este elemento sintáctico puede ser codificado por un codificador y descodificado por un descodificador. Además, el elemento sintáctico puede proporcionarse en una posición situada al principio de una cabecera del fragmento de una imagen RAP de enlace roto actual y antes de cualquier parámetro de cabecera del fragmento sometido a codificación de entropía de cada fragmento de la imagen RAP de enlace roto actual. Proporcionar el elemento sintáctico antes que los elementos sintácticos sometidos a codificación de entropía, por ejemplo, antes que cualquier elemento `ue(v)`, puede permitir que un dispositivo con menos capacidad, en algunos ejemplos, interprete el elemento sintáctico sin necesidad de codificación de entropía. En otros ejemplos, cuando una imagen RAP actual que se va a descodificar es una imagen RAP de enlace roto, todas las imágenes de referencia de una memoria intermedia de imágenes descodificadas se marcan como no usadas para referencia. Esta característica puede permitir que se aplique un proceso de descodificación estándar para las imágenes iniciales de una imagen BLA o BLC, incluso si la imagen BLA o BLC cambia la resolución espacial. En esta divulgación se describen estas y otras técnicas de ejemplo.

[0059] La figura 1 es un diagrama de bloques que ilustra un ejemplo de sistema 10 de codificación y descodificación de vídeo que puede utilizar las técnicas descritas en esta divulgación. Como se muestra en la figura 1, el sistema 10 incluye un dispositivo 12 de origen que genera datos de vídeo codificados que un dispositivo 14 de destino puede descodificar más tarde. El dispositivo 12 de origen y el dispositivo 14 de destino pueden comprender cualquiera de entre una amplia gama de dispositivos, incluidos ordenadores de escritorio, ordenadores plegables (es decir, portátiles), ordenadores de tableta, descodificadores, teléfonos tales como los denominados teléfonos «inteligentes», los denominados paneles «inteligentes», televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, un dispositivo de transmisión en continuo de vídeo o similares. En algunos casos, el dispositivo 12 de origen y el dispositivo de 14 destino pueden estar equipados para la comunicación inalámbrica.

[0060] En algunos casos, el codificador 20 de vídeo puede codificar una imagen o unas imágenes que se hallan después de un punto de acceso aleatorio o punto de adaptación de flujo, tal como un punto de conmutación de capa temporal. Por ejemplo, este podría ser un punto de conmutación para la adaptación de la velocidad de bits, la velocidad de tramas o la resolución espacial. Una o más de estas imágenes pueden ser imágenes iniciales de una imagen CRA en una secuencia de vídeo. Un descodificador puede descodificar correctamente las imágenes iniciales de una imagen CRA si la descodificación de la secuencia de vídeo comienza por una imagen RAP antes que la imagen CRA actual en la secuencia de vídeo. No obstante, las imágenes iniciales de una imagen CRA no se pueden descodificar correctamente cuando se produce un acceso aleatorio desde la imagen CRA. Por ejemplo, las imágenes iniciales pueden apuntar a bloques para referencia de predicción que no están disponibles. En consecuencia, la imagen inicial puede no ser descodificable en el descodificador 30 de vídeo. En consecuencia, un dispositivo 14 de destino puede típicamente descartar estas imágenes iniciales durante la descodificación de acceso aleatorio.

[0061] En otro ejemplo, para una imagen BLA o BLC, el codificador 20 puede codificar un señalizador, por ejemplo, `no_output_of_prior_pics_flag` (que de forma más correcta puede denominarse elemento sintáctico) de tal modo que no se facilita ninguna de las imágenes anteriores en la DPB. En algunos ejemplos, este señalizador, o elemento sintáctico, puede estar al principio de la cabecera del fragmento antes de la descodificación de entropía de tal modo que pueda descodificarse más fácilmente y la información esté disponible más pronto en el proceso de codificación. Por ejemplo, para que dispositivos menos avanzados, como un MANE, puedan tener acceso a la

información sin necesidad de un decodificador ya que, por ejemplo, no es necesario realizar la decodificación de entropía de `no_output_of_prior_pics_flag`. Como ilustración, el señalizador `no_output_of_prior_pics` se puede presentar como un elemento sintáctico no sometido a codificación de entropía, como, por ejemplo, un elemento de longitud fija $u(1)$, en lugar de un elemento sintáctico sometido a codificación de entropía, como, por ejemplo, un elemento de longitud variable $ue(v)$. El señalizador `no_output_of_prior_pics` se puede presentar, por ejemplo, justo después de un señalizador `first_slice_segment_in_pic` y antes que cualquier elemento sintáctico sometido a codificación de entropía.

[0062] En un ejemplo, el decodificador 30 de vídeo puede marcar imágenes almacenadas en una memoria intermedia de imágenes de referencia como no usadas para referencia cuando, por ejemplo, el uso de estas imágenes puede dar como resultado una imagen descodificada incorrectamente. Por ejemplo, una imagen de referencia que precede a una imagen BLA o BLC en orden de descodificación u orden de salida puede ser inutilizable para referencia para una imagen inicial que sigue a la imagen BLA o BLC en orden de descodificación. En consecuencia, la imagen de referencia puede marcarse como no usada para referencia 30 por el decodificador de vídeo como respuesta a la recepción de una imagen de referencia que precede a una imagen BLA o BLC en orden de descodificación u orden de salida y puede ser inutilizable para referencia para una imagen inicial que sigue a la imagen BLA o BLC en orden de descodificación.

[0063] En un ejemplo, un codificador 20 de vídeo puede estar configurado para incluir tipos de unidad NAL asignados para indicar cuándo las imágenes BLA o BLC tienen y no tienen imágenes iniciales. Por ejemplo, en una norma, los tipos de unidad NAL 16, `BLA_W_LP` (BLA con imagen inicial); 17, `BLA_W_DLP` (BLA con imagen inicial descodificable); y 18, `BLA_N_LP` (BLA sin imagen inicial) están incluidos.

[0064] En un ejemplo, el codificador 20 de vídeo puede procesar tipos de unidad NAL asignados para indicar cuándo las imágenes BLA tienen y no tienen imágenes iniciales. Por ejemplo, el codificador 20 de vídeo puede codificar imágenes de acuerdo con uno de una pluralidad de tipos diferentes de unidades de capa de abstracción de red (NAL). La pluralidad de tipos de unidades NAL incluye uno o más de (1) un fragmento codificado de una imagen de acceso de enlace roto (BLA), en la que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, (2) un fragmento codificado de una imagen BLA, en la que la imagen BLA es una imagen BLA con imágenes iniciales descodificables asociadas en el flujo de bits, y (3) un fragmento codificado de una imagen BLA, en la que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits.

[0065] El dispositivo 14 de destino puede recibir los datos de vídeo codificados. El dispositivo de destino puede descodificar los datos recibidos a través de un enlace 16. El enlace 16 puede comprender cualquier tipo de medio o dispositivo capaz de desplazar los datos de vídeo codificados desde el dispositivo 12 de origen hasta el dispositivo 14 de destino. En un ejemplo, el enlace 16 puede comprender un medio de comunicación para permitir al dispositivo 12 de origen transmitir datos de vídeo codificados directamente al dispositivo 14 de destino en tiempo real. Los datos de vídeo codificados pueden modularse de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y transmitirse al dispositivo 14 de destino. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrica o alámbrica, tal como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión físicas. 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 12 de origen hasta el dispositivo 14 de destino.

[0066] De forma alternativa, los datos codificados pueden facilitarse desde la interfaz 22 de salida hasta un dispositivo 34 de almacenamiento. De forma similar, una interfaz de entrada puede acceder a los datos codificados desde el dispositivo 34 de almacenamiento. El dispositivo 34 de almacenamiento puede incluir cualquiera de una diversidad de medios de almacenamiento de datos de acceso distribuido o local, tales como una unidad de disco duro, unos discos Blu-ray, unos DVD, unos CD-ROM, una memoria flash, una memoria volátil o no volátil o cualquier otro medio de almacenamiento digital adecuado para almacenar datos de vídeo codificados. En otro ejemplo, el dispositivo 34 de almacenamiento puede corresponder a un servidor de archivos o a otro dispositivo de almacenamiento intermedio que puede contener el vídeo codificado generado por el dispositivo 12 de origen. El dispositivo 14 de destino puede acceder a los datos de vídeo almacenados del dispositivo 34 de almacenamiento a través de transmisión en continuo o descarga. El servidor de archivos puede ser cualquier tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir esos datos de vídeo codificados al dispositivo 14 de destino. Entre los ejemplos de servidores de archivos se incluyen un servidor web (por ejemplo, para un sitio web), un servidor FTP, dispositivos de almacenamiento conectado en red (NAS) o una unidad de disco local. El dispositivo 14 de destino puede acceder a los datos de vídeo codificados a través de cualquier conexión de datos estándar, incluyendo una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión wifi), una conexión alámbrica (por ejemplo, DSL, módem de cable, etc.) o una combinación de ambos que sea adecuada para acceder a datos de vídeo codificados almacenados en un servidor de archivos. La transmisión de datos de vídeo codificados desde el dispositivo 34 de almacenamiento puede ser una transmisión en continuo, una transmisión de descarga o una combinación de ambas.

[0067] Las técnicas de esta divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas pueden aplicarse a la codificación de vídeo, como apoyo a cualquiera de una diversidad de aplicaciones de multimedia, tales como radiodifusiones de televisión por aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones de vídeo en continuo, por ejemplo, mediante Internet, codificación de vídeo digital para su almacenamiento en un medio de almacenamiento de datos, descodificación de vídeo digital almacenado en un medio de almacenamiento de datos, u otras aplicaciones. En algunos ejemplos, el sistema 10 puede estar configurado para admitir la transmisión de vídeo unidireccional o bidireccional para admitir aplicaciones tales como la transmisión en continuo de vídeo, la reproducción de vídeo, la radiodifusión de vídeo y/o la videotelefonía.

[0068] En el ejemplo de la figura 1, el dispositivo 12 de origen incluye una fuente 18 de vídeo, un codificador 20 de vídeo y una interfaz 22 de salida. En algunos casos, la interfaz 22 de salida puede incluir un modulador/desmodulador (módem) y/o un transmisor. En el dispositivo 12 de origen, la fuente 18 de vídeo puede incluir una fuente tal como un dispositivo de captación de vídeo, por ejemplo, una videocámara, un archivo de vídeo que contiene vídeo previamente captado, una interfaz de vídeo en tiempo real para recibir vídeo desde un proveedor de contenido de vídeo y/o un sistema de gráficos de ordenador para generar datos de gráficos de ordenador como el vídeo de origen, o una combinación de dichas fuentes. En un ejemplo, si la fuente 18 de vídeo es una videocámara, el dispositivo 12 de origen y el dispositivo 14 de destino 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 pueden aplicarse a aplicaciones inalámbricas y/o alámbricas.

[0069] El vídeo captado, precaptado o generado por ordenador puede ser codificado por el codificador 20 de vídeo. Los datos de vídeo codificados pueden transmitirse directamente al dispositivo 14 de destino mediante la interfaz 22 de salida del dispositivo 12 de origen. Los datos de vídeo codificados pueden almacenarse de forma adicional (o alternativa) en el dispositivo 34 de almacenamiento para un posterior acceso por el dispositivo 14 de destino u otros dispositivos, para su descodificación y/o reproducción.

[0070] El dispositivo 14 de destino incluye una interfaz 28 de entrada, un descodificador 30 de vídeo y un dispositivo 32 de visualización. En algunos casos, la interfaz 28 de entrada puede incluir un receptor y/o un módem. La interfaz 28 de entrada del dispositivo 14 de destino recibe los datos de vídeo codificados por el enlace 16. Los datos de vídeo codificados transmitidos a través del enlace 16, o proporcionados en el dispositivo 34 de almacenamiento, pueden incluir una diversidad de elementos sintácticos generados por el codificador 20 de vídeo, para su uso por un descodificador de vídeo, tal como el descodificador 30 de vídeo, en la descodificación de los datos de vídeo. Dichos elementos sintácticos se pueden incluir con los datos de vídeo codificados, transmitidos en un medio de comunicación, almacenar en un medio de almacenamiento o almacenar en un servidor de archivos.

[0071] En algunos ejemplos, el descodificador 30 de vídeo del dispositivo 14 de destino puede descodificar una imagen o varias imágenes que se hallan después de un punto de acceso aleatorio o un punto de adaptación de flujo, tal como un punto de conmutación de capa temporal. Por ejemplo, este podría ser un punto de conmutación para adaptación de velocidad de bits, velocidad de tramas (es decir, un punto de conmutación de capa temporal) o resolución espacial. Una o más de estas imágenes pueden ser imágenes iniciales. Las imágenes iniciales no se pueden descodificar correctamente cuando se produce un acceso aleatorio desde una imagen BLA o BLC.

[0072] En un ejemplo, para evitar la propagación de errores por imágenes de referencia que pueden no estar disponibles dependiendo de dónde comienza la descodificación, el descodificador 30 de vídeo no puede usar ninguna imagen que preceda a la imagen BLA o BLC en orden de descodificación u orden de salida (lo cual incluye las imágenes iniciales) como imágenes de referencia.

[0073] En diversos ejemplos, el descodificador 30 de vídeo puede marcar todas las imágenes de referencia de la DPB como no usadas para referencia antes de descodificar la imagen BLA cuando se descodifica una imagen BLA o BLC. Por ejemplo, el descodificador 30 de vídeo puede marcar imágenes de referencia de la memoria intermedia de imágenes descodificadas (DPB) como no usadas para referencia.

[0074] En otro ejemplo, el codificador 20 puede incluir en el flujo de bits, y el descodificador 30 puede recibir, un señalizador u otro elemento sintáctico, por ejemplo, un `no_output_of_prior_pics_flag`, para una imagen BLA que se va a descodificar. Cuando el señalizador es igual a 1, indica que no se facilita ninguna de las imágenes anteriores en la DPB para su visualización. En particular, cuando `no_output_of_prior_pics_flag` es igual a 1, el descodificador 30 vacía todas las memorias intermedias de almacenamiento de imágenes de la memoria intermedia de imágenes descodificadas sin facilitar las imágenes que contienen. En algunos ejemplos, este señalizador o elemento sintáctico puede presentarse muy al principio de la cabecera de fragmento antes de la descodificación de entropía, de tal forma que puede descodificarse más fácilmente, sin necesidad de codificación de entropía, y la información está disponible más pronto en el proceso de codificación. Por ejemplo, a fin de que los dispositivos menos avanzados, como un MANE, puedan tener acceso a la información sin necesidad de un descodificador ya que, por ejemplo, no se necesita realizar la descodificación de entropía de `no_output_of_prior_pics_flag`.

[0075] En otro ejemplo, el descodificador 30 de vídeo puede procesar tipos de unidad NAL asignados para indicar cuándo las imágenes BLA o BLC tienen y no tienen imágenes iniciales. (De nuevo, como se ha descrito anteriormente, las imágenes BLA son, conceptualmente y en general, iguales que las imágenes BLC, en la medida en que las imágenes BLA y BLC representan imágenes CRA de enlace roto.) En un ejemplo, el descodificador 30 de vídeo puede descodificar imágenes de acuerdo con uno de una pluralidad de tipos diferentes de unidades de capa de abstracción de red (NAL). La pluralidad de tipos de unidades NAL incluye uno o más de (1) un fragmento codificado de una imagen de acceso de enlace roto (BLA), en la que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, (2) un fragmento codificado de una imagen BLA, en la que la imagen BLA es una imagen BLA con imágenes iniciales descodificables asociadas en el flujo de bits, y (3) un fragmento codificado de una imagen BLA, en la que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits.

[0076] En algunos ejemplos, el descodificador 30 de vídeo está configurado para marcar todas las imágenes de referencia como no usadas para referencia si detecta un tipo de unidad BLA NAL. Si el descodificador 30 detecta un tipo de unidad BLA NAL, las imágenes de referencia no son válidas para descodificar la imagen BLA o cualquier imagen que sigue a la imagen BLA en orden de descodificación o de salida.

[0077] El dispositivo 32 de visualización puede estar integrado en, o ser externo a, el dispositivo 14 de destino. En algunos ejemplos, el dispositivo 14 de destino puede incluir un dispositivo de visualización integrado y también estar configurado para interconectarse con un dispositivo de visualización externo. En otros ejemplos, el dispositivo 14 de destino puede ser un dispositivo de visualización. En general, el dispositivo 32 de visualización 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.

[0078] El codificador 20 de vídeo y el descodificador 30 de vídeo pueden funcionar de acuerdo con una norma de compresión de vídeo, tal como la norma de codificación de vídeo de alta eficiencia (HEVC), actualmente en fase de elaboración, y pueden ajustarse al modelo de prueba HEVC (HM). De forma alternativa, el codificador 20 de vídeo y el descodificador 30 de vídeo pueden funcionar de acuerdo con otras normas de propiedad o industriales, tales como la norma ITU-T H.264, de forma alternativa denominada MPEG-4, parte 10, codificación avanzada de vídeo (AVC), o ampliaciones de dichas normas. Sin embargo, las técnicas de esta divulgación no están limitadas a ninguna norma de codificación particular. Otros ejemplos de normas de compresión de vídeo incluyen MPEG-2 e ITU-T H.263.

[0079] Aunque no se muestra en la figura 1, en algunos aspectos, el codificador 20 de vídeo y el descodificador 30 de vídeo pueden estar integrados, cada uno de ellos, en 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 pueden ajustarse al protocolo de multiplexador ITU H.223 o a otros protocolos, tales como el protocolo de datagrama de usuario (UDP).

[0080] El codificador 20 de vídeo y el descodificador 30 de vídeo pueden implementarse, cada uno de ellos, como cualquiera de entre una variedad de circuitos de codificador adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables *in situ* (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de estos. Cuando las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio no transitorio legible por ordenador adecuado, y ejecutar las instrucciones en hardware mediante uno o más procesadores para realizar las técnicas de esta divulgación. Cada uno del codificador 20 de vídeo y del descodificador 30 de vídeo pueden estar incluidos en uno o más codificadores o descodificadores, cualquiera de los cuales puede estar integrado como parte de un codificador/descodificador (CÓDEC) combinado en un dispositivo respectivo.

[0081] El equipo JCT-VC está trabajando en la elaboración de la norma HEVC. La iniciativa de normalización HEVC se basa en un modelo en evolución de un dispositivo de codificación de vídeo denominado modelo de prueba HEVC (HM). El HM supone varias capacidades adicionales de los dispositivos de codificación de vídeo respecto a dispositivos existentes de acuerdo con, por ejemplo, la norma ITU-T H.264/AVC. Por ejemplo, mientras que la norma H.264 proporciona nueve modos de codificación de intrapredicción, el HM puede proporcionar hasta treinta y tres modos de codificación de intrapredicción.

[0082] En general, el modelo de explotación del HM describe que una trama o imagen de vídeo puede dividirse en una secuencia de bloques de árbol o unidades de codificación de máximo tamaño (LCU), que incluyen muestras tanto de luma como de croma. Un bloque de árbol tiene un fin similar al de un macrobloque de la norma H.264. Un fragmento incluye un número de bloques de árbol consecutivos en orden de codificación. Una trama o imagen de vídeo puede dividirse en uno o más fragmentos. Cada bloque de árbol puede dividirse en unidades de codificación (CU) de acuerdo con un árbol cuaternario. Por ejemplo, un bloque de árbol, como un nodo raíz del árbol cuaternario, puede dividirse en cuatro nodos hijo, y cada nodo hijo puede, a su vez, ser un nodo padre y dividirse en otros

cuatro nodos hijo. Un nodo hijo final no dividido, como un nodo hoja del árbol cuaternario, comprende un nodo de codificación, es decir, un bloque de vídeo codificado. Los datos sintácticos asociados a un flujo de bits codificado pueden definir un número máximo de veces que puede dividirse un bloque de árbol, y también pueden definir un tamaño mínimo de los nodos de codificación.

5

[0083] 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 ser de forma cuadrada. El tamaño de la CU puede variar desde 8x8 píxeles hasta el tamaño del bloque de árbol, con un máximo de 64x64 píxeles o más. Cada CU puede contener una o más PU y una o más TU. Los datos sintácticos asociados a una CU pueden describir, por ejemplo, la división de la CU en una o más PU. Los modos de división pueden diferir dependiendo de si la CU está codificada en modo de salto o directo, codificada en modo de intrapredicción o codificada en modo de interpredicción. Las PU pueden dividirse para tener forma no cuadrada. Los datos sintácticos asociados a una CU también pueden describir, por ejemplo, la división de la CU en una o más TU de acuerdo con un árbol cuaternario. Una TU puede tener forma cuadrada o no cuadrada.

10

15

[0084] La norma HEVC admite transformaciones de acuerdo con las TU, que pueden ser diferentes para diferentes CU. El tamaño de las TU típicamente se basa en el tamaño de las PU de una CU dada, definida para una LCU dividida, aunque puede que no sea siempre así. Las TU son típicamente del mismo tamaño o de un tamaño más pequeño que las PU. En algunos ejemplos, las muestras residuales correspondientes a una CU pueden subdividirse en unidades más pequeñas mediante una estructura de árbol cuaternario conocida como «árbol cuaternario residual» (RQT). Los nodos hoja del RQT pueden denominarse unidades de transformada (TU). Los valores de diferencia de píxeles asociados a las TU pueden transformarse para generar coeficientes de transformada, que pueden cuantificarse.

20

25

[0085] En general, una PU incluye datos relacionados con el proceso de predicción. Por ejemplo, cuando la PU se codifica mediante el modo intra, la PU puede incluir datos que describen un modo de intrapredicción para la PU. En otro ejemplo, cuando la PU se codifica mediante el modo inter, 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, un componente horizontal del vector de movimiento, un componente vertical del vector de movimiento, una resolución para el vector de movimiento (por ejemplo, precisión de un cuarto de píxel o precisión de un octavo de píxel), una imagen de referencia a la que apunta el vector de movimiento y/o una lista de imágenes de referencia (por ejemplo, lista 0, lista 1 o lista C) para el vector de movimiento.

30

35

[0086] En general, se usa una TU para los procesos de transformada y cuantificación. Una CU determinada que tiene una o más PU también puede incluir una o más unidades de transformada (TU). Tras la predicción, el codificador 20 de vídeo puede calcular valores residuales correspondientes a la PU. Los valores residuales comprenden valores de diferencia de píxeles que se pueden transformar en coeficientes de transformada, cuantificar y examinar mediante las TU, para generar coeficientes de transformada en serie para la codificación de entropía. Esta divulgación usa típicamente el término «bloque de vídeo» para referirse a un nodo de codificación de una CU. En algunos casos específicos, esta divulgación también puede usar el término «bloque de vídeo» para referirse a un bloque de árbol, es decir, una LCU o una CU, que incluye un nodo de codificación y unas PU y TU.

40

45

[0087] Una secuencia de vídeo incluye típicamente una serie de tramas o imágenes de vídeo. Un grupo de imágenes (GOP) comprende, en general, una serie de una o más de las imágenes de vídeo. Un GOP puede incluir datos sintácticos en una cabecera del GOP, en una cabecera de una o más de las imágenes, o en otras ubicaciones, que describen un número 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. Un codificador 20 de vídeo actúa típicamente sobre bloques de vídeo dentro de fragmentos de vídeo individuales con el fin de codificar los datos de vídeo. Un bloque de vídeo puede corresponder a un nodo de codificación dentro de una CU. Los bloques de vídeo pueden tener tamaños fijos o variables y pueden diferir en tamaño de acuerdo con una norma de codificación especificada.

50

55

[0088] En un ejemplo, el HM admite la predicción en diversos tamaños de PU. Si se supone que el tamaño de una CU particular es 2Nx2N, el HM admite intrapredicción en tamaños de PU de 2Nx2N o NxN e interpredicción en tamaños de PU simétricos de 2Nx2N, 2NxN, Nx2N o NxN. El HM también admite la división asimétrica para la interpredicción en tamaños de PU de 2Nx n U, 2Nx n D, n Lx2N y n Rx2N. En la división asimétrica, una dirección de una CU no está dividida, mientras que la otra dirección está dividida en el 25 % y el 75 %. La parte de la CU correspondiente a la división del 25 % está indicada por una «n» seguida de una indicación de «arriba», «abajo», «izquierda» o «derecha». Así pues, por ejemplo, «2Nx n U» se refiere a una CU 2Nx2N que está dividida horizontalmente con una PU 2Nx0,5N encima y una PU 2Nx1,5N debajo.

60

65

[0089] En esta divulgación, «NxN» y «N por N» pueden usarse indistintamente para hacer referencia a las dimensiones de píxeles de un bloque de vídeo en términos de dimensiones verticales y horizontales, por ejemplo, 16x16 píxeles o 16 por 16 píxeles. En general, un bloque 16x16 tendrá 16 píxeles en una dirección vertical ($y = 16$) y 16 píxeles en una dirección horizontal ($x = 16$). Asimismo, un bloque NxN tiene, en general, N píxeles en una dirección vertical y N píxeles en una dirección horizontal, donde N representa un valor entero no negativo. Los

píxeles de un bloque pueden disponerse en filas y columnas. Además, no es necesario que los bloques tengan necesariamente el mismo número de píxeles en la dirección horizontal y en la dirección vertical. Por ejemplo, los bloques pueden comprender $N \times M$ píxeles, donde M no es necesariamente igual a N .

5 **[0090]** Tras la codificación intrapredictiva o interpredictiva mediante las PU de una CU, el codificador 20 de vídeo puede calcular datos residuales para las TU de la CU. Las PU pueden comprender datos de píxeles en el dominio espacial (también denominado dominio del píxel) y las TU pueden comprender coeficientes en el dominio de la transformada tras la aplicación de una transformada, por ejemplo, una transformada discreta de coseno (DCT), una transformada de enteros, una transformada de *wavelet* o una transformada conceptualmente similar a los
10 datos de vídeo residuales. Los datos residuales pueden corresponder a diferencias de píxel entre píxeles de la imagen no codificada y los valores de predicción correspondientes a las PU. El codificador 20 de vídeo puede formar las TU incluyendo los datos residuales para la CU y, a continuación, transformar las TU para generar coeficientes de transformada para la CU.

15 **[0091]** Tras cualquier transformada para generar coeficientes de transformada, el codificador 20 de vídeo puede realizar la cuantificación de los coeficientes de transformada. La cuantificación se refiere, en general, a un proceso en el que los 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 una parte o la totalidad de los coeficientes. Por ejemplo, un valor de n bits
20 puede redondearse a la baja a un valor de m bits durante la cuantificación, donde n es mayor que m .

[0092] En algunos ejemplos, el codificador 20 de vídeo puede utilizar un orden de examen predefinido para examinar los coeficientes de transformada cuantificados a fin de generar un vector en serie que se pueda someter a codificación de entropía. En otros ejemplos, el codificador 20 de vídeo puede realizar un examen adaptativo.
25 Después de examinar los coeficientes de transformada cuantificados para formar un vector unidimensional, el codificador 20 de vídeo puede realizar la codificación de entropía del vector unidimensional, por ejemplo, de acuerdo con la codificación de longitud variable adaptativa según el contexto (CAVLC), la codificación aritmética binaria adaptativa según el contexto (CABAC), la codificación aritmética binaria adaptativa según el contexto basada en la sintaxis (SBAC), la codificación de entropía de división de intervalo de probabilidad (PIPE) u otros
30 procedimientos de codificación de entropía. El codificador 20 de vídeo también puede realizar la codificación de entropía de los elementos sintácticos asociados a los datos de vídeo codificados, para su uso por el descodificador 30 de vídeo en la descodificación de los datos de vídeo.

[0093] Para realizar la CABAC, el codificador 20 de vídeo puede asignar un contexto de un modelo de contexto a un símbolo que se va a transmitir. El contexto puede indicar, por ejemplo, si los valores contiguos del símbolo son distintos de cero o no. Para realizar la CAVLC, el codificador 20 de vídeo puede seleccionar un código de longitud variable para un símbolo que se va a transmitir. Las palabras de código en la VLC pueden construirse de tal manera que los códigos relativamente más cortos correspondan a símbolos más probables, y los códigos más largos correspondan a símbolos menos probables. De esta manera, el uso de la VLC puede lograr un ahorro en bits con respecto, por ejemplo, al uso de palabras de código de igual longitud para cada símbolo que se va a transmitir. La determinación de la probabilidad puede basarse en un contexto asignado al símbolo.
40

[0094] La figura 2 es un diagrama de bloques que ilustra un ejemplo de codificador 20 de vídeo que puede implementar las técnicas para codificar imágenes RAP de enlace roto descritas en esta divulgación. El codificador 20 de vídeo puede realizar intracodificación e intercodificación de bloques de vídeo dentro de fragmentos de vídeo. La intracodificación se basa en la predicción espacial para reducir o eliminar la redundancia espacial en el vídeo dentro de una trama o imagen de vídeo determinada. La intercodificación se basa en la predicción temporal para reducir o eliminar la redundancia temporal en el vídeo dentro de tramas o imágenes adyacentes de una secuencia de vídeo. El modo intra (modo I) puede referirse a cualquiera de varios modos de compresión espacial. Los modos inter, tales como la predicción unidireccional (modo P) o la predicción bidireccional (modo B), pueden referirse a cualquiera de varios modos de compresión temporal.
50

[0095] En el ejemplo de la figura 2, el codificador 20 de vídeo incluye un módulo 35 de división, un módulo 41 de predicción, un módulo 63 de filtro, una memoria 64 de imágenes de referencia, un sumador 50, un módulo 52 de transformada, un módulo 54 de cuantificación y un módulo 56 de codificación de entropía. El módulo 41 de predicción incluye un módulo 42 de estimación de movimiento, un módulo 44 de compensación de movimiento y un módulo 46 de intrapredicción. Para la reconstrucción de bloques de vídeo, el codificador 20 de vídeo incluye además un módulo 58 de cuantificación inversa, un módulo 60 de transformada inversa y un sumador 62. El módulo 63 de filtro 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 63 de filtro que se ilustra en la figura 2 es un filtro en bucle, en otras configuraciones, el módulo 63 de filtro puede implementarse como un filtro posbucle.
60

[0096] Como se representa en la figura 2, el codificador 20 de vídeo recibe datos de vídeo, y el módulo 35 de división 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 de bloques de vídeo, por ejemplo, de acuerdo con una
65

estructura de árbol cuaternario de LCU y CU. El codificador 20 de vídeo ilustra, en general, los componentes que codifican bloques de vídeo de un fragmento de vídeo que se va a codificar. El fragmento puede dividirse en múltiples bloques de vídeo (y, posiblemente, en conjuntos de bloques de vídeo denominados mosaicos). El módulo 41 de predicción 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 41 de predicción 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 usar como imagen de referencia.

[0097] Como se ha analizado anteriormente, en algunos casos, el codificador 20 de vídeo puede codificar un punto de acceso aleatorio o un punto de adaptación de flujo, tal como un punto de conmutación de capa temporal, tal como una imagen BLA o BLC. Por ejemplo, la codificación puede tener lugar dentro del módulo 56 de codificación de entropía, que puede realizar codificación de entropía y no entropía. Una o más de estas imágenes pueden ser imágenes iniciales de la imagen CRA. Las imágenes iniciales de una imagen CRA se pueden descodificar correctamente si la descodificación comienza por una imagen RAP anterior a la imagen CRA actual. No obstante, las imágenes iniciales de una imagen CRA no se pueden descodificar correctamente cuando se produce un acceso aleatorio desde la imagen CRA. Por ejemplo, las imágenes iniciales pueden apuntar a bloques para referencia de predicción que no están disponibles. En consecuencia, la imagen inicial puede no ser descodificable en el descodificador 30 de vídeo. En consecuencia, estas imágenes iniciales se descartan típicamente durante la descodificación de acceso aleatorio.

[0098] En un ejemplo, el codificador 20 de vídeo puede proporcionar un señalizador en la cabecera de fragmento, por ejemplo, `no_output_of_prior_pics_flag` o un elemento sintáctico de modo que no se facilita ninguna de las imágenes anteriores, es decir, imágenes anteriores a la imagen BLA o BLC, en la DPB. En algunos ejemplos, este señalizador (o elemento sintáctico) puede hallarse al principio de la cabecera de fragmento antes de la codificación de entropía, de modo que puede descodificarse más fácilmente en el descodificador 30, por ejemplo, y la información puede estar disponible más pronto en el proceso de codificación. El elemento sintáctico o señalizador puede ser codificado, por ejemplo, en una cabecera de fragmento para una imagen BLA o BLC, por el módulo 56 de codificación de entropía (que puede realizar codificación no de entropía). Esto puede ser útil, por ejemplo, para dispositivos intermedios tales como un MANE, de modo que la información proporcionada por el elemento sintáctico o señalizador puede estar disponible para el dispositivo intermedio sin descodificación de entropía, aunque también puede ser útil para que un descodificador tenga acceso a dicha información antes de la descodificación de entropía.

[0099] Por ejemplo, el codificador 20 de vídeo (por ejemplo, el módulo 56 de codificación de entropía) puede incluir un elemento sintáctico que indica que una memoria intermedia de almacenamiento de imágenes se vacía sin facilitar ninguna imagen de la memoria intermedia de almacenamiento de imágenes. El elemento sintáctico, una vez establecido, puede hacer que las imágenes, anteriores a una imagen actual en orden de descodificación y que residen en la memoria intermedia de almacenamiento de imágenes en el momento de la descodificación de la imagen actual, se vacíen de la memoria intermedia de almacenamiento de imágenes sin ser facilitadas. En algunos ejemplos, el elemento sintáctico puede ser uno de una pluralidad de elementos sintácticos. De forma adicional, la pluralidad de elementos sintácticos puede incluir elementos sintácticos de cabecera de fragmento sometidos a codificación de entropía y elementos sintácticos de cabecera de fragmento no sometidos a codificación de entropía. En un ejemplo, el elemento sintáctico, que indica que una memoria intermedia de almacenamiento de imágenes se vacía sin facilitar ninguna imagen de la memoria intermedia de almacenamiento de imágenes, está incluido en la cabecera del fragmento antes de cualquier elemento sintáctico de cabecera de fragmento sometido a codificación de entropía de modo que no se realiza la codificación de entropía del propio elemento sintáctico.

[0100] En algunos ejemplos, el elemento sintáctico puede ser `no_output_of_prior_pics_flag` y el señalizador `no_output_of_prior_pics_flag` puede estar incluido en la cabecera del fragmento justo después de un señalizador `first_slice_in_pic_flag`. El señalizador `first_slice_in_pic_flag` puede ser un señalizador que indica si un fragmento es el primer fragmento, en orden de descodificación, de una imagen.

[0101] En un ejemplo, un codificador 20 de vídeo puede estar configurado para incluir tipos de unidad NAL asignados para indicar cuándo las imágenes BLA o BLC tienen y no tienen imágenes iniciales. Por ejemplo, en una norma, los tipos de unidad NAL 16, `BLA_W_LP` (BLA con imagen inicial); 17, `BLA_W_DLP` (BLA con imagen inicial descodificable); y 18, `BLA_N_LP` (BLA sin imagen inicial) están incluidos. El módulo 56 de codificación de entropía (que puede realizar una codificación no de entropía) puede codificar estos tipos de unidad NAL. En consecuencia, basándose en el tipo de unidad NAL, un descodificador puede saber cuándo una imagen BLA tiene una imagen inicial y cuándo la imagen inicial no es descodificable, por ejemplo, cuando la descodificación comienza por una imagen RAP anterior. En consecuencia, esta información podría usarse para determinar cuándo las imágenes iniciales pueden marcarse como no usadas para referencia, lo que puede hacer que el descodificador se active para marcar imágenes de referencia de la memoria intermedia de imágenes descodificadas como no usadas para referencia.

[0102] El módulo 46 de intrapredicción, dentro del módulo 41 de predicción, 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 va a codificar para proporcionar compresión espacial. El módulo 42 de estimación de movimiento y el módulo 44 de compensación de movimiento, dentro del módulo 41 de predicción, realizan la codificación interpredictiva del bloque de vídeo actual con respecto a uno o más bloques para referencia de predicción en una o más imágenes de referencia, para proporcionar compresión temporal.

[0103] El módulo 42 de estimación de movimiento 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 de la secuencia como fragmentos P, fragmentos B o fragmentos GPB. El módulo 42 de estimación de movimiento y el módulo 44 de compensación de movimiento pueden estar altamente integrados, pero se ilustran por separado con fines conceptuales. La estimación de movimiento, realizada por el módulo 42 de estimación de movimiento, 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 el desplazamiento de una PU de un bloque de vídeo de una trama o imagen de vídeo actual a un bloque predictivo de una imagen de referencia.

[0104] Un bloque predictivo es un bloque que se observa que corresponde estrechamente con la PU del bloque de vídeo que se va a codificar en términos de diferencia de píxel, que puede determinarse mediante una suma de una diferencia absoluta (SAD), una suma de diferencia al cuadrado (SSD) u otras métricas de diferencia. En algunos ejemplos, el codificador 20 de vídeo puede calcular valores para posiciones de píxel de subentero de imágenes de referencia almacenadas en la memoria 64 de imágenes de referencia. Por ejemplo, el codificador 20 de vídeo puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones de píxel fraccionario de la imagen de referencia. Por lo tanto, el módulo 42 de estimación de movimiento puede realizar una búsqueda de movimiento con respecto a las posiciones de píxel completo y a las posiciones de píxel fraccionario, y facilitar un vector de movimiento con una precisión de píxel fraccionario.

[0105] El módulo 42 de estimación de movimiento calcula un vector de movimiento para una PU de un bloque de vídeo en un fragmento intercodificado, comparando la posición de la PU con la posición de un bloque predictivo de una imagen de referencia. La imagen de referencia puede seleccionarse de una primera lista de imágenes de referencia (lista 0) o una segunda lista de imágenes de referencia (lista 1), cada una de las cuales identifica una o más imágenes de referencia almacenadas en la memoria 64 de imágenes de referencia. El módulo 42 de estimación de movimiento envía el vector de movimiento calculado al módulo 56 de codificación de entropía y al módulo 44 de compensación de movimiento.

[0106] La compensación de movimiento, realizada por el módulo 44 de compensación de movimiento, puede implicar extraer 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 44 de compensación de movimiento puede localizar el bloque predictivo al que apunta el vector de movimiento en una de las listas de imágenes de referencia. El codificador 20 de vídeo forma un bloque de vídeo residual restando los valores de píxel del bloque predictivo a los valores de píxel del bloque de vídeo actual que se está codificando, generando valores de diferencia de píxel. Los valores de diferencia de píxel forman datos residuales para el bloque, y pueden incluir componentes de diferencia de luma y croma. El sumador 50 representa el componente o los componentes que realizan esta operación de resta. El módulo 44 de compensación de movimiento también puede generar elementos sintácticos asociados a los bloques de vídeo y al fragmento de vídeo para su uso por el descodificador 30 de vídeo en la descodificación de los bloques de vídeo del fragmento de vídeo.

[0107] El módulo 46 de intrapredicción puede realizar la intrapredicción de un bloque actual, como alternativa a la interpredicción realizada por el módulo 42 de estimación de movimiento y el módulo 44 de compensación de movimiento, descrita anteriormente. En particular, el módulo 46 de intrapredicción puede determinar un modo de intrapredicción que se va a usar para codificar un bloque actual. En algunos ejemplos, el módulo 46 de intrapredicción puede codificar un bloque actual usando diversos modos de intrapredicción, por ejemplo, durante pasadas de codificación independientes, y el módulo 46 de intrapredicción (o el módulo 40 de selección de modo, en algunos ejemplos) puede seleccionar un modo de intrapredicción adecuado que se va a usar de entre los modos probados. Por ejemplo, el módulo 46 de intrapredicción puede calcular valores de distorsión de velocidad mediante un análisis de distorsión de velocidad para los diversos modos de intrapredicción probados, y seleccionar el modo de intrapredicción que tiene las mejores características de distorsión de velocidad de entre los modos probados. El análisis de distorsión de velocidad determina, en general, una cantidad de distorsión (o error) entre un bloque codificado y un bloque original no codificado que se ha codificado para generar el bloque codificado, así como una velocidad de bits (es decir, un número de bits) usada para generar el bloque codificado. El módulo 46 de intrapredicción puede calcular razones a partir de las distorsiones y velocidades para los diversos bloques codificados, a fin de determinar qué modo de intrapredicción presenta el mejor valor de distorsión de velocidad para el bloque.

[0108] En cualquier caso, tras seleccionar un modo de intrapredicción para un bloque, el módulo 46 de intrapredicción puede proporcionar información que indica el modo de intrapredicción seleccionado para el bloque al módulo 56 de codificación de entropía. El módulo 56 de codificación de entropía puede codificar la información que indica el modo de intrapredicción seleccionado de acuerdo con las técnicas de esta divulgación. El codificador 5 20 de vídeo puede incluir datos de configuración en el flujo de bits transmitido. El flujo de bits puede incluir una pluralidad de tablas de índices de modos de intrapredicción y una pluralidad de tablas de índices de modos de intrapredicción modificadas (también denominadas tablas de correlación de palabras de código), definiciones de contextos de codificación para diversos bloques e indicaciones del modo de intrapredicción más probable, una 10 tabla de índices de modos de intrapredicción y una tabla de índices de modos de intrapredicción modificada que se va a usar para cada uno de los contextos.

[0109] Después de que el módulo 41 de predicción 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 residual del bloque residual pueden 15 incluirse en una o más TU y aplicarse al módulo 52 de transformada. El módulo 52 de transformada transforma los datos de vídeo residual en coeficientes de transformada residual mediante una transformada, tal como una transformada de coseno discreta (DCT) o una transformada conceptualmente similar. El módulo 52 de transformada puede convertir los datos de vídeo residual de un dominio de píxel a un dominio de transformada, tal como un dominio de frecuencia.

[0110] El módulo 52 de transformada puede enviar los coeficientes de transformada resultantes al módulo 54 de 20 cuantificación. El módulo 54 de cuantificación cuantifica los coeficientes de transformada para reducir más la velocidad de bits. El proceso de cuantificación puede reducir la profundidad de bits asociada con una parte o la totalidad de los coeficientes. El grado de cuantificación puede modificarse ajustando un parámetro de 25 cuantificación. En algunos ejemplos, el módulo 54 de cuantificación puede realizar, a continuación, un examen de la matriz que incluye los coeficientes de transformada cuantificados. De forma alternativa, el módulo 56 de codificación de entropía puede realizar el examen.

[0111] Tras la cuantificación, el módulo 56 de codificación de entropía realiza la codificación de entropía de los 30 coeficientes de transformada cuantificados. Por ejemplo, el módulo 56 de codificación de entropía 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 de entropía de división de intervalo de probabilidad (PIPE) u otra metodología o técnica de codificación de entropía. Tras la codificación de entropía realizada por el módulo 56 de codificación 35 de entropía, el flujo de bits codificado puede transmitirse al descodificador 30 de vídeo, o archivar para su posterior transmisión o recuperación por el descodificador 30 de vídeo. El módulo 56 de codificación de entropía también puede realizar la codificación de entropía de los vectores de movimiento y de los otros elementos sintácticos para el fragmento de vídeo actual que se está codificando.

[0112] El módulo 58 de cuantificación inversa y el módulo 60 de transformada inversa aplican una cuantificación 40 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 44 de compensación de movimiento 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 44 de compensación de 45 movimiento 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 generado por el módulo 44 de compensación de movimiento para generar un bloque de referencia para su almacenamiento en la memoria 64 de 50 imágenes de referencia. El módulo 42 de estimación de movimiento y el módulo 44 de compensación de movimiento 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.

[0113] El codificador 20 de vídeo de la figura 2 representa un ejemplo de codificador de vídeo configurado para 55 codificar un señalizador u otro elemento sintáctico para indicar cuándo las imágenes de referencia antes de una imagen BLA en orden de descodificación no pueden usarse como imágenes de referencia de acuerdo con las técnicas de esta divulgación.

[0114] La figura 3 es un diagrama de bloques que ilustra un descodificador 30 de vídeo de ejemplo que puede 60 implementar las técnicas descritas para un apoyo mejorado de adaptación y empalme de flujos basándose en las imágenes RAP de enlace roto descritas anteriormente. En el ejemplo de la figura 3, el descodificador 30 de vídeo incluye un módulo 80 de descodificación de entropía, un módulo 81 de predicción, un módulo 86 de cuantificación inversa, un módulo 88 de transformada inversa, un sumador 90, un módulo 91 de filtro y una memoria 92 de 65 imágenes de referencia. El módulo 81 de predicción incluye un módulo 82 de compensación de movimiento y un módulo 84 de intrapredicción. En algunos ejemplos, el descodificador 30 de vídeo puede realizar una pasada de descodificación que en general es recíproca a la pasada de codificación descrita con respecto al codificador 20 de vídeo de la figura 2.

[0115] Durante el proceso de descodificación, el descodificador 30 de vídeo recibe un flujo de bits de vídeo codificado que representa bloques de vídeo de un fragmento de vídeo codificado y elementos sintácticos asociados, desde el codificador 20 de vídeo. El descodificador 30 de vídeo puede recibir el flujo de bits de vídeo codificado desde una entidad 29 de red. La entidad 29 de red puede ser, por ejemplo, un servidor, un elemento de red adaptable a los medios (MANE), un editor/empalmador de vídeo u otro dispositivo de este tipo configurado para implementar una o más de las técnicas descritas anteriormente. La entidad 39 de red puede incluir o no un codificador 20 de vídeo. Como se ha descrito anteriormente, algunas de las técnicas descritas en esta divulgación pueden ser implementadas por la entidad 29 de red antes de que la entidad 29 de red transmita el flujo de bits de vídeo codificado al descodificador 30 de vídeo. En algunos sistemas de descodificación de vídeo, la entidad 29 de red y el descodificador 30 de vídeo pueden ser partes de dispositivos separados, mientras que en otros casos la funcionalidad descrita con respecto a la entidad 29 de red puede ser realizada por el mismo dispositivo que comprende el descodificador 30 de vídeo.

[0116] Como se ha analizado anteriormente, en algunos ejemplos, el descodificador 30 de vídeo puede descodificar una imagen RAP de enlace roto, tal como una imagen BLA, o imágenes que se hallan después de un punto de acceso aleatorio o un punto de adaptación de flujo, tal como un punto de conmutación de capa temporal, como unas imágenes iniciales. Las imágenes iniciales no se pueden descodificar correctamente cuando se produce un acceso aleatorio desde la imagen RAP de enlace roto.

[0117] En un ejemplo, para evitar la propagación de errores por imágenes de referencia que pueden no estar disponibles dependiendo de dónde comienza la descodificación, el descodificador 30 de vídeo no puede usar ninguna imagen que preceda a la imagen RAP de enlace roto en orden de descodificación u orden de salida (lo cual incluye las imágenes iniciales) como imagen de referencia. Por ejemplo, el módulo 81 de predicción no puede usar ninguna imagen almacenada en la memoria 92 de imágenes de referencia que preceda a la imagen RAP de enlace roto en orden de descodificación u orden de salida (lo cual incluye las imágenes iniciales) como imagen de referencia.

[0118] En diversos ejemplos, el descodificador 30 de vídeo puede marcar todas las imágenes de referencia de la DPB como no usadas para referencia antes de descodificar la imagen BLA como no usada para referencia. Por ejemplo, el módulo 80 de descodificación de entropía, que puede realizar descodificación de entropía y no entropía, puede marcar imágenes de referencia de la memoria 92 de imágenes de referencia, a veces denominada memoria intermedia de imágenes descodificadas (DPB), como no usadas para referencia. El descodificador 30 de vídeo (por ejemplo, el módulo 80 de descodificación de entropía) puede determinar que una imagen actual es una imagen BLA o BLC y marcar las imágenes de referencia de una memoria intermedia de almacenamiento de imágenes como no usadas para referencia antes de descodificar la imagen BLA o BLC. Determinar que una imagen actual es una imagen BLA o BLC puede incluir determinar que la imagen actual es una imagen CRA y determinar que la imagen actual es una imagen RAP. La imagen CRA actual es una imagen BLA cuando la imagen actual es tanto una imagen CRA como una imagen RAP. En algunos ejemplos, esta determinación por el descodificador 30 podría realizarse basándose en que la imagen tiene un tipo de unidad BLA NAL.

[0119] En otro ejemplo, cuando se descodifica una imagen BLA, el descodificador 30 puede recibir un señalizador o elemento sintáctico en el flujo de bits codificado, por ejemplo, `no_output_of_prior_pics_flag`, de modo que no se facilita ninguna de las imágenes anteriores de la DPB. En algunos ejemplos, este señalizador puede presentarse al principio de una cabecera de fragmento, de un fragmento de una imagen BLA, antes de la descodificación de entropía de modo que puede descodificarse más fácilmente y la información está disponible más pronto en el proceso de codificación. El módulo 80 de descodificación de entropía, que puede realizar descodificación de entropía y no entropía, puede descodificar el señalizador o elemento sintáctico. Colocar el señalizador u otro elemento sintáctico al principio de la cabecera de fragmento antes de la codificación de entropía puede permitir que dispositivos menos avanzados, como un MANE, tengan acceso a la información sin necesidad de un descodificador de entropía ya que, por ejemplo, no es necesario realizar la descodificación de entropía de `no_output_of_prior_pics_flag`.

[0120] En un ejemplo, el descodificador 30 de vídeo puede colocar un elemento sintáctico, por ejemplo, `no_output_of_prior_pics_flag`, en el flujo de bits para que lo reciba un descodificador. El elemento sintáctico puede indicar que una memoria intermedia de almacenamiento de imágenes se vacía sin facilitar ninguna imagen de la memoria intermedia de almacenamiento de imágenes. El elemento sintáctico, una vez establecido, puede hacer que las imágenes, anteriores a una imagen actual en orden de descodificación y que residen en la memoria intermedia de almacenamiento de imágenes en el momento de la descodificación de la imagen actual, se vacíen de la memoria intermedia de almacenamiento de imágenes sin ser facilitadas. En algunos ejemplos, el elemento sintáctico puede ser uno de una pluralidad de elementos sintácticos. De forma adicional, la pluralidad de elementos sintácticos puede incluir uno o más elementos sintácticos de cabecera de fragmento sometidos a codificación de entropía y uno o más elementos sintácticos de cabecera de fragmento sometidos a codificación no de entropía. En un ejemplo, el elemento sintáctico que indica que una memoria intermedia de almacenamiento de imágenes se vacía sin facilitar ninguna imagen de la memoria intermedia de almacenamiento de imágenes se incluye en la cabecera de fragmento, por ejemplo, como elemento `u(1)`, antes de cualquier elemento sintáctico de cabecera de

fragmento sometido a codificación de entropía, por ejemplo, antes de cualquier elemento ue(v). En algunos ejemplos, el elemento sintáctico puede ser `no_output_of_prior_pics_flag` y el señalizador `no_output_of_prior_pics_flag` se puede incluir en la cabecera de fragmento justo después de `first_slice_in_pic_flag` y antes de cualquier elemento sometido a codificación de entropía. El señalizador `first_slice_in_pic_flag` puede ser un señalizador que indica si un fragmento es el primer fragmento, en orden de descodificación, de una imagen.

[0121] En otro ejemplo, el descodificador 30 de vídeo puede procesar tipos de unidad NAL asignados para indicar cuándo las imágenes BLA tienen y no tienen imágenes iniciales. El descodificador 30 de vídeo puede estar configurado para incluir tipos de unidad NAL asignados para indicar cuándo las imágenes BLA tienen y no tienen imágenes iniciales. Por ejemplo, el módulo 80 de descodificación de entropía, que puede realizar descodificación de entropía y no entropía, puede procesar los tipos de unidad NAL.

[0122] En un ejemplo, el descodificador 30 de vídeo (por ejemplo, el módulo 80 de descodificación de entropía) puede descodificar imágenes de acuerdo con una de una pluralidad de diferentes tipos de unidades de capa de abstracción de red (NAL). La pluralidad de tipos de unidades NAL incluye uno o más de (1) un fragmento codificado de una imagen de acceso de enlace roto (BLA), en la que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, (2) un fragmento codificado de una imagen BLA, en la que la imagen BLA es una imagen BLA con imágenes iniciales descodificables asociadas en el flujo de bits, y (3) un fragmento codificado de una imagen BLA, en la que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits. En un ejemplo, la imagen inicial comprende una imagen que precede a una imagen de acceso aleatorio (RAP) en un orden de visualización, pero sigue a la imagen de acceso aleatorio en orden de descodificación.

[0123] El módulo 80 de descodificación de entropía del descodificador 30 de vídeo realiza la descodificación de entropía del flujo de bits para generar coeficientes cuantificados, vectores de movimiento y otros elementos sintácticos. El módulo 80 de descodificación de entropía reenvía los vectores de movimiento y otros elementos sintácticos al módulo 81 de predicción. El descodificador 30 de vídeo puede recibir los elementos sintácticos en el nivel del fragmento de vídeo y/o el nivel del bloque de vídeo.

[0124] Cuando el fragmento de vídeo se codifica como un fragmento intracodificado (I), el módulo 84 de intrapredicción del módulo 81 de predicción 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 82 de compensación de movimiento del módulo 81 de predicción genera bloques para referencia de predicción para un bloque de vídeo del fragmento de vídeo actual, basándose en los vectores de movimiento y otros elementos sintácticos recibidos desde el módulo 80 de descodificación de entropía. Los bloques para referencia de predicción pueden generarse a partir de una de las imágenes de referencia de una de las listas de imágenes de referencia. El descodificador 30 de vídeo puede construir las listas de tramas de referencia, lista 0 y lista 1, mediante técnicas de construcción predeterminadas, basándose en imágenes de referencia almacenadas en la memoria 92 de imágenes de referencia.

[0125] El módulo 82 de compensación de movimiento determina la información de predicción para un bloque de vídeo del fragmento de vídeo actual analizando los vectores de movimiento y otros elementos sintácticos, y usa la información de predicción para generar los bloques para referencia de predicción para el bloque de vídeo actual que se está descodificando. Por ejemplo, el módulo 82 de compensación de movimiento usa algunos de los elementos sintácticos recibidos para determinar un modo de predicción (por ejemplo, intrapredicción o interpredicción) usado para codificar los bloques de vídeo del fragmento de vídeo, un tipo de fragmento de interpredicción (por ejemplo, fragmento B, fragmento P o fragmento GPB), información de construcción para una o más de las listas de imágenes de referencia para el fragmento, vectores de movimiento para cada bloque de vídeo intercodificado del fragmento, estado de interpredicción para cada bloque de vídeo intercodificado del fragmento y otra información, para descodificar los bloques de vídeo en el fragmento de vídeo actual. Cuando las imágenes de la DPB están marcadas como no usadas para referencia, no hay imágenes de referencia disponibles. En consecuencia, el descodificador 30 de vídeo no podrá descodificar una imagen inicial que se refiere a una imagen de referencia anterior para la interpredicción.

[0126] El módulo 82 de compensación de movimiento también puede realizar la interpolación basándose en filtros de interpolación. El módulo 82 de compensación de movimiento puede usar filtros de interpolación como los usados por el codificador 20 de vídeo 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 82 de compensación de movimiento puede determinar los filtros de interpolación usados por el codificador 20 de vídeo a partir de los elementos sintácticos recibidos y usar los filtros de interpolación para generar bloques para referencia de predicción.

[0127] El módulo 86 de cuantificación inversa 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 80 de descodificación de entropía. El proceso de cuantificación inversa puede incluir el uso de un parámetro de cuantificación calculado por el codificador 20 de vídeo para cada bloque de vídeo del fragmento de vídeo, a fin de

determinar un grado de cuantificación y, asimismo, un grado de cuantificación inversa que debería aplicarse. El módulo 88 de transformada inversa 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.

[0128] Después de que el módulo 82 de compensación de movimiento genere el bloque predictivo para el bloque de vídeo actual basándose en los vectores de movimiento y otros elementos sintácticos, el descodificador 30 de vídeo forma un bloque de vídeo descodificado sumando los bloques residuales del módulo 88 de transformada inversa y los correspondientes bloques para referencia de predicción generados por el módulo 82 de compensación de movimiento. El sumador 90 representa el componente o los componentes que realizan esta operación de suma. Si se desea, también pueden usarse filtros de bucle (ya sea en el bucle de codificación o después del bucle de codificación) para suavizar las transiciones de píxeles o mejorar de otro modo la calidad del vídeo. El módulo 91 de filtro 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 91 de filtro que se ilustra en la figura 3 es un filtro en bucle, en otras configuraciones, el módulo 91 de filtro puede implementarse como un filtro posbucle. Los bloques de vídeo descodificados de una trama o imagen determinada se almacenan a continuación en la memoria 92 de imágenes de referencia, que almacena imágenes de referencia usadas para una subsiguiente compensación de movimiento. La memoria 92 de imágenes de referencia almacena también vídeo descodificado para su presentación posterior en un dispositivo de visualización, tal como el dispositivo 32 de visualización de la figura 1.

[0129] De esta manera, el descodificador 30 de vídeo de la figura 3 representa un ejemplo de descodificador de vídeo configurado para descodificar los ID de conjunto de parámetros codificados de acuerdo con las técnicas de esta divulgación.

[0130] La figura 4 es un diagrama de bloques que ilustra un conjunto de dispositivos de ejemplo que forman parte de la red 100. En este ejemplo, la red 100 incluye los dispositivos 104A, 104B de encaminamiento (dispositivos 104 de encaminamiento) y el dispositivo 106 de transcodificación. Los dispositivos 104 de encaminamiento y el dispositivo 106 de transcodificación 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 en la red 100. Además, pueden proporcionarse dispositivos de red adicionales a lo largo de un trayecto de red entre el dispositivo servidor 102 y el dispositivo cliente 108. El dispositivo servidor 102 puede corresponder al dispositivo 12 de origen (figura 1), mientras que el dispositivo cliente 108 puede corresponder al dispositivo 14 de destino (figura 1), en algunos ejemplos.

[0131] En general, los dispositivos 104 de encaminamiento implementan uno o más protocolos de encaminamiento para intercambiar datos de red a través de la red 100. En algunos ejemplos, los dispositivos 104 de encaminamiento pueden estar configurados para realizar operaciones de *proxy* o de memoria caché. Por lo tanto, en algunos ejemplos, los dispositivos 104 de encaminamiento pueden denominarse dispositivos *proxy*. En general, los dispositivos 104 de encaminamiento ejecutan protocolos de encaminamiento para descubrir rutas a través de la red 100. Al ejecutar dichos protocolos de encaminamiento, el dispositivo 104B de encaminamiento puede descubrir una ruta de red desde sí mismo hasta el dispositivo servidor 102, a través del dispositivo 104A de encaminamiento.

[0132] Las técnicas de esta divulgación pueden ser implementadas por dispositivos de red tales como los dispositivos 104 de encaminamiento y el dispositivo 106 de transcodificación, pero también pueden ser implementadas por el dispositivo cliente 108. De esta manera, los dispositivos 104 de encaminamiento, el dispositivo 106 de transcodificación y el dispositivo cliente 108 representan ejemplos de dispositivos configurados para realizar las técnicas de esta divulgación. Por otro lado, los dispositivos de la figura 1, y el codificador ilustrado en la figura 2 y el descodificador ilustrado en la figura 3 son también dispositivos a modo de ejemplo que pueden estar configurados para realizar las técnicas de esta divulgación.

[0133] Por ejemplo, el dispositivo servidor 102 puede incluir un codificador para codificar una imagen o unas imágenes que se hallan después de un punto de acceso aleatorio o un punto de adaptación de flujo, tal como un punto de conmutación de capa temporal, u otro punto de adaptación de flujo. Por ejemplo, este podría ser un punto de conmutación para adaptación de velocidad de bits, velocidad de tramas (es decir, un punto de conmutación de capa temporal) o resolución espacial. Del mismo modo, el dispositivo cliente 108 puede descodificar una imagen o unas imágenes que se hallan después de un punto de acceso aleatorio o un punto de adaptación de flujo, tal como un punto de conmutación de capa temporal. De nuevo, este podría ser un punto de conmutación para adaptación de velocidad de bits, velocidad de tramas (es decir, un punto de conmutación de capa temporal) o resolución espacial. Una o más de estas imágenes pueden ser imágenes iniciales. Las imágenes iniciales no pueden descodificarse correctamente en el dispositivo cliente 108 cuando se produce un acceso aleatorio desde la imagen BLA.

[0134] En un ejemplo, para evitar la propagación de errores por imágenes de referencia que pueden no estar disponibles dependiendo de dónde comienza la descodificación, el dispositivo cliente 108 no puede aplicar imágenes almacenadas en la memoria intermedia de imágenes descodificadas (DPB) que preceden a la imagen BLA en orden de descodificación o en orden de salida (lo cual incluye las imágenes iniciales) como imágenes de referencia.

[0135] En diversos ejemplos, el dispositivo cliente 108 puede marcar todas las imágenes de referencia de la DPB como no usadas para referencia antes de descodificar la imagen BLA. Por ejemplo, el dispositivo cliente 108 puede marcar imágenes de referencia de la memoria intermedia de imágenes descodificadas (DPB) como no usadas para referencia.

[0136] En otro ejemplo, el dispositivo servidor 102, el dispositivo cliente 108 o ambos pueden incluir un elemento sintáctico o señalizador en una cabecera de fragmento y codificar el señalizador dentro de un flujo de bits, por ejemplo, un señalizador `no_output_of_prior_pics`, de modo que ninguna de las imágenes anteriores de la DPB procesadas por el decodificador 30 de vídeo se use para descodificar las imágenes iniciales o de salida de la DPB, por ejemplo, para la presentación en un monitor. En algunos ejemplos, este señalizador puede aparecer al principio de la cabecera de fragmento antes de la descodificación de entropía para que pueda descodificarse más fácilmente y la información esté disponible más pronto en el proceso de codificación. En un ejemplo, uno de estos dispositivos de elementos de red puede convertir una CRA en una BLA cuando, por ejemplo, se necesita un empalme o una conmutación de canales o adaptación de flujo, tal como una conmutación de capa temporal. Tener el señalizador accesible sin codificación de entropía permite que los elementos de red sin la capacidad de descodificación de entropía tengan acceso al señalizador.

[0137] La pluralidad de tipos de unidades NAL incluye uno o más de (1) un fragmento codificado de una imagen de acceso de enlace roto (BLA), en la que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, (2) un fragmento codificado de una imagen BLA, en la que la imagen BLA es una imagen BLA con imágenes iniciales descodificables asociadas en el flujo de bits, y (3) un fragmento codificado de una imagen BLA, en la que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits. En un ejemplo, la imagen inicial comprende una imagen que precede a una imagen de acceso aleatorio (RAP) en un orden de visualización pero sigue a la imagen de acceso aleatorio en orden de descodificación.

[0138] En diversos ejemplos, los dispositivos 104A, 104B de encaminamiento y el dispositivo 106 de transcodificación que componen la red 100 también pueden realizar algún procesamiento en una imagen o unas imágenes que se hallan después de un punto de acceso aleatorio o punto de adaptación de flujo, tal como un punto de conmutación de capa temporal. Por ejemplo, este podría ser un punto de conmutación para adaptación de velocidad de bits, velocidad de tramas (es decir, un punto de conmutación de capa temporal) o resolución espacial. Como se ha analizado anteriormente, una o más de estas imágenes pueden ser imágenes iniciales que no pueden descodificarse correctamente.

[0139] En un ejemplo, uno o más de los dispositivos 104A, 104B de encaminamiento, y el dispositivo 106 de transcodificación no pueden usar ninguna imagen que precede a la imagen CRA en orden de descodificación u orden de salida (que incluye las imágenes iniciales) como imagen de referencia. En otro ejemplo, uno o más de los dispositivos 104A, 104B de encaminamiento, y el dispositivo 106 de transcodificación pueden marcar todas las imágenes de referencia de la DPB como no usadas para referencia antes de descodificar la imagen BLA. Por ejemplo, uno o más de los dispositivos 104A, 104B de encaminamiento, y el dispositivo 106 de transcodificación pueden marcar imágenes de referencia de la memoria intermedia de imágenes descodificadas (DPB) como no usadas para referencia. En otro ejemplo, uno o más de los dispositivos 104A, 104B de encaminamiento, y el dispositivo 106 de transcodificación o un servidor de transmisión en continuo pueden usar un señalizador, por ejemplo, `no_de_prior_pics_flag` para que no se facilite ninguna de las imágenes anteriores en la DPB. De forma adicional, los dispositivos 104A, 104B de encaminamiento y el dispositivo 106 de transcodificación pueden procesar los tipos de unidades NAL asignados para indicar cuándo las imágenes BLA tienen y no tienen imágenes iniciales.

[0140] La figura 5 es un diagrama que ilustra ejemplo de acuerdo con las técnicas descritas en esta divulgación. La figura 5 ilustra ejemplos de casos en los que las imágenes iniciales son descodificables y no descodificables. La capacidad de descodificación de las imágenes iniciales puede basarse en la ubicación del bloque predictivo. De forma adicional, la capacidad de descodificación de las imágenes iniciales puede basarse en si la imagen CRA actual es una imagen CRA que no es una imagen BLA o si la imagen CRA actual es una imagen CRA que también es una imagen BLA. (Las imágenes BLA son un subconjunto de las imágenes CRA.)

[0141] La parte 200 de la figura 5 ilustra una serie de imágenes en orden de descodificación. Inicialmente, el decodificador 30 de vídeo (figuras 1 y 3) o el dispositivo cliente 108 (figura 4) por ejemplo, pueden descodificar una imagen RAP en la ubicación 202 como primera imagen de una secuencia de vídeo. El decodificador 30 de vídeo o el dispositivo cliente 108 pueden descodificar a continuación una imagen o parte de una imagen que puede actuar como imagen de referencia en la ubicación 204. Como se ilustra en la figura 5, la ubicación 204 es una ubicación posible de la imagen de referencia en orden de descodificación. Si la imagen de referencia está situada en la ubicación 204 y la imagen de la ubicación 206 es una imagen CRA que no es una imagen BLA, entonces la

imagen inicial de la ubicación 208 será descodificable. Por el contrario, si la imagen de referencia está situada en la ubicación 204 y la imagen de la ubicación 206 es una imagen CRA que también es una imagen BLA, entonces la imagen inicial de la ubicación 208 no será descodificable. (Las imágenes BLA son un subconjunto de las imágenes CRA.)

5

[0142] Si la imagen RAP de la ubicación 202 es la imagen RAP, donde comienza la descodificación, y la imagen CRA actual en la ubicación 206 no es una imagen BLA, entonces, como se ha descrito anteriormente, la imagen inicial de la ubicación 208 es descodificable. Por el contrario, si la imagen CRA actual en la ubicación 206 es la RAP, entonces la imagen CRA actual de la ubicación 206 también es una imagen BLA y la imagen inicial de la ubicación 208 no es descodificable. Esto se debe a que el bloque predictivo de la ubicación 204 no está disponible para una imagen BLA que es una imagen CRA actual en la ubicación 206. En consecuencia, el bloque predictivo de la ubicación 204 puede (1) marcarse como no usado para referencia y (2) el señalizador no_de_prior_pics_flag puede indicar que las imágenes anteriores, hasta e incluyendo la imagen inicial de la ubicación 208, no deberían facilitarse.

10

15

[0143] Como se ilustra en la figura 5, la ubicación 210 es otra ubicación posible de la imagen de referencia en orden de descodificación. Si la imagen de referencia está situada en la ubicación 210, entonces la imagen inicial de la ubicación 208 será descodificable.

20

[0144] La parte 212 de la figura 5 ilustra una serie de imágenes en orden de salida. Inicialmente, el descodificador 30 de vídeo (figuras 1 y 3) o el dispositivo cliente 108 (figura 4) es una imagen RAP 202. El descodificador 30 de vídeo o el dispositivo cliente 108 pueden descodificar, a continuación, una imagen o parte de una imagen que puede actuar como bloque predictivo en la ubicación 204. Como se ilustra en la figura 5, la ubicación 204 es una ubicación posible del bloque predictivo en el orden de salida.

25

[0145] En el orden de salida, la imagen inicial 208 puede facilitarse antes que la imagen CRA actual de la ubicación 206, como se ilustra en la figura 5. Como se ilustra en la figura 5, la ubicación 210 es otra ubicación posible del bloque predictivo en orden de salida.

30

[0146] La figura 6 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar imágenes RAP de acuerdo con uno o más ejemplos descritos en esta divulgación. Como se ilustra en la figura 6, en algunos ejemplos, un codificador de vídeo, por ejemplo, el codificador 20 de vídeo o el descodificador 30 de vídeo puede codificar una imagen BLA para un punto de acceso aleatorio o punto de adaptación de flujo, tal como un punto de conmutación de capa temporal en una secuencia de vídeo. Por ejemplo, este podría ser un punto de conmutación para la adaptación de la velocidad de bits, la velocidad de tramas o la resolución espacial. La imagen BLA puede incluir una o más imágenes iniciales. Las imágenes iniciales no pueden descodificarse correctamente (por ejemplo, mediante el descodificador 30, un MANE u otro dispositivo de descodificación) cuando se produce un acceso aleatorio desde la imagen BLA.

35

40

[0147] En un ejemplo, para evitar la propagación de errores de imágenes de referencia que pueden no estar disponibles dependiendo de dónde comienza la descodificación, el codificador de vídeo no puede usar ninguna imagen que precede a la imagen BLA en orden de descodificación o en orden de salida (lo cual incluye las imágenes iniciales) como imagen de referencia.

45

[0148] Por ejemplo, el descodificador 30 de vídeo puede recibir y descodificar un elemento sintáctico que indica que una memoria intermedia de almacenamiento de imágenes se vacía sin facilitar ninguna imagen de la memoria (600) intermedia de almacenamiento de imágenes. El codificador 20 o un elemento de red intermedio puede incluir el elemento sintáctico en el flujo de bits de vídeo codificado. El descodificador 30 de vídeo puede determinar si el elemento sintáctico indica que no debe producirse una salida de imágenes anteriores, por ejemplo, el descodificador 30 de vídeo puede verificar el elemento sintáctico (602) para determinar si está establecido, es decir, si es igual a 1. Cuando el descodificador 30 de vídeo recibe un elemento sintáctico que está establecido, el descodificador 30 de vídeo puede hacer que las imágenes anteriores a una imagen actual en orden de descodificación y que residen en la memoria intermedia de almacenamiento de imágenes en el momento de la descodificación de la imagen actual se vacíen de la memoria intermedia de almacenamiento de imágenes sin ser facilitadas (604).

50

55

[0149] En algunos ejemplos, el elemento sintáctico puede ser uno de una pluralidad de elementos sintácticos. De forma adicional, la pluralidad de elementos sintácticos puede incluir elementos sintácticos de cabecera de fragmento sometidos a codificación de entropía y elementos sintácticos de cabecera de fragmento no sometidos a codificación de entropía. En un ejemplo, el elemento sintáctico que puede indicar que una memoria intermedia de almacenamiento de imágenes está vacía sin facilitar ninguna imagen de la memoria intermedia de imágenes descodificadas, está incluido en la cabecera de fragmento antes de cualquier elemento sintáctico de cabecera de fragmento sometido a codificación de entropía. En otro ejemplo, el elemento sintáctico, que puede indicar que los datos de la memoria intermedia de almacenamiento de imágenes deben ignorarse y/o sobrescribirse sin facilitar ninguna imagen de la memoria intermedia de almacenamiento de imágenes, está incluido en la cabecera de fragmento antes de cualquier elemento sintáctico de cabecera de fragmento sometido a codificación de entropía.

60

65

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

[0150] La figura 7 es un diagrama de flujo que ilustra un ejemplo de procedimiento de acuerdo con uno o más ejemplos descritos en esta divulgación. En el ejemplo ilustrado, un codificador de vídeo puede determinar que la imagen actual es una imagen CRA (700). El codificador de vídeo también puede determinar que la imagen actual es una imagen RAP (702). El codificador de vídeo puede determinar que la imagen actual es una imagen BLA (700). En otros ejemplos, un codificador de vídeo puede usar un tipo de unidad NAL para determinar si la imagen actual es una imagen BLA.

[0151] En diversos ejemplos, el codificador de vídeo puede marcar todas las imágenes de referencia de la DPB como no usadas para referencia antes de descodificar la imagen BLA. Por ejemplo, el descodificador 30 de vídeo puede marcar imágenes de referencia de la memoria 92 de imágenes de referencia, algunas veces denominada memoria intermedia de imágenes descodificadas (DPB), como no usadas para referencia. En consecuencia, las imágenes no se usarán para intercodificar, lo que evitará posibles errores, y en algunos ejemplos permitirán abordar problemas con la adaptación espacial. De forma adicional, en general estas imágenes no se facilitarán, por ejemplo, a un monitor o pantalla, donde pueden visualizarse.

[0152] El descodificador 30 de vídeo puede determinar que una imagen actual es una imagen BLA y marcar una imagen de referencia de una memoria intermedia de almacenamiento de imágenes como no usada para referencia antes de descodificar la imagen BLA. En un ejemplo, el marcado de la imagen de referencia en la memoria intermedia de almacenamiento de imágenes puede tener lugar cuando la imagen BLA comprende una imagen BLA no descodificable, lo que se puede determinar en algunos ejemplos, basándose en el tipo de unidad NAL. En algunos ejemplos, uno o más de (1) el marcado de imágenes en una DPB como no usadas para referencia, (2) el uso de un elemento sintáctico como un señalizador `no_output_of_prior_pics_flag` y (3) el uso tipos de unidad NAL que indican imágenes BLA se pueden usar de manera independiente o en cualquier combinación.

[0153] La figura 8 es un diagrama de flujo que ilustra un ejemplo de procedimiento de acuerdo con uno o más ejemplos descritos en esta divulgación. Un codificador de vídeo puede procesar tipos de unidad NAL asignados para indicar cuándo las imágenes BLA tienen y no tienen imágenes iniciales. El codificador de vídeo puede estar configurado para incluir tipos de unidad NAL asignados para indicar cuándo las imágenes BLA tienen y no tienen imágenes iniciales. Por ejemplo, en una norma, los tipos de unidad NAL 16, `BLA_W_LP` (BLA con imagen inicial); 17, `BLA_W_DLP` (BLA con imagen inicial descodificable); y 18, `BLA_N_LP` (BLA sin imagen inicial) están incluidos.

[0154] En un ejemplo, el codificador de vídeo puede codificar imágenes de acuerdo con uno de una pluralidad de diferentes tipos de unidades de capa de abstracción de red (NAL) que incluyen uno o más de los siguientes. Por ejemplo, el codificador de vídeo puede determinar que una imagen BLA no tiene imágenes iniciales asociadas (800) y codificar un fragmento codificado de la imagen BLA o toda la imagen BLA usando un tipo de unidad NAL que indica que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en un flujo (802) de bits. El codificador de vídeo puede determinar que una imagen BLA tiene imágenes (804) iniciales descodificables asociadas y codificar un fragmento codificado o la imagen BLA completa usando un tipo de unidad NAL que indica que la imagen BLA es una imagen BLA con imágenes iniciales descodificables asociadas en el flujo (806) de bits. El codificador de vídeo determina que una imagen BLA tiene imágenes (808) iniciales asociadas y puede codificar un fragmento codificado de la imagen BLA o toda la imagen BLA usando un tipo de unidad NAL que indica que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en el flujo (810) de bits. En un ejemplo, si el descodificador 30 de vídeo detecta un tipo de unidad BLA NAL, entonces el descodificador 30 de vídeo puede, por ejemplo, marcar imágenes de la DPB como no usadas para referencia.

[0155] En algunos ejemplos, uno o más de (1) el marcado de imágenes en una DPB como no usadas para referencia, (2) el uso de un elemento sintáctico como un señalizador `no_output_of_prior_pics_flag` y (3) el uso tipos de unidad NAL que indican imágenes BLA se pueden usar de manera independiente o en cualquier combinación. Por ejemplo, en algunos casos se pueden usar los tres anteriores. En otros ejemplos, los tipos de unidad NAL de ejemplo podrían usarse junto con el marcado de imágenes de la DPB como no usadas para referencia. En otro ejemplo, podría usarse el elemento sintáctico que señala que no se facilitan imágenes anteriores y el marcado de las imágenes como no usadas para referencia. En otro ejemplo, podrían usarse los elementos sintácticos que señalizan que no se facilitan imágenes anteriores y los tipos de unidades NAL.

[0156] Como se usa en el presente documento, BLC se refiere al acceso aleatorio limpio de enlace roto, BLCL se refiere a una imagen BLC con imágenes iniciales asociadas en el flujo de bits y BLCNL se refiere a una imagen BLC sin imágenes iniciales asociadas en el flujo de bits. Como se analiza en el presente documento, las imágenes BLC en general son las mismas que las imágenes BLA. CRA se refiere a un acceso aleatorio limpio, CRAL se refiere a una imagen CRA con imágenes iniciales asociadas en el flujo de bits, y CRANL se refiere a una imagen CRA sin imágenes iniciales asociadas en el flujo de bits. IDR se refiere a una actualización de descodificación instantánea, LPR se refiere a una imagen inicial asociada con una imagen de punto de acceso aleatorio, NSP se

refiere a «normal sin nada especial», RAP se refiere a un punto de acceso aleatorio y RPS se refiere a un conjunto de imágenes de referencia. Como se usa en el presente documento, TLA se refiere a acceso a capa temporal, TLAL se refiere a una imagen TLA que también es una imagen LRP, TLANL se refiere a una imagen TLA que no es una imagen LRP.

5

[0157] Una unidad de acceso BLA se refiere a una unidad de acceso en la que la imagen codificada es una imagen BLA. Una imagen BLC es una imagen RAP en la que la cabecera de fragmento de los fragmentos codificados incluye sintaxis RPS mientras que el RPS se deriva como vacío sin usar la sintaxis RPS. Una unidad de acceso BLCL es una unidad de acceso en la que la imagen codificada es una imagen BLCL. Las imágenes BLCL son imágenes BLA para las cuales las imágenes LPR asociadas están presentes en el flujo de bits. En algunos ejemplos, una unidad de acceso BLCL puede ser equivalente a una combinación de BLA_W_DLP y BLA_W_LP. Una unidad de acceso BLCNL es una unidad de acceso en la que la imagen codificada es una imagen BLCNL. En algunos ejemplos, una unidad de acceso BLCNL puede ser equivalente a una BLA_N_LP. Las imágenes BLCNL son imágenes BLA para las cuales las imágenes LPR asociadas no están presentes en el flujo de bits.

10

15

[0158] En un ejemplo, una unidad de acceso CRA es una unidad de acceso en la que la imagen codificada es una imagen CRA. Las imágenes CRA son imágenes RAP en las que la cabecera de fragmento de los fragmentos codificados incluye la sintaxis RPS, y la sintaxis RPS se usa para la derivación del RPS. Una unidad de acceso CRAL es una unidad de acceso en la que la imagen codificada es una imagen CRAL. Las imágenes CRAL son imágenes CRA para las cuales las imágenes LPR asociadas están presentes en el flujo de bits. Una unidad de acceso CRANL es una unidad de acceso en la que la imagen codificada es una imagen CRANL. Las imágenes CRANL son imágenes CRA para las cuales las imágenes LPR asociadas no están presentes en el flujo de bits.

20

25

[0159] En un ejemplo, una unidad de acceso IDR es una unidad de acceso en la que la imagen codificada es una imagen IDR. Las imágenes IDR son imágenes RAP en las que la cabecera de fragmento de los fragmentos codificados no incluye la sintaxis RPS, y se deriva el RPS como vacío.

30

[0160] En un ejemplo, si todas las unidades de acceso precedentes en orden de decodificación no están presentes, siempre que cada conjunto de parámetros al que se refiere la imagen codificada y todas las imágenes codificadas subsiguientes en orden de decodificación estén presentes antes de su activación, una imagen IDR y todas las imágenes codificadas subsiguientes en orden de decodificación pueden decodificarse correctamente. De forma alternativa, en otro ejemplo, las imágenes IDR pueden ser como se define en HEVC con la adición de lo anterior.

35

[0161] En otros ejemplos, la imagen IDR puede definirse como se indica a continuación más la nota anterior, y una imagen IDR puede ser una imagen codificada que contiene solo fragmentos I. De forma adicional, para una imagen IDR de ejemplo, todas las imágenes codificadas que siguen a la imagen IDR en orden de decodificación no usan interpredicción a partir de ninguna imagen que precede a la imagen IDR en orden de decodificación. En un ejemplo, cualquier imagen que precede a la imagen IDR en orden de decodificación también precede a la imagen IDR en orden de salida.

40

[0162] Una imagen inicial es una imagen codificada que no es una imagen RAP y que sigue a otra imagen particular en orden de decodificación y precede a la imagen particular en orden de salida. Una imagen LPR es una imagen inicial que está asociada con una imagen RAP o una imagen inicial de una imagen RAP.

45

[0163] El recuento de orden de imágenes puede ser una variable asociada a cada imagen codificada y tiene un valor que aumenta al aumentar la posición de la imagen en orden de salida con relación a la imagen RAP anterior en orden de decodificación.

50

[0164] En un ejemplo, una unidad de acceso RAP es una unidad de acceso en la que la imagen codificada es una imagen RAP. Una imagen RAP puede ser una imagen codificada que contiene solo fragmentos I. Para una imagen RAP, todas las imágenes codificadas que siguen a la imagen RAP tanto en orden de decodificación como en orden de salida no usan interpredicción a partir de ninguna imagen que precede a la imagen RAP en orden de decodificación u orden de salida. La salida de cualquier imagen que precede a la imagen RAP en orden de decodificación debe preceder a la salida de la imagen RAP. Si todas las unidades de acceso precedentes en orden de decodificación no están presentes, siempre que cada conjunto de parámetros al que se refiere la imagen codificada y todas las imágenes codificadas subsiguientes en orden de decodificación estén presentes antes de su activación, una imagen RAP y todas las imágenes codificadas subsiguientes en orden de decodificación y orden de salida se puede decodificar correctamente.

55

60

[0165] De forma alternativa, la imagen RAP se puede definir de conformidad con el análisis anterior y tal como se indica a continuación. Una imagen RAP puede ser una imagen codificada que contiene solo fragmentos I y para la cual todas las imágenes codificadas que siguen a la imagen RAP tanto en orden de decodificación como en orden de salida no usan interpredicción a partir de ninguna imagen que precede a la imagen RAP en orden de

65

descodificación u orden de salida. Cualquier imagen que precede a la imagen RAP en orden de descodificación también precede a la imagen RAP en orden de salida.

5 **[0166]** Una unidad de acceso TLA es una unidad de acceso en la que la imagen codificada es una imagen TLA. Una imagen TLA es una imagen codificada para la cual la imagen TLA y todas las imágenes codificadas con temporal_id son mayores o iguales a temporal_id de la imagen TLA. Una imagen TLA que sigue a la imagen TLA en orden de descodificación no usará interpredicción a partir de ninguna imagen con temporal_id mayor o igual a temporal_id de una imagen TLA que precede a la imagen TLA en orden de descodificación. Una unidad de acceso TLAL es una unidad de acceso en la que la imagen codificada es una imagen TLA.

10 **[0167]** En algunos ejemplos, se pueden definir los siguientes tipos de unidades VCL NAL diferenciados. Como primer ejemplo, se puede proporcionar un tipo de unidad NAL para un fragmento codificado de una imagen IDR (por ejemplo, nal_unit_type = 5). Para este tipo de unidad NAL, se aplica el concepto de imagen IDR de HEVC WD6. Una característica única de este tipo de unidad VCL NAL en comparación con otros tipos de unidades VCL NAL es que no hay ninguna sintaxis de conjunto de imágenes de referencia (RPS) incluida en el encabezado del fragmento.

20 **[0168]** Algunos ejemplos incluyen un fragmento codificado de una imagen BLCNL (una imagen BLC sin imágenes iniciales asociadas en el flujo de bits, por ejemplo, nal_unit_type = 2). Comparado con un fragmento codificado de una imagen IDR, un fragmento codificado de una imagen BLCNL incluye sintaxis RPS en la cabecera del fragmento, pero la sintaxis RPS no se usa para la derivación de RPS, sino que todos los subconjuntos RPS se derivan como vacíos.

25 **[0169]** Algunos ejemplos incluyen un fragmento codificado de una imagen BLCL (una imagen BLC con imágenes iniciales asociadas en el flujo de bits, por ejemplo, nal_unit_type = 3). En comparación con una imagen BLCNL, hay imágenes iniciales asociadas con una imagen BLCL en el flujo de bits.

30 **[0170]** Algunos ejemplos incluyen un fragmento codificado de una imagen CRANL (una CRA sin imágenes iniciales asociadas en el flujo de bits, por ejemplo, nal_unit_type = 15). En comparación con un fragmento codificado de una imagen BLCNL, un fragmento codificado de una imagen CRANL incluye sintaxis RPS en la cabecera de fragmento, y la sintaxis RPS se usa para la derivación de RPS.

35 **[0171]** Algunos ejemplos incluyen un fragmento codificado de una imagen CRAL (una imagen CRA con imágenes iniciales asociadas en el flujo de bits, por ejemplo, nal_unit_type = 4). En comparación con una imagen CRANL, hay imágenes iniciales asociadas con una imagen CRAL en el flujo de bits.

40 **[0172]** Algunos ejemplos incluyen un fragmento codificado de una imagen TLANL (una imagen TLA que no es una imagen LFR, por ejemplo, nal_unit_type = 16). Algunos ejemplos incluyen un fragmento codificado de una imagen TLAL (una imagen TLA que también es una imagen LPR, por ejemplo, nal_unit_type = 17). Algunos ejemplos incluyen un fragmento codificado de una imagen NSP (una imagen normal sin nada especial - ninguna de las anteriores, nal_unit_type = 1).

45 **[0173]** Antes de descodificar los datos de fragmento de cada imagen BLC (BLCL o BLCNL), todas las imágenes de referencia de la memoria intermedia de imágenes descodificadas (DPB) deben ser marcadas como «no usadas para referencia» por el descodificador 30, como se ha descrito anteriormente. Solo con esto, el descodificador 30 puede aplicar directamente el proceso de descodificación actual como se especifica actualmente en HEVC WD 6 para las imágenes iniciales de una imagen CRA que comienza el flujo de bits, para las imágenes iniciales de una imagen BLA, independientemente de si la imagen BLA cambia la resolución espacial.

50 **[0174]** Sin lo anterior, si una imagen BLA no cambia la resolución espacial, el descodificador 30 puede aplicar directamente el proceso de descodificación actual como se especifica actualmente en HEVC WD 6 para imágenes iniciales de una imagen CRA que comienza el flujo de bits, para imágenes iniciales de la imagen BLA. Sin embargo, si una imagen BLA cambia la resolución espacial, el proceso de descodificación actual como se especifica actualmente en HEVC WD 6 para imágenes iniciales de una imagen CRA que comienza el flujo de bits no se puede aplicar directamente para imágenes iniciales de la imagen BLA, ya que puede darse la situación en la que la resolución espacial sea diferente para una imagen actual y una imagen de referencia para la imagen actual.

55 **[0175]** Una forma de asegurarse de que todas las imágenes de referencia de la DPB se vayan a marcar como «no usadas para referencia» antes de descodificar los datos del fragmento de cada imagen BLA es derivar el RPS de cada imagen BLA como vacío, independientemente de si la señalización RPS de la cabecera de fragmento indica un RPS no vacío. Por ejemplo, incluso si hay un RPS, el descodificador 30 de vídeo puede anularlo y entonces deriva o trata el RPS como vacío si la imagen es una imagen BLA.

60 **[0176]** En realidad, si la señalización RPS de la cabecera del fragmento indica un RPS vacío para una imagen BLA o una imagen CRA (CRAL o CRANL), entonces la imagen debería haberse codificado como una imagen IDR.

65

[0177] En algunos ejemplos, una imagen RAP puede definirse como una imagen codificada que contiene solo fragmentos I. Para una imagen RAP, todas las imágenes codificadas que siguen a la imagen RAP tanto en orden de descodificación como en orden de salida no usan interpredicción a partir de ninguna imagen que precede a la imagen RAP en orden de descodificación u orden de salida. De forma adicional, la salida de cualquier imagen que precede a la imagen RAP en orden de descodificación puede preceder a la salida de la imagen RAP.

[0178] Una forma de asegurar que la salida de cualquier imagen que precede a la imagen RAP en orden de descodificación precederá a la salida de la imagen RAP es que un codificador 20 de vídeo establezca `no_output_of_prior_pics_flag` igual a 1 en un flujo de bits codificado transmitido a, por ejemplo, un descodificador 30 de vídeo, como se ha analizado anteriormente. En otro ejemplo, un descodificador 30 de vídeo puede inferir que `no_output_of_prior_pics_flag` es igual a 1 (independientemente de su valor) para una imagen BLA. De esta manera, las operaciones de empalme en una imagen BLA, en las que en el flujo de bits empalmado el valor POC de una imagen anterior a la imagen BLA es mayor que el valor POC de la imagen BLA, están permitidas. En particular, si el valor POC de la imagen BLA se deriva como igual a su POC LSB (suponiendo que POC MSB sea igual a 0), lo anterior es fácil que suceda. Otra forma de asegurar esto es asegurarse de que los tiempos de salida de las imágenes que preceden a la imagen BLA en orden de descodificación sean anteriores a los de la imagen BLA.

[0179] Algunos ejemplos permiten que el codificador 20 de vídeo y/o un empalmador de flujo de bits determinen cuál de las formas descritas en el presente documento se va a usar. En consecuencia, el codificador 20 de vídeo, por ejemplo, puede incluir `no_output_of_prior_pics_flag` en la cabecera del fragmento de las imágenes BLA en algunos ejemplos. En otros ejemplos, el codificador 20 de vídeo puede incluir tipos de unidad NAL que indican si las imágenes almacenadas en una memoria intermedia en el descodificador 30 de vídeo se pueden usar para referencia. En consecuencia, el descodificador 30 de vídeo puede marcar imágenes como no usadas para referencia, por ejemplo, cuando el uso de dichas imágenes en la descodificación dará como resultado la descodificación incorrecta de una imagen.

[0180] Para permitir la reescritura simple de una CRA en una imagen BLA por un elemento de red, además de poner el señalizador `no_output_of_prior_pics_flag` en la cabecera de fragmento de una imagen CRA, este también se puede incluir lo más al principio posible en la cabecera de fragmento. Esto puede ser antes de cualquier parámetro de cabecera de fragmento sometido a codificación de entropía, por ejemplo, justo después de `first_slice_in_pic_flag` en algunos ejemplos, como se ha analizado anteriormente.

[0181] En un ejemplo, dos imágenes BLA consecutivas tienen el mismo POC LSB, y solo es posible diferenciarlas mediante `random_access_pic_id` (o renombrado como `rap_pic_id`). Por lo tanto, podría ser preferible usar una codificación de longitud fija para `rap_pic_id`, y también incluirlo lo más al principio posible en la cabecera de fragmento, preferentemente no después de cualquier parámetro de cabecera de fragmento sometido a codificación de entropía, por ejemplo, justo después de `first_slice_in_pic_flag` y `no_output_of_prior_pics_flag`, por ejemplo, para imágenes CRA e imágenes BLA. Otros elementos sintácticos de cabecera de fragmento que pueden usarse para la detección de límites de imagen, por ejemplo, `pic_parameter_set_id` y POC LSB (es decir, `pic_order_cnt_lsb`) pueden ser similares.

[0182] Todos los conjuntos de parámetros de un tipo particular (por ejemplo, SPS) o todos los tipos pueden estar presentes al principio del flujo de bits, es decir, incluidos en la primera unidad de acceso del flujo de bits. Si es así, es conveniente buscar todos los conjuntos de parámetros de un tipo particular y enviarlos fuera de banda. Por ejemplo, un codificador puede incluir un parámetro de protocolo de descripción de sesión (SDP) que se usa durante la negociación de la sesión. Por lo tanto, puede ser beneficioso para un codificador incluir una indicación en el flujo de bits de que todos los conjuntos de parámetros de un tipo particular o todos los tipos están presentes al principio del flujo de bits. La indicación puede incluirse en un mensaje SEI, un delimitador de unidad de acceso o un conjunto de parámetros. Por ejemplo, un tipo de unidad NAL diferenciada puede usarse para el SPS, el conjunto de parámetros de imagen (PPS) o el conjunto de parámetros de adaptación (APS) para indicar que todos los SPS (o PPS o APS) están presentes al principio del flujo de bits.

[0183] El elemento sintáctico `slice_type` puede estar presente condicionalmente en la cabecera del fragmento, basándose en si el tipo de unidad NAL indica que la imagen que contiene el fragmento es una imagen IDR, una imagen CRA o una imagen BLA. Por ejemplo, si el tipo de unidad NAL indica que la imagen que contiene el fragmento es una imagen IDR, una imagen CRA o una imagen BLA, `slice_type` no está presente en la cabecera del fragmento. De lo contrario, el codificador inserta `slice_type` en la cabecera del fragmento. Cuando no está presente, el valor de `slice_type` indica que el sector es un fragmento I.

[0184] La figura 9 es un diagrama de flujo que ilustra un procedimiento de ejemplo para descodificar un fragmento de acuerdo con uno o más ejemplos descritos en esta divulgación. En un ejemplo, el descodificador 30 de vídeo puede descodificar datos de vídeo, incluyendo la recepción de un fragmento de una imagen actual que se va a descodificar para una secuencia (900) de datos de vídeo. El descodificador 30 de vídeo puede recibir, en una cabecera de fragmento, al menos un elemento sintáctico sometido a codificación de entropía y al menos un elemento sintáctico no sometido a codificación de entropía. El elemento sintáctico no sometido a codificación de

entropía puede hallarse antes del elemento sintáctico sometido a codificación de entropía en la cabecera del fragmento. De forma adicional, el elemento sintáctico no sometido a codificación de entropía puede indicar si las imágenes anteriores a la imagen actual en orden de descodificación deben vaciarse de una memoria intermedia de almacenamiento de imágenes sin facilitarse (902). En un ejemplo, el elemento sintáctico puede ser un

5 señalizador `no_output_of_prior_pics_flag`. El señalizador `no_output_of_prior_pics_flag` puede establecerse en «1», por ejemplo, para indicar cuándo las imágenes anteriores a la imagen actual en orden de descodificación deben vaciarse de una memoria intermedia de almacenamiento de imágenes sin facilitarse. El descodificador 30 de vídeo puede descodificar el fragmento basándose en el elemento sintáctico no sometido a codificación de entropía (904).

10 **[0185]** La figura 10 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar un fragmento de acuerdo con uno o más ejemplos descritos en esta divulgación. El codificador 20 puede codificar datos de vídeo. El codificador 20 de vídeo puede codificar un fragmento de una imagen actual para una secuencia de datos de vídeo (1000).

15 **[0186]** El codificador 20 de vídeo codifica, en una cabecera del fragmento, al menos un elemento sintáctico sometido a codificación de entropía y al menos un elemento sintáctico no sometido a codificación de entropía, en el que el elemento sintáctico no sometido a codificación de entropía se halla antes del elemento sintáctico sometido a codificación de entropía en la cabecera de fragmento e indica si las imágenes anteriores a la imagen actual en

20 orden de descodificación deben vaciarse de una memoria intermedia de almacenamiento de imágenes sin facilitarse (1002). En un ejemplo, el elemento sintáctico puede ser un señalizador `no_output_of_prior_pics_flag`. El señalizador `no_output_of_prior_pics_flag` puede establecerse en «1», por ejemplo, para indicar cuándo las imágenes anteriores a la imagen actual en orden de descodificación deben vaciarse de una memoria intermedia de almacenamiento de imágenes sin facilitarse.

25 **[0187]** En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o cualquier combinación de estos. Si se implementan en software, las funciones pueden almacenarse o transmitirse, como una o más instrucciones o código, en o a través de un medio legible por ordenador y ejecutarse mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir

30 medios de almacenamiento legibles por ordenador, que corresponden a un medio tangible tal como unos medios de almacenamiento de datos o unos medios de comunicación que incluyen cualquier medio que facilita 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) unos

35 medios de almacenamiento tangibles legibles por ordenador que son no transitorios, o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que uno o más ordenadores o uno o más procesadores puedan acceder para recuperar instrucciones, código y/o estructuras de datos para implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

40 **[0188]** En otros ejemplos más, esta divulgación contempla un medio legible por ordenador que comprende una estructura de datos almacenada en el mismo, en el que 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.

45 **[0189]** A modo de ejemplo, y no de limitación, dichos medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda usarse para almacenar un código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. Además, cualquier conexión recibe debidamente la denominación de

50 medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde un sitio web, un servidor u otra fuente remota mediante un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o unas 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 incluidas en la definición de medio. Sin embargo, debería entenderse que los medios de

55 almacenamiento legibles por ordenador y los medios de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales ni otros medios transitorios, sino que, en su lugar, se dirigen a medios de almacenamiento tangibles no transitorios. El término disco, como se usa en el presente documento, incluye un disco compacto (CD), un disco láser, un disco óptico, un disco versátil digital (DVD), un disco flexible y un disco Blu-ray, de los cuales el disco flexible habitualmente reproduce datos magnéticamente, mientras que el resto de discos reproduce datos ópticamente con láseres. Las combinaciones de los anteriores también deben incluirse dentro del alcance de los

60 medios legibles por ordenador.

[0190] Las instrucciones pueden ser ejecutadas por uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables *in situ* (FPGA) u otros circuitos lógicos integrados o

65 discretos equivalentes. En consecuencia, el término «procesador», como se usa en el presente documento, puede referirse 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 puede proporcionarse dentro de módulos de hardware y/o software dedicados configurados para codificar y descodificar, o incorporados en un códec combinado. Asimismo, las técnicas podrían implementarse por completo en uno o más circuitos o elementos lógicos.

5

[0191] Las técnicas de la presente divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluidos un teléfono inalámbrico, un circuito integrado (CI) o un conjunto de CI (por ejemplo, un conjunto de chips). En la presente divulgación se describen diversos componentes, módulos o unidades para poner de manifiesto aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no se requiere necesariamente su realización por diferentes unidades de hardware. En cambio, como se ha descrito anteriormente, diversas unidades pueden combinarse en una unidad de hardware de códec o proporcionarse por medio de un grupo de unidades de hardware interoperativas, incluidos uno o más procesadores como se ha descrito anteriormente, junto con software y/o firmware adecuados.

10

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

15

REIVINDICACIONES

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

5 recibir (900), en una unidad de capa de abstracción de red (NAL), un segmento de una imagen actual (206) que se va a descodificar para una secuencia (200) de datos de vídeo, en el que la imagen actual comprende una imagen de acceso de enlace roto (BLA) y la unidad NAL tiene un tipo de unidad NAL que es uno de una pluralidad de tipos de unidades NAL que incluyen:

10 un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, y
un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits;

15 recibir (902), en una cabecera de segmento del segmento, uno o más elementos sintácticos sometidos a codificación de entropía y elementos sintácticos no sometidos a codificación de entropía; y
descodificar (904) el segmento basándose en los elementos sintácticos no sometidos a codificación de entropía,

caracterizado por que:

20 los elementos sintácticos no sometidos a codificación de entropía de la cabecera de segmento del segmento comprenden un elemento sintáctico no_output_of_prior_pics_flag que está incluido en la cabecera de segmento inmediatamente después de un elemento sintáctico que indica si el segmento es un primer segmento, por orden de descodificación, de la imagen actual, y
25 el elemento sintáctico no_output_of_prior_pics_flag es anterior a todos los elementos sintácticos sometidos a codificación de entropía en la cabecera de segmento e indica si unas imágenes anteriores a la imagen actual por orden de descodificación se deben extraer de una memoria intermedia de imágenes descodificadas sin facilitarlas.

30 2. El procedimiento de la reivindicación 1, en el que el uno o más elementos sintácticos sometidos a codificación de entropía comprenden una pluralidad de elementos sintácticos sometidos a codificación de entropía.

3. El procedimiento de la reivindicación 1, que comprende además descodificar el elemento sintáctico no sometido a codificación de entropía en un descodificador.

35 4. El procedimiento de la reivindicación 3, en el que el descodificador comprende un Elemento de Red Adaptable a los Medios (MANE).

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

determinar que la imagen actual es una imagen de acceso de enlace roto (BLA); y

40 marcar al menos una imagen de referencia de la memoria intermedia de imágenes descodificadas como no usada para referencia antes de descodificar la imagen BLA.

6. Un procedimiento de codificación de datos de vídeo, comprendiendo el procedimiento:

45 codificar (1000) un segmento de una imagen actual para una secuencia de datos de vídeo en una unidad de capa de abstracción de red (NAL), en el que la imagen actual comprende una imagen de acceso de enlace roto (BLA) y la unidad NAL tiene un tipo de unidad NAL que es uno de una pluralidad de tipos de unidades NAL que incluyen:

50 un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, y
un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits; y

55 codificar (1002), en una cabecera de segmento del segmento, uno o más elementos sintácticos sometidos a codificación de entropía y elementos sintácticos no sometidos a codificación de entropía,

caracterizado por que:

60 los elementos sintácticos no sometidos a codificación de entropía de la cabecera de segmento del segmento comprenden un elemento sintáctico no_output_of_prior_pics_flag que está incluido en la cabecera de segmento inmediatamente después de un elemento sintáctico que indica si el segmento es un primer segmento, por orden de descodificación, de la imagen actual,

el elemento sintáctico `no_output_of_prior_pics_flag` es anterior a todos los elementos sintácticos sometidos a codificación de entropía en la cabecera de segmento e indica si imágenes anteriores a la imagen actual por orden de descodificación se deben extraer de una memoria intermedia de imágenes descodificadas sin facilitarlas.

- 5
7. El procedimiento de la reivindicación 6, en el que el uno o más elementos sintácticos sometidos a codificación de entropía comprenden una pluralidad de elementos sintácticos sometidos a codificación de entropía
- 10
8. El procedimiento de la reivindicación 6, que comprende además codificar el elemento sintáctico no sometido a codificación de entropía en un codificador.
9. Un medio de almacenamiento legible por ordenador que tiene instrucciones almacenadas que, cuando se ejecutan, hacen que uno o más procesadores de un dispositivo lleven a cabo un procedimiento de acuerdo con cualquiera de las reivindicaciones 1 a 8.
10. Un dispositivo (14) para descodificar datos de vídeo, comprendiendo el dispositivo:
- 15
- medios para recibir (900), en una unidad de capa de abstracción de red (NAL), un segmento de una imagen actual que se va a descodificar para una secuencia de datos de vídeo, en el que la imagen actual comprende una imagen de acceso de enlace roto (BLA) y la unidad NAL tiene un tipo de unidad NAL que es uno de una pluralidad de tipos de unidades NAL que incluyen:
- 20
- un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, y
un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits;
- 25
- medios para recibir (902), en una cabecera de segmento del segmento, uno o más elementos sintácticos sometidos a codificación de entropía y elementos sintácticos no sometidos a codificación de entropía; y medios para descodificar (904) el segmento basándose en los elementos sintácticos no sometidos a codificación de entropía,
caracterizado por que:
- 30
- los elementos sintácticos no sometidos a codificación de entropía de la cabecera de segmento del segmento comprenden un elemento sintáctico `no_output_of_prior_pics_flag` que está incluido en la cabecera de segmento inmediatamente después de un elemento sintáctico que indica si el segmento es un primer segmento, por orden de descodificación, de la imagen actual,
- 35
- el elemento sintáctico `no_output_of_prior_pics_flag` es anterior a todos los elementos sintácticos sometidos a codificación de entropía en la cabecera de segmento e indica si imágenes anteriores a la imagen actual por orden de descodificación se deben extraer de una memoria intermedia de imágenes descodificadas sin facilitarlas.
- 40
11. El dispositivo de la reivindicación 10 que comprende medios para llevar a cabo un procedimiento de acuerdo con cualquiera de las reivindicaciones 2 a 5.
12. Un dispositivo (12) para codificar datos de vídeo, comprendiendo el dispositivo:
- 45
- medios para codificar (1000) un segmento de una imagen actual para una secuencia de datos de vídeo en una unidad de capa de abstracción de red (NAL), en el que la imagen actual comprende una imagen de acceso de enlace roto (BLA) y la unidad NAL tiene un tipo de unidad NAL que es uno de una pluralidad de tipos de unidades NAL que incluyen:
- 50
- un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA con imágenes iniciales asociadas en un flujo de bits, y
un segmento codificado de una imagen BLA, en el que la imagen BLA es una imagen BLA sin imágenes iniciales asociadas en el flujo de bits; y
- 55
- medios para codificar (1002), en una cabecera de segmento del segmento, uno o más elementos sintácticos sometidos a codificación de entropía y elementos sintácticos no sometidos a codificación de entropía,
caracterizado por que:
- los elementos sintácticos no sometidos a codificación de entropía de la cabecera de segmento del segmento comprenden un elemento sintáctico `no_output_of_prior_pics_flag` que está incluido en la

- 5 cabecera de segmento inmediatamente después de un elemento sintáctico que indica si el segmento es un primer segmento, por orden de descodificación, de la imagen actual, el elemento sintáctico `no_output_of_prior_pics_flag` es anterior a todos los elementos sintácticos sometidos a codificación de entropía en la cabecera de segmento e indica si imágenes anteriores a la imagen actual por orden de descodificación se deben extraer de una memoria intermedia de imágenes descodificadas sin facilitarlas.
13. El dispositivo de la reivindicación 12 que comprende medios para llevar a cabo un procedimiento de acuerdo con la reivindicación 7 o la reivindicación 8.

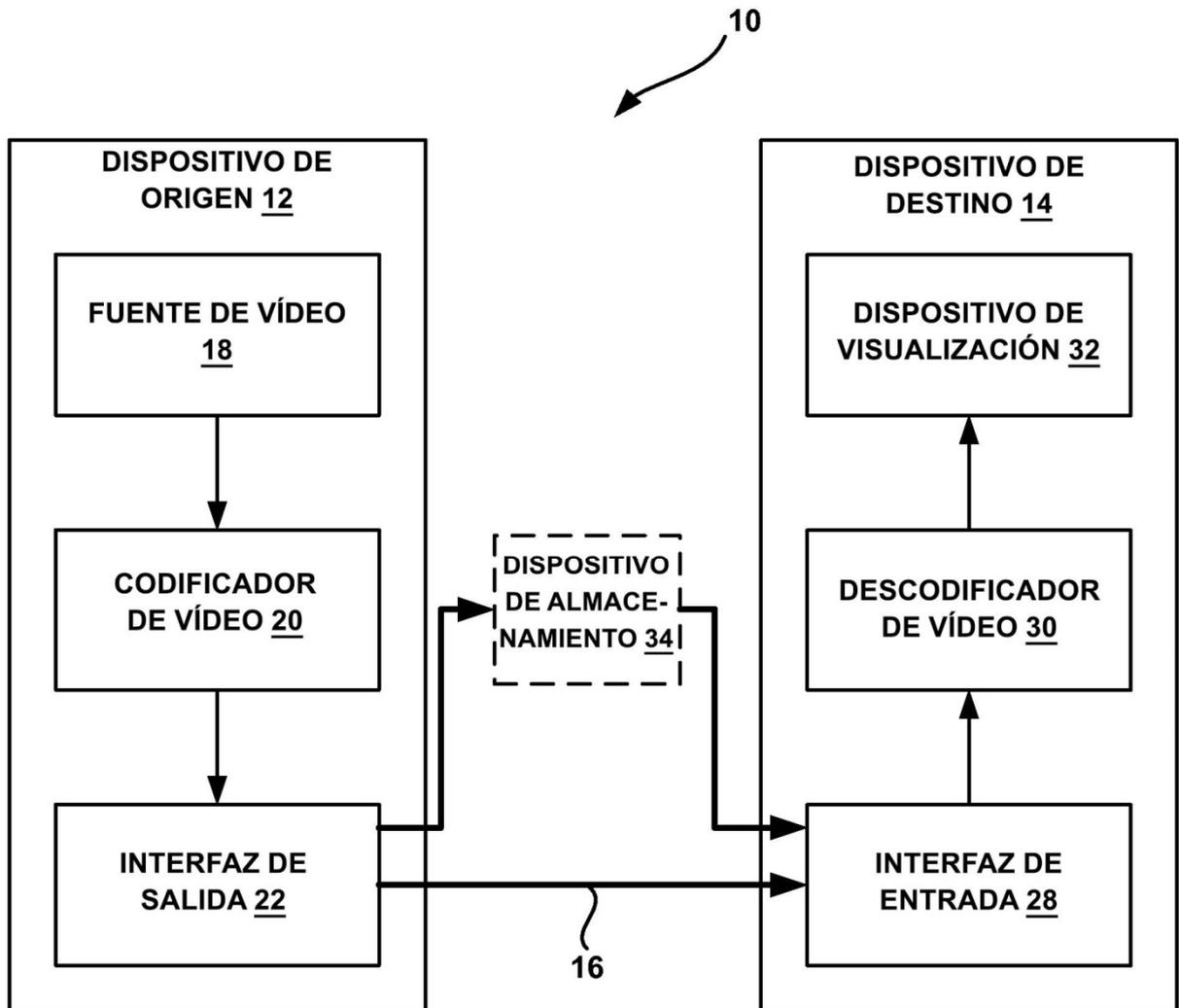


FIG. 1

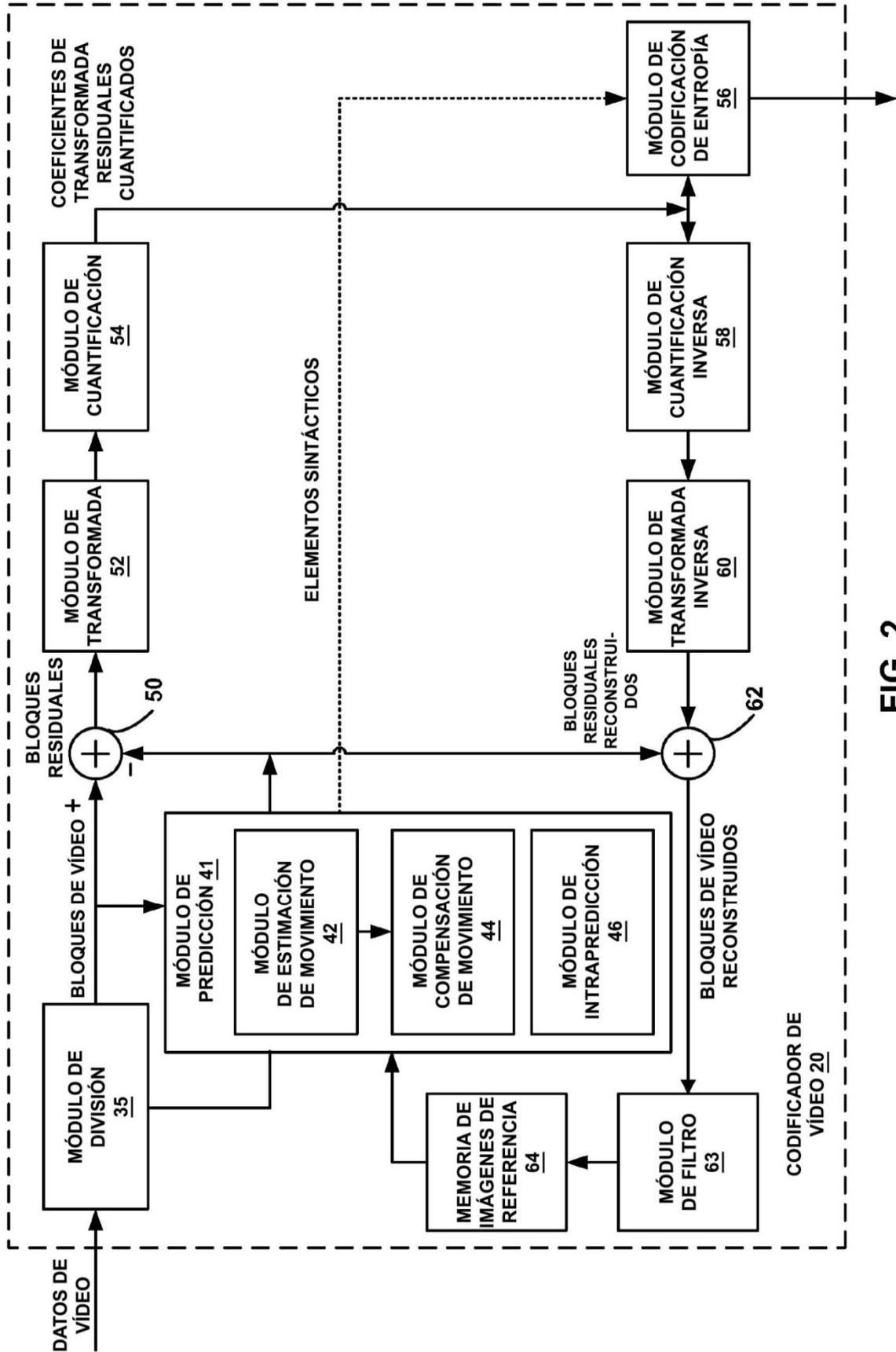


FIG. 2

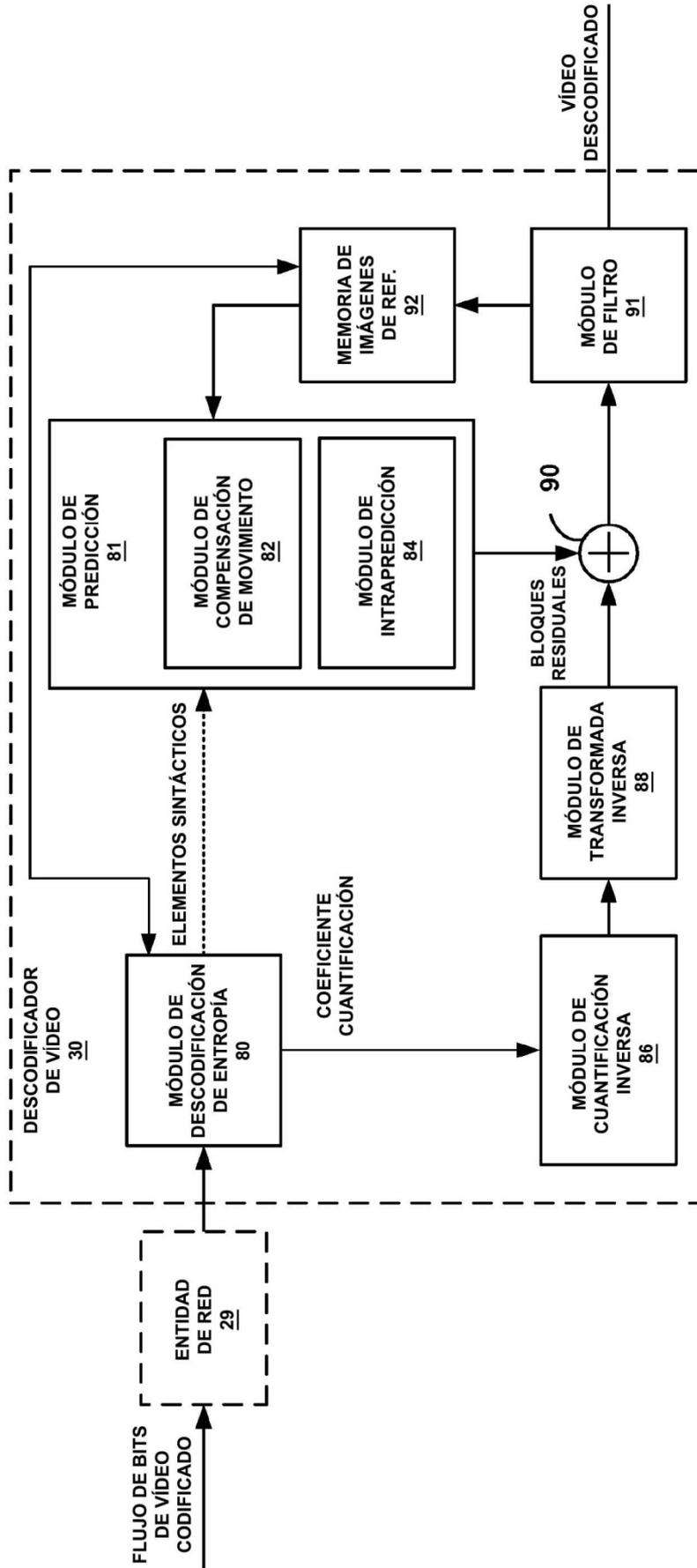


FIG. 3

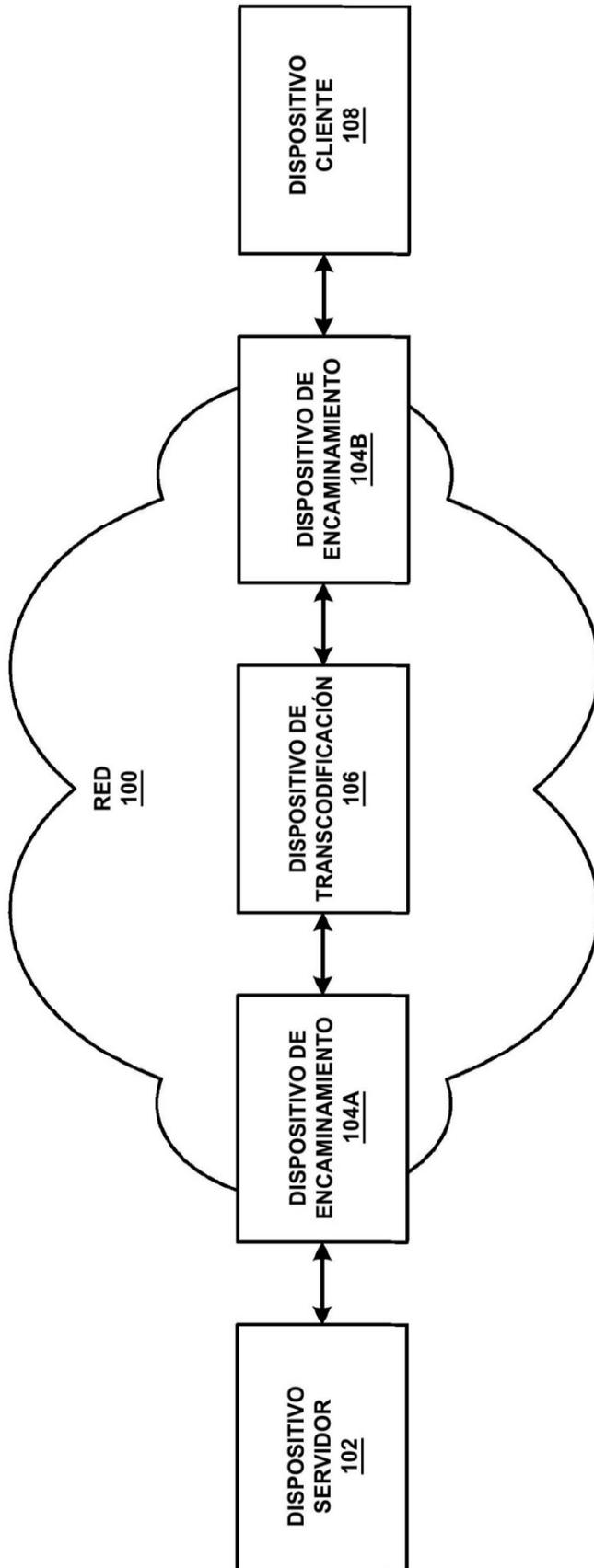


FIG. 4

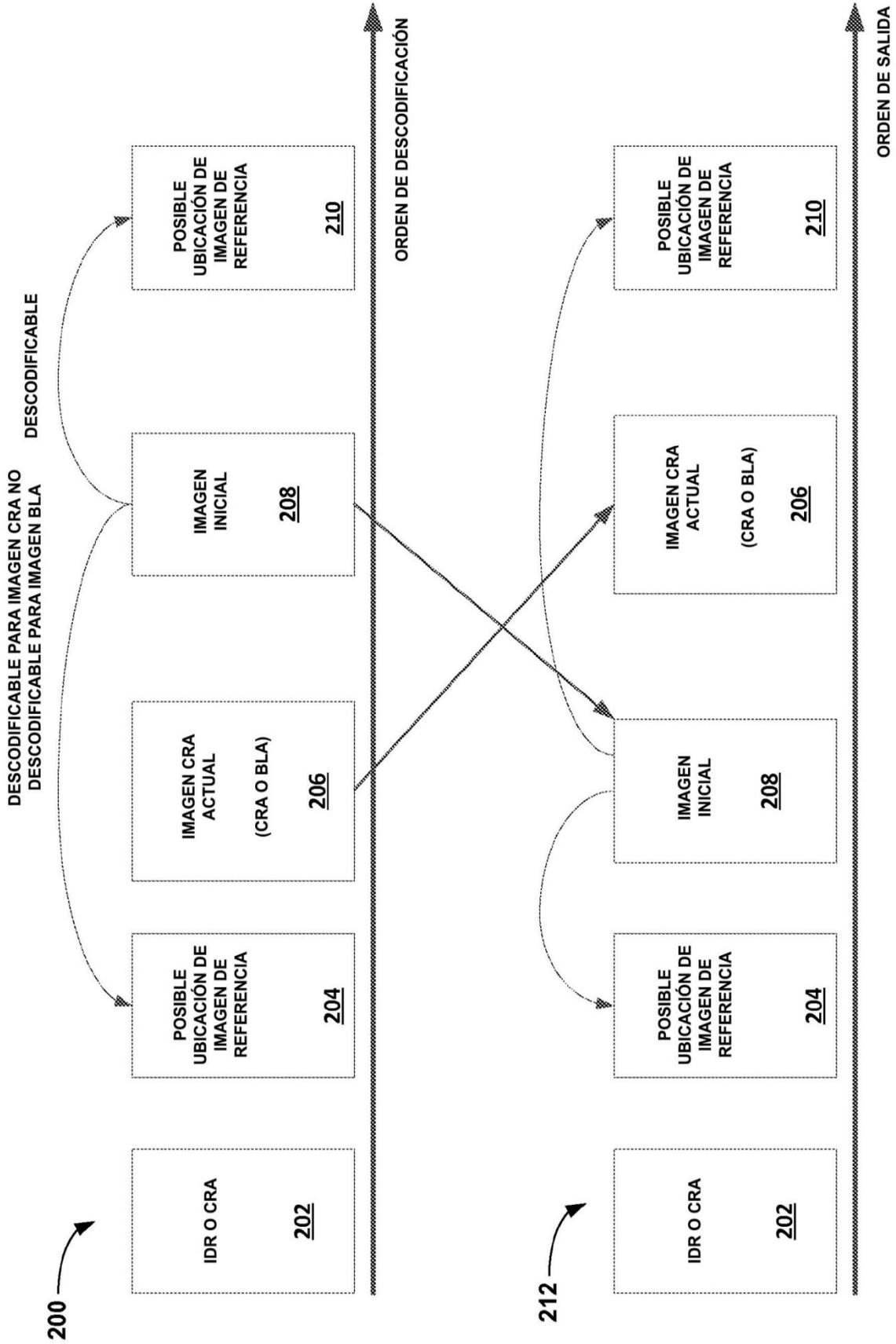


FIG. 5

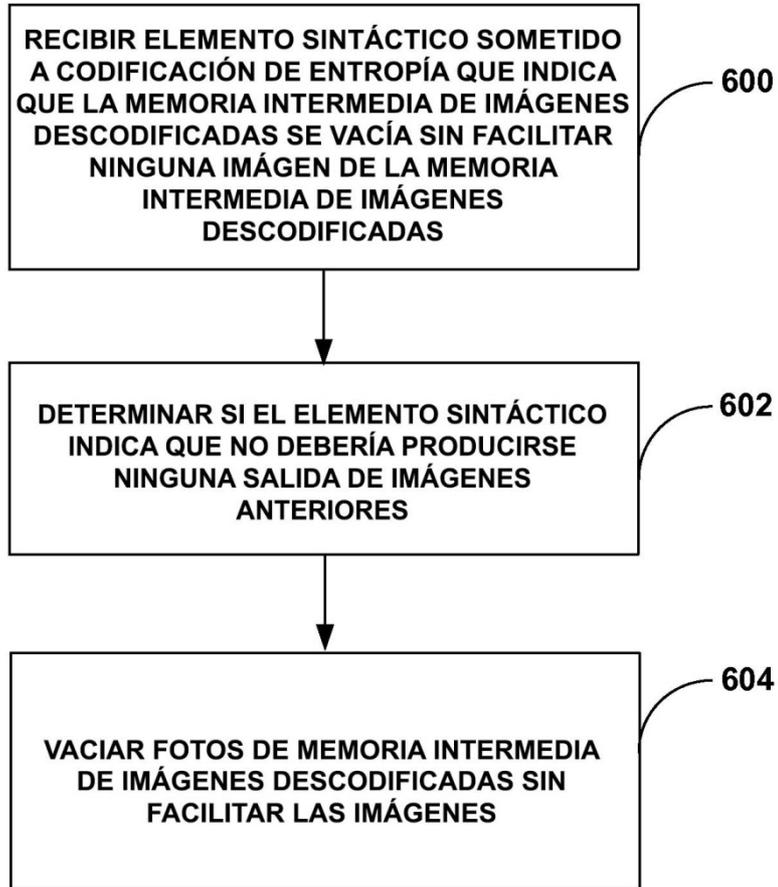


FIG. 6

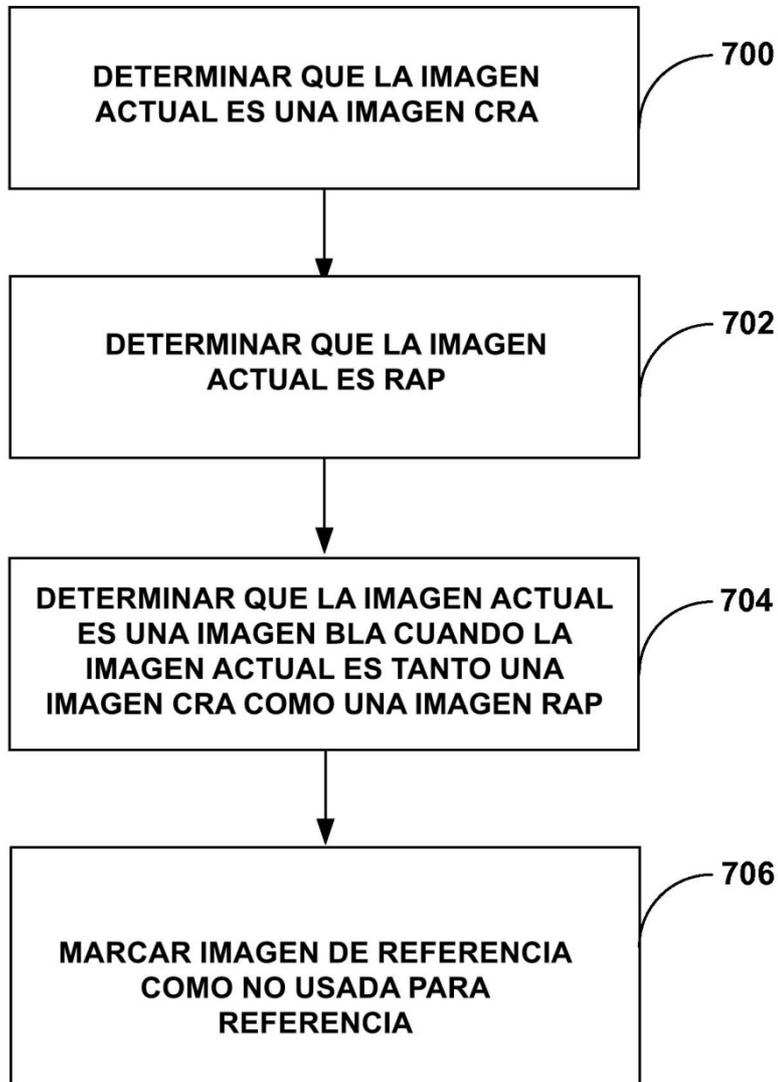


FIG. 7

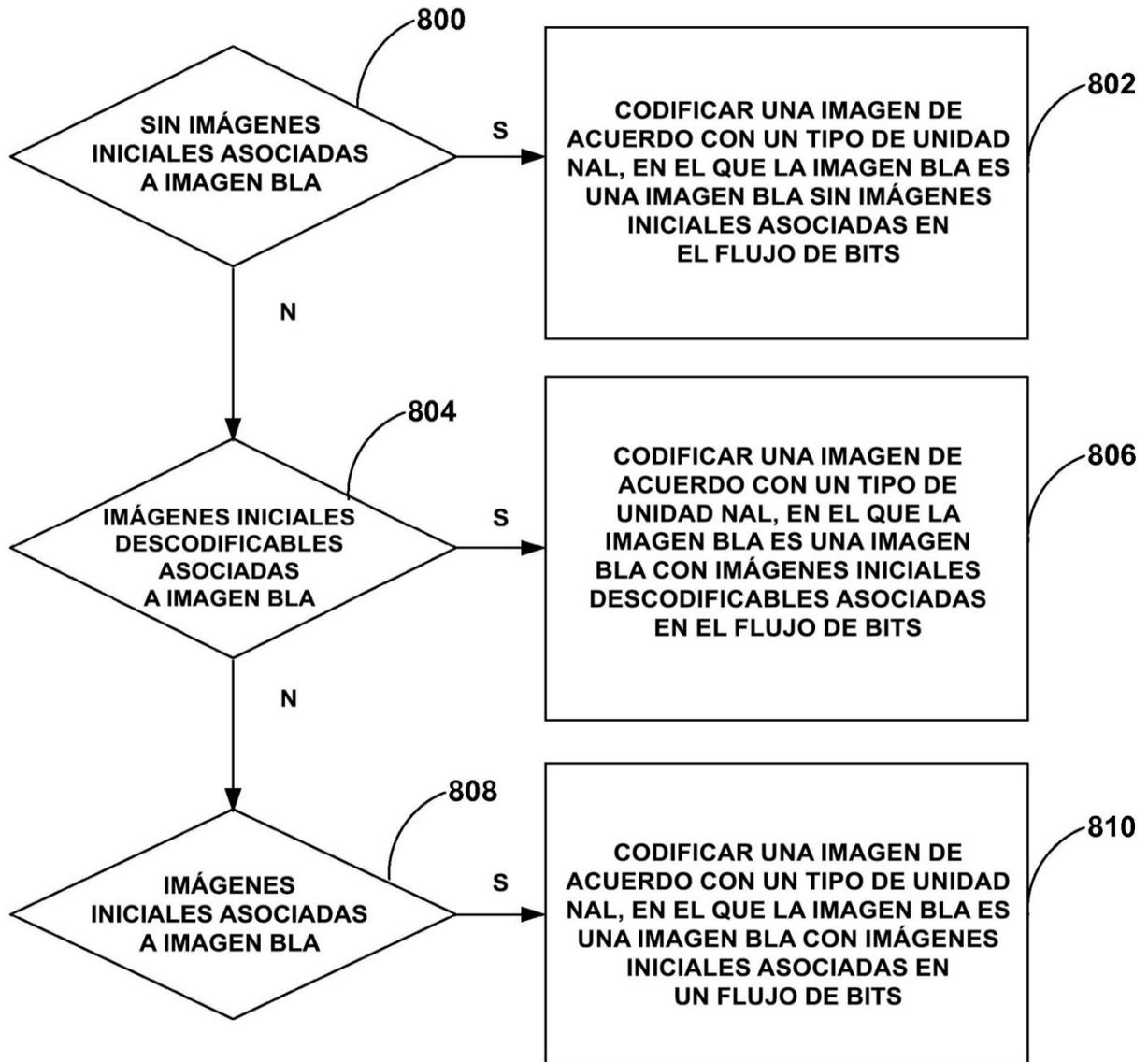


FIG. 8

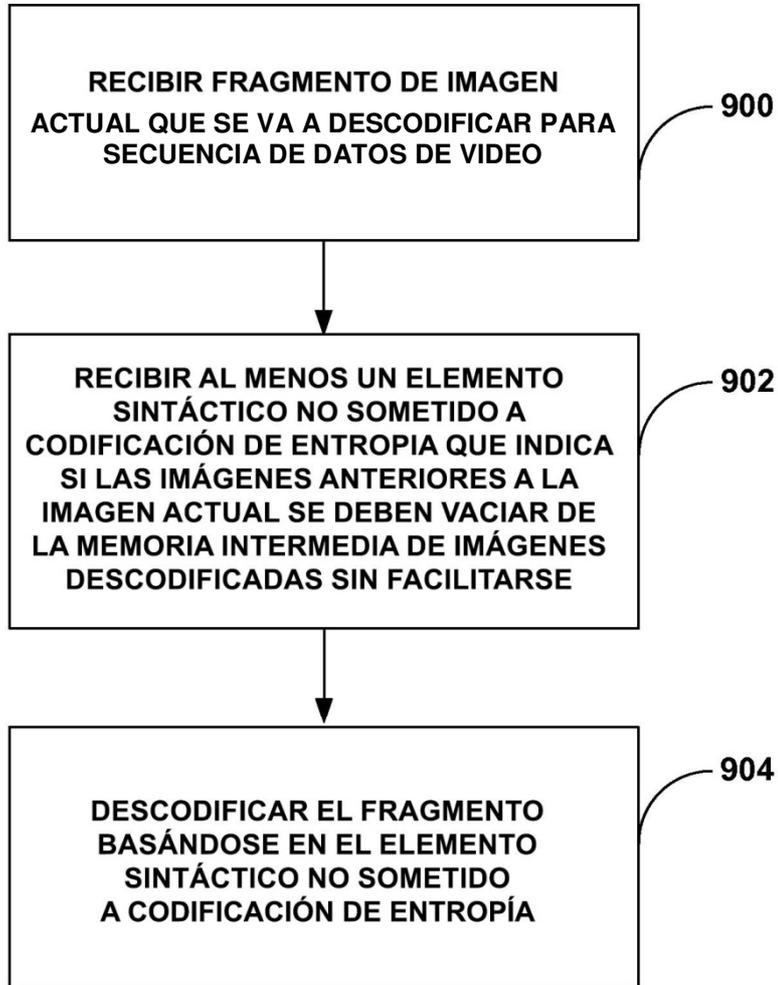


FIG. 9

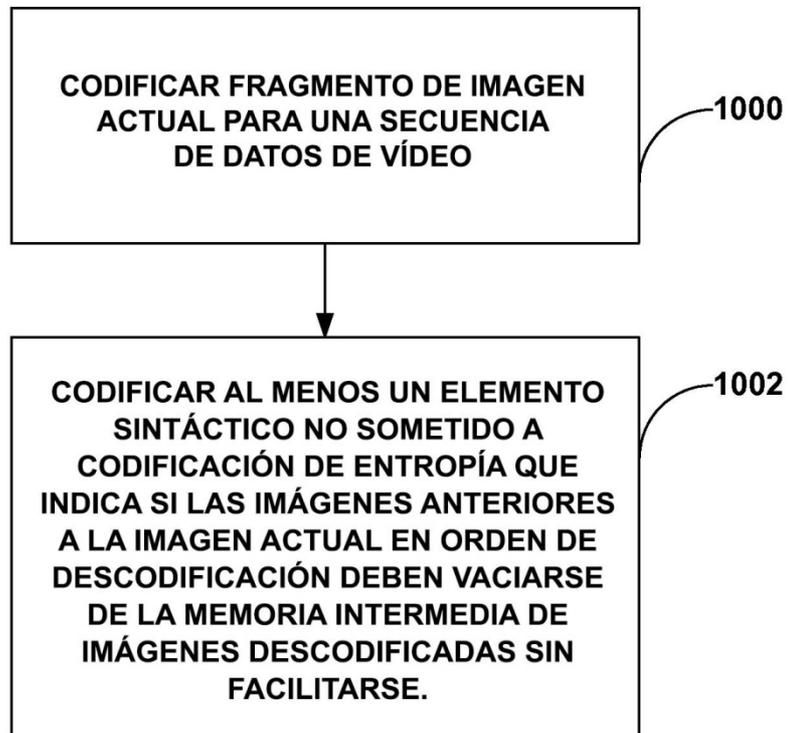


FIG. 10